

Project Research Report

ROBOT DETECTION SYSTEM DESIGN ABOUT FRONT-FOLLOWING TECHNOLOGY

PART 1

**Jinwei Lin
(JY Lin, Shenzhen, China)**

December 06, 2018

 ORCID: <https://orcid.org/0000-0003-0558-6699>

INTRODUCTION

1.What is front-following technology?

Front-following technology is a type of human-following technology. It is a robot technology and human-computer interaction technology with few researches and great research value and plenty of application scenarios. The human-following technique can be divided into three types: back-following, side-by-side and front-following, depending on the location of the robot when the robot follows humans. Side-by-side and front-following are technically more difficult than back-following. One of the basic reasons is that when implementing side-by-side and front-following, the robot often need to predict people's future actions, while back-following is basically unnecessary.

2.Why do we need to explore front-following technology?

Front-following is more technically difficult to implement than the other two human following technologies, but front-following technology is more practical and can be applied in more areas to solve more practical problems. Front-following technology has many advantages not found in back-following and side-by-side technologies. For example, Front-following technology can reduce the psychological burden of the followers and make the target person feel safer. There are two main aspects to security here. One is that if the users use robots of back-following technology, they will worry about whether the robots behind them will crash and hit them. On the other hand, if the robot adopts the front-following method, the user can supervise the robot in front of them that is performing the following tasks in the first perspective. Such a following mode is more useful when the user uses the robot to carry the load and needs to view the goods at any time.

3.What is the content of this project research report?

The research report of this project describes the algorithm research and theoretical analysis of Front-following, an advanced human-computer interaction technology. In order to better reflect and describe the implementation principle of Front-following technology and the author's self-innovation technology, we adopt the combination of partial and whole, start from the basic algorithm, and then gradually increase the depth and difficulty of the explained algorithm. During the elaboration of the technical principles, various algorithms will call each other to implement the joint use of the algorithms. In this project report, all of the algorithms are designed independently by the author. Since this project research report was completed independently by the author, it's compose lasted for 36 days, what is more, it was not enough time to write the report every day. Therefore, there are inevitable problems or insufficient in neglect or omission

in this report. We hope that you will give us advice and guidance if you find any shortcomings. The purpose of writing this project research report is to share our research results and research ideas and strategies with you through this report. We also hope that our report can serve as a reference and auxiliary advice for relative researchers. Thank you very much for your reading! Thanks!

DESIGN ANALYSIS

I. Project purpose

This project aims to set up a human interaction robot based on front-following technology, which can realize the function of following the target person in front of the target person. The prototype of the project robot's product mechanical drive is positioned in a simple front-following four-wheeled robot.

2. Project requirements

Design and build a physical robot. The robot has a front-following function. It has the function of predicting the path and direction of the target person. The robot can use sensors, but not RGBD cameras or other types of cameras. The design principle of the robot should try to ensure that the personal privacy of the target person and make them are protected to the utmost extent.

3. Starting points and key points

3.1 Separation and combination system design ideas

When designing a relatively large technical solution or research solution, we usually do not consider the entire technical solution as an inseparable whole, from beginning to end, design step by step. Such a design strategy may make the whole design scheme very cumbersome in the revision and improvement of the later project, and may cause the design scheme to be disconnected, and the implementation strategy of such a design strategy is not very efficient too. Therefore, we adopt the idea of separation design to improve project design efficiency. When the research of each part of the system is completed, each part is combined to carry out the overall analysis and optimization of the project.

We divided the entire technical solution into 8 major studies, namely:

1. Mechanical and circuit structure design;

2. Sensor system construction;
3. Structure of the detection model;
4. Robot following and control model;
5. Introduction of gait analysis;
6. Filter analysis;
7. Exception handling model;
8. Functional and technical expansion`n.

Due to the time limit for writing this report, this project only addresses the first four parts of the above sections. The last four parts will be arranged in the second edition of the project research report for detailed explanation, so stay tuned.

3.2 Key points for efficient technology solutions

1. Considering the actual use, the horizontal interface volume area of the robot should be similar to or slightly larger than the target person's volume.
2. When the robot is working at a close distance, the distance between the robot and the target person should not be too large.
3. The program should try to ensure that the robot is directly in front of the target person to achieve best experimental results.
4. When designing a robot, it should be fully realized that the final robot is to be put into practical use. Therefore, first of all, when designing the technical solution, we should fully consider the error of the detected data and the signal noise as well as the influence of specific environmental factors. For this practical application of technical solutions, the best way to assess their strengths and weaknesses is field experiment verification. Through the analysis of specific experimental results, efforts to optimize and upgrade the improved technical solutions.
5. The postures of human beings' walking are almost regular and habitual, and the walking gait of a person can reflect personal characteristics. It can even reflect some physiological and pathological features of the individual. Related to this is the field of gait analysis and gait recognition research. Therefore, we can take gait analysis into account when modeling human walking models.

3.3 Introducing new concepts:

Detection error and prediction error

In order to better analyze the errors caused by predicting the future path of humans, we divide the errors caused by predicting future human paths into two types: detection error and prediction error. The detection error refers to the error between the detected data and the real data when the human body parts are detected in real time within a fixed time. The prediction error refers to the error between the predicted human future

path and the real human future path when the future path of the human is predicted in real time within a fixed time. Thinking from the definitions above, these two kinds of errors are relative. From the definitions we can find that the detection error measures the error in a short time, and the estimated error is the error measured over a long period of time. These differences are especially important when we set the size of the path that predicts the future path of humans. A direct reason is that the larger the distance of the predicted future path of the human being, the greater the initiative of the robot, because the robot can be shorter after knowing the predicted data information of the future path that is predicted to be sufficient for humans. A more complete response (e.g., stay in front of humans, avoid big turns, and automatically return to the front of humans after a turn). Conversely, if the predicted distance of the human future path is smaller, the initiative of the robot will be smaller, and the time left for the robot to react will be shorter.

But this does not mean that the larger the path to predicting the future path of mankind, the better. Combined with the definition of detection error and prediction error, if the predicted future path of human is shorter, the corresponding detection error will be larger, and the corresponding prediction error will be smaller. If the predicted distance of the human future path is longer, the corresponding detection error will be smaller, and the corresponding prediction error will be larger. This is because, the shorter the distance we set for the future path to be predicted, the less test data we need to analyze, and the fewer the number of detections required. The corresponding detection data contains the detection error. Big. Because when the amount of detected data is relatively large, we can balance or reduce the corresponding detection error by means of data processing. Accordingly, if the amount of detected data is smaller, the uncertainty of the corresponding detected data is larger.

In general, we can only predict the future path of a short distance, that is, the predicted future path cannot be too large. Because the walking path and direction of travel of a human being are changeable, a walking human being may change his or her destination or direction of travel at any time, and this generally does not tell the robot who actively follows him or her. We cannot expect to predict a large future path of humans by detecting a small segment of human path, because this is full of errors.

However, we can't set the path of human prediction of the future path very short for the reason that to reduce the prediction error, because this will lead to an increase in the corresponding detection error. Since the robot does not know the detection error at this moment (n.b., this moment is meaning the real time in this report) during the execution of the follow-up process, although the robot can compare and predict the prediction error of the previous moment by comparing the predicted data with the predicted data. However, please note that the prediction error of the previous moment has no practical help for the prediction of the future path of the human being at the next moment. The detection error at this moment is only helpful for correcting the future path prediction of human beings at this moment. However, since the robot cannot obtain the detection error at this moment by itself when detecting at this moment, the robot needs to get the detection errors with external forces at this moment. For the reason that this report focuses on the existence of robots in the form of a single individual, the

expansion of getting the detection errors with external forces will be explored through expanding discussion sections.

The main purpose of why we set the detection error and prediction error is to measure the accuracy of the prediction results. We use known data to predict unknown data, which means that we predict human future path data from human path data that has been detected. In general, the more known data is used, the more or the more accurate the future data can be predicted. However, due to the variability and temperament of the human walking path, this rule does not apply well in reality. In our proposed prediction model, human path data or gait data is used to predict future path data or future gait data of the human. For convenience of presentation, we will use the existing path data or gait data of humans to be predicted as sample data. We relate the detection error to the prediction error to the sample data. Therefore, the larger the data amount of the sample data selected at the present time, the larger the data amount of the path data of the future path of the human being predicted, and of course, the corresponding prediction error will become larger, but the corresponding detection error will be smaller. In other words, the amount of sample data must be appropriate, not too much or too little. The best solution is to find the data volume of the most suitable prediction data so that the prediction error and the detection error can reach a balanced state. That is, when predicting the future path of humans, the amount of sample data has an optimum value, so that the sum of the detection error and the prediction error is the smallest.

Promotion, we found through analogy and related experiments that all the phenomena that predict the future data through the detection data will have a relative tuning relationship between the detection error and the prediction error. Interested readers can further conduct relevant research.

Multiple pre-judgment

In many current papers on Front-following technology researches, most of the detection methods are directly detect human location data and use them as predictive model data. There are also some papers that study the detection of specific parts of the human body to obtain test data (e.g., human legs or torso). But they are biased to detect only a certain part of the human body being tested. This can cause a lot of error in some extreme cases. Different from them, our detection model proposed by the technical solution will simultaneously detect multiple parts of the human body, and one of the advantages of this is that the accuracy of the prediction result can be effectively improved, and the error caused by some extreme situations can be well avoided. The specific details will be explained later.

Gait prediction and gait advance

The concept of path prediction has been mentioned in related researches introduced in many papers. In the Front-following techniques, the robot must predict the future path of the target person in real time in order to respond accordingly. In these studies, the researchers focused on calculating the most probable path of the target person in the future by a good algorithm, and calculating the best tracking execution path based on this path, and then driving the robot to track the motion. These studies focus on detecting the future path of the entire segment rather than dividing it into a number of small segments. One of the reasons for this is that studying a large whole path is enough to balance the error caused by studying only a small path. Therefore, the size of this small segment is worth thinking about. That is what the optimal size of the known walking path used to predict the target person should be. For ease of explanation, we will refer to the short path used to predict the future path of the target person as the sample path. From the concept of detection error and sample error, we know that the sample path is composed of sample data. So, again, the size of the sample path has the most appropriate amount. But what is discussed here is not the prediction of the future path of the target person, but the prediction of the future gait of the target person, which we call gait prediction. The shift in research objectives from sample data to subdivided sample data is the shift from the prediction path to the predicted gait. Obviously, the gait prediction research is to predict the gait action that the target person will take in the next short period of time for a certain moment gait of the target person.

For the reason that we can only predict future gait data information by gait data information that has been detected, gait prediction also has a relative relationship between detection error and prediction error. But in fact, the time for humans to take a normal step is usually very short. And under normal circumstances, human gait movements are relatively stable within the action range of human taking a step. Rarely occurs that human gait movements change dramatically within the action range of human taking a step. This is related to human walking habits and gait habits. Therefore, human gait prediction has high predictive directionality and guiding significance. We will add gait prediction to the predictive model, and only predict the possible action of the human gait in the range of motion of the step, that is, and predict the change of the future gait of the target person's leg from the time of lifting to landing.

Gait presupposes means that in a time when a certain human leg has been lifted but has not yet landed, the gait prediction mechanism predicts the future gait of the human next moment, and the robot adjusts the gait in real time and performs corresponding movements. It also in order to achieve the process of staying ahead of human gait in the next step. Therefore, the gait is a highly real-time following approach, which is more than enough to keep the robot in front of humans for follow-up without producing significant motion shifts. Our defined path prediction and path following are macroscopic prediction and following, while gait prediction and gait leading are microscopic. In specific predictive modeling, we will combine macro and micro predictions. In the specific control modeling, we will also combine the macro leading

and micro leading. This is also based on the design concepts and principles of multiple pre-judgment. See the Predictive Models and Control Models section for more details.

Pre-judgment restrictions based on ergonomics

We know that the physical and skeletal structure of the human body limits the movement of human limbs and other parts of the body to a certain extent. Human walking involves multiple bones and muscle movements in humans. This makes the body's body trunk and limb movement have certain restrictions. Under this constraint law, we can predict some human behaviors and incorporate them into the prediction model to enhance the accuracy of the prediction model. For example, when a human raises his leg, the probability of the leg moving forward is much greater than the probability of backwards. This kind of knowledge can be applied to gait prediction. The idea of restricting the pre-judgment is that the action pre-judgment is carried out under certain restriction rules, so that the accuracy of the pre-judgment will be greatly improved. The restricted structure of anatomy can help us optimize the predictive model. See the Predictive Models and Control Models section for more details.

MECHANICAL STRUCTURE AND CIRCUIT DESIGN

Mechanical structure is an important part of a robot. The mechanical structure is equivalent to the skeleton of the robot, and the circuit is equivalent to the muscles and blood vessels of the robot. Robots combined with mechanical structures and electronic circuits already have a hardware foundation. In addition, robots need a soul. This soul is the algorithm. The algorithm belongs to the software part. Only the combination of mechanics, circuits and algorithms, the robot has the basis to achieve basic functions. Since this report focuses on the algorithmic implementation principle of the robot, the mechanical and circuit design will be discussed in detail in future upgrades, and only a brief description will be given here.

The robot designed in this report is based on four-wheel drive. The premise of the implementation of the robot algorithm should be that the mechanical and electronic parts can respond well to the requirements of the software algorithm. Some of the mechanical and electronic designs will be discussed in the next section on sensor systems.

CONSTRUCTION OF SENSOR SYSTEM

Analysis

To build a well-functioning sensor system, it is necessary to fully understand and analyze the functions and implementation conditions of various sensor candidates. Based on the requirements of our research project, we can use the following sensors to build the sensor system of the robot:

1. Laser ranging sensors

Laser sensors are mainly used to detect obstacles and interferences in humans and the environment. This kind of sensor is widely used in the design of robot following system, and it is also the most accurate range measuring sensor with mature technology. Widely used in ranging and positioning.

2. Thermal infrared imaging sensors

The biggest function of the thermal infrared imaging sensor is to be able to detect the thermal infrared rays emitted by the organism and to form a thermal infrared image. Since the personal privacy information about the target person displayed by the thermal infrared image is very small, it is feasible in ensuring personal privacy of the human being. Although the current thermal infrared technology may have a problem of insufficient accuracy, the thermal infrared imaging sensor is still an excellent sensor to choose from without precise positioning. We recommend the use of thermal infrared imaging sensors in this project to distinguish organisms from non-living organisms. Further, according to the detection area and temperature characteristics, humans are distinguished from other organisms, even under the premise of high progress. The target person is distinguished from other unrelated humans.

3. Ultrasonic sensor

The most prominent advantage of ultrasonic sensors is their low cost, and correspondingly, the cost is not high enough. Can be used as an alternative sensor solution. Ultrasonic sensors are optional in this project. The ultrasonic sensor is not described in the sensor system construction section, but will be described in the section on the detection and acquisition of the upgraded version report.

Based on the design idea of separation and combination system and the design idea of multiple pre-judgment, we will scan the target person by scanning the whole body. Our detection method is not only for a certain part of the body of the target person, but for the whole body of the target person. In the whole body detection, based on the separation and combination system design idea, we divide the whole body of the target person into four main parts for detection: head detection, torso detection, legs detection,

and feel detection. When detecting the body structure of the target person, the sensor we use is LRF, for the reason that to ensure the accuracy and reliability of the test data.

Design of Sensor System

In this project, we propose two sensor model system models, one based on a rectangular four-vertex angle model and one based on a rectangular four-sided center point model. Both modes have their own advantages and disadvantages. The model based on the rectangular four-vertex angle design has high detection precision for the detection areas at four corners, can perform double detection of a wide range, and can perform detection data superposition processing to achieve less detection error. The disadvantage of this model is that the installation structure is more complicated. The model design based on the rectangular four-sided center has the advantages of simple installation, stable scanning structure, strong front scanning capability, dual-mode data processing, and the ability to detect and track the influencing factors of the surrounding environment while the robot following target person. This scanning mode also has the advantage of performing a wide range of double detection data superposition to achieve less error. Moreover, the scanning detection accuracy of the scanning area in the four corners is also high, but not higher than the rectangular four-corner model. Due to the limitation of writing time, the project report will first discuss and compare the differences between the two scanning modes in the later algorithm control part. Then, in the following algorithms, all of them will be explained and analyzed based on the rectangular four-sided central model. And the model is very suitable for the dual-mode scanning algorithm proposed in this report. For other detailed analysis based on the rectangular four-corner model, please pay attention to the new version of the project design report.

Sensor System I

This section describes a sensor system based on a rectangular four-corner angle. In this mode, the robot's sensor system consists mainly of eight LRFs, abstracting the robot into a cube, and the eight LRF sensors are mounted in two corners above the cube at two or two corners. One apex angle is provided for each LRF group, and one LRF is composed of two LRFs that are in the same parallel to the Z axis. For convenience of explanation, we will give the sensor system model of the four apex angles of the robot rectangle as the four-angle sensor system model, referred to as the four-corner model. As shown in Figure 1:

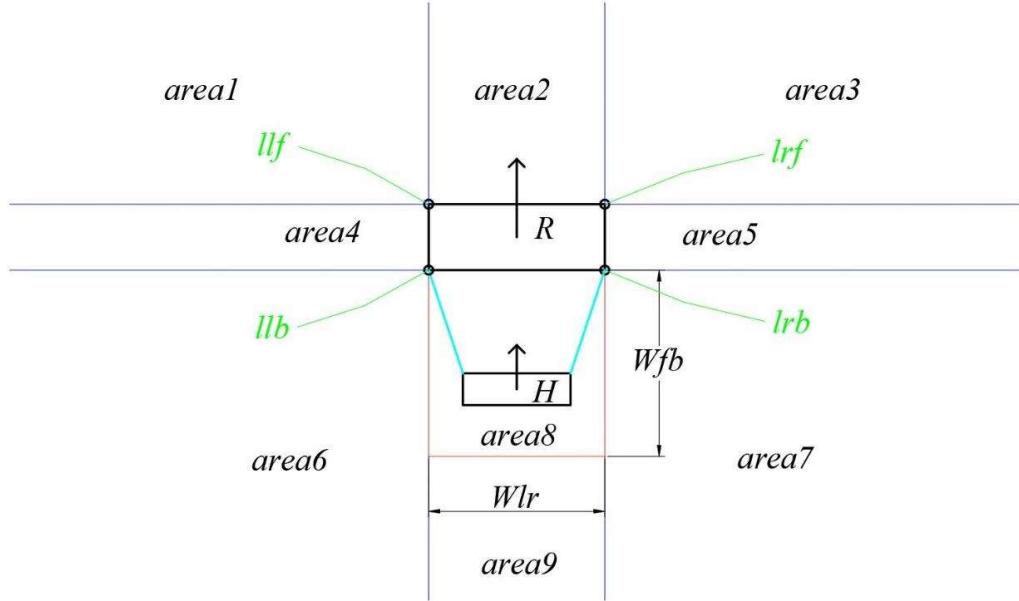


Figure 1. Distribution of LRF detection areas based on four-vertex model robots.

We plot in 2D. Abstract the robot into a rectangle called a robot rectangle and mark it with R . We also abstract the target person into a rectangle called a human rectangle, marked with H . Here we define the width of the robot rectangle to be slightly larger than the human rectangle corresponding to the target person. Eight LRFs are mounted in the four corners of the robot rectangle. The two LRFs in the left front corner are labeled llf , and the two LRFs in the right front corner are labeled lrf . The two LRFs in the left rear corner are labeled llb , and the two LRFs in the right rear corner are labeled lrb . The horizontal and vertical lines have been changed in the four corners of the robot. The robot and the human space are divided into nine areas, which are marked as $area1$, $area2$, ..., $area9$. There is a restricted area behind the robot rectangle, which is just $area8$. The restricted area is a rectangle whose left to right width is equal to the width of the LRF at the back of the robot rectangle, labeled Wlr . The front and rear widths of the restricted area are marked as Wfb . The design premise of the restricted area is to be able to completely include humans in the restricted area, but to leave a certain amount of free space for human activities. Please note that the restricted area is only a virtual algorithm area, and the real human activity is not restricted by this area. The idea of a restricted area is to define a virtual area. If the target person is in this area, then the robot is locked. If the target person is not in this area, then the robot at this time is said to be in a disengaged state. Obviously, when the robot is locked, the robot shows good forward tracking to humans. When the robot is in the disengaged state, the robot cannot show good forward tracking to humans. At this time, the robot following can no longer be called front-following. In Figure 1, when the robot is in $area8$, the robot is said to be in a locked state. When the robot is in another area, the robot is said to be in a disengaged state.

The purpose of setting up the restricted area is very clear, to serve the highly accurate front-following algorithm. If a robot is always in a disengaged state in the

process of following the human forward, then the robot is obviously not a good front-following robot. Namely, a good robot front-following algorithm should keep the robot as locked as possible. Our robot control algorithm is based on this design.

In the above analysis, we assume that the cross-sectional area of the robot is larger than the human body's, which is more common in the actual design of general robots. There was another situation what we can't ignore under these circumstances is that the cross-sectional area of the robot is smaller or very close to humans. In this case, the above modeling does not apply. In order to solve this problem, we propose the concept of proportional expansion area model. The purpose of the expansion area model is to make the virtual robot and the rear of the real robot on the same plane, and then construct the virtual robot, so that the cross-sectional area of the virtual robot is larger than that of the human body using the virtual robot instead of the real robot for algorithm calculation. This will facilitate the setting and operation of restricted areas, as shown in the following figure:

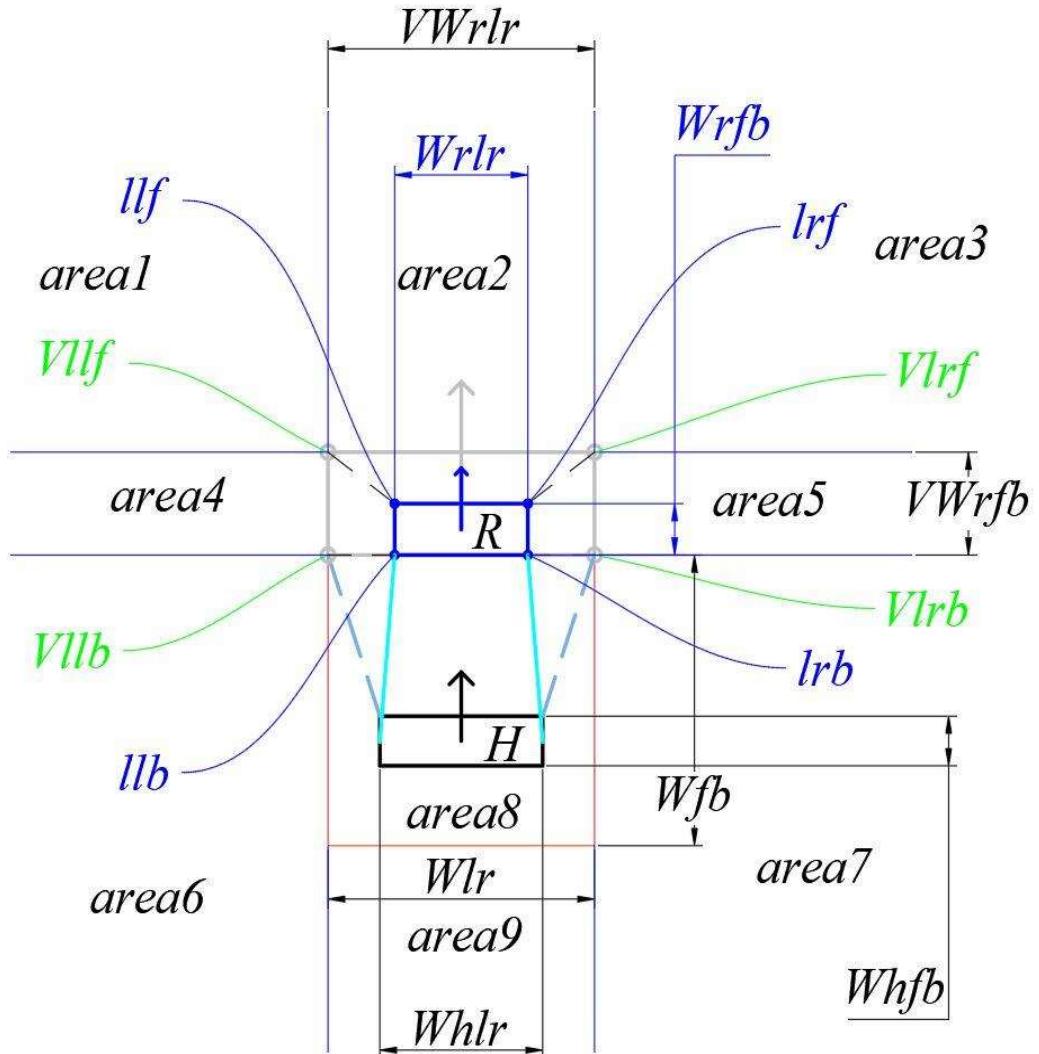


Figure 2. Proportional extended area model.

As shown in Figure 2, the blue rectangle represents the real robot rectangle, marked with R. The gray rectangle represents the virtual robot rectangle. The positions of the LRFs of the two corners of the four corners of the real robot are labeled llf, llb, lrf, lr, respectively. Let the four vertices of the virtual robot be P₁, P₂, P₃, and P₄. The four vertices defining the restricted area are P₃, P₄, P_a, P_b.

After the scale expansion, the positions of the LRF groups in the four corners of the virtual robot rectangle are Vllf, Vllb, Vlrf, Vlr. The front and rear width of the virtual robot rectangle is VWrb, and the left and right width is VWrlr. It is assumed that the front and rear widths of the restricted area are Wfb, and the left and right widths are Wlr. The front and back width of the human rectangle is Whfb, and the left and right width is Whlr. The front and back width of the real robot rectangle is Wrfb, and the left and right width is Wr. In this case, the left and right widths of the real robot rectangle are smaller than the left and right widths of the human rectangle, that is, Wr < Whlr. Wfb and Wlr are automatically set up following the control mechanism. The process of setting is as follows:

First, we need to pre-set two experimental experience values, namely p and ξ , ε , and specify:

$$\left\{ \begin{array}{l} p = \frac{VWfb}{Wrfb} = \frac{VWrlr}{Wr} = \frac{Wlr}{Wr} \\ k1 = \frac{Wrfb}{Whfb} \\ k2 = \frac{Wrlr}{Whlr} \\ p > 1 \\ p = \xi + k2 \\ \varepsilon = \frac{Wfb}{Wlr} \end{array} \right. \quad (1)$$

Here ξ and ε should be the constant ratios of the robots that are designed to be specific to a particular mechanical and circuit design. Therefore, for a structure-determined robot, the values of ξ and ε should be deterministic or not change (n.b.: the post-upgrade or modification of the robot is considered here). However, since the human Whlr that the robot follows each time is not necessarily the same, the corresponding p value is not necessarily the same each time the robot performs the following process.

In theory, $p > 1$. The optimal range of p should be $1.2 \leq p \leq 2$. K1 is not necessarily equal to k2, and our focus is on k2. After experimental demonstration, we can determine the proportional constant of a dividing line, such as 1.2. If k2 satisfies $k2 > 1.2$, there is no need to scale up, otherwise scale expansion is required. The process of scale expansion is as follows:

1. *The robot calls the sensor system to scan the left and right width Whlr of the corresponding human rectangle when the human stands;*
2. *Find k2 from (1), and find p in combination with the inherent ξ of the robot;*
3. *From the formula (1), combined with the inherent ε of the robot to find Wfb, Wlr;*
4. *Determine the VWfb and VWrlr by the formula (1) in combination with the known*

$Wrfb$ and $Wrlr$ and p ;

5. Set the geometric center coordinates of the real robot rectangle to $(0,0)$, as shown in Figure 3, then there are:

$$\left\{ \begin{array}{l} x1' = -x2' = x3' = -x4' = xa' = -xb' = -\frac{1}{2}VWlr = -\frac{1}{2}Wlr \\ y1' = y2' = \frac{1}{2}VWfb = \frac{p}{2}Wfb \\ y3' = y4' = -\frac{1}{2}VWfb = -\frac{p}{2}Wfb \\ ya' = yb' = -(Wfb + \frac{1}{2}VWrfb) = -(\varepsilon \cdot Wlr + \frac{1}{2}VWrfb) \end{array} \right. \quad (2)$$

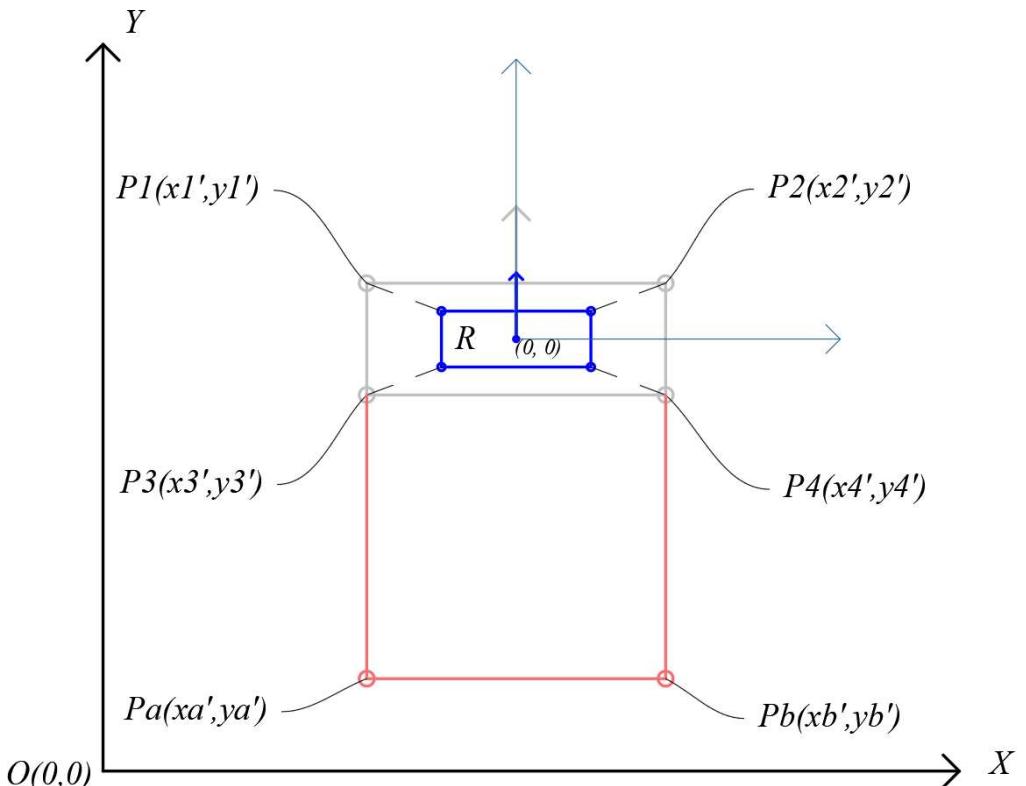


Figure 3. Schematic diagram of the key points of the scale-up regional model I.

6. After obtaining the initial coordinates of each point, move the whole robot model in the model down by $1/2(VWrfb - Wrfb)$ units in the negative direction of the Y-axis. As shown in Figure 4, the final coordinates of each point are:

$$\left\{ \begin{array}{l} x1 = x3 = xa = x1' = x3' = xa' \\ x2 = x4 = xb = x2' = x4' = xb' \\ y1 = y2 = y1' + \frac{1}{2}(VWrfb - Wrfb) \\ y3 = y4 = y3' + \frac{1}{2}(VWrfb - Wrfb) \\ ya = yb = ya' + \frac{1}{2}(VWrfb - Wrfb) \end{array} \right. \quad (3)$$

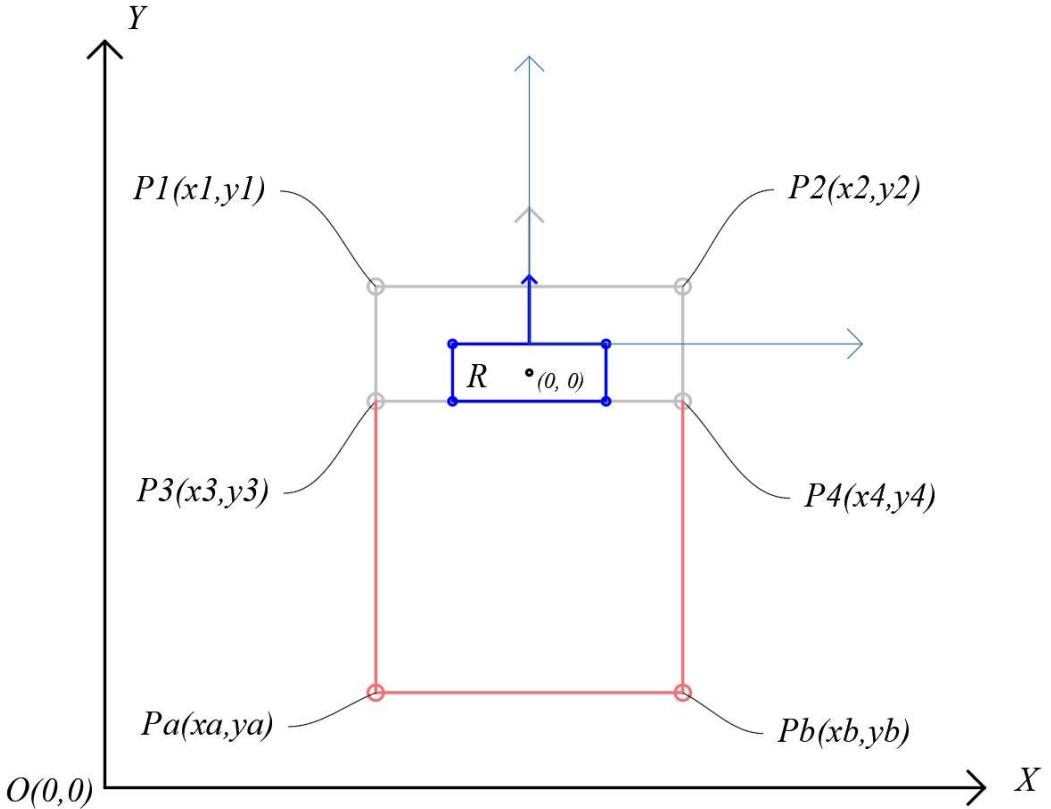


Figure 4.0. Proportion 2 of the key points of the proportional expansion area model, moving down the origin.

Please note that in general, the distance between the two LRFs in the LRF group in the two groups is very close on the Z axis. In general, we can ignore the height distance difference between them. However, in the pursuit of high-precision algorithms, it is also necessary to consider the difference in height distance between two groups of LRFs. The specific part will be covered in the algorithm analysis section.

Now, we can determine the coordinates of each vertex of the restricted area and the virtual robot rectangle. And we can determine the extent and size of the virtual robot rectangle and its corresponding restricted area. From the above analysis, we know that only the restricted area corresponding to the real robot rectangle can not satisfy the case of including the human rectangle while retaining a certain space, and then it is necessary to scale the real robot rectangle.

After determining the size of virtual robots and restricted areas, an obvious problem will be: if our robot only have four LRF groups distributed in four corners, how can we assign working areas to these four LRF groups to enable them to effectively develop their due detection capabilities? To solve this problem, we propose a 45-degree partition method. The specific implementation process of the 45-degree division method is as follows:

1. Using the four vertices of the virtual robot rectangle as the starting point, make

a ray with an angle of 45 degrees from the left and right horizontal boundary lines;

2. As shown in Figure 5, the area where the robot rectangle is located is divided into four areas according to the two-dimensional space: area1, area2, area3, area4. Then, two LRFs in each LRF group are allocated into two adjacent regions. For example, one LRF partition in the LRF group at point P1 in FIGS. 3 and 4 is specified to detect area1, and another LRF partition is used to detect area4. A similar partition designation is made for the LRF groups at the remaining three vertices. The final result of the division is that each detection area is checked by two LRFs.

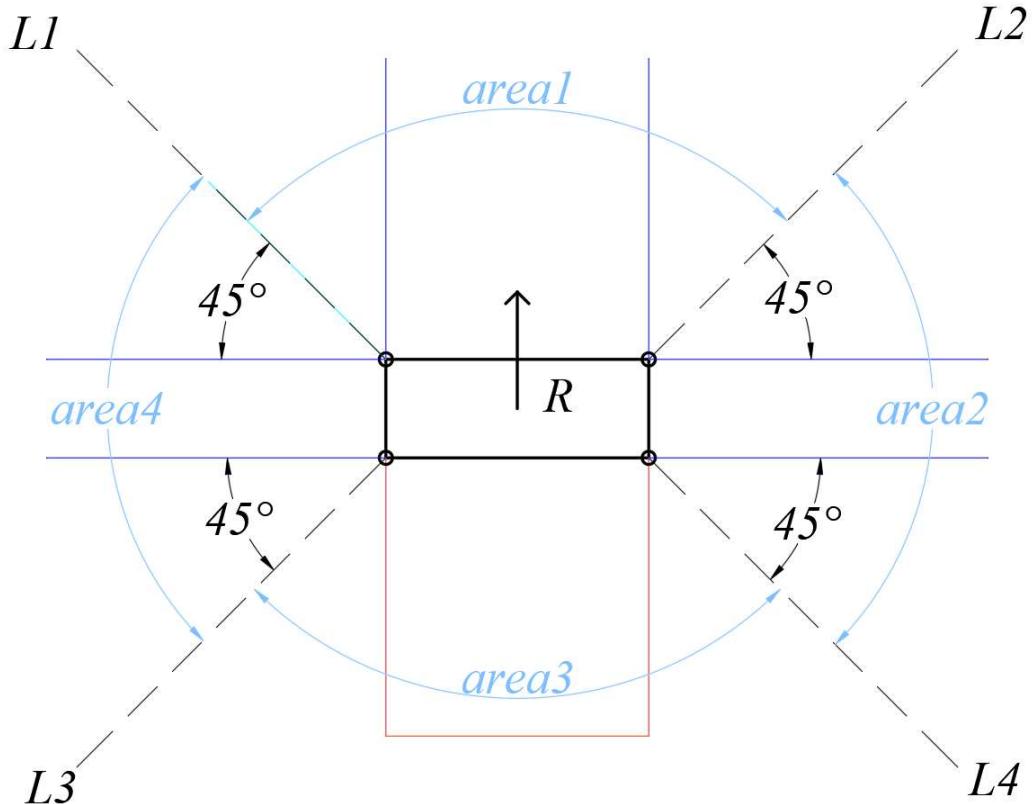


Figure 5. 45-degree partitioning method based on two-dimensional space.

3. As shown in Figure 6, assume that there are 8 points to be measured in the space, which are represented by P1, P2, ..., P8. Each checkpoint is subject to the following rules and definitions:

3.1. If the point to be measured is in area1, the two LRFs in front of the virtual robot are responsible for the detection. In fact, it is detected by the two LRFs in front of the real robot.

3.2. If the point to be measured is in area2, the two LRFs on the right side of the virtual robot are responsible for the detection. In fact, the two LRFs on the right side of the real robot are responsible for the detection.

3.3. If the point to be measured is in area3, the two LRFs behind the virtual robot are responsible for the detection. In fact, it is detected by the two LRFs behind the real robot.

3.4. If the point to be measured is in area4, the two LRFs on the left side of the virtual robot are responsible for the detection. In fact, it is detected by the two LRFs on the left side of the real robot.

3.5. With this division method, a special type of point will appear: a point (eg, P1) on the 45-degree dividing line, assuming that the detected object represented by the point (eg, P1) is just entering the detection scanning area. And at the moment of scanning, it is just on a certain 45 degree dividing line. When the robot detection system allocates LRFs for such points, it will drive the two LRFs of the same LRF group corresponding to the point to be jointly responsible for detection.

3.6. For those points located on the horizontal dividing line or vertical dividing line of the virtual robot rectangle, such as P8, we will ignore the influence of the horizontal dividing line or vertical dividing line of the virtual robot rectangle, and directly distribute the points according to the corresponding detection area. (area4) The two LRFs assigned to the left of the robot are responsible for the detection. For those points that are very close to the 45-degree dividing line, the LRF groups are still strictly distributed to detect those points according to the area where the center of the points are located and the division principle. For example, point P6 will be divided into area2, that is, the two LRFs on the right side of the robot are responsible for the detection;

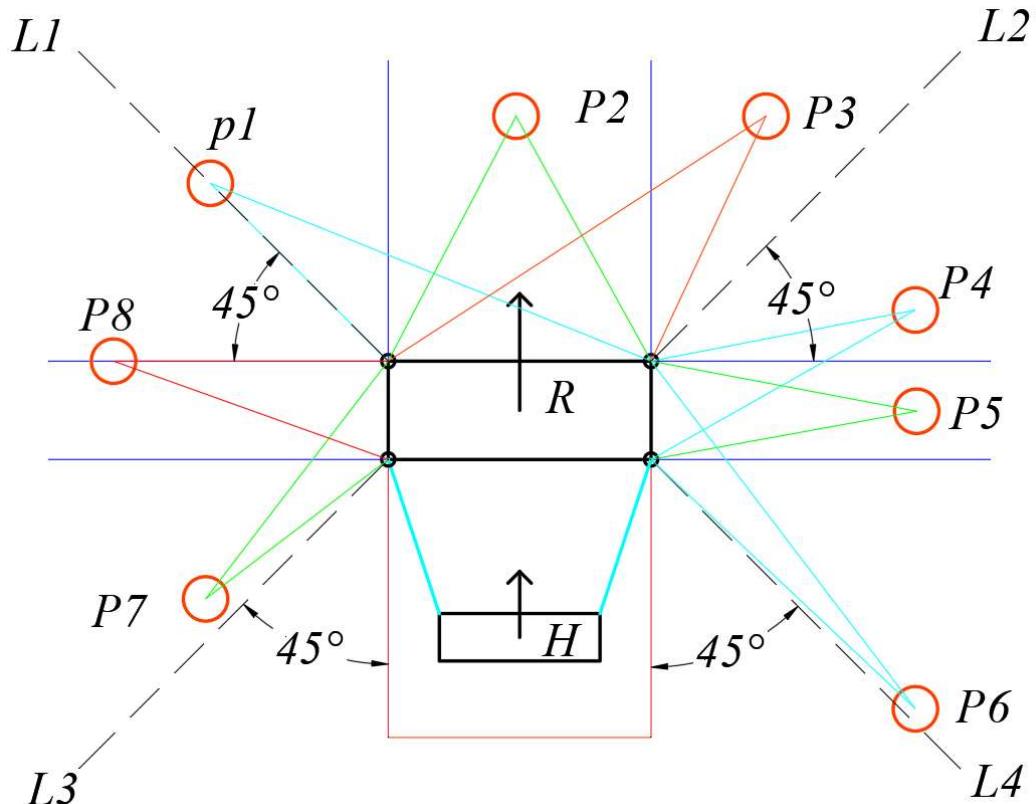


Figure 6. Distribution of LRF detection regions using 45-degree region partition method.

Sensor System 2

In this part, we will introduce a new sensor system. The system is also composed of eight LRFs, and each two LRFs constitute one LRF group, and then four LRF groups are respectively installed at the center of the four sides of the robot rectangle. Although in subjective feeling, the robot only changed in the position of the four LRF groups compared to the previous sensor system, the two sensor systems will make a huge difference in algorithm characteristics and performance characteristics. For the sake of convenience, we refer to this sensor model based on the uniform distribution of the four-sided center as the four-sided central model.

As shown in Figure 7, similarly, we make an infinitely extended straight line along the four sides of the robot rectangle, and divide the two-dimensional space corresponding to the space in which the robot is located into eight regions. Then, the restricted area is introduced again, and the corresponding two-dimensional space is divided into nine areas in total.

Then, four LRF groups are respectively disposed at the center of the four sides of the robot rectangle, and each LRF group is composed of two LRFs in the same linear direction parallel to the Z axis. The four LRF groups are respectively recorded as: (ll1, ll2), (lf1, lf2), (lr1, lr2), (lb1, lb2).

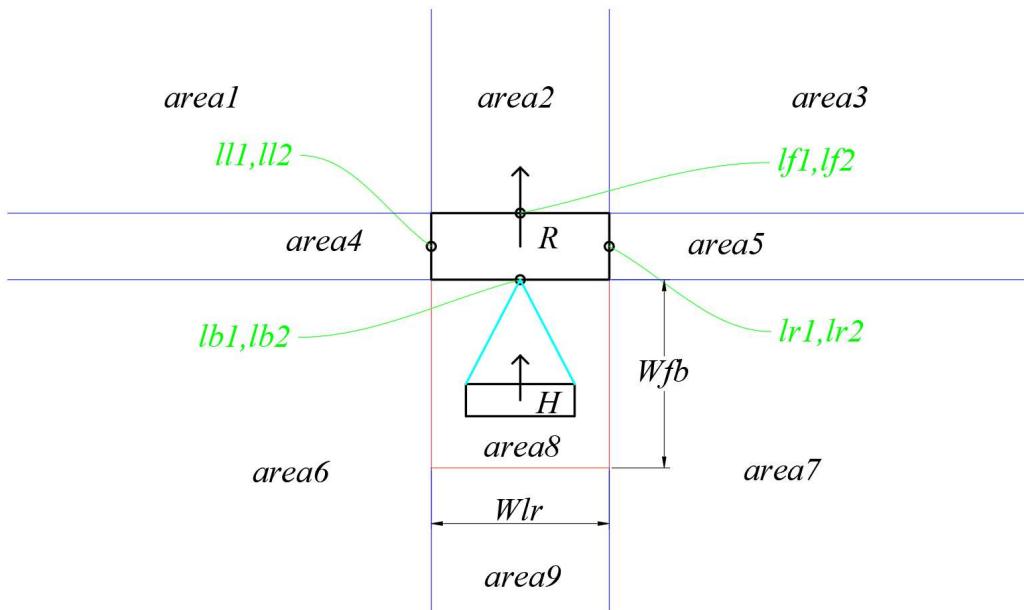


Figure 7. Robotic sensor system based on rectangular four-sided center model.

As shown in Figure 8, in terms of parameter configuration, in addition to the positional parameters of the sensor, other model parameters based on the rectangular four-sided center model are the same as the four-vertex model described above. Therefore, when defining the algorithm parameters of the sensor model in the center of the four sides, we only need to redefine the position parameters of the sensor, and the

other model parameters will remain unchanged. We set the LRF group ($Vlf1, Vlf2$) at the center point of the front edge of the virtual robot rectangle, set the LRF group ($Vlb1, Vlb2$) at the center point of the rear edge, and set the LRF group ($Vll1$) at the center point of the left edge. $Vll2$, set the LRF group ($Vlr1, Vlr2$) at the center point of its right edge. We set the LRF group ($lf1, lf2$) at the center point of the front edge of the real robot rectangle, set the LRF group ($lb1, lb2$) at the center point of the rear edge, and set the LRF group ($ll1$) at the center point of the left edge. $ll2$, set the LRF group ($lr1, lr2$) at the center point of its right edge.

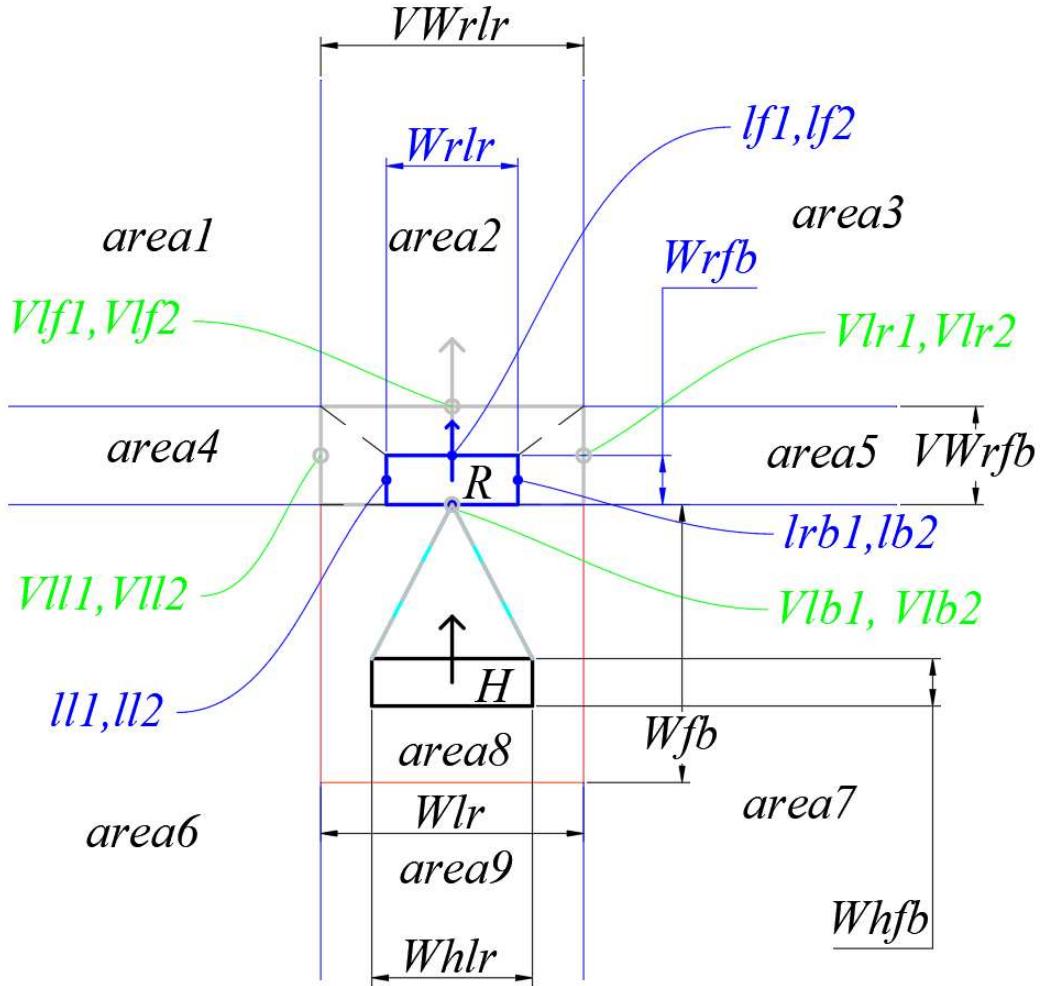


Figure 8. Parameter configuration of a sensor system based on a four-sided center model.

After defining the parameters, we also introduce a scale expansion mechanism for the four-sided central model, so that the four-sided central model's design methods and ideas can be applied to robots of different body types and structural designs.

Similarly, after scale expansion, we define the LRF groups located in the four corners of the virtual robot at the positions $Vllf, Vllb, Vlrf, Vlrb$. The front and rear width of the virtual robot is $VWrfb$, and the left and right width is $VWrlr$. It is assumed that the front and rear width of the restricted area is Wfb , and the left and right width is

W_{lr} . The front and back width of the human rectangle is W_{hfb} , and the left and right width is W_{hrl} . The front and back width of the real robot rectangle is W_{rfb} , and the left and right width is W_{rlr} . In the case of proportional expansion, the left and right width of the real robot rectangle should be less than the left and right width of the human rectangle, that is, $W_{rlr} < W_{hrl}$. The values of W_{fb} and W_{lr} are automatically set by the follow control mechanism. The corresponding setting process is as follows:

First, we need to pre-set three experimental experience values, defined as p and ξ , ε , and specify: p and ξ , ε satisfy (1).

Similarly, ξ and ε should be the constant ratios obtained after many experiments, which are suitable for the robot group of a same particular mechanical and circuit structure. Therefore, for a group of robots based on a same particular structure, the values of ξ and ε should be deterministic. However, since the W_{hrl} of the human being followed by the robot is not necessarily the same each time in following the task, the p values corresponding to the robots in the different following tasks are not necessarily the same.

In theory, $p > 1$. The optimal range of P should be $1.2 \leq p \leq 2$. K_1 is not necessarily equal to k_2 , and our focus is on k_2 . If k_2 satisfies $k_2 > 1.2$, there is no need to do the scale expansion, otherwise scale expansion is required. The process of scale expansion is as follows:

1. First, the robot calls the sensor system to scan the corresponding left and right widths W_{hrl} when the human is standing;

2. Find k_2 from (1), and find p in combination with the ξ parameter value inherent in the robot;

3. From the formula (1), combined with the inherent ε parameter value of the robot to find W_{fb} , W_{lr} ;

4. Determine the VW_{fb} and VW_{lr} by the formula (1) in combination with the known W_{rfb} and W_{rlr} and p ;

5. Set the coordinates of the geometric center of the real robot rectangle to $(0,0)$, as shown in the left figure of Figure 9, there are:

$$\begin{cases} x_{1'} = x_{3'} = 0 \\ y_{1'} = -y_{3'} = \frac{1}{2}VW_{rfb} \\ x_{2'} = -x_{4'} = x_b' = -x_a' = \frac{1}{2}VW_{rlr} \\ y_{2'} = y_{4'} = 0 \\ y_a' = y_b' = -(\frac{1}{2}VW_{fb} + W_{fb}) \end{cases} \quad (4)$$

6. After obtaining the initial coordinates of each point, move the rectangular model of the robot in the model downward by $\frac{1}{2}(VW_{rfb} - W_{rfb})$ units in the negative direction of the Y-axis, as shown in the right figure of Figure 9. The final coordinates of the point are:

$$\left\{ \begin{array}{l} x_1 = x_3 = 0 \\ x_2 = -x_4 = x_b' = -x_a' = \frac{1}{2}VWrlr \\ y_1 = \frac{1}{2}VWrfb + \frac{1}{2}(VWrfb - Wrfb) \\ y_3 = -\frac{1}{2}VWrfb + \frac{1}{2}(VWrfb - Wrfb) \\ y_2 = y_4 = 0 + \frac{1}{2}(VWrfb - Wrfb) \\ ya = yb = -\left(\frac{1}{2}VWfb + Wfb\right) + \frac{1}{2}(VWrfb - Wrfb) \end{array} \right. \quad (5)$$

After simplification, the above expressions are:

$$\left\{ \begin{array}{l} x_1 = x_3 = 0 \\ x_2 = -x_4 = x_b' = -x_a' = \frac{1}{2}VWrlr \\ y_1 = VWrfb - \frac{1}{2}Wrfb \\ y_3 = -Wrfb \\ y_2 = y_4 = \frac{1}{2}(VWrfb - Wrfb) \\ ya = yb = -\left(\frac{1}{2}Wrfb + Wfb\right) \end{array} \right. \quad (6)$$

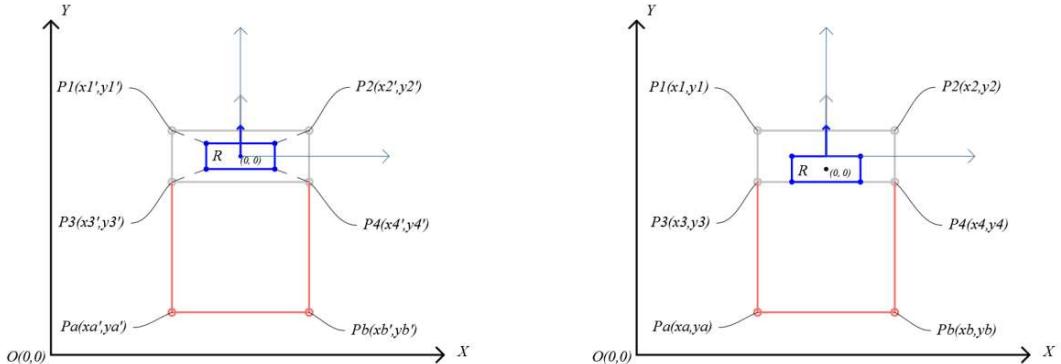


Figure 9. Proportional expansion mechanism of the four-sided central model.

Next, we will assign the corresponding detection area to the four-sided central model. As shown in Figure 10, we make an infinitely extended parallel line along the four sides of the robot rectangle, and divide the two-dimensional space corresponding to the space in which the robot rectangle is located into eight detection area sections. The eight detection area sections are labeled as area1, area2, ..., area8, respectively. The eight detection areas can be further divided into two categories, one is an overlapping detection area, and the other is a separate detection area. The overlap detection area refers to: the detection area can be detected by the central LRF group of the adjacent two sides of the robot rectangle, such as area2, area4, area6, area8 in FIG 10. The separate detection zone refers to an area that can only be detected by the LRF group at the center of one side of the robot rectangle, such as area1, area3, area5, area7 in FIG

10. For example, the overlap detection area area8 will be detected by the left and front LRF groups of the robot rectangle, and the area2 will be detected by the front and right LRF groups of the robot rectangle. The area4 is detected by the right and rear LRF groups of the robot rectangle, and the area6 is detected by the left and rear LRF groups of the robot rectangle. Area1 is detected by the LRF group in front of the robot rectangle, area3 is detected by the LRF group on the right side of the robot rectangle, area5 is detected by the LRF group behind the robot rectangle, and area7 is detected by the LRF group on the left side of the robot rectangle. (N.B.: There is a black forest illusion in Figure 10, such that the extensions of the corresponding parallel sides of the robot rectangle do not appear to be parallel, but in reality the two parallel lines are parallel).

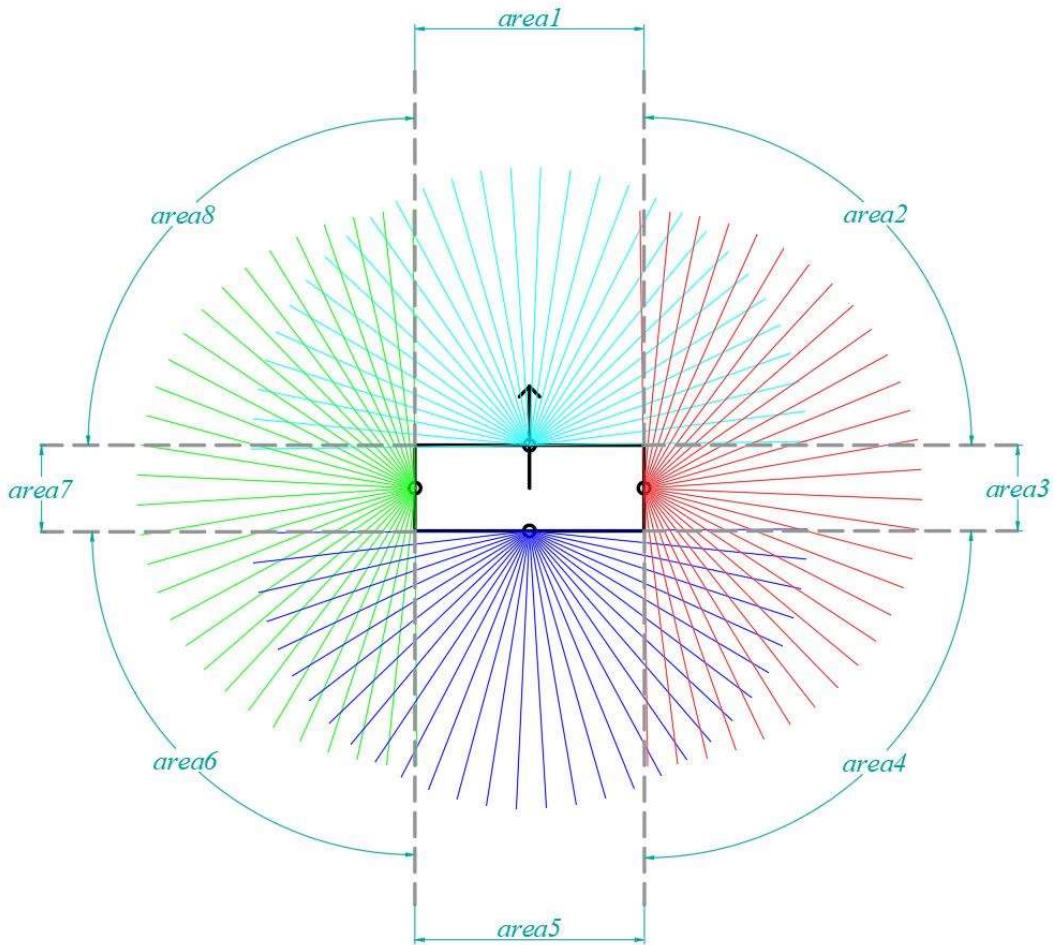


Figure 10. Configuration of the detection area of the four-sided center model.

The advantage of using a four-sided central model for sensor system configuration is that we can completely and seamlessly cover the space in which the robot is located, without the occurrence of a missing detection area. Moreover, in Front-Following, in reality, the four regions corresponding to the front, the back, the left, and the right of the robot are less likely to be disturbed by the outside than the four diagonal regions of the rectangle. In the following follow-up system algorithm design, we will mention the

importance of the detection capability of the LRF groups in the diagonal detection areas (i.e., overlapping detection areas) when the robot performs curve motion following. The overlap detection areas are detected by the dual LRF groups with respect to the individual detection areas, so the detection accuracy of the overlap detection areas is higher than that of the individual detection areas. Through the four-sided center model, we can also find that the areas of the overlapping detection areas are much larger than the areas of the individual detection areas.

For example, as shown in FIGURE 11, P3, P4, P6, and P7 of the overlap detection area are both double-detected, and their detection accuracy will be higher than other points.

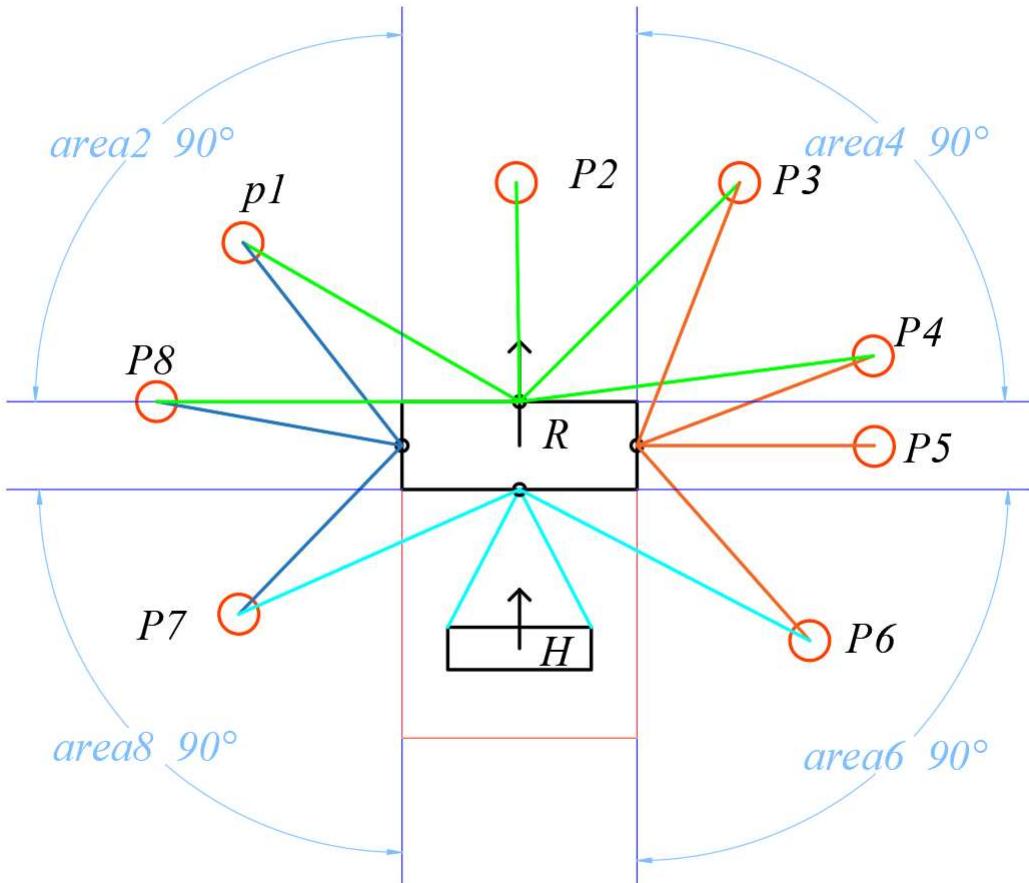


Figure 11. Schematic diagram of double detection of the four-sided center model.

Figure 11 also shows a special kind of point distribution, which is the point on the dividing line, such as P8. Whether the point on the dividing line is double-detected depends on the mechanical structure of the robot's sensor. In general, the demarcation point is also double-detected. So we also count P8 into the category of double detection points.

LRF Group

The LRF group is a new design concept that we have established. In the study, we set the horizontal ground where the robot is located to the XY plane, and the direction perpendicular to the horizontal ground to the Z-axis direction. The LRF group is a combination of two LRF detectors in the same Z-axis direction. The two LRF detectors in the LRF group work independently and do not interfere with each other. The two LRF detectors of the same LRF group are in the same straight line when viewed in the Z-axis direction.

Structural Design

As shown in Figure 12, two LRF sensors are distributed in a straight line parallel to the Z axis. In general, the hardware performance of these two LRFs is not required to be the same. Therefore, the robot designer can install different performance LRF sensors according to actual needs. In general, the LRFs in the LRF group satisfies the freedom of motion action as long as there are two degrees of freedom. One degree of freedom is used to rotate in a plane parallel to the Z axis. V_{zr} in Figure 1 represents a rotating electric machine capable of precise angle control, which corresponds to this degree of freedom. Another degree of freedom is the degree of freedom of the LRF to make a rotational motion parallel to the plane of the support bar. This degree of freedom can control the rotating electric machine to perform precise angular rotation. The support bar refers to a rotatable crossbar that supports the LRF, such as L1 and L2 in Figure 12. As shown in Figure 12, if we look in the direction parallel to the XY plane, the lights emitted by the two LRFs in the LRF group seen is in a horizontal straight line, but they are actually a plane. These two planes become the laser ray plane. This plane is made up of a number of planar areas made from the laser lights emitted from the LRFs.

The lasers emitted by our robots are all lasers with invisible frequencies, and the power of these lasers should not be too high. It should meet the premise of protecting human organs from damage, such as protecting human eyes and skin. This is also the original intention of the project, to achieve the protection of human privacy as much as possible, so that human beings do not interfere too much, have a good experience.

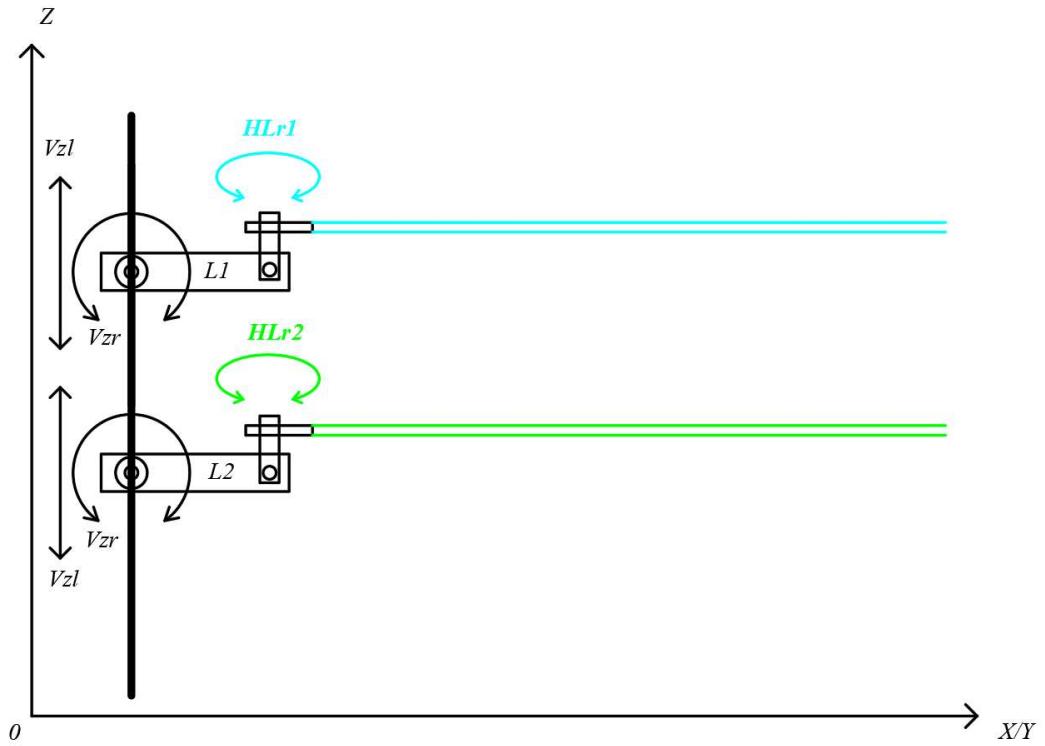


Figure 12. Schematic diagram of the structure and operation of the LRF group.

Detection Mode

Analysis

Why do we want to introduce the new concept and design idea of the LRF group? This is closely related to the detection mode of our robot's sensor system. We defined two scan modes for the LRF group. One is the normal scan mode and the other is the locking scan mode. The following algorithm and design analysis will be performed on a robot based on a sensor system with a four-sided central model.

Data Storage Mode

After scanning with the detector, the computer system of the robot needs to store

and process the acquired scan data. Here we define two storage processing methods for the robot's computer system, one is obscured scanning storage, and the other is map planning storage. If the computer system performs obscured scan storage, the scanned data will not be permanently recorded, and the scan data of the scan area other than the edge of the largest scan area will be recorded when the distance from the scan area is large to a certain extent. It will be cleared by the system. The purpose of this is to save the memory resources of the computer system of the robot and reduce the burden of data processing of the computer system. Forgotten storage is suitable for robots with uncertain working areas. Such robots often work in uncertain surroundings, so the storage of surrounding environment information does not help the robot's following actions and cruise movements. The robot system in the map planning storage mode records all the data in the scanning area within the farthest edge distance that the LRF can detect, and generates a corresponding area map by the map generation algorithm. Map-planned storage is suitable for fixed or relatively fixed robots at work. Such robots often move in the same area. So the more familiar the area is, the more the robot has sufficient data to make a more reasonable judgment.

Normal Mode

In the normal scan mode, the two LRFs in the LRF group are in the same working mode, that is, a 180-degree angle scan. When the target person is not in the detection area corresponding to the LRF group, the sensor system will automatically start the normal scan mode. As shown in Figure 13. The representative obstacle marked "O", marked as "P" represents other humans except the target person, called irrelevant human. The two LRFs of the LRF group are represented by LRF1 and LRF2, respectively. The green and sky blue rays in the figure represent the detected rays emitted by LRF1 and LRF2, respectively. Obstacle or unrelated humans may be mobile or stationary. In order to save computing resources, it is possible to perform obscured scan storage only for obstacles or unrelated humans. If the robot is working in a specific area, the robot will perform map planning storage.

In the normal mode, the two LRFs of the same LRF group are in the same working mode, so the data processing method of the double scan data superposition detection can be used to reduce the detection errors of the LRFs. The basic idea of double-scan data overlay detection is to detect the same position in the detection area by using two LRFs at different positions, and then perform average error processing on the scan results. The easiest way to handle the average errors is to find the arithmetic means of the coordinates.

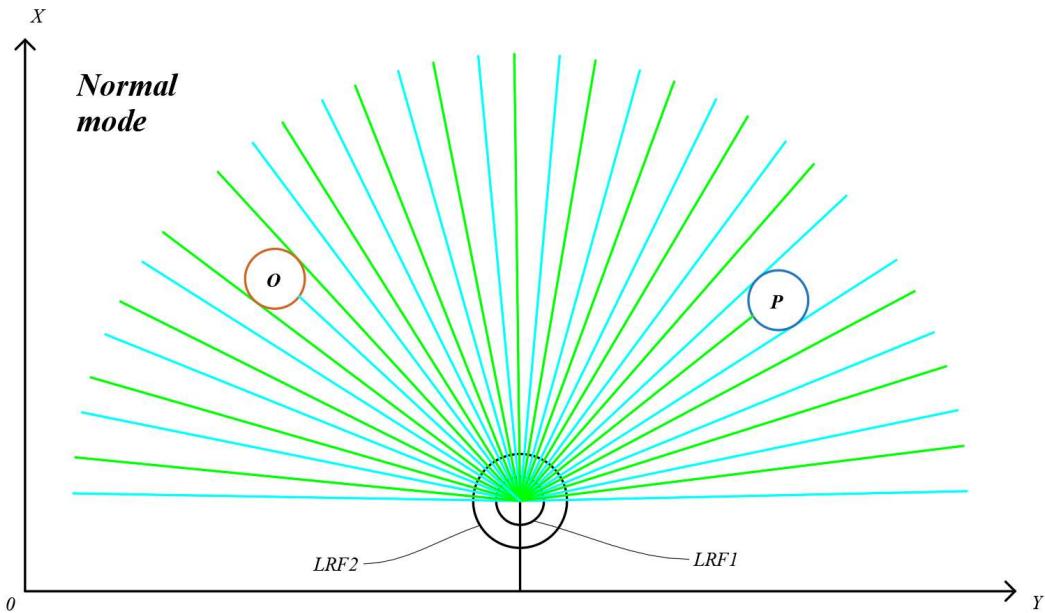


Figure 13. Normal scan mode for the LRF group.

Locking Mode

The second scan mode is the locking scan mode. In this mode, once the robot finds that the target person enters the detection area corresponding to a certain LRF group, the robot's scanning processing mechanism automatically sets the LRF group corresponding to the scanning area to enter the locking scan mode. In this mode, the algorithm mechanism will assign an LRF in the LRF group for a locked tracking scan of the target person. At the same time, the algorithmic mechanism will allocate another LRF in the LRF group for scanning the obstacles and unrelated humans. As shown in FIGURE 14, the LRFs in the locking scan mode scan only in the area corresponding to the range of the detection angle at which the target person is located and the corresponding nearby area. The basic design principle of the lock-type scanning is to perform round-trip angle scans on the scanning area including the target person and the vicinity thereof through the LRF, thereby acquiring the detection scan data information of the edge of the target person in real time. In the process of locked scanning, if the target person moves, his or her motion data information will be recorded in high speed and high precision, and stored in the corresponding database to form tracking path data information with a certain format. This data information will be used to predict the future paths of the target person. In general, normal scanning does not record the detected data information. Once the robot in the locked scan mode detects that the target person has moved in position, the scan angle range of the locked scan will be adjusted

to the corresponding angle area in time. In Figure 14, the target person is represented by a circle with an "H" mark. When the target person moves from point H to point H', the corresponding scanning angle of the target person is also changed from 10 degrees to 18 degrees. The detection and acquisition data system controls the LRF in the locked scan mode in real time to perform corresponding angle adjustment and scanning to ensure that the target person is always in the locked scan angle range during the whole scanning process. At the same time, the path information of the target person will be collected and recorded in real time. We use a straight line parallel to the Y-axis as the starting line, counterclockwise as the rotation direction, and rotate until the first boundary line of the scanning area. The angle of rotation is recorded as the deflection angle corresponding to the scanning area. Then, Θ_1 is the deflection angle when the target person is at the "H" point, and Θ_2 is the deflection angle when the target person is at the "H" point. Since the scanning angle is only the angle at which the first boundary line scanned to the object is recorded, the shape and position information of the object cannot be completely reflected. Therefore, in order to better describe and solve the problem of scanning angle range, we define the following concepts:

- 1. The first boundary angle:** the angle from the reference polar axis to the angle at which the detected light is tangent to the edge of the object being detected for the first time. It is also the angle between the detected light and the reference polar axis when the detected light first contacts the detected object, and is denoted as Ψ_a ;
- 2. The angle of the second boundary:** the angle from the reference pole axis to the angle at which the detected light is tangent to the edge of the object being detected. It is also the angle between the detected light and the reference polar axis when the last time the detected light touches the detected object, which is denoted as Ψ_b ;
- 3. Scan angle range:** the absolute value of the difference between the angle between the first boundary and the second boundary.
- 4. Scanning angle interval:** an angle interval formed by the angle between the first boundary angle and the second boundary, including an angle between the first boundary angle and the second boundary. Generally, the first boundary angle is the starting point.

In a normal round-trip rotation scan, there are generally only two boundary lines, the start boundary line and the end boundary line. Correspondingly, we call them the first dividing line and the second dividing line respectively. The width of the width of the corresponding detected portion of the detected object in a certain scanning direction can be determined by the two boundary lines. As shown in FIGURE 14, the scanning angle interval corresponding to the target person H is (Θ_1, Θ_1+10) , and the scanning angle interval corresponding to the target person H' is (Θ_2, Θ_2+18) . N.B.: The angle between the first boundary angle and the second boundary is the important parameters in the detection algorithm.

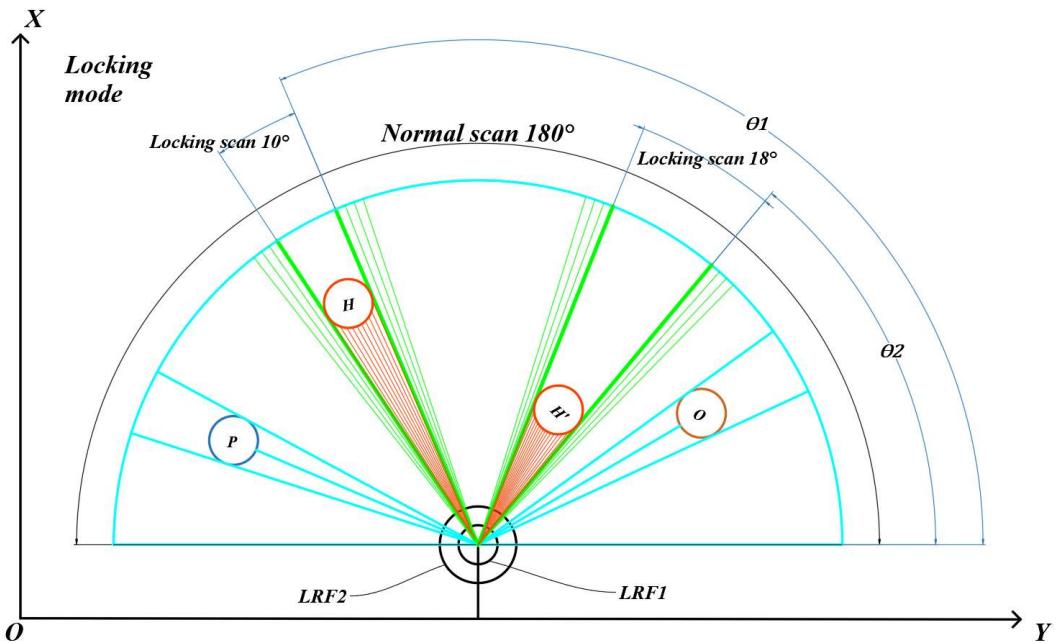


Figure 14. Schematic diagram of the locked scan mode of the LRF groups.

Combination Coordinate System

After the scan mode is determined, the next step will be to scan the detected object with the determined scan model and obtain the corresponding scan data of the detected object. To accurately scan the object being inspected and obtain the corresponding highly accurate scan data information, it is necessary to establish a scientific and rigorous coordinate system. Next, we will introduce a combined coordinate system suitable for the robot of this project. The combined coordinate system is constructed by combining Cartesian coordinate system and spherical coordinate system.

After the scan mode is determined, the next step will be to scan the detected object with the determined scan model and obtain the corresponding scan data of the detected object. To accurately scan the object being detected and obtain accurate scan data information, it is necessary to establish a scientific and rigorous coordinate system. Next, we will introduce a combined coordinate system suitable for the robot of this project. The combined coordinate system is constructed by combining Cartesian coordinate system and spherical coordinate system.

As shown in Figure 15, the pale blue GXGYGZ is a global positioning coordinate system. That is to say, GXGYGZ is determined by geomagnetic field, GPS and other global navigation systems or indoor positioning and navigation. The coordinate system does not change with the position and orientation of the robot. For GXGYGZ, which is a global positioning coordinate system, we set GX's direction to be positive north, GY's

direction to be positive east, and GZ's direction to be vertical sea level. The origin O of GXGYGZ global positioning coordinate system is the geometric center origin when the robot starts to depart after accepting the task. V(0) is the forward orientation of the robot when it starts to depart after accepting the task. Once the robot starts to perform a new follow task, the forward orientation of the robot will be set to the Y axis, and the right orientation of the robot will be set to the X axis. Similarly, the starting point of the coordinate system is the geometric center point O of the robot, so as to construct the scanning and positioning coordinate system XYZ of the robot. The origin of the XYZ scanning positioning coordinate system is set to (0, 0, 0). The origin coordinates of GXGYGZ coordinate system are set as (gx, gy, gz), where GX is the latitude value of the corresponding position points when the robot starts the task, GY is the longitude value of the corresponding position points when the robot starts the task, and GZ is the sea level height value of the corresponding position points when the robot starts the task. It can be seen that the coordinate axis Z of the XYZ coordinate system coincides with the coordinate axis GZ of the GXGYGZ coordinate system, but the coordinate axis (X, Y) of the XYZ coordinate system does not necessarily coincide with the coordinate axis (GX, GY) of the GXGYGZ coordinate system. We define the angle between the GX axis and the X axis as the counterclockwise plane angle between the GXGYGZ coordinate system and the XYZ coordinate system, expressed in Θ_2 . The angle between the GY axis and the Y axis is also defined as the counterclockwise plane angle between the coordinate system GXGYGZ and the coordinate system XYZ, which is expressed in Θ_1 . That is $\Theta_2 = \Theta_1$. Assuming that the robot is in the Pi point in space during the process of following the target person, the coordinates of the Pi point are expressed as $Pi(x_i, y_i, z_i)$ in the XYZ coordinate system. R0 is the reference axis parallel to the GX axis, and the direction is the same as that of GX. In the process of scanning, $Pi(x_i, y_i, z_i)$ is used as the origin point $Pi(r_0, \varphi_0, \theta_0)$ of the robot in the process of performing the following task, and the spherical coordinate system is established. R0 is the projection of the polar axis of the spherical coordinate system on the ZY plane, that is, the polar axis itself. Note that R0 always keeps its direction parallel to the GX axis of GXGYGZ, pointing at the same direction, while the robot follows the target person. $v(\varphi_i, \theta_i)$ is the velocity vector of the robot at Pi point. PRra and PRrb respectively detect the projection of laser ray RA and Rb on the XY plane. The anticlockwise angles of PRra and PRrb and polar axis R0 are a and b, respectively. The spherical coordinates of Pa (x_a, y_a, z_a) and Pb (x_b, y_b, z_b) in the XYZ coordinate system are $Pa(r_a, \varphi_a, \theta_a)$ and $Pb(r_b, \varphi_b, \theta_b)$, respectively.

Next we will discuss the conversion between the GXGYGZ positioning coordinate system, the XYZ following coordinate system and the $R\Phi\Theta$ scanning spherical coordinate system.

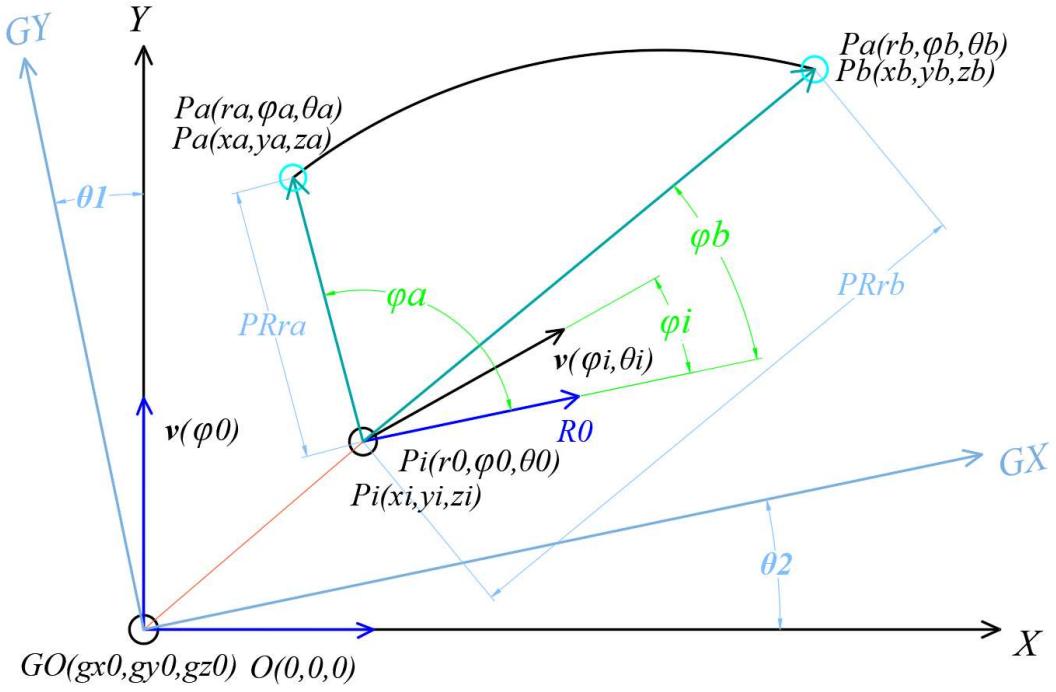


Figure 15. Design schematic of a combined coordinate system for a scanned data model.

- Set the starting point of the robot as the starting point. The GXGYGZ coordinate corresponding to the origin of origin is $(gx0, gy0, gz0)$, and $\vartheta 1 = \vartheta 2 = \vartheta g$ measured by the angle sensor.

1. For a point $Pi(xi, yi, zi)$ in the XYZ coordinate system, the coordinates of the point PGi corresponding to the GXGYGZ coordinate system are:

$$\begin{cases} gx = \sqrt{xi^2 + yi^2} \times \cos(\arccos\left(\frac{xi}{\sqrt{xi^2 + yi^2}}\right) - \theta g) + gx0 \\ gy = \sqrt{xi^2 + yi^2} \times \sin(\arccos\left(\frac{xi}{\sqrt{xi^2 + yi^2}}\right) - \theta g) + gy0 \\ gz = zi + gz0 \end{cases} \quad (7)$$

2. For the point $PGi (gx, gy, gz)$ on the GXGYGZ coordinate system, to convert to a point $Pi(xi, yi, zi)$ in the XYZ coordinate system, the corresponding coordinates are:

$$\begin{cases} xi = \sqrt{(gx - gx0)^2 + (gy - gy0)^2} \div \cos(\arcsin\left(\frac{gx}{\sqrt{gx^2 + gy^2}}\right) + \theta g) \\ yi = \sqrt{(gx - gx0)^2 + (gy - gy0)^2} \div \sin(\arcsin\left(\frac{gx}{\sqrt{gx^2 + gy^2}}\right) + \theta g) \\ zi = gz - gz0 \end{cases} \quad (8)$$

- Set the robot's geometric center as the origin when the robot first starts performing the

following task. The corresponding coordinates of the origin in the XYZ coordinate system are $(0, 0, 0)$. In the process of the robot following the target character, the coordinate point of the XYZ following coordinate system corresponding to a certain LRF in a certain LRF of the robot is $P_i(x_i, y_i, z_i)$, and some of the XYZ coordinate system space detected by the LRF The XYZ coordinate of a probe point is $P_a(x_a, y_a, z_a)$. Taking the P_i point as the origin, the P_i point is parallel to the reference polar axis R_0 of the GX axis, and the angle between R_0 and the X axis is ϑ_g . It can be seen that the angle between R_0 and the X axis is the angle between the GX axis and the X axis (i.e., ϑ_g). Taking P_i as the origin and R_0 as the reference polar axis, the $R\phi\theta$ detection spherical coordinate system is established. That is to say, the origin of the $R\phi\theta$ detection spherical coordinate system is not set by the geometric center of the robot. (IE:Therefore, there is a three-dimensional spatial translation process when transforming the coordinate system. For the specific processing and related algorithms, please refer to the analysis part of the mutual conversion between the coordinate systems of the combined coordinate system.) At this time, the laser is detected by the sensor. The zenith angle corresponding to the beam irradiation point P_a is ϑ_a , the azimuth angle is φ_a , and the radial distance is r_a (ie, the length of the laser ranging beam). N.B.: In the combined coordinate system, the GXGYGZ positioning coordinate system and the XYZ following coordinate system are determined after setting the starting point of the robot to the origin of the respective coordinate system. The origin of the coordinate system will not change in the entire follow-up task of the robot. The origin of the $R\phi\theta$ detection spherical coordinate system ($r_0=0, \varphi_0=0, \vartheta_0=0$) changes as the position of the LRF of the robot changes. I.E.: Changes with the XYZ following coordinates of the LRF of the LRF group of the robot or the change of the GXGYGZ positioning coordinates. That is to say, the $R\phi\theta$ detection spherical coordinate system is dynamically changed to have real-time performance.

- Because the GXGYGZ positioning coordinate system and the origin of the XYZ following coordinate system are set relative to the geometric center of the robot when considering the precise origin position setting, and the origin of the $R\phi\theta$ detecting spherical coordinate system is set being relative to an exit point of the detected light of an LRF in the LRF group of robot. Therefore, if the $R\phi\theta$ detection spherical coordinate system wants to convert the corresponding coordinates with the other two coordinate systems, the origin translation operation must be performed on the $R\phi\theta$ detection spherical coordinate system. It is assumed that the XYZ coordinates corresponding to the light exit point of the LRF used to establish the origin of the $R\phi\theta$ detecting spherical coordinate system are (X_l, Y_l, Z_l) . The $R\phi\theta$ coordinate corresponding to a certain detection point detected by the $R\phi\theta$ detection spherical coordinate system with the light exit point of the LRF as the origin is $(r_t, \varphi_t, \vartheta_t)$. The detection point is converted into the coordinates of the XYZ coordinate system with the light exit point of the LRF as the origin (x_{lt}, y_{lt}, z_{lt}) , and the detected point is converted into the coordinate of the XYZ coordinate system with the geometric center point of the robot as the origin (x, y, z) , then have the following relationship:

$$\begin{cases} x = x_{lt} - XL \\ y = y_{lt} - YL \\ z = z_{lt} - ZL \end{cases} \quad (10)$$

1. For a point $Pa(xa, ya, za)$ in the XYZ coordinate system, on the $R\Phi\Theta$ spherical coordinate system with the point $Pi(xi, yi, zi)$ as the origin, the point corresponding to the Pa point is the point PSa . $(ra, \varphi a, \theta a)$, the coordinates of the point PSa are:

$$\begin{cases} ra = \sqrt{(xa - XL - xi)^2 + (ya - YL - yi)^2 + (za - ZL - zi)^2} \\ \varphi a = \varphi a \\ \theta a = \theta a \end{cases} \quad (11)$$

2. Then, for the point $PSa(ra, \varphi a, \theta a)$ on the $R\Phi\Theta$ spherical coordinate system with the point $Pi(xi, yi, zi)$ as the origin, in the XYZ coordinate system with the geometric center of the robot as the origin, the coordinate point corresponding to point PSa is $Pa(xa, ya, za)$, and the coordinates of point Pa are:

$$\begin{cases} rx = ra \times \sin\theta a \times \cos\varphi a \\ ry = ra \times \sin\theta a \times \sin\varphi a \\ rz = ra \times \cos\theta a \\ xa = rx \div \cos\theta g - ry \times \sin\theta g + xi + XL \\ ya = ry \div \cos\theta g + rx \times \sin\theta g + yi + YL \\ za = rz + zi + ZL \end{cases} \quad (12)$$

- In the following principles and algorithm descriptions in this report, unless otherwise specified, both of the “XYZ coordinate system” or “XYZ following coordinate system” refer to the XYZ Cartesian coordinate system with the geometric center of the robot as the origin.
- The combined coordinate system we built consists of three coordinate combinations and is built in three dimensions. We propose to use the coordinate system in three-dimensional space instead of the two-dimensional space, mainly because, from the current technological development, our robots live in the three-dimensional world and work in the three-dimensional world. Therefore, the three-dimensional coordinate system is more suitable for the robot detection system. Of course, building a three-dimensional coordinate system is more complicated than constructing a two-dimensional coordinate system, but the three-dimensional coordinate system will be more realistic, can handle more problems, and acquire more detection data for more accurate and complex detection and control, as well as provide more complete and accurate data and information for the control or predictive models.

- Another noteworthy question is how the sensor system acquires the value of the Z axis. The Z value in the GXGYGZ positioning coordinate system is directly obtained from the height sensor detection in the robot body. The Z value of the XYZ coordinate system is algorithmically converted from the value detected by the height sensor in the robot body. Below we will focus on why we should introduce RΦΘ to detect the spherical coordinate system instead of the cylindrical coordinate system or other coordinate system when dealing with this problem.
- If the analysis is from static and dynamic angles, it can be considered that the RΦΘ detection spherical coordinate system is dynamic because its origin position coordinates are changed every moment during the execution of the robot following the character. The GXGYGZ positioning coordinate system and the XYZ following coordinate system are static, because in the process of the robot performing the following character, its origin position coordinate is unchanged after the start time is determined.
- The RΦΘ detection spherical coordinate system is the most important component of the combined coordinate system. When n detection, we introduce RΦΘ to detect the spherical coordinate system instead of other coordinate systems. The important purpose is to closely combine with the operating mechanism and structure of the sensor system we designed. We believe that the introduction of RΦΘ to detect the spherical coordinate system can make the detection system of the robot of this project reach its maximum potential. In this project, each LRF in the LRF group in the robot sensor system has at least two degrees of freedom. In general, only two degrees of freedom are required. One degree of freedom is used for rotational motion in a plane parallel to the Z-axis, and the other degree of freedom is used to rotate within the cylinder centered on the Z-axis, the points of rotation being located in a series of planar clusters parallel to the XOY plane . As shown in Figure 16, it is assumed that the point $P_{di}(r_i, \varphi_i, \vartheta_i)$ is a point detected by the LRF, and rotate1 and rotate2 respectively indicate the rotation modes corresponding to the two degrees of freedom. The origin of the XYZ coordinate system is set to LRF. At the same time, LRF is also the origin of the RΦΘ detection spherical coordinate system, and is also the emission point of the LRF detection light. By the LRF's rotary control unit, we can easily get the values of φ_i and ϑ_i by rotating the angle sensor of the unit, and it is directly obtained. And because the original function of the LRF is to measure the point-to-point linear distance, and r_i represents the length of the detected light emitted by the LRF, this measurement distance can be directly obtained. Therefore, three important parameters of r_i , φ_i and ϑ_i can be obtained directly. This greatly improves the convenience and accuracy of the test data. This is why we want to introduce RΦΘ to detect the spherical coordinate system for the final detection.

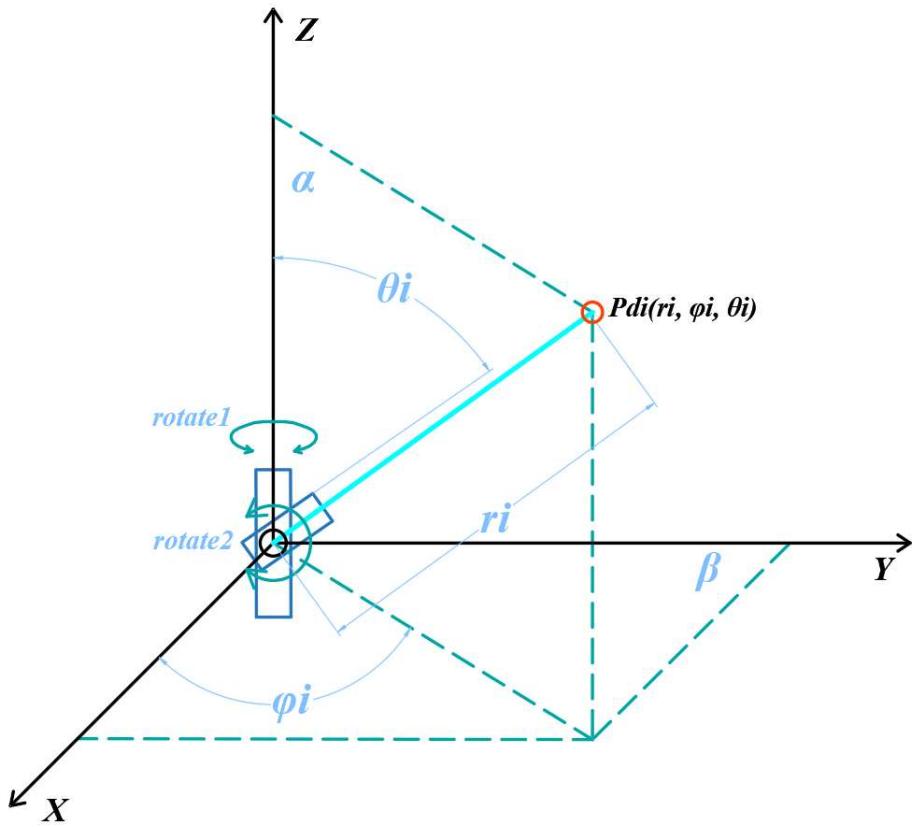


Figure 16. Relationship between RΦΘ detection spherical coordinate system and LRF detection group.

- After we knowing how to realize transforming the coordinates of a point in the GXGYGZ positioning coordinate system to the coordinates of a corresponding point in the XYZ following coordinate system, and how to realize transforming the corresponding coordinates of a point in the XYZ following coordinate system to the coordinates of a corresponding point in RΦΘ detecting the spherical coordinate system, we can establish a bridge of the transformation of the coordinates of the two corresponding points between the two coordinate systems. As shown in Figure 17, the XYZ following coordinate system will function as a link bridge in the coordinate transformation of the three coordinate systems. The solid line in Figure 17 indicates that direct conversion is possible, and the broken line indicates indirect conversion. If the coordinate points in the GXGYGZ positioning coordinate system are to be converted into the corresponding coordinate points in the RΦΘ detection spherical coordinate system, they can be first converted into the corresponding coordinate points in the XYZ following coordinate system, and then converted into the corresponding coordinate point in the RΦΘ detection spherical coordinate system. Similarly, if the coordinate points in the RΦΘ detection spherical coordinate system are to be converted into the corresponding coordinate points in the GXGYGZ positioning coordinate system, they can be first converted into the corresponding coordinate points in the XYZ following coordinate system, and then converted into the corresponding coordinate point in the GXGYGZ positioning coordinate system.

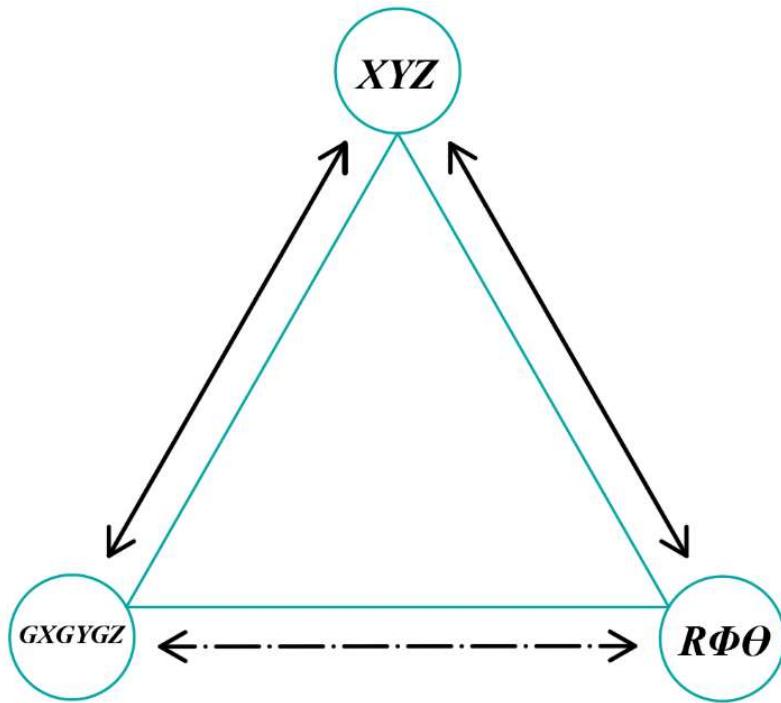


Figure 17. Mutual conversion diagram of each coordinate system of the combined coordinate system.

The derivation process in which the coordinates of the corresponding points in the combined coordinate system are mutually converted is not the focus of this project report, so it is omitted in this version of the report. A schematic diagram of the derivation calculation is given here, as shown in Figure 18, for the reader's reference.

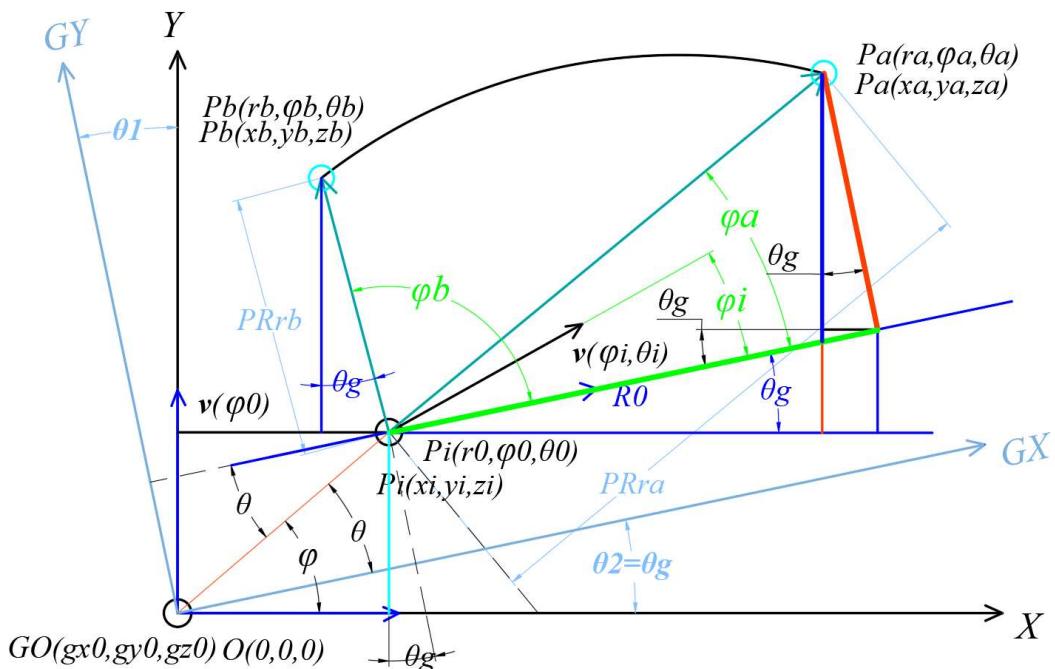


Figure 18. Schematic diagram of the derivation process of the combined coordinate system.

Detection algorithm

After determining the combined coordinate system, we can easily describe the detection algorithm. In this part, we will first carry out the description and analysis of the most basic as well as the most important algorithm of the robot of this project (I.E.: LRF specific point scanning algorithm). Then in the scan mode algorithm part, we will elaborate and analyze the algorithms of the two scan modes (I.E.: the normal scan mode and locking scan mode) of the robot of this project.

LRF specific point scanning algorithm

LRF specific point scanning algorithm, referred to as a specific point scanning algorithm. The algorithm is executed by controlling the LRF at a specific position to scan any point in the effective detection area corresponding to the LRF to be detected. Therefore, the LRF specific point scanning algorithm belongs to the control type. Since during the scanning process, our main concern is the scanning trajectory. The scan trajectory is composed of a number of track points distributed in space. Therefore, the scanning of a specific point in the space is realized, that is, the scanning of the set of specific points in the space is realized, that is, the scanning of a specific trajectory curve in the space is realized. Therefore, the LRF specific point scanning algorithm is a fundamental algorithm for the robot of this project.

Because the specific point scanning algorithm is closely related to the construction of the combined coordinate system, we arrange and analyze this part of the algorithm behind the combined coordinate system. From the previous analysis of the combined coordinate system, we know that by combining the coordinate system, we can control the LRF to scan any specific point in the effective detection space, which is essentially the LRF scan detection of a specific point in the detection area.

We divided the LRF scan into two categories for discussion:

1. *Get the coordinates of the specific point to be scanned from the system, generally the coordinates of the GXGYGZ positioning coordinate system, and then call the robot's LRF mechanical control algorithm to control the LRF to implement a specific rotary motion action to complete the scanning of a specific point or points set.*
2. *If the system does not give the coordinates of a particular point to be scanned, but gives the trajectory equation for the scan. In this case, it is common to set a scan start point and a scan end point for the LRF specific point scan algorithm. Then starting from the domain defined by the equation, set the scan interval and*

divide the domain of the scan equation. Then, the scanning of a specific trajectory is completed by a method of multi-point line formation (I.E.: a method of fitting a polyline to a curve).

In the third section of this section, we will analyze a basic space curve, which is a spatial straight line segment. Our analysis of the spatial straight line segment is based on RΦΘ detection spherical coordinate system. We propose to use the RΦΘ detection spherical coordinate system for analysis because it is more convenient and efficient for analysis. The reason and basis for the detection of the spatial straight segment detection algorithm is that any spatial curve can be approximated by a sufficient number of spatial straight segments.

Another point to note is that in the second section of this part, the calculated meta-component we set is the coordinate interval of the spatial curve. The calculated meta-component we set in the scanning algorithm of the spatial straight line segment in the third section is the scanning angle interval of the spatial straight line segment. This is because the algorithm in the second section is based on the XYZ Cartesian type coordinate system, and the third section algorithm is based on the RΦΘ detection spherical coordinate system. That is to say, the method of setting the coordinate interval is more suitable for the XYZ Cartesian type coordinate system, and the method of setting the angular interval is more suitable for the RΦΘ detection spherical coordinate system.

Let's take a detailed algorithm analysis:

I. For the case of giving the coordinates of a specific point to be scanned:

In this part of the algorithm, the algorithm in this case is the simplest. It is assumed that the mechanical and electronic control parts of the LRF are able to operate well and perform the command results of the control algorithm intact.

Input:

1. *GXGYGZ coordinates (x_{gs} , y_{gs} , z_{gs}) or XYZ coordinates (x_s , y_s , z_s) corresponding to the points to be scanned;*
2. *Real-time real-time XYZ coordinates (x_l , y_l , z_l) of the LRF used for scan detection;*

Operation:

1. Judgment: If the input is GXGYGZ coordinates, perform the coordinate transformation:

$$\begin{cases} (xt, yt, zt) = \Gamma gt(xgs, ygs, zgs) \\ (rs, \varphi s, \theta s) = \Gamma tr(xt, yt, zt) \end{cases} \quad (13)$$

2. For the current LRF rotation point the corresponding angles φs and θs .

Analysis:

Due to the scanning characteristics of the LRFs, in the RΦΘ detection sphere coordinates, the deflection angle φ and the zenith angle θ of the points in the point set on the same ray are respectively equal. So for the scanning of points on the same ray, the angles of rotation of the LRFs are respectively equal as the pointing.

3. Calculate the difference between the target deflection angle φt of the LRF and the current deflection angle φs of the LRF, and the difference is represented by $\Delta\varphi$. Calculate the difference between the target zenith angle θt and the current zenith angle θs , and use $\Delta\theta$ to represent the difference:

$$\begin{cases} \Delta\varphi = \varphi t - \varphi s \\ \Delta\theta = \theta t - \theta s \end{cases} \quad (14)$$

Judge and execute:

- If $\Delta\varphi >= 0$, control the LRF to rotate counterclockwise $|\Delta\varphi|$ angle along the direction of the deflection angle φ ;
- If $\Delta\varphi < 0$, control the LRF to rotate clockwise $|\Delta\varphi|$ angle in the opposite direction of the deflection angle φ ;
- If $\Delta\theta >= 0$, control the LRF to rotate counterclockwise $|\Delta\theta|$ angle along the direction of the zenith angle θ ;
- If $\Delta\theta < 0$, control the LRF to rotate clockwise $|\Delta\theta|$ angle along the direction of the zenith angle θ ;

Result:

The rotation direction of the LRF scanning ray changes from $(\varphi s, \theta s)$ to $(\varphi t, \theta t)$, that is, the set of scanning ray points of the LRF is composed of rays (origin = (xl, yl, zl) , direction = $(\varphi s, \theta s)$) changes to the ray (origin = (xl, yl, zl) , direction = $(\varphi t, \theta t)$).

At this point, we can use this algorithm above to control the LRF to detect and scan a certain point or points set in the detection area corresponding to the LRF.

For the LRF at a certain point, we define the rotational orientation of the LRF using

the set of deflection angles and zenith angles (φ, θ) of the LRF position in the $R\Phi\Theta$ detection sphere coordinates. By analyzing the above algorithm, we know that due to the characteristics of the scanning ray of the LRF, for the LRF at a certain position, if the rotation of the LRF is directed to (φ, θ) , then the points that the LRF can detect are in the same ray on the same direction (φ, θ) , and we call this set of points a collection of ray points. The most obvious feature of the ray point set is that the LRF scan of the point to be measured near the LRF origin masks the LRF scan of the point to be measured that is relatively far from the LRF origin. That is to say, in the same set of ray points, only one point to be measured is scanned, and the point to be detected that is scanned is the position closest to the origin of the LRF probe beam emission. This feature of the collection of ray points will continue to be discussed in the third section of this section.

2. For the case where the system does not give the coordinates of the specific points to be scanned:

This situation is a little more complicated than the above one. Let the general equation of a certain space curve L corresponding to the XYZ following coordinate system be:

$$\begin{cases} F(x, y, z) = 0 \\ G(x, y, z) = 0 \end{cases} \quad (15)$$

Where $F(x, y, z)$ represents a curved surface A , and $G(x, y, z)$ represents another curved surface B , so the spatial curve L is the intersection of the spatial curved surface A and the spatial curved surface B .

That is, the domain of definition of the spatial curve L is:

$$\begin{cases} x \in [Sx, Tx] \\ y \in [Sy, Ty] \\ z \in [Sz, Tz] \end{cases} \quad (16)$$

Because the distribution of spatial curves in three-dimensional space is continuous, the specific point scanning algorithm of LRF is to control the scanning angle and steering of the corresponding LRF according to the specific set of points. So we used a method of segmenting the continuous curve to divide the domain of the spatial curve to be scanned. The strategy for implementing the fixed-number division of the three-coordinate domain of any spatial curve L is as follows:

1. Take mx points in the middle distance of the X -defined field $[Sx, Tx]$, that is, divide $[Sx, Tx]$ into $mx+1$ segments on average. The distance of each segment is equal, it is:

$$dx = \frac{|Tx-S|}{mx+1} \quad (17)$$

That is, the X coordinate of the point of the i-th X-defined domain is:

$$xi = Sx + (i - 1) \times dx \quad (18)$$

2. Take the my point in the middle of the Y-defined field [Sy, Ty], that is, divide [Sy, Ty] into my+1 segments. The distance between each segment is equal, which is:

$$dy = \frac{|Ty-Sy|}{my+1} \quad (19)$$

That is, the Y coordinate of the point of the i-th X-defined domain is:

$$yi = Sy + (i - 1) \times dy \quad (20)$$

3. Take mz points in the middle distance of the Z-defined domain [Sz, Tz], that is, divide [Sz, Tz] into mz+1 segments on average. The distance between each segment is equal, which is:

$$dz = \frac{|Tz-S|}{mz+1} \quad (21)$$

That is, the Z coordinate of the point of the i-th X-defined domain is:

$$zi = Sz + (i - 1) \times dz \quad (22)$$

Note that when dividing the X, Y, and Z domain of definition of the interval curve, it is only necessary to divide the two definition domains of the three definition domains corresponding to the X, Y, and Z coordinates, and it is not necessary and cannot directly divide the three definition domains synchronously. Because if the three coordinate quantities of X, Y, and Z are directly divided, the most likely result will be that the corresponding (x, y, z) coordinates cannot correctly correspond to the points on the spatial curve. So, we should avoid this division. Under normal circumstances, we must only need to divide the defined interval corresponding to the two coordinate components in the XYZ coordinate definition domain, and then substitute the corresponding coordinates of the corresponding two coordinate components into the curve equation to uniquely determine the third corresponding coordinate of the coordinate components. Then, the three coordinates (x, y, z) thus obtained are the three

coordinates of the point above the spatial curve L.

In the real-time execution of the algorithm, we let $mx = my = mz = mk$, so that the obtained number of X coordinate points is the same as the number of Y coordinate points. Then, the general equation of the space curve is transformed into the equation of Z for X, Y:

$$z = f(x, y) \quad (23)$$

Corresponding to $(x1, y1), \dots, (x(k+1), y(k+1))$ into the equation $f(x, y)$, the corresponding Z-axis coordinates are obtained:

$$z1, z2, \dots, z(k+1) \quad (24)$$

Thus, the coordinates of each segment point can be completely obtained:

$$(x1, y1, z1), (x2, y2, z2), \dots, (x(k+1), y(k+1), z(k+1)) \quad (25)$$

Then convert the above XYZ coordinates into corresponding $R\Phi\Theta$ detection sphere coordinates:

$$\left\{ \begin{array}{l} (r1, \varphi1, \theta1) = \Gamma tr(x1, y1, z1) \\ (r2, \varphi2, \theta2) = \Gamma tr(x2, y2, z2) \\ \dots \\ (ri, \varphi_i, \theta_i) = \Gamma tr(xi, yi, zi) \\ \dots \\ (r(k+1), \varphi(k+1), \theta(k+1)) = \Gamma tr(x(k+1), y(k+1), z(k+1)) \end{array} \right. \quad (26)$$

Then, $(r1, \varphi1, \theta1), \dots, (r(k+1), \varphi(k+1), \theta(k+1))$ are sequentially input into the scan control algorithm of a specific point, so that the LRFs sequentially scan the space points. Mobile scanning detection is performed at each specific point on the curve by the LRFs. So in fact, this scanning method is also to detect and scan the set of points extracted from the segmentations of the space curve.

Therefore, in fact, the scanning algorithm that our designed LRF scans for the spatial curve is essentially a segmentation process of the spatial curve, and the smooth curve of the space is converted into a non-smooth polyline for processing. So why do we want to segment the spatial curve into multi-segment curves for scanning? There are two main reasons for this:

The first reason is that even though we designed the algorithm to make the LRF scan simulate a smooth spatial curve and get a lot of continuous detection points, we don't need so much consecutive points for data extraction and analysis. Because the

coordinate data of the adjacent detection points are very similar under the high-speed scanning of the LRF, and there is no obvious specificity, the coordinate data of these points contribute little to the correct and accurate processing result of the algorithm. In addition, excessive extraction of adjacent similarities can also result in wasted computing resources for the robot.

The second reason is that because of the design and construction of LRF, even if we use continuous moving LRF detection light to scan a detected object continuously, the final actual result will be that when the robot detection mechanism extracts the LRF detected data, it also extracts the data in the form of data segments. Because continuous curves are made up of countless points, in theory, it is impossible for us to store the detection information of countless points. According to the current theory of physics, even between two very close points, it can still be divided into countless points. Therefore, it is necessary to extract data points by using piecewise data extraction method. So, what we usually say is that the continuous detection of a detected object, in fact, the detection light has actually scanned countless detected points, but because of the abstract form of points, the final data we get can only be segmented. But when the distance between segments is very small, we think that the curve formed by these points is continuous.

3. The most common scanning mode: straight line scanning between two coordinate points

In this section, we will introduce the most common and basic scanning method in the LRF fixed-point scanning algorithm. This is a specific point scan along a specific straight line segment, also known as a straight line scan between two coordinate points. Because of the LRF scan function design of the robot version described in the research report, we mainly use straight-line segment scanning or round-trip folding multi-segment scanning. Since the folding round-trip multi-segment scan can be split into multiple linear segment scans, the specific point scan of the straight line segment is the basis and foundation of the LRF specific point scan algorithm. So we specifically take this part of the algorithm out for analysis. Since the simplest way to determine a finite straight line segment is determined by the two endpoints of the straight line segment, we will construct the algorithm by determining the straight line segment from two points. The specific algorithm flow is as follows:

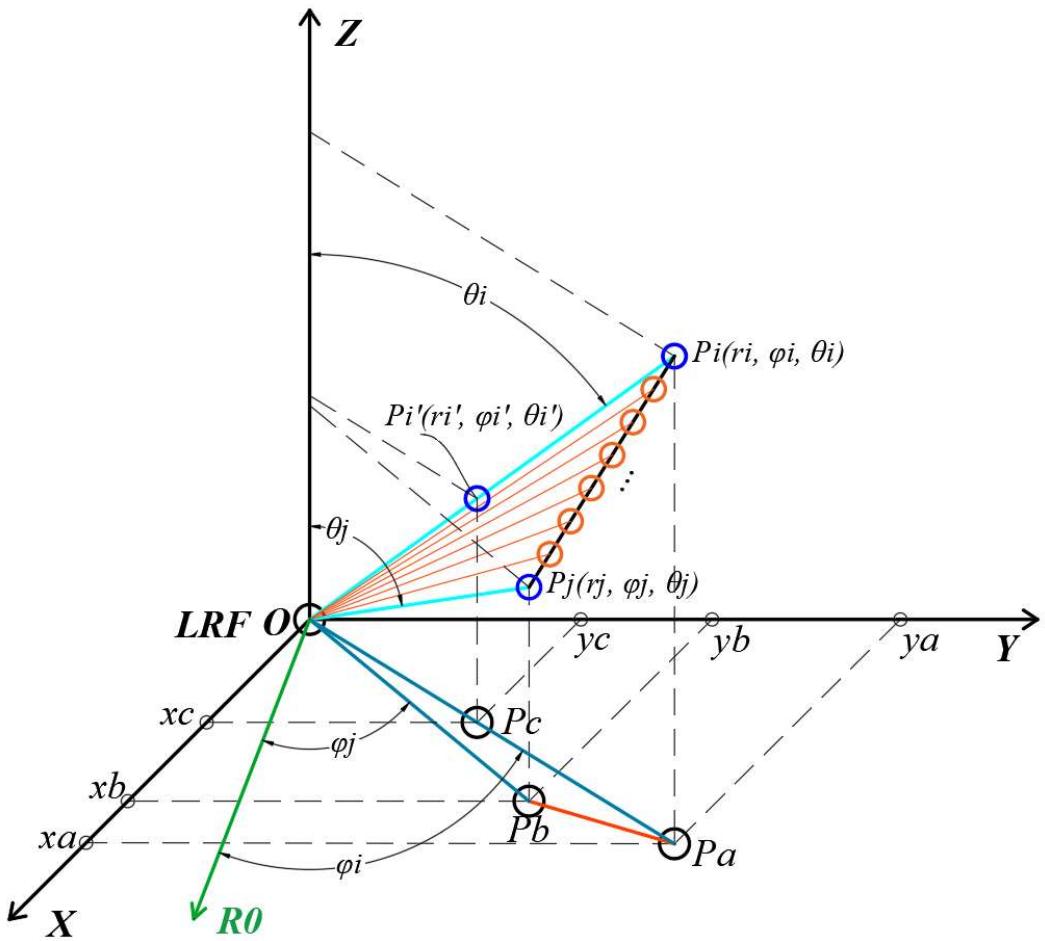


Figure 19. Straight line scan between two coordinate points of space. Modeled by introducing an XYZ reference coordinate system.

1. As shown in Figure 19, we set the position point where the LRF is located as the origin O, and thus establish the RΦΘ detection sphere coordinates. At the same time, an XYZ reference coordinate system is established. It should be noted that the XYZ reference coordinate system in FIGURE 19 is not established with the robot taking the position point at the start of following the character task. The three-coordinate components between the XYZ reference coordinate system and the XYZ following coordinate system are represented by the same symbol, but they belong to two different coordinate systems. The purpose of establishing the XYZ reference coordinate system at the location where the LRF is located is to better illustrate the straight line segmentation algorithm between the two coordinate points. When setting the XYZ reference coordinate system, ensure that the three coordinate axes of the XYZ reference coordinate system are parallel to the three coordinate axes of the XYZ following coordinate system. That is to say, the XYZ reference coordinate system is equivalent to a coordinate system obtained by shifting the XYZ following coordinate system from the origin to the position where the LRF is located. The following is the specific modeling process:

It is assumed that there are two points in the effective detection area space

corresponding to the LRF: points $P_i(r_i, \varphi_i, \theta_i)$ and points $P_j(r_j, \varphi_j, \theta_j)$. $R\Phi\Theta$ to detect the deflection angle polar axis of the spherical coordinate system. Point P_a and point P_b are projections of point P_i and point P_j on the XY plane, respectively. The straight line segment OP_i and the straight line segment OP_j represent the detected light rays emitted by the LRF to the point P_i and the point P_j , respectively. The point $P'_i(r'_i, \varphi'_i, \theta'_i)$ is a point on the straight line segment OP_i . Point P_c is the projection of point P'_i on the XY plane. The projection of the straight line segment P_iP_j in space on the XY plane is P_aP_b . The three coordinates of P_a , P_b , and P_c are: $(x_a, y_a, 0)$, $(x_b, y_b, 0)$, $(x_c, y_c, 0)$.

Suppose the space segment we want to scan is P_iP_j . Through analysis, we find that the points on the same line of the probe light have the same angle of φ . If the end of a straight line segment near the origin O is extended, the straight line segment of the space can pass through the O point, then the straight line segment of the space is called the straight line segment of the origin emission. The straight line segment of the origin emission must coincide with a detected light emitted from a point O. Therefore, the angles of φ of all the points on the straight line segment of the origin are the same. That is to say, if a straight line segment of a space is a straight line segment of the origin, when the LRF scans the straight line segment, the control of the φ angle of the LRF does not need to be changed.

2. In order to more vividly describe the fixed-point scanning algorithm for the straight line segment between two coordinate points in space, we use the top view angle along the Z-axis negative direction for analysis. As shown in Figure 20, the projection of the space to be scanned straight line P_iP_j and the scanning ray segment for scanning the P_iP_j straight line segment on the XY plane is shown.

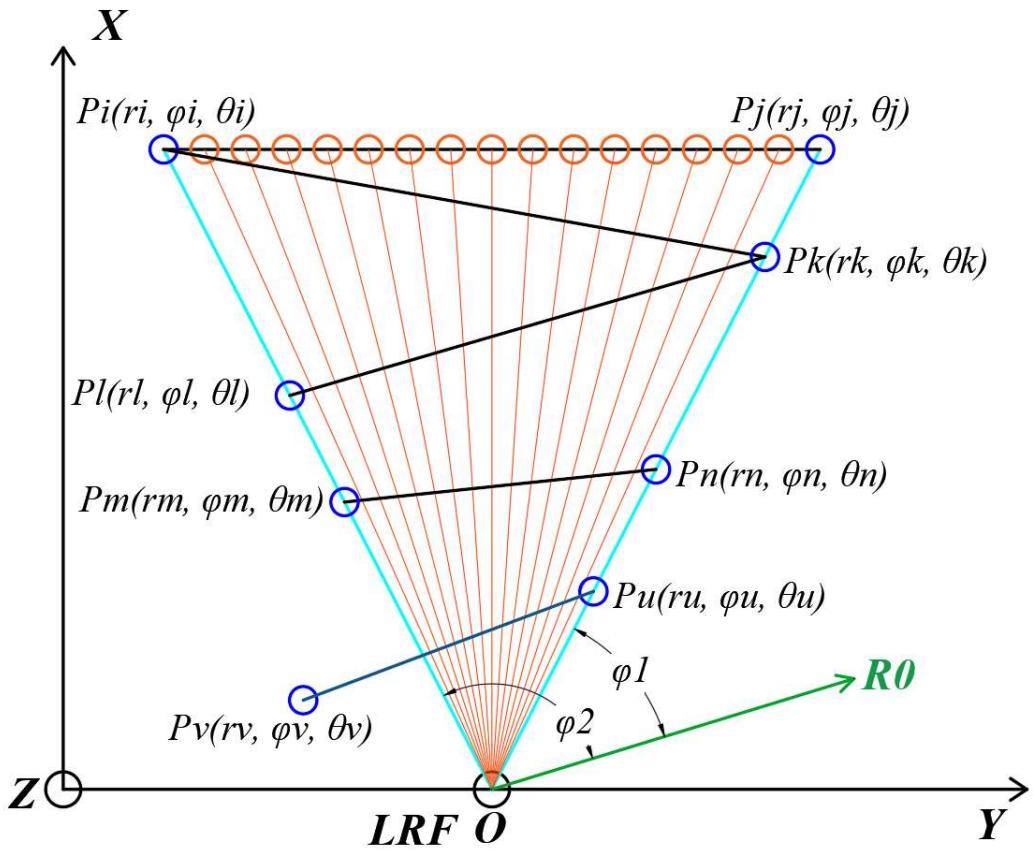


Figure 20. Straight line scan between two coordinate points of space. The projection view is viewed from the Z-axis direction.

At the same time, we define the following concepts:

Scanning triangle:

A triangular planar area formed by the origin O of the LRF and the two extreme points P₁ and P₂ of the straight line segment P₁P₂ in the space to be scanned. Define the O point as the vertex of the scanned triangle. The straight line segment OP₁ and the straight line segment OP₂ are correspondingly referred to as the left and right sides of the scanning triangle according to the left and right positions where they are located. P₁P₂ is called the scanning edge of the scanned triangle.

As shown in Figure 19, the scanning triangle corresponding to the spatial straight line segment P_iP_j is the triangle OP_iP_j. The left side of the scanning triangle OP_iP_j is OP_i, the right side is OP_j, and the scanning side is P_iP_j.

Triangle scan angle:

When projecting in a plan view in the negative direction of the Z axis, the angle formed by the projection of the right side and the left side of the scanning triangle on the XY plane is called the triangle scanning angle. As shown in FIGURE 20, with R₀

as a reference axis, the projection angles of the left and right sides of the scanning triangle P_iP_jO on the XY plane are φ_2 and φ_1 , respectively. Then the triangle scanning angle of the scanning triangle P_iP_jO is

$$\varphi_{tr} = \varphi_2 - \varphi_1 \quad (27)$$

Combining with Figure 20 for analysis, we get the following conclusions:

1. *The angles of φ on the same scanning ray are the same;*
2. *Each scanning triangle corresponds to a scanning flow mode of the LRF, which corresponds to the implementation of an LRF scanning control algorithm.*
3. *Line segments that are within the same scan triangle can be scanned by the scan mode corresponding to the scan triangle.*
4. *If the end points of a straight line segment of a space are respectively located on the left and right sides of the scanning triangle, the spatial straight line segment can be completely scanned by the LRF scanning process represented by the scanning triangle. And scanning of a straight line segment relatively close to the O point will block the scanning of a straight line segment relatively far from the O point.*
5. *If a part of a linear segment of a space is in the area of the scanned triangle and the other part is outside the area of the scanned triangle, the line segment cannot be completely scanned. As shown as P_uP_v in Figure 20.*
6. *Since there are infinite number of straight line segments as the end points on the left and right sides of the scanning triangle, the LRF scanning process represented by the scanning triangle is also suitable for infinite number of straight line segments as the end points on the left and right sides of the scanning triangle.*

Next, we will use the straight-line segment side-projection method to analyze whether the scanning algorithm can decompose the LRF scan of the spatial straight line segment to different coordinate components of the coordinate system when the LRF scans a specific spatial straight line segment. As shown in Figure 21, we project the spatial scanning straight line segment P_iP_j to the XOY plane, the YOZ plane and the XOZ plane, respectively. P_iP_j is the straight line segment to be tested in space, P_i1P_j1 is the projection of the straight line segment to be measured on the XOY plane, P_i2P_j2 is the projection of the straight line segment to be measured on the YOZ plane, and P_i3P_j3 is the projection of the straight line segment to be tested on the XOZ plane. We found that in the XYZ coordinate system, the projection of the spatial straight line segment on the three coordinate axes is still a straight line segment, and the relevant projection points all correspond one-to-one. Therefore, the decomposition projection of the straight line segment on the XYZ coordinate system is feasible and uniquely determined. The GXGYGZ positioning coordinate system and the XYZ following coordinate system in the combined coordinate system are both based on Cartesian coordinate system. These properties of the two coordinate systems have high

approximation, and the decompositions of the three-coordinate component of the spatial straight line segment corresponding principle of projection are also applicable both in the GXGYGZ positioning coordinate system and the XYZ following coordinate system.

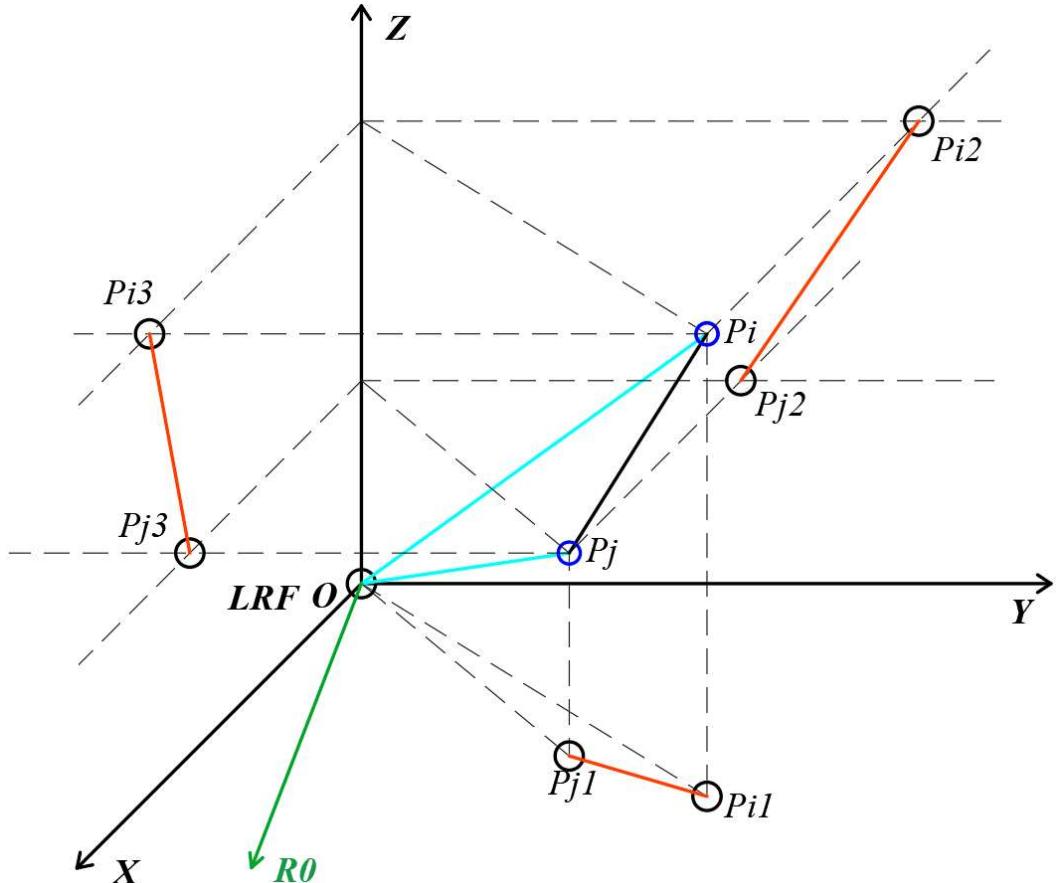


Figure 21. Projection of a straight line segment between two coordinate points of space in three sides of the XYZ coordinate system.

Since the decomposition projection of the spatial straight line segment is applicable to both the GXGYGZ positioning coordinate system and the XYZ following coordinate system, the next obvious problem to be considered is whether the decomposition projection of the spatial straight line segment is applicable in the RΦΘ detection spherical coordinate system. To explore this problem, we designed an analysis process as shown in Figure 22. R0 is the yaw angle reference axis of the RΦΘ detection spherical coordinate system. Point O represents the position of the LRF (I.E.: the emission point of the laser detection light). Point R0 coincides with the origin of the XYZ reference coordinate system. We cross the Z axis and the R0 axis at point O, making a plane ZOR0.

Definition: Plane ZOR0 is the zenith angle rotation projection plane of the RΦΘ detection spherical coordinate system. Definition: Plane XOY is RΦΘ to detect the deflection angle of the spherical coordinate system and to rotate the projection plane.

$PiPj$ is the straight line segment to be detected in space. Point Pk is a point on the straight line segment $PiPj$. The straight line segment $Pi1Pj1$ is a projection of the straight line segment $PiPj$ on the XOY plane. The straight line segment $PirPjr$ is the projection of the straight line segment $PiPj$ on the ZOR₀ plane. Point $Pk1$ is the projection of point Pk on the XOY plane. Point Pkr is the projection of point Pk on the ZOR₀ plane.

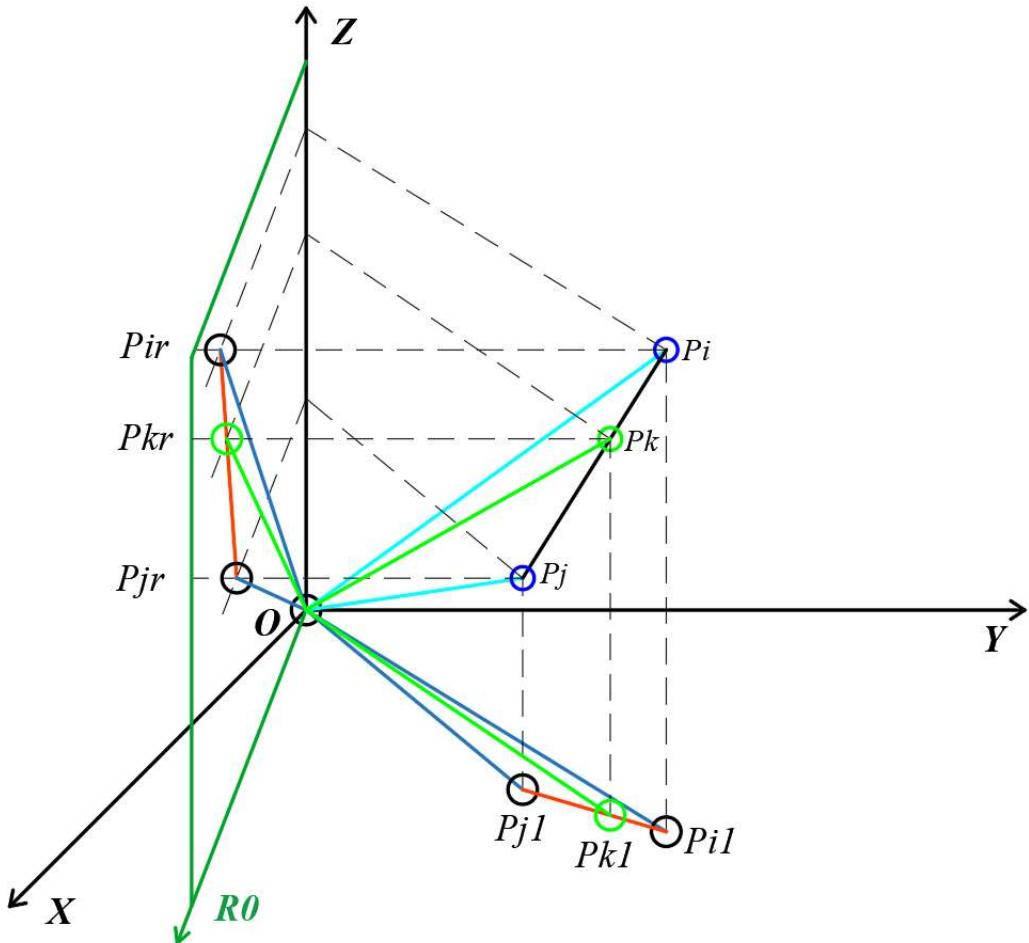


Figure 22. Projection of straight line segments between two coordinate points in space on both sides of the $R\Phi\Theta$ detection spherical coordinate system.

Because the characteristics of $R\Phi\Theta$ detection spherical coordinate system, to completely represent the data information of the spatial straight line segment, we respectively project the control straight line segments to the plane of zenith angle rotation projection and the plane of deflection angle rotation projection in the $R\Phi\Theta$ detection spherical coordinate system. After analysis, we found that the straight line segment exists in the form of straight segments in both planes of the zenith angle rotation projection plane and the deflection angle rotation projection plane. And because the two planes of the projection plane can be rotated by rotating the projection plane and the deflection angle by the zenith angle, the information of the complete

spatial straight line segment can be reflected. Therefore, we can decompose the LRF detection and scanning motion of the spatial straight line segment on two planes (I.E.: zenith angle rotation projection plane and deflection angle rotation projection plane).

Next, we will analyze the relationship between the deflection angle φ of the LRF and the zenith angle θ and the corresponding relationship between the LRF corresponding yaw angle φ and the zenith angle θ after the LRF detects the scanning motion of the spatial straight line segment. This analysis will be explained in conjunction with FIGURE 22 and FIGURE 23.

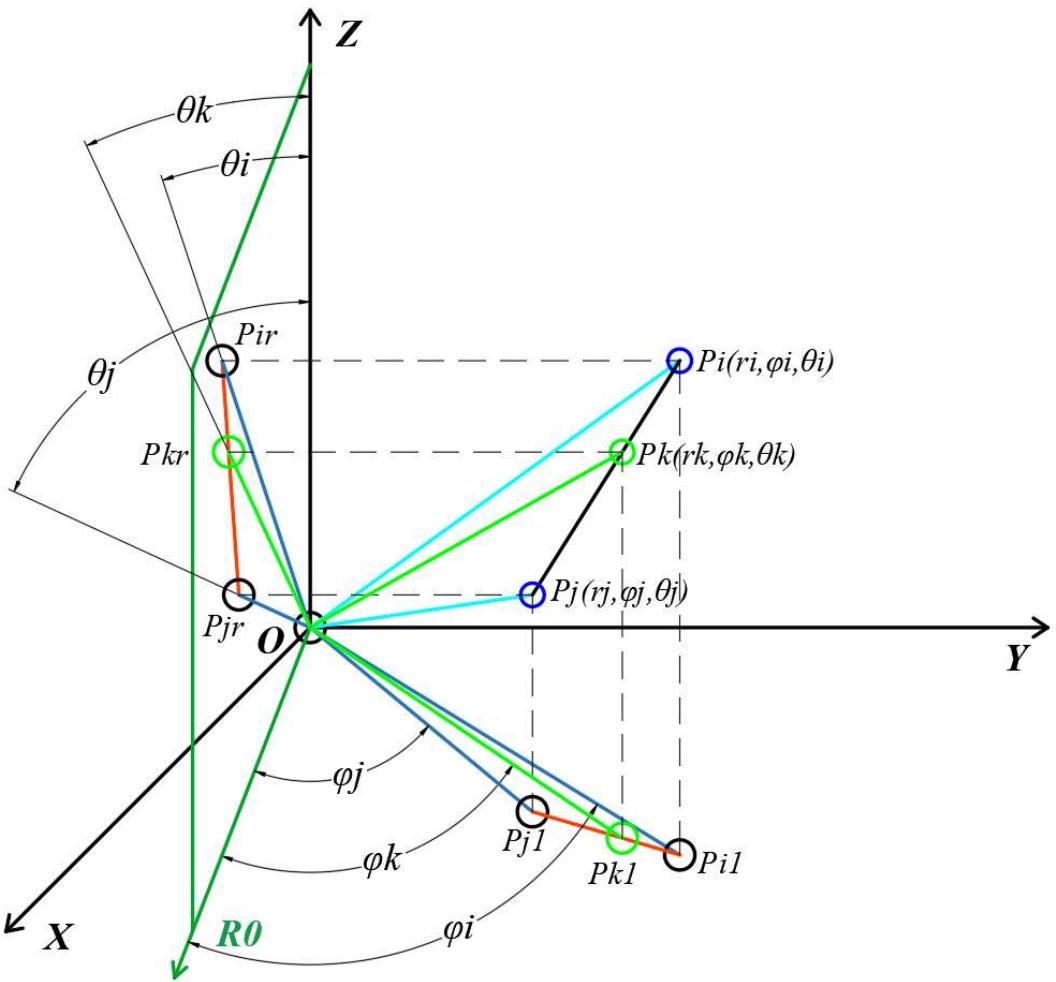


Figure 23. Scanning projection analysis of the celestial angle rotation projection plane and the deflection angle rotation projection plane of the spatial straight line segment in the $R\Phi\Theta$ detection spherical coordinate system.

First, for the sake of convenience, we define the following concept: The LRF scan of a spatial straight line segment is decomposed into two sub-scanning motions. The sub-scanning motion that is decomposed into the zenith angle rotation projection plane is called the zenith angle scan, and the sub-scan motion that is decomposed into the yaw angle rotation projection plane is called the yaw angle scan. From the properties of the

projection surface and the one-to-one correspondence of the coordinates, we know that the properties of the motion process of the zenith angle scan are the same as those of the motion process of the deflection angle scan. That is, if the zenith angle scan is a scan of uniform angular velocity, then correspondingly, the scan of the deflection angle is also a scan of uniform angular velocity. If the zenith angle scan is a scan of the uniform scan path line speed, then correspondingly, the yaw angle scan is also a scan of the uniform scan path line speed. (NB: The scan path line speed here refers to the path size of the track path scanned by the LRF scan light per unit time, and does not refer to the scan line speed relative to the scan angular velocity. We define the purpose of the scan path line speed. It is to more vividly represent the amount of change in the path of the scan path during the scanning process.) The basic principle of the above definition is to satisfy the one-to-one correspondence of the coordinates at the same motion time. The properties of the zenith angle scan and the yaw angle scan are the same, but their motion transform or rate of change is generally different. These differences are manifested in the fact that the LRF scan angular velocity or scan path linear velocity is different when the robot is moving.

After knowing the relationship between the zenith angle scan and the deflection angle scan after the LRF scans the spatial straight line segment, we can reduce the dimension of the three-dimensional scanning process of the spatial straight line segment by the LRF. That is, the scanning motion in the original LRF three-dimensional space is decomposed into scanning sub-movements in two two-dimensional spaces. Here, the design idea of the synthesis and decomposition of motion is adopted.

One question worth pondering is, why do we want to decompose the LRF scan of the linear motion segment of the three-dimensional space on the R $\Phi\Theta$ detection sphere coordinates instead of the XYZ-type Cartesian coordinate system? Another problem worth considering is that if we want to decompose the LRF scan motion of the straight line segment of the three-dimensional space on the R $\Phi\Theta$ detection spherical coordinate system, why do we only decompose the Φ component and the Θ component without decomposing the R component? The reason why the LRF scanning motion decomposed in the R $\Phi\Theta$ detection spherical coordinate system instead of the XYZ-type Cartesian coordinate system is that the R $\Phi\Theta$ detection spherical coordinate system can better reflect the detection characteristics of the LRF and the mechanical motion control. The reason why we did not decompose the LRF scanning motion on the R component is that in the current laser technology background, the R component is uncontrollable by humans in the LRF scan, and only the LRF scanning light does detect the detected object (I.E.: Objects can block that detecting light.), LRF will return the value of the R component. Therefore, the R component cannot be actively controlled by humans, and the R component is a value with a passive amount (I.E.: We are not yet able to freely control the length of the laser beam of the LRF). Another important reason is that the spherical coordinate system does not have a reference axis of the R component, so we do not consider decomposing the scanning motion of the LRF onto the R component. By decomposing the scanning motion of the LRF onto the Φ component and the Θ component, the LRF motion process can be completely reflected, and the LRF rotational motion process is an important research topic of our concern.

LRF scan algorithm for any straight line segment in space:

I. Algorithm design analysis:

In this part, we will analyze the fixed-point scanning algorithm of any straight line segment in space based on R Φ Θ detection spherical coordinate system and XYZ reference coordinate system:

Assume that the straight line segment of the space to be detected is PiPj. Point Pk is any point on the straight line segment PiPj to be tested. The coordinates of the point Pi are Pi(ri, φ_i , θ_i), the coordinates of the point Pj are Pj(ri, φ_j , θ_j), and the coordinates of the point Pk are Pk(rk, φ_k , θ_k). The projected straight line segment of the PiPj on the deflection angle rotation projection plane R0OY is Pi1Pj1. The projected straight line segment of PiPj on the zenith angle rotation projection plane R0OZ is PirPjr. Point Pk is any point on the straight line segment PiPj to be detected. The projection point of the point Pk on the zenith angle rotation projection plane R0OZ is Pkr, and the projection point on the deflection angle rotation projection plane R0OY is Pk1.

2. Define the following parameters:

OPi1:

A projected straight line segment of the spatial straight line segment OPi on the R0OY plane;

OPj1:

A projected straight line segment of the spatial straight line segment OPj on the R0OY plane;

OPk1:

A projected straight line segment of the spatial straight line segment OPk on the R0OY plane;

OPir:

A projected straight line segment of the spatial straight line segment OPi on the R0OZ plane;

OPjr:

A projected straight line segment of the spatial straight line segment OPj on the R0OZ plane;

OPKr:

A projected straight line segment of the spatial straight line segment OPk on the R0OZ

plane;

$\Omega\varphi$:

Scan angular velocity of LRF on the R0OY plane. Corresponding to the deflection angle scanning angular velocity in the mechanical and electronic control algorithms of the LRF.

$\Delta\varphi$:

The amount of angular change in the deflection angle of the scanning ray when the LRF is scanned on the R0OY plane.

$\omega\Theta$:

The angular velocity of the LRF on the R0OZ plane. Corresponding to the angular velocity of the zenith angle in the mechanical and electronic control algorithms of the LRF.

$\Delta\Theta$:

The amount of change in the angle of the zenith angle corresponding to the scanning light when the LRF is scanned on the R0OZ plane.

In order to facilitate the construction of the scanning algorithm, we define that both $\omega\varphi$ and $\omega\Theta$ are constant value. That is to say, the zenith angle scanning rotational motion of the LRF on the R0OZ plane and the yaw angle scanning rotational motion on the R0OY plane are uniform angular velocity motions.

3. Algorithm execution flow:

Input:

1. The coordinates of the two extreme points of the straight line segment to be detected in the space:

$$P_i = (r_i, \varphi_i, \theta_i), \quad P_j = (r_j, \varphi_j, \theta_j) \quad (28)$$

2. The distance from the LRF to the ends of the straight line segment of the space to be tested:

$$|OP_i| = r_i, \quad |OP_j| = r_j \quad (29)$$

3. The coordinates of the LRF currently scanning to the detection point:

$$P_k = (r_k, \varphi_k, \theta_k) \quad (30)$$

Calculation: Coordinates of each relevant point:

$$\left\{ \begin{array}{l} P_{i1} = (r_{i1}, \varphi_{i1}, \theta_{i1}) = (r_{i1}, \varphi_{i1}, 0) = \left(r_{i1} \cdot \cos \left(\frac{\pi}{2} - \theta_{i1} \right), \varphi_{i1}, 0 \right) = (r_{i1} \cdot \sin \theta_{i1}, \varphi_{i1}, 0) \\ P_{j1} = (r_{j1}, \varphi_{j1}, \theta_{j1}) = (r_{j1}, \varphi_{j1}, 0) = \left(r_{j1} \cdot \cos \left(\frac{\pi}{2} - \theta_{j1} \right), \varphi_{j1}, 0 \right) = (r_{j1} \cdot \sin \theta_{j1}, \varphi_{j1}, 0) \\ P_{ir} = (r_{ir}, \varphi_{ir}, \theta_{ir}) = (r_{ir}, 0, \theta_{ir}) = (r_{ir} \cdot \cos \varphi_{ir}, 0, \theta_{ir}) \\ P_{jr} = (r_{ir}, \varphi_{ir}, \theta_{ir}) = (r_{ir}, 0, \theta_{ir}) = (r_{ir} \cdot \cos \varphi_{ir}, 0, \theta_{ir}) \end{array} \right. \quad (31)$$

Calculation: $\Delta\varphi$ and $\Delta\theta$:

$$\left\{ \begin{array}{l} \Delta\varphi = |\Delta\varphi_i - \Delta\varphi_j| \\ \Delta\theta = |\Delta\theta_i - \Delta\theta_j| \end{array} \right. \quad (32)$$

Analysis:

After t time, the ratio of the LRF scanning angle range at the zenith angle to the total range of angles to be scanned is the same as the ratio of the LRF scanning angle range at the deflection angle to the total range of angles to be scanned:

$$\frac{\omega\varphi \cdot t}{\Delta\varphi} = \frac{\omega\theta \cdot t}{\Delta\theta} \quad (33)$$

which is:

$$\frac{\omega\varphi}{\Delta\varphi} = \frac{\omega\theta}{\Delta\theta} \quad (34)$$

That is, the deflection angle scanning angular velocity $\omega\varphi$ of the LRF and the zenith angular scanning angular velocity $\omega\theta$ must satisfy the following proportional relationship:

$$\frac{\omega\varphi}{\omega\theta} = \frac{\Delta\varphi}{\Delta\theta} \quad (35)$$

Implement:

Let the deflection angle scanning angular velocity $\omega\varphi$ of the LRF and the zenith angular scanning angular velocity $\omega\theta$ satisfy the above relationship (35), so that the LRF starts from the point P_i and starts scanning the straight line segment of the target space.

Judge and implement:

Whether $r_k=r_j$:

1. If yes, stop the LRF scan operation. The LRF scan of the spatial straight line segment is completed.

2. If not, continue the LRF scan until the LRF scan of the spatial straight line is complete (I.E.: Proposition 1 above).

N.B.: It is very simple to calculate the values of $\Delta\varphi$ and $\Delta\theta$ in the $R\Phi\Theta$ detection spherical coordinate system. This eliminates the need to calculate the triangle side length and the cosine theorem in trigonometric functions when calculating the angle in the Cartesian coordinate system. Applications such as more complex calculation steps. At the same time, this also reflects one of the advantages of the $R\Phi\Theta$ detection spherical coordinate system in the constructed coordinate system to directly define the coordinate data of the detection point of the scanning detection process.

4. Algorithm expansion:

The above algorithm is based on the situation that the coordinates of the starting point and the ending point of the scanning space line have been known. The final result of the algorithm is a proportional relationship between the deflection angle scanning angular velocity $\omega\Theta$ of the LRF and the zenith angular scanning angular velocity $\omega\varphi$. That is to say, as long as we set the deflection angle scanning angular velocity $\omega\varphi$ of the LRF and the zenith angular scanning angular velocity $\omega\varphi$ to satisfy this proportional relationship, the LRF will continue to scan along the direction of the spatial straight line segment. That is to say, the above algorithm is an algorithm for the LRF to scan a spatial straight line segment that already knows the coordinates of the starting point and the coordinates of the ending point, and can be easily extended to the spatial line of the LRF to know only the coordinates of the starting point and the direction of the straight line. The segment performs an extended scan algorithm.

Next, we extend the parameter setting of the scanning algorithm of the spatial straight line segment of the LRF that knows the coordinates of the starting point and the coordinates of the ending point to explain the extended algorithm.

This extension algorithm has two basic parameters that we need to pay attention to:

1. The starting point coordinate of the straight line segment of the space to be detected $P_i(x_i, y_i, j_i)$;
2. The direction of the straight line segment of the space to be detected.

The first parameter is easily available and is usually given directly before the scanning algorithm is enabled. The acquisition of the second parameter is the key to this extension algorithm. To solve this problem, we define that the direction of the spatial straight line segment is represented by the two end points of a sub-line segment of the spatial straight line segment. Therefore, in the extended algorithm, we use the direction of the coordinates of the starting point of the straight line segment with the scanning space and the coordinates of any a point on the straight line segment of the space as the direction of the straight line segment of the space. For example, if the

spatial straight line segment to be scanned is $P_i P_n$, the point P_i has a coordinate of $P_i(x_i, y_i, z_i)$, and the point $P_n(x_n, y_n, z_n)$ is an unknown end point in the extending direction of the spatial straight line segment. The point $P_k(x_k, y_k, z_k)$ is an already known point in the direction of extension of the straight line segment of the space. Then we define the direction of the straight line segment of the undetermined terminating end point as $\langle P_i, P_k \rangle$.

5. Algorithm operation:

Inherit the characteristics of the scanning algorithm of the LRF to the spatial straight line segment that already knows the coordinates of the starting point and the coordinates of the ending point. The point $P_k(x_k, y_k, z_k)$ is set as the virtual end point $P_j(x_j, y_j, z_j)$ of the spatial straight line segment, that is,

$$P_j(x_j, y_j, z_j) = P_k(x_k, y_k, z_k) \quad (36)$$

Performing the same algorithm as the LRF scan algorithm for a spatial straight line segment that already knows the coordinates of the starting point and the coordinates of the ending point, only needs to transform the steps that are determined and executed into:

Judge and implement:

Whether have accepted the termination of the scan instruction:

1. If yes, stop the LRF scan operation. Complete the LRF scan of a straight line segment of unknown length.
2. If not, continue the LRF scan operation until the instruction to terminate the LRF scan is received, and complete the LRF scan of a straight line segment of unknown length(I.E.: Proposition 1 above is realized).

Through the above algorithm, we can easily control the LRF to scan the straight line segments in any space from beginning to end. It is also convenient to control the LRF to scan a straight line segment in space along a certain straight line from a certain point. However, it is not our ultimate goal to implement a control LRF to scan any line segment in space or to scan along a particular line direction. The final result we want is the scan data that the LRF gets when scanning the straight line segment of the space. Since the scanned straight line segment has an infinite number of scanning position points, we can only effectively extract the scanned data of the partially scanned data points. The data point extraction strategy we designed is: continuous scanning, when eligible, immediately extracting. That is to say, the LRF scan is continuous, but we will not continuously extract the data points. (NB: The specific reasons have been discussed in detail in the previous two sections of this section, please refer to the previous analysis

section) Instead, a strategy of extracting data points at intervals is used. The specific data extraction algorithm is as follows:

Define parameters:

1. λ :

The spatial density of the extracted data points. The number of scanned data points that need to be extracted within the scan path per unit length is represented by λ .

2. $d\phi$:

The per unit angle of the data points extracted from the zenith angle of the LRF scan;

3. $d\Theta$:

The per unit angle of the data points extracted from the deflection angle of the LRF scan;

4. $P_i(r_k, \varphi_k, \theta_k)$:

Point P_i is the starting end point of the straight line segment of the detected space. $R\Phi\Theta$ detection spherical coordinates;

5. $P_j(r_j, \varphi_j, \theta_j)$:

Point P_j is the terminative end point of the straight line segment of the detected space. $R\Phi\Theta$ detection spherical coordinates;

6. $P_o(x_o, y_o, z_o)$:

Point P_o is the real-time XYZ following coordinate point of the LRF when scanning;

7. D_{ik} :

The linear distance between the currently detected point P_k and the starting detected point P_i ;

8. L_l :

The length of the space of the straight line segment to be detected;

9. L_n :

The number of segments of the detected straight line segment;

10. dl :

The linear distance between two adjacent scanned data points.

Define variables:

Pk(r_k, φ_k, θ_k):

Point P_k is the RΦΘ detection sphere coordinates of the current scanning point of the LRF.

Define a storage collection:**Ds[] :**

Extracts a set of scan data points for storing coordinate data of scanned data points acquired during equal-angle interval scanning.

(N.B.: If the XYZ following coordinates or GXGYGZ coordinates of the LRF position at the time of scanning are not recorded, the RΦΘ detection sphere coordinates extracted from the scanned point need to be converted into corresponding XYZ following coordinates or GXGYGZ coordinates before storage.)

Algorithm description:

We designed two algorithms for the data extraction of the detection points of the fixed-point scanning algorithm using LRF to scan the spatial straight line segment. The first algorithm is realized by calculating the linear length of the spatial straight line segment, combined with the extracted data point density λ . The second algorithm is to calculate the zenith angle range and the deflection angle range between the starting end point and terminal end point of the detected spatial straight line segment, by combining the angular interval $d\varphi$ of the extracted data points of the LRF scanning zenith angle with the angular interval $d\Theta$ of the extracted data points of the LRF scanning deflection angle. The second algorithm inherits the characteristics of the LRF fixed-point scanning algorithm for spatial straight line segments. Moreover, the second algorithm does not need to know the coordinates of the end point of the straight line segment of the detected space. Therefore, the second algorithm is suitable for knowing the terminal end point of detected spatial straight line segment or not. Therefore, we recommend the second algorithm.

Algorithm I:**Calculation:**

1. The length of the line segment to be detected:

$$Ll = \sqrt{(\Gamma_{rtx}(Pj) - \Gamma_{rtx}(Pi))^2 + (\Gamma_{rty}(Pj) - \Gamma_{rty}(Pi))^2 + (\Gamma_{rtz}(Pj) - \Gamma_{rtz}(Pi))^2} \quad (37)$$

2. The number of the detected straight segments:

$$Ln = Ll \times \lambda - 1 \quad (38)$$

3. The straight line distance between two adjacent scanned data points:

$$dl = \frac{Ll}{Ln} = \frac{Ll}{Ll \times \lambda - 1} \quad (39)$$

4. The linear distance between the current detected point Pk and the starting detected point Pi is Dik:

$$Dik = \sqrt{(\Gamma_{rtx}(Pk) - \Gamma_{rtx}(Pi))^2 + (\Gamma_{rty}(Pk) - \Gamma_{rty}(Pi))^2 + (\Gamma_{rtz}(Pk) - \Gamma_{rtz}(Pi))^2} \quad (40)$$

Judgment:

1. Whether Dik can be divisible by dl:

$$\begin{cases} \xi = Dik \bmod dl \\ \varepsilon = \frac{Dik}{dl} \end{cases} \quad (41)$$

- a) If $\xi=0$, the current detection point is the $\varepsilon+1$ th data point to be extracted. Record the RΦΘ detection ball coordinate information of the detection point:

$$Ds[\varepsilon] = ((xo, yo, zo), (xk, yk, zk)) \quad (42)$$

- b) If $\xi=0$ does not hold, then the current detection point is not the data point that needs to be extracted, then the point data is not extracted.

2. Whether Dik is equal to Ll:

- a) If Dik is equal to Ll, the detection data extraction is ended and the algorithm is completely executed.
- b) If Dik is not equal to Ll, continue to detect data extraction and the algorithm continues to execute.

Algorithm 2:**Import:**

Obtaining the angular change amount $\Delta\theta$ of the zenith angle and the angular change amount $\Delta\varphi$ of the deflection angle, corresponding to the starting point and end point of the straight line segment to be scanned, respectively.

Calculation:

The angular variation range $\Delta\theta_k$ of the zenith angle and the angular variation range $\Delta\varphi_k$ of the deflection angle between the current detection point and the starting point of the spatial straight line segment:

$$\begin{cases} \Delta\varphi_k = |\Delta\varphi_i - \Delta\varphi_k| \\ \Delta\theta_k = |\Delta\theta - \Delta\theta_k| \end{cases} \quad (43)$$

Judgment:

1. Whether $\Delta\varphi_k$ can be divisible by $d\varphi$:

$$\begin{cases} \xi\varphi = \Delta\varphi_k \bmod d\varphi \\ \varepsilon\varphi = \frac{\Delta\varphi_k}{d\varphi} \end{cases} \quad (44)$$

2. Whether $\Delta\theta_k$ can be divisible by $d\theta$:

$$\begin{cases} \xi\theta = \Delta\theta_k \bmod d\theta \\ \varepsilon\theta = \frac{\Delta\theta_k}{d\theta} \end{cases} \quad (45)$$

Through the above algorithm design, we can easily get:

$$\varepsilon\varphi = \varepsilon\theta \quad (46)$$

Analysis:

1. If $(\xi\varphi=0) \wedge (\xi\theta=0)$ holds, the current detection point is $\varepsilon\varphi+1$ (IE: $\varepsilon\theta+1$). The data points to be extracted are recorded, and the RΦΘ detection spherical coordinates data of the current detection point is recorded.

$$Ds[\varepsilon] = ((x_o, y_o, z_o), (x_k, y_k, z_k)) \quad (47)$$

2. If $(\xi\varphi=0) \wedge (\xi\theta=0)$ does not hold, then the current detection point does not need to be extracted data points, then the point RΦΘ detection spherical coordinates data is not extracted and recorded.

Judgment:

1. If the coordinates of the end point of the detected space straight line segment are known, it is judged whether: $(\Delta\varphi_k = \Delta\varphi) \wedge (\Delta\theta_k = \Delta\theta)$ holds:

- a) If it is established, the extraction of detection data is ended and the algorithm is executed.
- b) If not, continue the extraction of detection data and the algorithm continues to execute.

2. If the algorithm does not know the end point coordinates of the detected space straight line segment, then determine: whether to accept the termination command:

The application of this algorithm:

1. The LRF scanning implementation of two endpoints of a straight line segment in space are known.

- a) Set the straight line segment to be scanned as P_iP_j ;
- b) Using the fixed-point scanning algorithm described above, obtain the proportional relationship between the deflection angular scan angular velocity ω_L and the zenith angular scanning angular velocity ω_φ of the LRF;
- c) Set the corresponding scanning angle or scanning time to complete the scanning;
- d) Using the above algorithm for extracting scanned point data, extracting data information of the required scanned point data for analysis for other algorithm programs.

2. The LRF scanning implementation without knowing the end point of the straight line segment in the space and only knowing the direction of the line to be scanned and the starting end point.

- a) Set the straight line segment to be scanned as P_iP_j ;
- b) Using the fixed-point scanning algorithm described above, obtain the proportional relationship between the deflection angular scan angular velocity ω_L and the zenith angular scanning angular velocity ω_φ of the LRF;
- c) Set the corresponding scanning angle or scanning time to complete the scanning;

- d) Using the above algorithm for extracting scanned point data, extracting data information of the required scanned point data for analysis for other algorithm programs.

Scan mode algorithm

As explained in the previous section, the LRF group of the sensor system of the robot of this project has two scanning modes. One is the normal scan mode and the other is the locking scan mode. Below we will introduce the detection algorithms of these two scanning modes.

Normal scan mode

As shown in FIGURE 24, when the sensor system of the robot of this project performs a normal scanning, two LRFs in one LRF group respectively emit detection light. The LRF's detection light is emitted evenly. In the process of detecting light rays by the LRF, a straight line parallel to the Z-axis is used as a rotation axis to perform a reciprocating 180-degree oscillating rotational motion. Introducing the concept of angular range, we can see that when the detection light scans the detected object, only the part of the detected object that is within the angular range will be detected by the LRF light, and the latter part will not be detected by the LRF light due to the detecting light being blocked. N.B.: The density of the detected light emitted by the LRF shown in Figure 24 is only shown as schematic diagram, and the actual light density is much larger than this. Therefore, when the LRF group in the normal mode detects the detected object, it only emits the detection light in the form of a round-trip sweep of 180 degrees at a specific time, and scans only the front surface of edge can be detected of the detected object toward the detected light.

By introducing the concept of the previous combined coordinate system, we know that the main coordinate system used by the robot when detecting is the $R\Phi\Theta$ detection spherical coordinate system. Before describing the specific detection algorithm, we will analyze the detection characteristics of the LRF group.

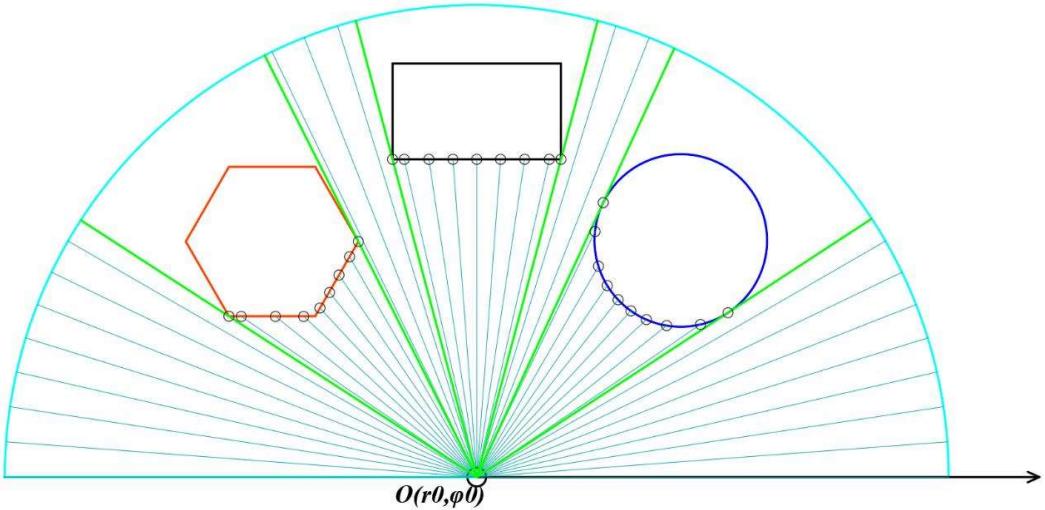


Figure 24. Schematic diagram of normal scan mode detection.

Detection characteristics of the LRF group

Edge issues

As shown in Figure 25, it is assumed that at time T1, the robot is at the P0 point and the normal mode is activated for detection. There are three detected objects in the detection area at this time. Since the original function of the LRF is to perform distance detection, with this feature, we separate the detected objects at different positions according to the different distance data of the detected object. So what we care about most is the edge issue. At a particular moment, only the edge of the detected object that can receive the detected light from the LRF group from the front can be detected. We define that the edge of an object that can receive the detected light at a certain time T is called the detectable edge of the object at time T. Introducing the previous definition, in Figure 25, it is assumed that the detection light is scanned in the counterclockwise direction and then scanned in the clockwise direction, so that the reciprocating scanning motion is continuously performed. In the process of scanning the LRF, we use $P_{ijk}(r_{jk}, \varphi_{jk}, \theta_{jk})$ to represent the kth detected point on the detectable edge of the jth detected object when the robot is at the P_i point.

Theoretically, when the LRF's detected light traverses a detected object, a large number of edge points are detected. But in fact, under the effective LRF precision, the distance between adjacent pairs of some detection points is very small, and there is no useful identification information between them. We call these detection points approximately the same point. If the approximately same data of points are collected and processed, it obviously increases the computational burden of the system operator, and this has little effect on the accuracy of the final result. Because the physical

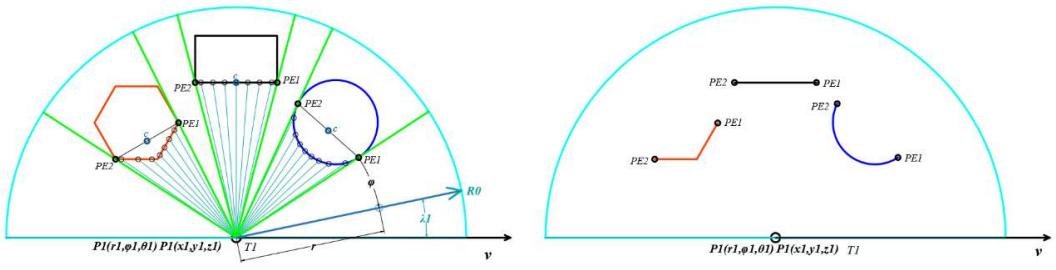


Figure 25. The detectable edges in normal scan mode. I

properties that can be reflected by adjacent approximately identical points are very similar, the differences between them can be almost ignored. So we need to perform a "dot screening" operation on dense detected points.

Point screening is the scanning of data from one detection point at a distance. This determined distance is called "screening distance". In the RΦΘ detection spherical coordinate system, we define the "screening angle" instead of the "screening distance". Obviously, in the RΦΘ detection spherical coordinate system, the "screening angle" is more suitable than the "screening distance". The screening angle is the Φ angular interval corresponding to the screening distance in the RΦΘ detection spherical coordinate system. The specific screening algorithm is described in the next algorithm section.

Dynamic RΦΘ detection of the spherical coordinate system

The right diagram of Figure 25 shows the detectable edge data of the detected object actually detected by the robot at point P1 (x_1, y_1, z_1) at time T1, which is also detected by the detection system in real time at the time. In the right diagram of Figure 26, the detectable edge data of the detected object actually detected by the robot at point P2 (x_2, y_2, z_2) at time T2 is shown, which is also the locality of the detection system at the time. The detected data. In the figure, we use PE1 to indicate the tangent point of the first boundary line of the detected object and the detectable edge of the object, that is, the first boundary detection point. Use PE2 to indicate the tangent point of the second boundary line of the detected object and the detected edge of the object, that is, the second boundary detection point. That is, the PE1 and PE2 points of each detected object are the endpoints of the detectable edge of the object at that moment. Let the center of the line connecting PE1 point and PE2 point be point c. The point c is used for the description and modeling of the object movement recognition process in the detection algorithm mentioned later. R0 in the figure indicates the reference axis parallel to the GXGYGZ positioning coordinate system, and the RΦΘ with the P1 point as the origin detects the spherical coordinate system. It can be known from the definition of the RΦΘ detection spherical coordinate system in the combined coordinate system that the RΦΘ detection spherical coordinate system is a dynamic coordinate system. That is to say, the origin of the RΦΘ detecting spherical coordinate system

changes at any time during the execution of the following task by the robot. However, the RΦΘ detection spherical coordinate system has an invariant or an angle between R0 and the X axis. In the figure, v is the motion vector direction of the robot.

When using the normal mode of the LRF group for detection, it is not difficult to find that there is a very significant feature of the LRF detection. When the robots in different spatial positions perform the same object, the detected edge data is likely to be detected. It is different. That is to say, the robots in different positions are likely to detect the edge data of different parts of the detected object. For example, the robot shown in Figure 26 has a detection edge that has a rectangular right edge more than the detection edge obtained by the robot shown in Figure 25. We should consider this in designing the probing algorithm. When designing the detection algorithm, we should also pay special attention to the timely conversion of the detected important data of the detected points into the corresponding XYZ following coordinate system data. Since the RΦΘ detection spherical coordinate system is dynamic, the coordinate data it can provide is relative (i.e., the coordinate data is relative to the spatial position point Pi point where the LRF group at a particular moment is located). The RΦΘ detection spherical coordinate system can be established at the position of the corresponding LRF group at any position where the robot is located when the robot performs the following task. The RΦΘ detection spherical coordinate system is independent of the moving speed of the robot, and is only related to the X-axis of the XYZ following coordinate system established at the beginning of the following task (i.e., parallel to the X-axis of the XYZ following coordinate system). As shown in Figure 27, the origin P2 of the RΦΘ detecting spherical coordinate system can be determined by the robot at any position (e.g., P2 point.).

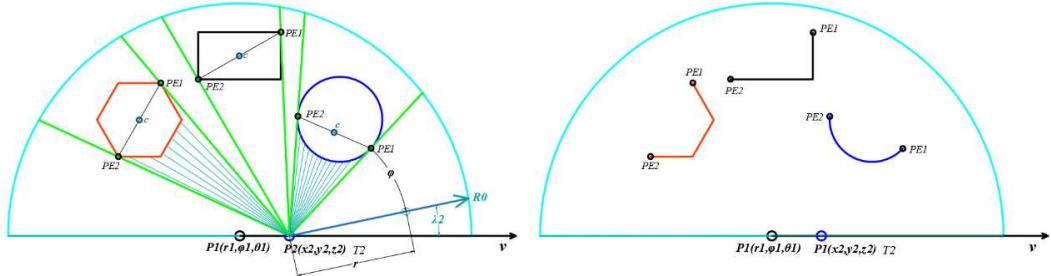


Figure 27. The detectable edge of the normal scan mode. 2

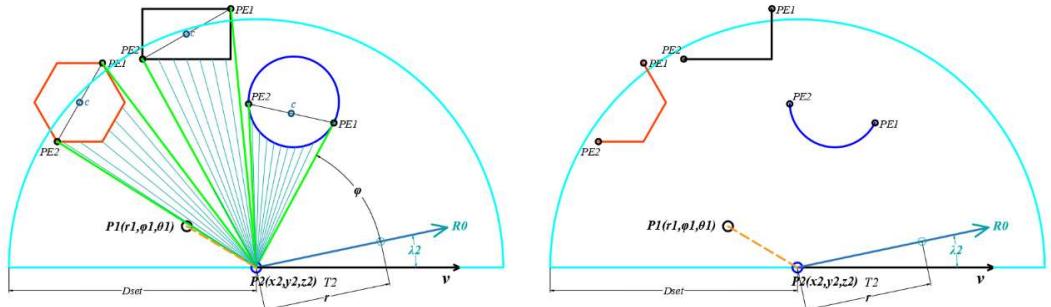


Figure 28. The detectable edge of the normal scan mode. 3

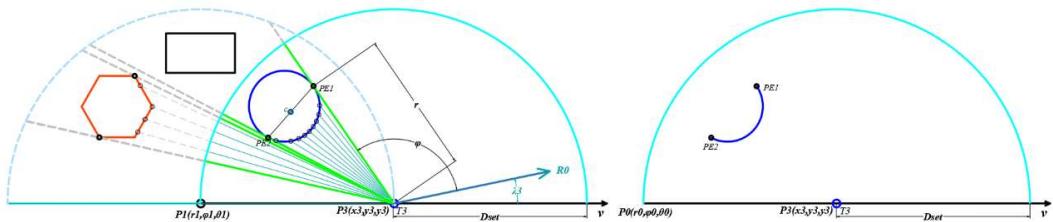


Figure 29. Normal mode detection edge in mode with limited detection radius. 5

Reasonable detection radius

The characteristics of the robot using the LRF group for detection are discussed above. Another issue worth discussing with the LRF group is that since the current LRF is based on laser ranging, we need to consider how large the radius we should set for LRF of the robot to determine how large range the robot needs to detect. Or, from another perspective, we need to set a fixed detection radius for the LRF group. In order to deal with this problem, we have defined two detection radius modes: undefined detection radius mode and limited detection radius mode. As shown in Figure 28, an undefined detection radius mode is shown. The robot at point P3 is not set to a specific detection radius. Therefore, in the detection area, objects within the effective detection distance of the LRF laser are detected. Figure 29 shows a defined detection radius mode. The robot at point P3 is set to a specific detection radius Dset. Therefore, in the area of the semicircle in a plan view with a radius of Dset, objects within the effective detection distance of the LRF laser are detected. However, among the detected objects, only the detection data of the object within the semicircle area of limited detection radius is acquired by the detection system. That is to say, when the robot is in the limited detection radius mode of the normal detection mode, although the LRF group has a detection effect on the objects in the effective detection range corresponding to the LRF group, the detection mechanism only collects the detection data of the location of the detected object within the range of the detection radius. For example, the detected data of the detectable edge of the object represented by the left hexagon in Figure 29 is not collected by the system.

Why do we have to set a limited detection radius? This is because, for many robots that perform Front-Following tasks, the detection data of objects that are too far away from these robots does not play a significant role in predicting the future actions of the detected objects and the future Front-Following actions of the robot itself. In other words, the impact of these effects can be ignored relative to the cost of processing resources and storage resources. Of course, there are also some special cases. For example, for some robots that do not consider the cost and focus on solving the task performance problem, the detection mode with undefined detection radius can improve the robot's understanding of the environment factors. This will also increase the amount of data that can be processed when the robot performs map planning storage.

In general, when the following robot performs the normal detection mode, the data storage mode is generally performed using obscured scan storage. At the same time, the detection radius mode of the robot is also generally set to define the detection radius mode. If the robot needs to perform map planning, it can perform map planning storage on the detected data, and the detection radius mode can be set to an undefined detection radius mode or a limited detection radius mode.

Test data merge and update

If the robot is only detecting the environment around the robot at a fixed point in space, and only using the LRF detector or LRF group for detection, then the robot can never fully grasp the complete environmental condition data around it. If the robot is fixed at a certain point in the detection area, even if the robot performs a large number of scanning movements at this fixed position, the final result is likely that the robot cannot understand all of the environmental condition data. This is because a moving LRF-detecting robot will always have more opportunities to learn more about the surrounding environment than a stationary LRF-detecting robot. However, there is no guarantee that the moving LRF robot will be able to understand the overall environmental conditions. We can only say that the moving LRF detecting robot has more opportunities to learn more detailed environmental conditions than the inactive LRF detecting robot. In practical applications, the detection data transformation is very fast, and the amount of data is also very large. How to extract the really useful detection data within the specified time is a test of the detection algorithm.

For robots that perform oblivious storage scans, they do not need to save the environment data, as long as their computer operation unit is fast enough to complete the extraction and analysis of environmental data in a short period of time. Then the robot does not need to establish a navigation map. However, for robots that perform map-planned storage, it is necessary to establish a navigation map, analyze and store the related environmental data. Therefore, for a robot performing map-planned storage, a good extraction and storage mechanism for detecting data is necessary.

Therefore, we design a data processing mode for robots that implement Map planning storage, that is, the merging and updating mode. In this mode, the robot will continuously detect the environment in the process of following the target person, and continuously obtain the relevant environmental data, and then analyze and merge the environmental data obtained from different locations. So when the robot completes a follow-up task, the map it plans is likely to be updated. Figure 30 shows the surrounding environment detection data of the robot after the detection process of Figure 25, 26, 27, 28 and 29 is merged and updated.

When a robot completes a follow-up task, if the Front-Following path of the robot does not change much compared with the previous Front-Following paths, the environmental data near a certain point may not be updated too much. See the

algorithmic analysis section for specific update processing.

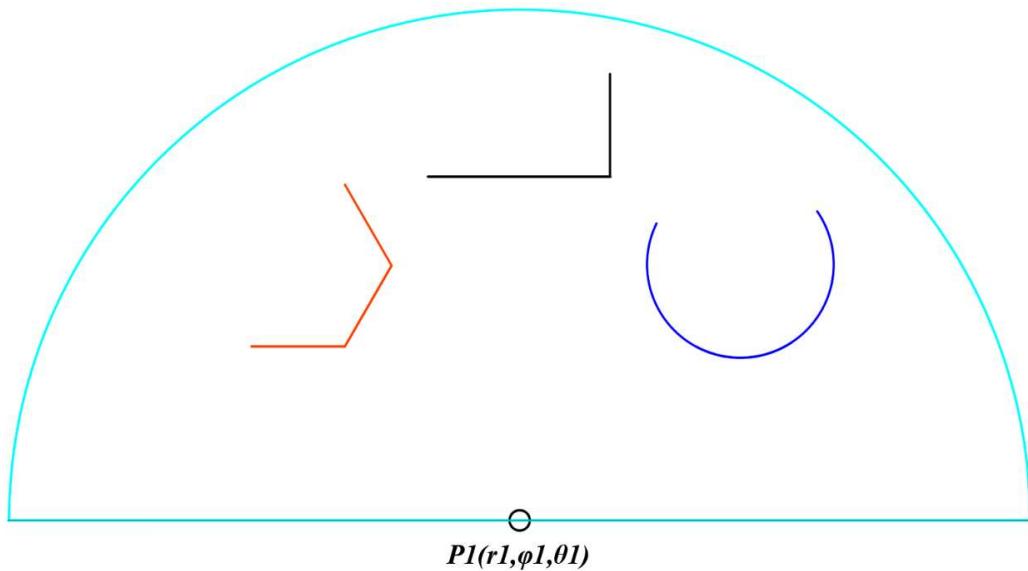


Figure 30. The merging and processing of scan results in normal scan mode.

Detection Mode Algorithm Analysis

Algorithmic analysis of normal detection mode

Suppose at T1, the XYZ coordinates of a LRF group of robots performing Front-Following tasks are $P_i(x_i, y_i, z_i)$. The angle between the extension line of the corresponding robot rectangle edge and the reference polar axis with the changeless direction is λ . And v is the geometric center direction of motion of the robot. And ω is the scanning angular velocity of an LRF used for this scanning analysis in the LRF group. According to the definition of LRF group, point $P_i(x_i, y_i, z_i)$ is the center of the LRF's 180 degree scanning motion. D_{set} is the corresponding limited detection radius. We will not discuss whether to set up the detection radius of D_{set} at first, but we will discuss it later. (i.e., ignore the discussion of D_{set} first.) We use a circle to represent the j th object to be detected. Firstly, LRF rotates the scanning in counterclockwise direction, and repeats the scanning action after reaching 180 degrees. Therefore, the definition: the coordinates of the k th detection point of the j th object detected by LRF scanning rays in the LRF group at the point $P_i(x_i, y_i, z_i)$ are $P_{ijk}(r_{jk}, \phi_{jk}, \theta_{jk})$. As shown in Figure 31, the first detected point at the intersection of the first boundary line of the j th object with the object itself is the first detection point, and the coordinates are $P_{ij1}(x_{j1}, y_{j1}, z_{j1})$. The detected point at the intersection of the second boundary line of the j th object with the object itself is the last one. Assuming that there are n detected points in

total, the coordinates of the nth detected point are $P_{ijn}(x_{jn}, y_{jn}, z_{jn})$. As shown in Figure 31, as we know from the previous definition, the offset angle of the detected object is 45 degrees and the range of the offset angle is 46 degrees. As shown in Figure 32, as we know from the previous definition, the offset angle of the detected object is 63 degrees and the range of the offset angle is 56 degrees.

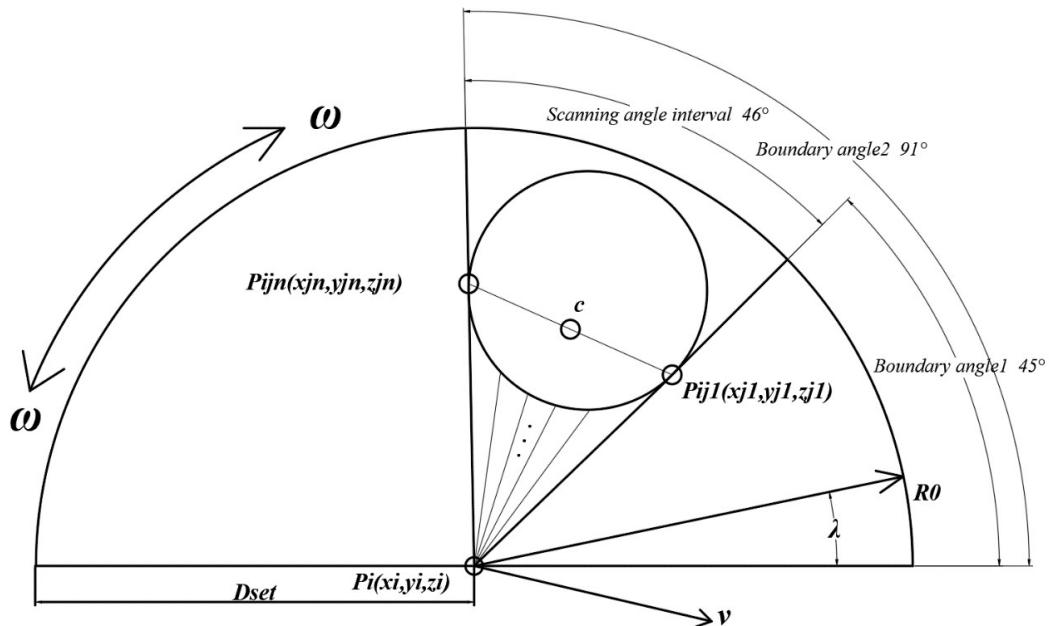


Figure 31. Algorithm analysis in normal scan mode. I

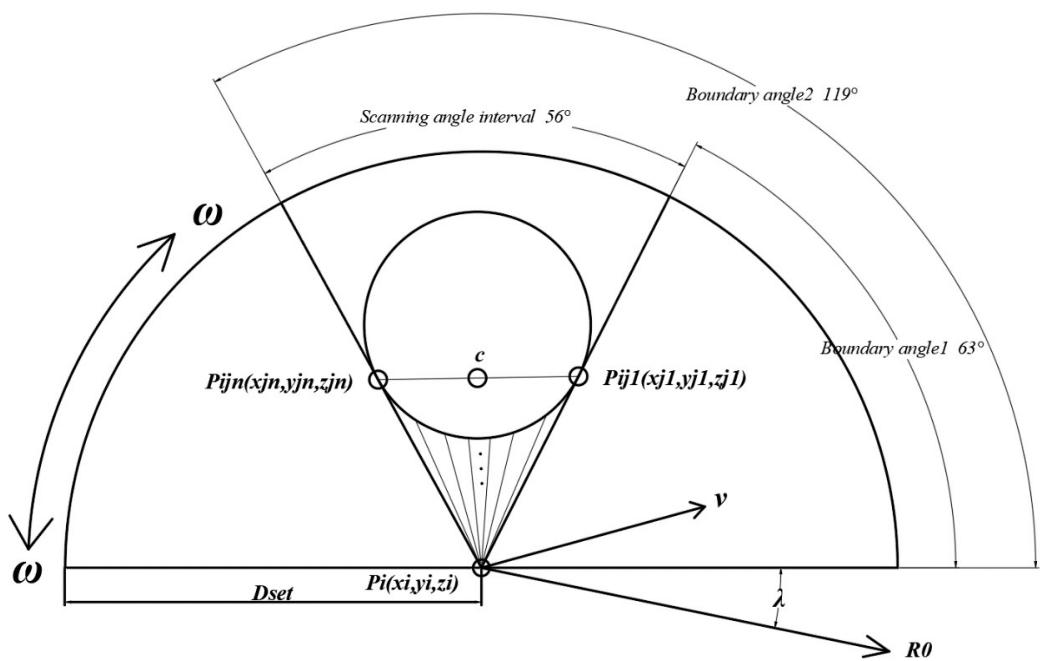


Figure 32. Algorithm analysis in normal scan mode. 2

In summary, the following conclusions are obtained:

1. Define the screening angle of the detection system as Δd , that is, after extracting the coordinate data of the first detection point, the detection system extracts the coordinate data of one detection point every angular interval of Δd . Let the angle of the first boundary of the jth detected object be β_1 , and the angle of the second boundary be β_2 , then, the scanning angle interval of the jth detected object is

$$\Omega_j = |\beta_1 - \beta_2| \quad (48)$$

2. Then, in the scan detection interval Ω_j , the number of all scan detection points in the scan detection interval Ω_j is

$$n = \frac{\Omega_j}{\Delta d} \quad (49)$$

3. The time required for the LRF scanning to complete the scan of the object to be detected is

$$t = \frac{\Omega_j}{\omega} \quad (50)$$

There is a key point what we can't ignore is that while the LRF is performing a rotational scan, the robot is also likely to be moving. In general, the angular velocity of the LRF scanning is very fast. We define the time taken for the LRF to scan a 180 degrees semi-circular detection area as the cycle T_c . Use π to represent 180 degrees, then there is

$$T_c = \frac{\pi}{\omega} \quad (51)$$

Under normal circumstances, T_c is a small amount of time. During T_c time, the motion displacement of the robot is almost negligible. However, in some algorithms that pursue high precision, it is necessary to consider the relative displacement of the robot in a short time interval. In the scanning algorithm of this report that has high requirements for time intervals, we assume that the robot is moving in a uniform shifting curve during the T_c time.

As shown in Figure 33, it is considered that the speed at which the robot makes a curved motion in the three-dimensional space during the time T_c is \vec{v} . Because the T_c time is very short, when considering the calculus, we think that the robot does the shifting curve motion in the T_c time, that is, the curve motion with the magnitude and direction of the acceleration. Set in T_c time, the robot moves from point $P_i(x_i, y_i, z_i)$ to point $P'_i(x'_i, y'_i, z'_i)$. In the process of the robot performing the following task, it is

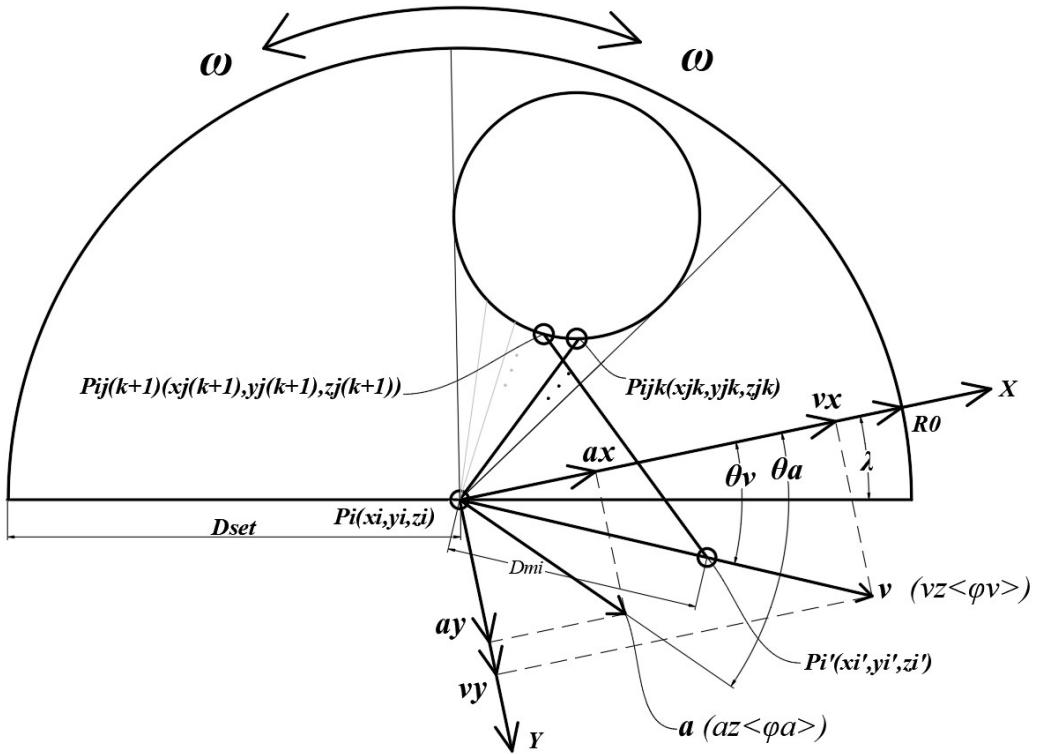


Figure 33. Algorithm analysis in normal scan mode. 3. Consider the motion of the robot during the scan.

assumed that no relative movement occurs between the robot and each LRF group mounted on the robot. So LRF and robot can be seen as the same particle. When set at P_i point, the speed vector of the robot is \vec{v} , the acceleration vector is \vec{a} , the angle between \vec{v} and the X axis (i.e., the R_0 reference axis) is θ_v , the angle between \vec{a} and X axis (i.e., the R_0 reference axis) is θ_a . Since the robot of this project uses a three-dimensional coordinate system, we define the angle between \vec{v} and Z axis to be φ_v , and the angle between \vec{a} and Z axis is φ_a . Since it is difficult to visually represent the Z axis in a two-dimensional map parallel to the XY plane, we define "az< φ_a " to indicate that the component of \vec{a} on the Z axis is a_z , and the angle between \vec{a} and the Z axis is φ_a . We define "vz< φ_v " to indicate that the component of \vec{v} on the Z axis is v_z , and the angle between \vec{v} and the Z axis is φ_v .

Introducing the design idea of synthesis and decomposition of motion, at the point of the point $P_i(x_i, y_i, z_i)$, the velocity component of \vec{v} along the X axis is v_x , and the velocity component of \vec{v} along the Y axis is v_y . Let the velocity component \vec{a} along the X axis is a_x and the velocity component \vec{a} along the Y axis is a_y . Assume that after Δt time, the robot moves from point $P_i(x_i, y_i, z_i)$ to point $P'_i(x'_i, y'_i, z'_i)$. The modules of \vec{a} and \vec{v} (i.e., equivalent in size or length) are a and v , respectively. The sub-displacements along the X-axis, Y-axis and the Z-axis are s_x , s_y and s_z , respectively, and the total displacement is S .

Then there are

$$\left\{ \begin{array}{l} vx = v \times \cos\theta v \\ vy = v \times \sin\theta v \\ vz = v \times \cos \\ ax = a \times \cos\theta a \\ ay = a \times \sin\theta a \\ az = a \times \cos\varphi a \\ sx = vx \cdot \Delta t + \frac{1}{2} ax(\Delta t)^2 \\ sy = vy \cdot \Delta t + \frac{1}{2} ay(\Delta t)^2 \\ sz = vz \cdot \Delta t + \frac{1}{2} az(\Delta t)^2 \\ S = \sqrt{(sx)^2 + (sy)^2 + (sz)^2} \end{array} \right. \quad (52)$$

Perform the integral operations from 0 to Tc on sx, sy, and sz, respectively. Then there are

$$\left\{ \begin{array}{l} sx = \int_0^{Tc} \left(vx \cdot t + \frac{1}{2} ax(t)^2 \right) dt \\ sy = \int_0^{Tc} \left(vy \cdot t + \frac{1}{2} ay(t)^2 \right) dt \\ sz = \int_0^{Tc} \left(vz \cdot t + \frac{1}{2} az(t)^2 \right) dt \end{array} \right. \quad (53)$$

Integrate S from 0 to Tc, there is

$$S = \int_0^{Tc} \left((vx \cdot \Delta t + \frac{1}{2} ax(t)^2)^2 + (vy \cdot \Delta t + \frac{1}{2} ay(t)^2)^2 + (vz \cdot \Delta t + \frac{1}{2} az(t)^2)^2 \right)^{\frac{1}{2}} dt \quad (54)$$

After Tc time, the robot is at the point Pi'(xi', yi', zi'). There are corresponding coordinates of Pi':

$$\left\{ \begin{array}{l} xi' = xi + sx \\ yi' = yi + sy \\ zi' = zi + sz \end{array} \right. \quad (55)$$

For the offset motion of the robot at Tc time, since the Tc is very small in general, we can also model it as a uniform linear motion. Processing with a nearly uniform linear motion will make the modeling algorithm easier. This project gives a more accurate method for modeling the uniform speed curve motion. If we want to pursue precision, the motion of the uniform shift curve is already high enough for the actual LRF operation. The establishment of a uniform linear motion is not much different from the modeling effect of the robot as no movement.

Another emphasis to note is that the detection data of the detected object is the same regardless of the motion of the robot during the Tc time. It is also said that during the execution of a certain Front-Following task by the robot, the detected data is independent of the motion state of the robot. Proof is shown as follows:

Assuming that Pd(xd, yd, zd) is a certain point to be measured on the object to be tested, the coordinates of the Pd point detected by the robot at the point Pi(xi, yi, zi) are P1(r1, φ1, θ1). The coordinates of the Pd point detected by the robot at the point Pi'(xi', yi', zi') are P2 (r2, φ2, θ2). Although the data of the RΦΘ coordinate system is detected, the data acquisition system will instantly convert the acquired data of the RΦΘ coordinate system into the coordinate data of the XYZ coordinate system:

$$\begin{cases} PT1 = \Gamma rt(P1) \\ PT2 = \Gamma rt(P2) \end{cases} \quad (56)$$

Because for the robot performing a Front-Following task, the XYZ coordinate system determines the origin after starting the task, and for the same object in the XYZ following coordinate system, if the object does not move, then the XYZ coordinates of the same point position on the object are constant. If the object has moved, the XYZ the XYZ coordinates of the same point position on the object are also relatively changed. That is,

$$PT1 = PT2 \quad (57)$$

which is

$$\Gamma rt(P1) = \Gamma rt(P2) \quad (58)$$

The above-mentioned PT1 and PT2 respectively represent the coordinates of the P1 point and the P2 point, after the coordinate conversion from RΦΘ detection spherical coordinate system to the XYZ coordinate system. Γrt is a coordinate conversion operator.

Combined with the concept of the previous combined coordinate system, the XYZ following coordinate system is also not absolutely stationary. During the a robot's executing Front-Following task, the origin of the XYZ following coordinate system remains unchanged. However, when the robot performs the next following task, the XYZ following coordinate system is very likely to change. Therefore, if permanent and versatile storage of the detected environmental data is to be performed, the detection data of the XYZ following coordinate system must be converted into the data of the corresponding GXGYGZ positioning coordinate system. The conversion process is similar to the above process of converting the detection data of the RΦΘ detection spherical coordinate system into the data of the XYZ following coordinate system:

Assuming that the detected coordinate data of a point P in the XYZ following coordinate system is Pt(xp, yp, zp), the detection data finally stored in the robot will be

the data $P_g = \Gamma_{tg}(x_p, y_p, z_p)$, based on the GXGYGZ positioning coordinate system.

Algorithmic analysis of locking detection mode

An important feature of locking detection mode is to use the LRF to lock the target person so that the target person is always in the effective scan area of the LRF scan. For this scanning feature, many well-defined algorithm features can be added to the locking scan algorithm. By the definition of the locking scan mode, we know that locking scan mode is to mobilize the four LRF groups located at the center of the four sides of the robot rectangle to perform 360-degree wrap-around locking on the target character to achieve real-time acquisition of the target character. The purpose of the accuracy of the positioning information. As shown in Figure 34, we divide the three-dimensional space into eight detection areas according to the straight line extension of the four sides of the robot rectangle. In this report, there is no area defined above the positive Z axis of the robot and below the negative Z axis. This is because, as shown in FIGURE 34, the division into eight detection areas has been able to satisfy the project requirements of this report. More detection areas can be added in the upgrade version report.

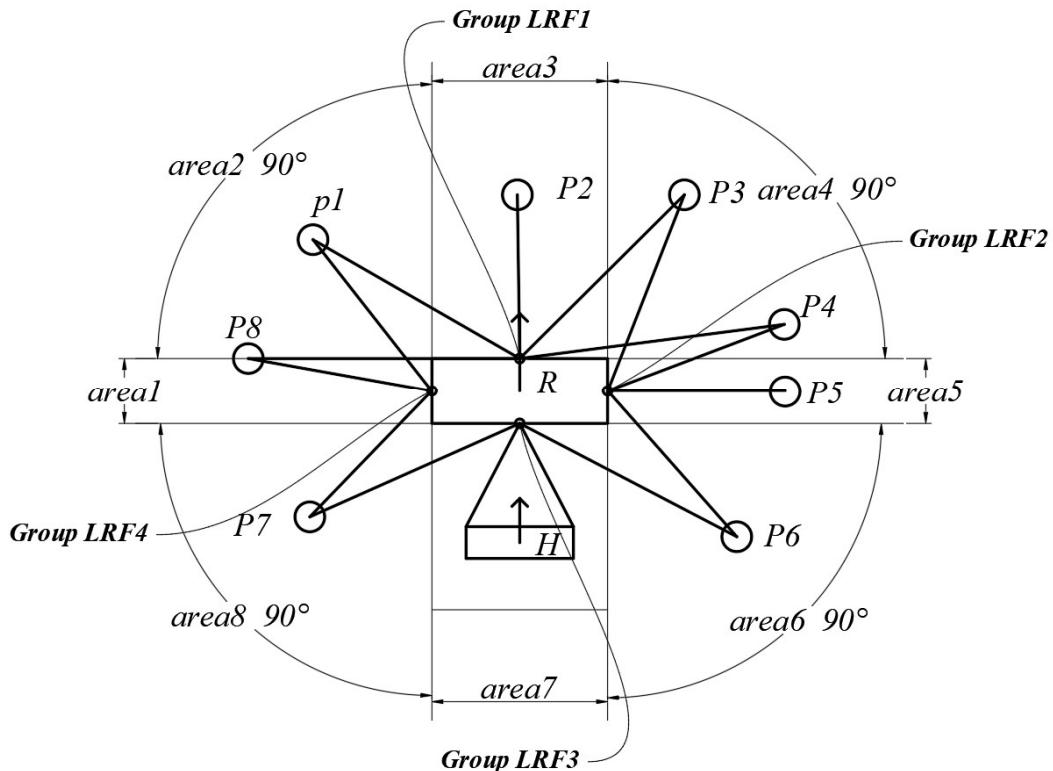


Figure 34. Algorithm analysis in locking scan mode. I. The division of 8 large detection areas.

As shown in Figure 34, it is assumed that the robot sequentially detects the target

characters at eight times of T1, T2, T3, T4, T5, T6, T7, and T8, respectively. The position of the target person is detected at points T1, T2, T3, T4, T5, T6, T7, T8, respectively. (I.E.: T1, T2, T3, T4, T5, T6, T7, T8 means that the time also represents the position point.) Then the working system of the LRF group of the detection system will be:

T1: The target person enters area1 and belongs to a separate detection area. One LRF in Group LRF4 is set for locking scan mode. There is a total of one LRF to perform a locking scan of the target person;

T2: The target person enters area2 and belongs to the overlap detection area. One LRF in Group LRF4 and one LRF in Group LRF1 are set to the locking scan mode. A total of two LRFs perform a locking scan of the target person;

T3: The target person enters area3 and belongs to a separate detection area. One LRF in Group LRF1 is set for locking scan mode. There is a total of one LRF to perform a locking scan of the target person;

T4: The target person enters area4, belongs to the overlap detection area. One LRF in Group LRF1 and one Group LRF2 are set to the locking scan mode. A total of two LRFs perform a locking scan of the target person;

T5: The target person enters area5 and belongs to a separate detection area. One LRF in Group LRF2 is set for locking scan mode. There is a total of one LRF to perform a locking scan of the target person;

T6: The target person enters area6, belongs to the overlap detection area. One LRF in Group LRF2 and one Group LRF3 are set to the locking scan mode. A total of two LRFs perform a locking scan of the target person;

T7: The target person enters area7 and belongs to a separate detection area. One LRF in Group LRF3 is set for locking scan mode. There is a total of one LRF to perform a locking scan of the target person;

T8: The target person enters area8, belongs to the overlap detection area. One LRF in Group LRF3 and one Group LRF4 are set to the locking scan mode. A total of two LRFs perform a locking scan of the target person;

In a summary, when the target person is located in the dual detection zone, the detection algorithm respectively activates one of the two LRF groups corresponding to the dual detection zone to detect the target person. Therefore, when the target person is in the double detection zone, a total of two LRFs will be set to detect the target person. When the target person is in a separate detection area, the detection algorithm activates one LRF in an LRF group corresponding to the separate detection area to scan the target person. At this time, another LRF of the LRF group will be assigned to perform a

normal detection task. That is, when the target person is in the separate detection area, the target person is locked by one LRF in one LRF group and scanned for detection. During the execution of the following tasks by the robot, the eight detection areas divided by the four-sided central model perform normal scanning. This is because, at any time, the robot has to pay close attention to the factors that may affect its surroundings. Therefore, in this project, we set up the robot's 360-degree wraparound normal scan detection method. From the content analysis of the sensor system chapter, we know that the detection on the single detection zone is less affected, and the detection on the dual detection zone is more affected. This design is consistent with the working configuration of the LRF group of this project.

Locking scans mode can also be referred to as feature scan. This is because the locking scan is a centralized high-precision scan. In general, the angular velocity of the LRF of the LRF group in the normal mode is fixed, so the purpose of the setting is to facilitate the detection of the surrounding environment by the same standard. Of course, sometimes we don't need all the LRF groups to use the same angular velocity for detection. In particular, we need to increase the angular velocity of the LRF of the LRF group in the normal detection mode in some detection areas. In this version of project report, we use the same angular velocity for setting all LRFs in normal detection mode.

The running algorithm of the locking scan will be specifically described below.

As shown in Figure 35, we use a circle to represent the target person. Set an LRF in the LRF group to perform a locking scan of the target person. The XYZ coordinate of the position where the LRF is point $P_i(x_i, y_i, z_i)$. Introducing the concepts of the normal scan mode, the first boundary angle and the second boundary angle are also defined. It is assumed that the first boundary angle and the second boundary angle respectively intersect the target person at point $P_{ij1}(x_{j1}, y_{j1}, z_{j1})$ and point $P_{ijn}(x_{jn}, y_{jn}, z_{jn})$. It is defined that at the scanning time, the point $P_{ij1}(x_{j1}, y_{j1}, z_{j1})$ is the first boundary point, and the point $P_{ijn}(x_{jn}, y_{jn}, z_{jn})$ is the second boundary point. Let the target person's first boundary angle be Ψ_{e1} and the second boundary angle be Ψ_{e2} , then the target person's scanning angle range is

$$\Delta\Psi = |\Psi_{e2} - \Psi_{e1}| \quad (59)$$

The locking scan angle range of the target person is:

$$(\Psi_{e1}, \Psi_{e2}) \quad (60)$$

The LRF scan beam does not always fold back immediately when it reaches the first or second boundary of the target person because of the limitation of the LRF mechanical response speed. This is because the locking scan differs from the normal scan mode in that the normal scan mode has a determined foldback angle interval (I.E.:

$(-\pi, \pi)$). However, the angular interval of the locking scan mode is likely to change every time the LRF performs a foldback cycle. That is $(\Psi e_1, \Psi e_2)$ does not necessarily remain the same unless the target person keeps moving. To solve this problem, we defined the concept of a scan buffer.

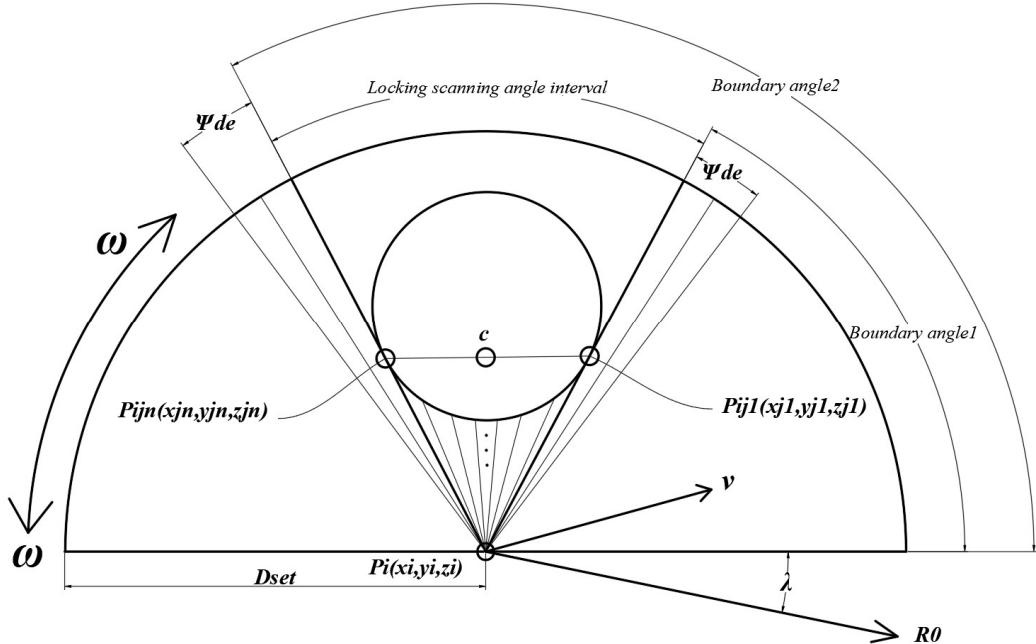


Figure 35. Algorithm analysis in locking scan mode. 2.

As shown in FIGURE 35, the Ψe angle interval distributed to the left of the detected target person is referred to as a left angle buffer, and the Ψe angle interval distributed on the right side of the detected target person is referred to as a right angle buffer. In general, the interval size of the left angle buffer and the right angle buffer is set to the same value. We define that the size of the two angle buffers is Ψe , then the scan angle range of the LRF will actually be

$$(\Psi e_1 - \Psi e, \Psi e_2 + \Psi e) \quad (61)$$

That is to say, when the LRF performs the locking scan, the actual scan angle interval of the LRF will be larger than the angle interval determined by the first boundary angle and the second boundary angle of the target person. The purpose of this design is to reduce the error caused by the mechanical action delay of the LRF, and to reduce the occurrence of false positives.

When we set up the left and right buffers, the LRF can detect more data. The data we collect when the LRF performs a lock scan is called a lock scan data set, which is represented by D_{set} . Obviously, D_{set} contains the detection data we need for the target person, and also contains some detection data that is not belong to the target person. Therefore, we set two distance thresholds: TH_{hn} and TH_{hf} , which represent the near

and far thresholds, respectively. The near threshold THhn is used to set the allowable change distance near the LRF, and the far threshold THhf is used to define the allowable change distance away from the LRF. Normally, THhn and THhf are two deterministic constants that are set to extract the detection data of a useful target person. The midpoint of the straight line connecting the set point Pij1 (x_{j1} , y_{j1} , z_{j1}) and the point Pijn (x_{jn} , y_{jn} , z_{jn}) is c point, and the c point coordinate is c(x_c , y_c , z_c). That are:

$$\begin{cases} x_c = \frac{x_{j1} + x_{jn}}{2} \\ y_c = \frac{y_{j1} + y_{jn}}{2} \\ z_c = \frac{z_{j1} + z_{jn}}{2} \end{cases} \quad (62)$$

That is, the distance from the LRF emission point to point c is:

$$Dci = \sqrt{(xi - xc)^2 + (yi - yc)^2 + (zi - zc)^2} \quad (63)$$

We use Dci as the reference distance and use the distance thresholds THhn and THhf to define the target detection effective detection interval Lhset:

$$Lhset = (Dci - THhn, Dci + THhf) \quad (64)$$

That is to say, only when the data points satisfying the condition that the detection distance is in the Lhset interval in the Dlset set are regarded as the data points of the target person by the detection program. Define Dhset as the set of valid scan detection data for the target person. Let Pt(x_t , y_t , z_t) be a point in Dlset, then only when

$$Dci - THhn \leq \sqrt{(xt - xi)^2 + (yt - yi)^2 + (zt - zi)^2} \leq Dci + THhf \quad (65)$$

the data acquisition system will store the coordinate data of the detection point into Dhset.

About the distance threshold: THhn and THhf setting method. After examining experiments of different humans of different sizes, this paper gives the following definitions:

Let Detwo be the radial distance between the two boundary points of the target person to detect the edge, then

$$Detwo = \sqrt{(x_{j1} - x_{jn})^2 + (y_{j1} - y_{jn})^2 + (z_{j1} - z_{jn})^2} \quad (66)$$

We define two body parameters bn, bf and specify:

$$\begin{cases} THhn = bn \cdot Detwo \\ THhf = bf \cdot Detwo \end{cases} \quad (67)$$

Therefore, we associate the setting of THhn and THhf with Detwo by the above formula. After examining the human body shape and walking posture, we recommend setting the value of bn to 0.6 and the value of bf to 0.8, which will make the detection result more reasonable. In practical applications, we must make reasonable settings according to the specific application environment of the robot (e.g., the number of unrelated humans in the environment where the robot is located) to ensure the highest accuracy of the robot detection data.

CONSTRUCTION OF DETECTION MODEL

Analysis: Subsections Modeling Thought

In the construction of the detection model, we introduce Subsections Modeling Thought. This design idea is derived from the separation and combination system design ideas mentioned at the beginning of this report. The main idea of Subsections Modeling Thought is to divide a large model into multiple small models for modeling, and finally combine all of the small models to form a new large model. This thought allows the associations or not between individual small models. When building a large model, the association between the original small models will affect the construction of the large model. Being good at making good use of the association between small models can increase the robustness of the large model and the efficiency of the work execution of the model, so as to achieve the gaining of additional experimental results.

This section will introduce the specific modeling ideas for the target person in this project. We divided the entire body of the target person into four parts for modeling analysis. As shown in Figure 36, the target person is divided into four detection portions of the head, the torso, the legs, and the feet. The project's human detection is based on the entire human body. The design idea adopted by this modeling algorithm is also the "separation and combination system design idea" mentioned in the design analysis part at the beginning of this report. "Separation" means that the entire human body is divided into four parts. Independent detection of each part of the target person. "Combination" is to comprehensively process the data of each detection part at the end of the analysis. The truth and accuracy of each other's data are demonstrated by the scientific and

existential nature of the test data. The data between the respective detection sections can be mutually demonstrated. The specific method of argumentation will be analyzed in detail in the discussion of the algorithm later.

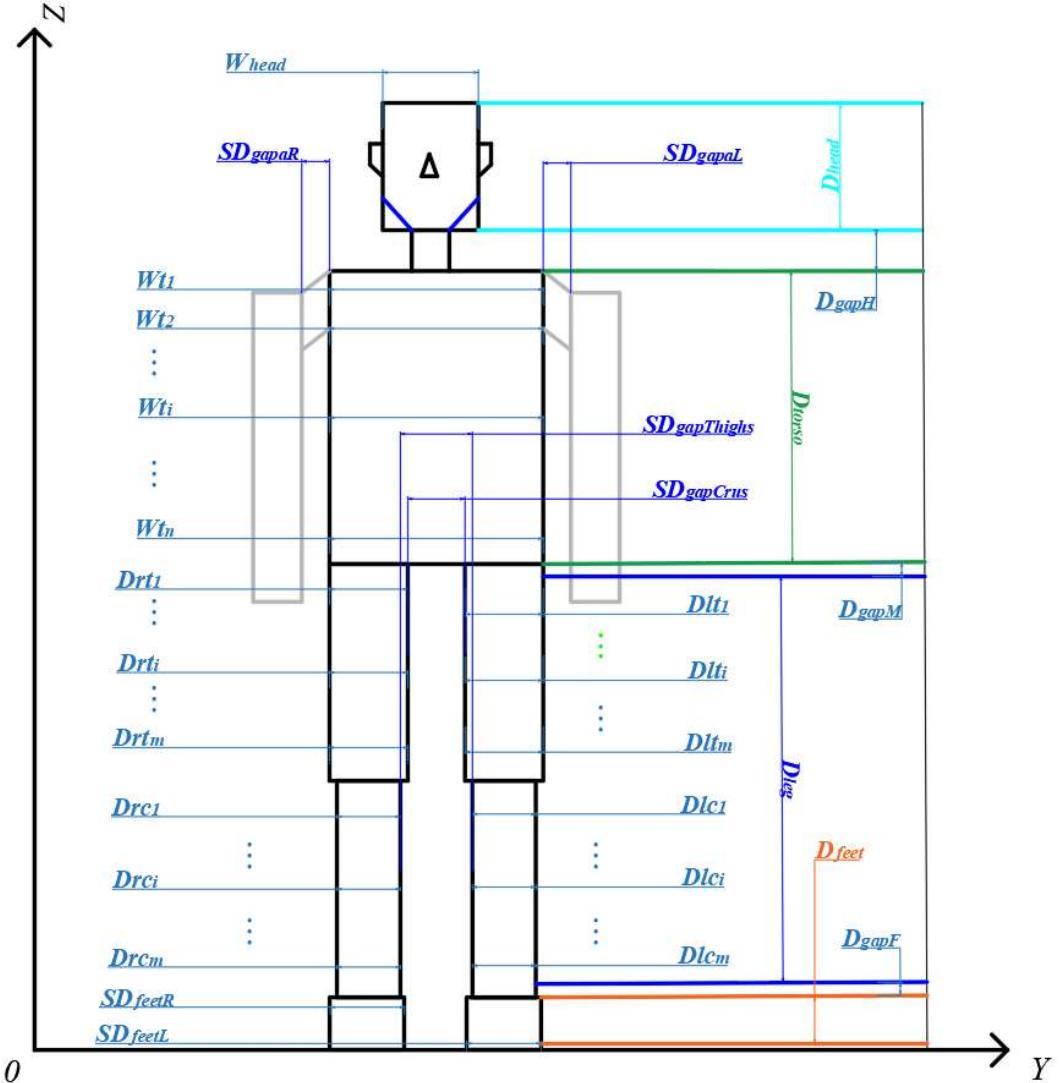


Figure 36. Partition of the target person detecting the local level.

The following authors will discuss why Subsections Modeling Thought should be introduced into the robot algorithm design of this project. As shown in FIGURE 32, the robot performing the Front-Following task must always maintain the action of detecting the target person to acquire the motion data information of the target person in real time. The robot then predicts the future path of the target person based on the acquired detection data to better perform the Front-Following task. Most of the research on Front-Following technology now obtains test data by detecting a single part of the human body. Some papers even try and find the most suitable human body part for detection. The authors believe that finding the most suitable human site for testing is a good experimental protocol. However, in actual experiments or applications, no human

part can fully reflect the walking state of the target person. Regardless of the location, there are always shortcomings or errors in one way or another. Moreover, no part is efficient and universal for the detection of all user usage scenarios, which will be mentioned in detail later in the algorithm analysis. Therefore, the author proposes a full-body detection of the target person in order to obtain the most accurate test data. The author divides the body of the target person into four detection areas, which are proposed through a lot of in-depth analysis and research on the human walking movement. The four detected areas divided by the author are very representative, and each area has a specific relationship with human walking activities, which can provide useful detection data for experiments and practical applications. When Subsections Modeling Thought is adopted, the experiments can reduce the detection error by correlating complementarity, expand the applicability of the algorithms to different target persons, and enhance the robustness and stability of the overall algorithm model. The algorithm analysis section that follows will continue to analyze and discuss this issue in depth. (NB: The target person's arms are not classified into the detection area. The specific reasons will be analyzed in detail later. In the extension part, the author will introduce the detection algorithm of the two hands to detect the specific user's movements, so that the robot can analyzes the user's status and intent, and make relevant response actions basing on the detection data.)

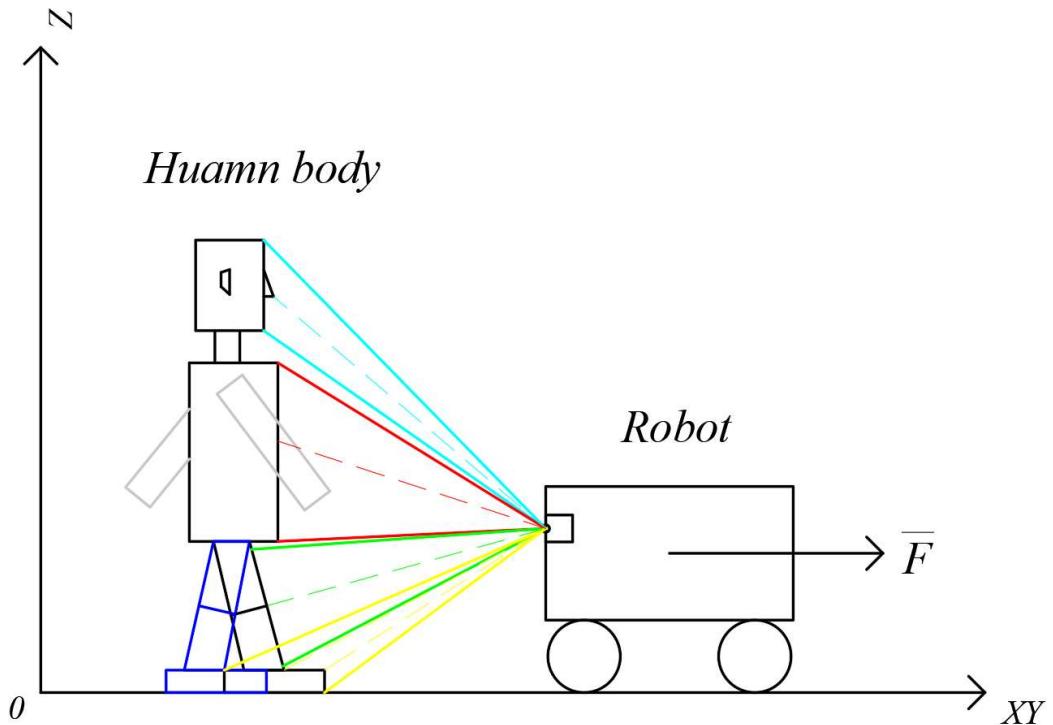


Figure 37. Detection partitioning according to different parts of the target person body.

Head :

The first part is the head. In the process of human walking, the human head can reflect a lot of information. However, there are few studies on human walking in the detection of human head. Part of the reason is that human head detection is susceptible to external interference and requires excellent algorithms to obtain some good calculation results. This is mainly because humans rely mainly on the eyes to obtain information on walking and road conditions while walking. Because most people will look at the road when they walk, that is, when people walk, they usually observe the road. If a person is not paying attention to the road while walking, then that person will be in a dangerous state. Although humans are not always staring at the road during walking, they spend most of their time looking at the front. The eyes are located in front of the human head. Due to the physiological structure of the human eye, when humans observe an object, under normal circumstances, humans will habitually observe the observed object positively. Therefore, in the process of human walking, the direction of the human head and the direction of human gaze are often in the same direction, and this direction is generally the direction of the destination position that humans will travel to. Therefore, the detection of the human head can provide important reference information for predicting the future walking path of human beings. The report will detail the related algorithms for detecting human heads in this project.

Torso :

The second part is the human torso. The human torso has been used as a detection site for robust human modeling part in many researchers' experiments. Some researchers have found through research that the human torso is used as the detection site, and the detection of the human torso will result in more accurate detection information. This is because when humans are walking, the human torso is relatively stable with respect to the constantly moving legs. The author will detail the related algorithms for detecting human torso in this project.

Legs :

The third part is the human legs. Human legs are robots that follow the most studied human body parts in human-related research. Because human legs can directly reflect the gait of human walking. So the study of human legs is easy to integrate with human gait-related knowledge. On the other hand, human legs play an important role in human walking. Although the feet of the human beings are finally landed, the human feet are largely restricted by human legs. By detecting human legs, researchers can get more accurate speed and acceleration when walking. In the following sections, the author will detail the related algorithms for human legs in this project.

Feet :

The fourth part is the human feet. Some readers may be wondering why the research on this part of the project has already tested the human legs. Why do the

researcher in this research want to detect the human feet? This is because human legs can directly reflect the gait and walking direction of human walking, especially when the human feet have landed. However, the detection of the feet of the human body is not easy, and the volume of the human feet is the smallest part relative to other detection parts of the human. That is to say, the detection of human feet may bring more errors, so why should the author insist on the detection of human feet in this project report? This is because, in terms of the error caused by the small size of the human feet, the human feet can provide a more accurate indication of the direction of human walking. Under normal circumstances, when humans walk, although the human legs are lifted, it is mainly the human legs that can reflect the direction of human gait. The pointing of the human feet in the raised state can provide a more direct indication of the walking direction, and the final landing detection part is the human feet. The status of the human feet mainly studied in this project is also the status of the human feet when they land. This is because the movement of the human feet at the time of landing is stable, and generally does not have too much offset action. Moreover, even if have an offset action after landing, in general, the feet also have important analytical significance for the human walking. In addition to the human feet have a more stable motion status after landing, the human feet are convenient to be detected, and have important significance of change and offset motion for the human walking. Another advantage of detecting human feet is that, in general, when humans are walking, a foot corresponding to a certain leg of human being points in the same direction as the tangential velocity of the corresponding leg. It is possible for human legs to perform various actions before landing, and even some actions may mislead the robot to make a wrong judgment. For the feet that have already landed or the feet that have not landed, their direction of movement is generally the same as the direction of human walking. When calculating, the direction of the moving speed of the human feet can be used as the tangential speed direction of the human walking path. And because of gravity, the feet will be subjected to pressure from human gravity and the support force and friction force from the land. If there is not important decision of the human happening, the direction of the feet will not change in most cases. Moreover, due to the physiological inertia and structure of humans, the feet are usually pointed at the tangential velocity of the human walking path before the humans walk landing normally. This is because normal humans do not walk on crooked feet deliberately while walking, because such a movement pattern of anti-human physiological structure will consume a lot of physical strength, and the whole movement will be very not flexible due to being restricted by the human physiological structure. Some serious situations can even cause humans to fall due to unstable gravity center and are not ergonomic. Therefore, human feet are an important indicator of the future path of human walking whether it is when the feet leaving the ground or landing. Therefore, the detection of human feet has important detection and prediction significances.

Head & Torso :

As shown in FIGURE 36, the body of detected target person is divided into target detected areas from top to bottom along the direction of the Z axis. The detection area corresponding to the human head is D_{head} . When detecting the human head, the widthwise width of the human head is defined as W_{head} , which is an important algorithm parameter. D_{torso} represents the human torso detection area. It can be seen that there is a D_{gapH} area between the D_{torso} area and the D_{head} area. D_{gapH} is called a separated area and the data analysis system will not collect the detection data of the interval. Researchers can think of a separated area as a data buffer or data partition. For the D_{torso} area, the LRF does not make a one-time scanning of the entire area, but divides D_{torso} into a number of small areas and then scans these small areas and then collects the useful detected data. Each area is scanned widthwise. The data of the i -th widthwise scan is the scan data obtained after widthwise scanning of the i -th small area, and is recorded as Wt_i . It is assumed that the detected area is divided into n small areas in total. All the detection data of the n widthwise scans are centrally collected and assigned to the set SD_{Wt} . (N.B.: Wt_i is a data set scanned at one time. I.E.: The SD_{Wt} data set is composed of multiple scanning data sets elements Wt_i at one time.)

$$SD_{Wt} = (Wt_1, \dots, Wt_i, \dots, Wt_n) \quad (68)$$

When the LRF performs a torso scan of the same human, the n is invariant. When the LRF performs torso scans on different humans, the n usually changes. The setting of the constant n is related to the body shape of the target person's torso. The specific setting method for n will be elaborated in the algorithm analysis section.

In the human torso scan, a thinking point that cannot be ignored is the influence of the activities of human arms on the accuracy of the scanning detection. Human arms may cause large errors in human detection. Because during the walking of human beings, the human arms are only an additional assistance, and the human arms are not necessarily in the state of assisting the human legs when walking, it is likely to be holding other things, or in the pocket and so on. The detection of the arms in these states does not help to analyze the walking state of the human being, and may even interfere with the nature of the human walking state. Therefore, the detection of the human torso set in this project does not include the detection for human arms. The latter algorithm will show how to remove the error interference from the arms. In addition, the author set the detection interval of human left and right arms and torso as SD_{gapL} and SD_{gapR} , respectively. In practical applications, it is not always possible to detect human SD_{gapL} and SD_{gapR} . Therefore, the scanning torso algorithm based on this project must have good classification processing ability to deal with the problem of correctly detecting the target person's SD_{gapL} and SD_{gapR} . If human SD_{gapL} and SD_{gapR} can be detected, it will greatly assist human torso scan.

Legs:

In the order of the scan analysis, the next part to analyze is the human legs which under the human torso. The detection area of human legs is called Dleg. Human legs are more specific and more closely reflect the human walking state than the human torso. However, in many cases, human legs do not necessarily exhibit a detected state of separation. However, if the legs are detected in a detected state that is not separated by two, an error will be caused in the detection accuracy of the detection model. For example, when the target person is walking in a long skirt, the LRF will have difficulty capturing the exact movement of the legs. This is one of the reasons why the author introduces the modeling of multi-site detection in human body (i.e.: if the detection model of the robot is only to detect the legs of the target person, then when the target person is wearing a long skirt, the detection algorithm of the robot will not be possible to obtain useful detection data of the legs of the target person. Correspondingly, the robot's prediction model will not make a very accurate prediction of the future path of the target person.). When it is not possible to accurately detect the user's legs, it is necessary to perform assistance judgment processing by using detection data of other detection parts of human beings.

Based on the actual analysis, when modeling the legs of the target person, researchers should also consider the existence of some special user groups. For example, a target person with a disabled leg. For this group of people, their walking gait may be very different from that of healthy human legs. How to identify these differences and formulate a good Front-Following and prediction strategies, instead of regarding them as external errors or the wrong actions of the target person, will be the content of the author's discussion in the algorithm analysis.

In the algorithm for detecting human legs, the author further subdivided the Dleg detection zone into four detection sections. Because the author considers that when humans walk, due to the characteristics of human bones and physiological forms, if the human leg detection area is divided according to the thigh and the calf, the human legs can be regarded as composed of four main parts: left thigh, left calf, right thigh and right calf. As shown in Figure 36, along the positive to negative direction of the Z-axis (i.e., top-to-bottom direction), the authors set up four data sets to store scan data for human legs:

1. A data set for storing left thigh scan data:

$$SD_{Drt} = (Drt_1, \dots, Drt_i, \dots, Drt_m) \quad (69)$$

2. A data set for storing left calf scan data:

$$SD_{Drc} = (Drc_1, \dots, Drc_i, \dots, Drc_m) \quad (70)$$

3. A data set for storing left thigh scan data:

$$SD_{Dlt} = (Dlt_1, \dots, Dlt_i, \dots, Dlt_m) \quad (71)$$

4. A data set for storing left calf scan data:

$$SD_{Dlc} = (Dlc_1, \dots, Dlc_i, \dots, Dlc_m) \quad (72)$$

In the above data set, $Dabc$ ($a=l$ or r , $b=t$ or c , $c=1, 2, \dots, m.$) represents the detection data set obtained after the c -th horizontal scan of the LRF recorded in the data set. I.E.: $Dabc$ is a one-time scan data set the detection containing data from multiple scan data points scanned by the LRF during a horizontal scan. In addition, considering the existence of gaps between human legs, the author defines two data sets: $SDgapThighs$ and $SDgapCrus$, which are used to indicate the detection data of the target person's thigh interval and calf interval, respectively.

1. A data set for storing thigh interval scan data:

$$SD_{gapThighs} = (D_{lT}, \dots, D_{lTi}, \dots, D_{lTm}) \quad (73)$$

2. A data set for storing calf interval scan data:

$$SD_{gapCrus} = (D_{lC}, \dots, D_{lCi}, \dots, D_{lCm}) \quad (74)$$

In the above data set, $DlAb$ ($A=T$ or C , $b=1, 2, \dots, m.$) represents the detection data set acquired after the b -th horizontal scan of the LRF recorded in the data set. I.E.: $DlAb$ is a one-time scan data set containing the test data of multiple scan data points scanned by the LRF during a horizontal scan.

The algorithm for human legs can be based on the four main parts and four data sets, combined with gait analysis, to set the corresponding detection algorithm. The specific algorithm will be explained later.

Feet:

Next is the detection of human feet. The detection area for humans feet is set to $Dfeet$. The area interval between the human feet and the leg detection area is set to $DgapF$. The author also sets up two data sets to store the test data of the $Dfeet$ area:

1. A data set for storing left foot scan data:

$$SD_{feetL} = (D_{fL1}, \dots, D_{fLi}, \dots, D_{fLm}) \quad (75)$$

2. A data set for storing right foot scan data:

$$SD_{feetR} = (D_{fR1}, \dots, D_{fRi}, \dots, D_{fRm}) \quad (76)$$

In the above data set, DfAb (A=L or R, b=1, 2, ..., m.) represents the detection data set acquired after the b-th horizontal scan of the LRF recorded in the data set. I.E.: Dfbc is a one-time scan data set containing the test data of multiple scan data points scanned by the LRF during a horizontal scan.

Another noteworthy thinking point is that since this project is based on cost-effective design principles, the author has only set up two LRFs for each LRF group. In the locked scan mode, the robot calls up to two LRFs in the LRF group to scan the target person, and at least one LRF is called to scan the target person. That is to say, it is possible for the LRF group to scan the target person whole body with only one LRF, which puts higher requirements on the design of the scanning algorithm. To solve this problem, the author proposes the concept of detecting rectangles. In addition, in order to enable the LRFs to perform stable detection for each part of the target person. The author proposes a method of step-by-step scanning, and the details of the specific algorithm will be explained later.

ALGORITHM AND MODEL FOR DRF

Detecting Rectangular Frames

In order to facilitate the following description of modeling and algorithms, in this research report, if there is no special explanation, the author define: The DRF mentioned in this report is the abbreviation of Detecting Rectangular Frames. If there is a “DRF” that represents other meanings, the exact labeling instructions will be given. Simply put, DRF is a three-dimensional scan space that encloses the target person.

As shown in Figure 39, the entire DRF area is mainly divided into three areas: Area1, Area2, and Area3. Area 1 is called the bottom scan area, Area 2 is called the body scan area, and Area 3 is called the top scan area. (NB: Figure 46 shows the three-dimensional DRF detection area in the form of a two-dimensional figure. The actual DRF is a three-dimensional rectangular frame space. As shown in Figure 38, Area3 and

Area2 should theoretically be in the same perpendicular to XY. The cube area of the face. Area1 should be a cube area parallel to the plane of the support surface (i.e.: parallel to the Z axis.). Area2 is subdivided into two areas of Area21 and Area22. Area 1 is called the bottom area of the DRF, and the width of the bottom area of the DRF is represented by Dd. Area 2 is called the real scan area of the DRF. Area21 is called the central scanning area. The scanning data of Area21 can best reflect the movement information of human walking, including: head, torso, legs, and feet. Area 22 is called the side scan area. In Area 2, the area on both sides of Area 21 is Area22. The Area 22 area mainly performs the detection of the two-legged movement of the target person and the movement of the two legs and the movement of the two feet. We define the lateral width of Area 2 (IE: parallel to the XY plane) by the width of the distance between the left and right ends when the target person horizontally spreads the hands horizontally, and define the longitudinal width of Area 2 with the height of the target character standing perpendicular to the support surface (i.e.: parallel to the Z axis). Therefore, the width and height of the Area 2 of the DRF are related to the size of the target person.

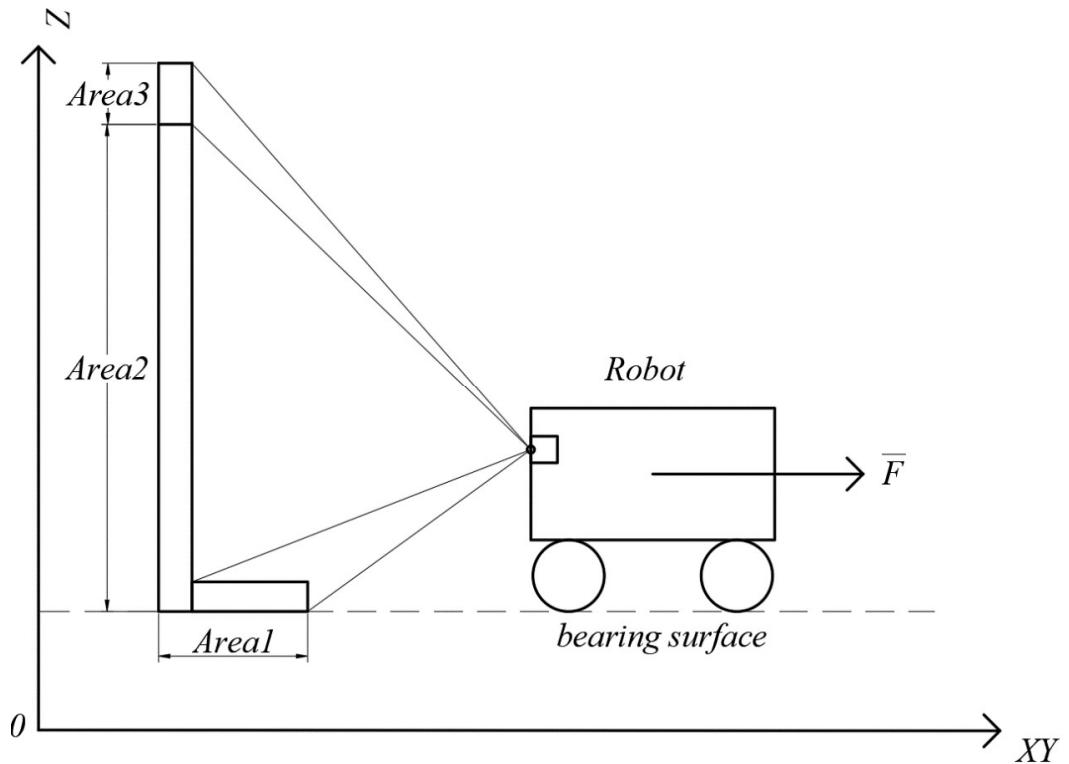


Figure 38. Relative distribution of Area1, Area2 and Area3 of RDF in 3D controls.

Generally speaking, when the same robot performs different Front-Following tasks, the body formation data of the target persons followed by the robot are also different. Therefore, correspondingly, the DRF established by the robot performing different Front-Following tasks will be different. These differences are mainly reflected in Area2. This is because, among the three detection areas of the DRF, only Area 2 reflects the

body type information of the target person. In other words, only Area2 is related to the figure characteristics of the target person. Therefore, Area2 is also the most important detection area in DRF.

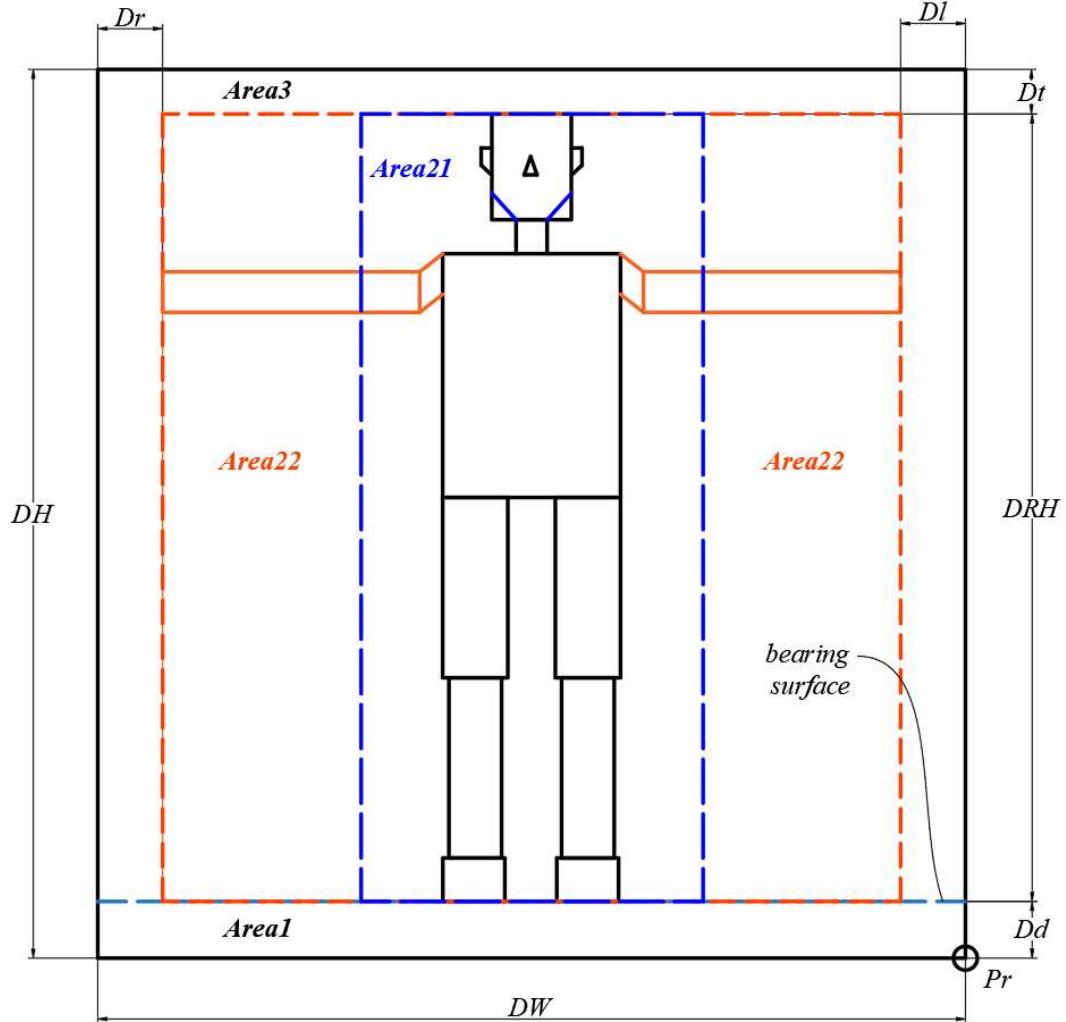


Figure 39. Division of the standard scan partition of the DRF.

Why is the height of Area2 of the DRF set to be the same as the height of the target person in this project, and the width of Area2 is the same as the width of the target person when the arms of the target person are opened? This is because the author considers that when a person is walking, or when the person is standing, if the person does not leave the ground without jumping, then the range of motion that the human can do will not exceed a rectangular space. The width of the rectangular space is determined by the maximum distance between the ends of the left and right ends when the human arms are horizontally opened. (IE: Due to human physiological structure, when the human arms are opened to maximize the distance between the two ends of the left and right arms, the human arms should be parallel with the support surface. Moreover, at this time the open arms of the human beings It should be in the same

straight line.) Therefore, under normal circumstances, the Area2 detection area can include all the actions that the target person takes while walking.

In view of the above analysis of the area setting of Area 2, a key point worthy of attention is that humans generally do not always keep their arms open when they are walking normally, because if they keep walking with open arms, these space-occupying actions will make people spend more work (i.e.: costing more energy.). Another important reason is that if the target person has been walking with open arms, these space-occupying actions may interfere with the walking of the passers-by, which will also affect the normal walking of the target person. When walking, the target person's more inclined action is that the arms naturally hang down and swing, which is also in line with the physiological structure of human beings, and is also in line with the physiological inertia of human beings. As shown in Figure 40. After doing the above thinking, a very obvious and important point of thinking will come out. Since the target person is walking, most of the actions that the target character can take are concentrated on Area21, so why should the author set Area22? Does adding Area22 not burden the working load of LRF and the mechanism for collection and processing of detection data? What this report can make clear here is that the author's main purpose in setting up the Area22 detection area is to build a robust human motion detection model. In fact, if the scanning area of the LRF is only strictly limited to Area21, then the author can indeed simplify the detection algorithm of Area2, but the optimization effect on the mechanical control burden of LRF is not obvious. Or, such a design may even increase the workload of the LRF. This is because, in conjunction with the discussion and analysis of LRF mechanical control at the beginning of the previous research report, this project report gives the following point of view: the control of the smaller detection area means that the angle of the round-trip scanning of the LRF is smaller. That is to say, the ability to accurately control the LRF is increased, and the angular velocity of the round-trip scanning is constant, so that the frequency of the round-trip is increased, which adds control loss and burden to the LRF to a certain extent. Another important reason is that if the researchers set the area of Area2 to a small size, such as just to be able to include the human body when the arms are drooping, this will likely cause some detection errors. Because the target person does not always maintain the most formal walking posture while walking, it is very likely that the target person will make some strange or little action related to walking detection while walking. EG: Sideways toward the robot, bending over to pick a certain item, or doing some long walks, etc., these actions will cause the original Area 2, which is set according to a small area, to not include all the important detection parts of the target person. The detection system then misses some detection actions that are likely to be very important, thus causing the detection errors.

In order to solve this problem and consider saving computing resources, the author established the partitioned Area2. Dividing Area2 into two areas, Area21 and Area22, can not only include all of the motions of target person's whole body that can affect or reflect human walking information during the walking process, but also the detection data of the two areas. What is more, by setting these two detection areas, the robot

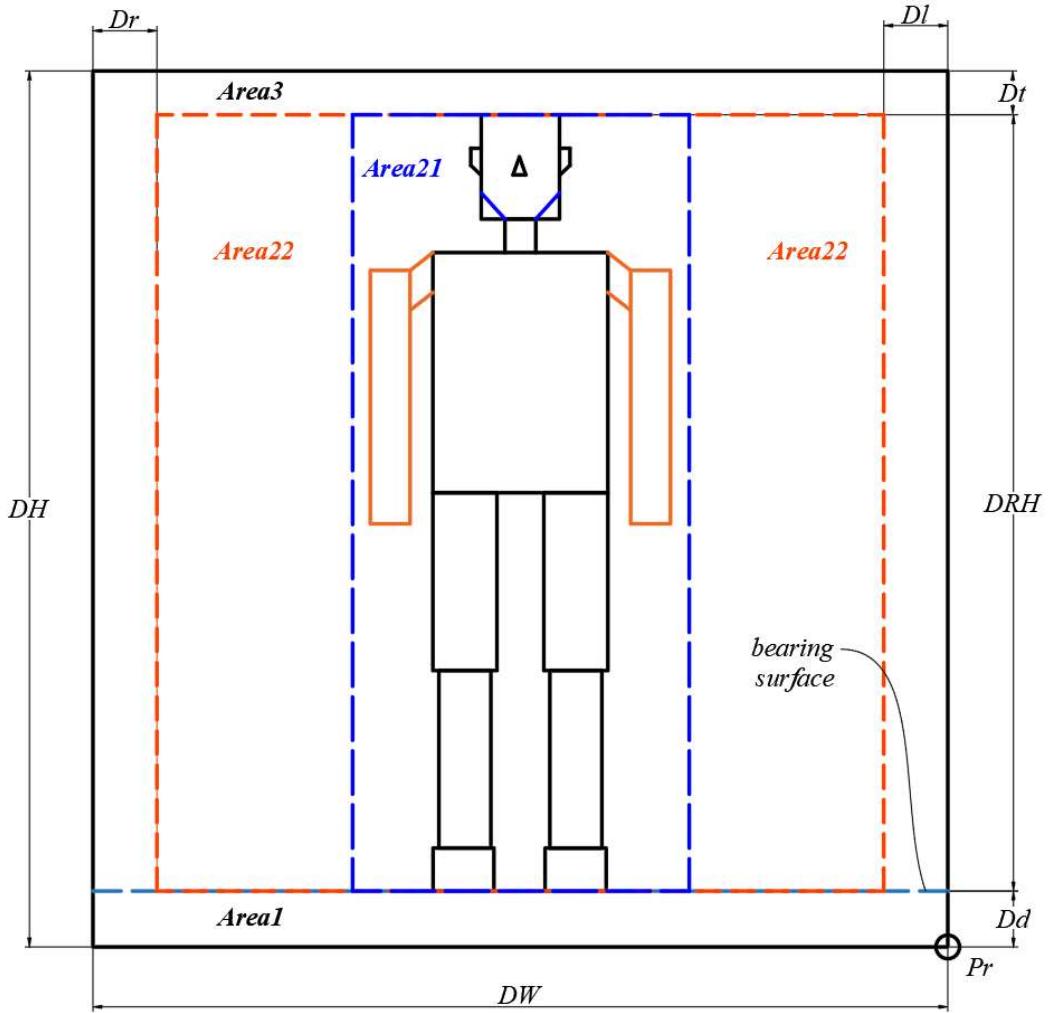


Figure 40. The scan partition of the DRF.

processing system can use different processing algorithms to optimize the resources and obtain more useful detection data, so that more detection functions will be available. The specific processing algorithms will be described later.

When establishing the DRF, the algorithm processing system of the robot will set different Area2 widths and heights according to the shape of different target people to set different Area2. The widths and heights of the Area1 area and the Area3 of the DRF are generally set fixedly. Because the target person is not recommended to perform some activities as like as picking up the toes when receiving the DRF modeling scan. It means that the value of the target person's height is relatively fixed. The DRF algorithm proposed in this project report has an automatic adjustment function. In the advanced function design of the detection algorithms, the author will add an adjustment function for the setting of the width of Area1 to make the DRF detection algorithms more robust. In a further algorithm design, the author will determine the maximum step size of the target person by scanning the side of the DRF to determine the maximum width of the Area1 area.

DRF modeling process

First, before the robot performs the following task, the target character will be required to stand behind the robot and face the robot face, as shown in FIG 39. The target person's arms are wide open, the two arms are on the same straight line, and are horizontally parallel with the support surface. (N.B.: For the sake of convenience, the “ground” appearing in this report also belongs to the category of the support surface that represents the support surface for broad sense.) The target person stands behind the robot, and the distance of the target person from the LRF group that directly front facing him or her is DH.

When the robot accepts the following task, it will firstly perform DRF modeling on the target person to be followed to obtain the real-time overall shape data information of the target person. As for the position of the target person to be located far from the robot, there is no mandatory requirement. An alternative solution is mounting an LED light on the robot or mounting a laser that emits visible light for emitting a pointing cursor on the ground to assist the target person in standing at the appropriate location to coordinate with the robot for DRF scanning. Of course, such a design may affect the user experience and make the target persons feel that they are constrained by the robot. In order to solve this problem, the author proposes a more intelligent algorithm, so that the target person does not need to be in a specific position, and the robot can get good DRF scan performing results, and the specific algorithm will be given out in the following algorithm section.

At the beginning of modeling, the robot's algorithm execution system will call the LRF control algorithm in the robot's control algorithm system. When the LRF control algorithm is executed, the two LRFs in the LRF group (i.e., generally located in the LRF group at the rear of the robot) facing the target person are automatically mobilized to perform DRF detection scan on the target person. The specific scanning process is as follows:

1. *The LRF used for scanning is first rotated such that the ranging beam is emitted in the negative direction of the Z-axis. Eventually the LRF probe beam is perpendicular to the support surface and parallel to the Z axis. Let the angle between the LRF probe beam and the Z axis be Θ , then $\Theta=0$ degrees.*
2. *The LRF will then support the ground for scanning along the direction in which the corners increase. When $\Theta=\varphi_0$, the LRF starts scanning the bottom region of the DRF. I.E.: Start scanning the area represented by Area1 in Figure 39.*
3. *When the LRF scans Area1, there is $\Theta=\varphi_0+\varphi_1$, then the corner continues to increase, and the LRF starts scanning the body area of the DRF. When $\Theta=\varphi_0+\varphi_1$, the LRF will scan the human body that is perpendicular to the ground along the direction in which the Θ increases. I.E.: Start scanning the area represented by Area2 in Figure 39.*

4. When the LRF scans Area2, $\Theta = \varphi_0 + \varphi_1 + \varphi_2$, and then as the corner continues to increase, the LRF starts scanning the top area of the DRF. At this time, $\Theta = \varphi_0 + \varphi_1 + \varphi_2$, and then the LRF will continue to scan the human body perpendicular to the ground along the direction in which the horn increases. I.E.: Start scanning the area represented by Area3 in Figure 39.
5. When the LRF scans Area3, $\Theta = \varphi_0 + \varphi_1 + \varphi_2 + \varphi_3$, then the LRF starts to rotate in the opposite direction, the corner angle begins to decrease, and the LRF starts scanning the top area of the DRF again. At this time, $\Theta = \varphi_0 + \varphi_1 + \varphi_2$, LRF will continue to scan the human body perpendicular to the ground in the direction of the reduction of the corner. I.E.: Start scanning the area represented by Area3 in Figure 39.
6. When the LRF scans Area3 again, $\Theta = \varphi_0 + \varphi_1 + \varphi_2$, and then as the corner continues to decrease, the LRF starts scanning the body area of the DRF again. At this time, $\Theta = \varphi_0 + \varphi_1 + \varphi_2$, LRF will continue to scan the human body perpendicular to the ground in the direction of the reduction of the corner. I.E.: Start scanning the area represented by Area2 in Figure 39.
7. When the LRF scans Area2 again, $\Theta = \varphi_0 + \varphi_1$, and then as the corner continues to decrease, the LRF starts scanning the bottom area of the DRF again. At this time, $\Theta = \varphi_0 + \varphi_1$, the LRF will continue to scan the corresponding area on the ground in the direction in which the corner is reduced. I.E.: Start scanning the area represented by Area1 in Figure 39.
8. When the LRF scans Area1 again, $\Theta = \varphi_0$, and the LRF group has completed a scan cycle. That is, a normal DRF human body forward modeling scan cycle is completed.

Repeat the above process twice, transfer the collected detection data to the robot's detection data processing program, and start data processing.

When the robot completes the DRF forward modeling of the target person's human body, the detection data of the D0 phase is recorded, but when the data processing is performed, the data of this part is removed. Therefore, the true DRF human body forward modeling only completes the scan of the scan angle from $\Theta = \varphi_0$ to $\Theta = \varphi_0 + \varphi_1 + \varphi_2 + \varphi_3$, and then completes the reverse direction from $\Theta = \varphi_0 + \varphi_1 + \varphi_2 + \varphi_3$ to $\Theta = \varphi_0$. The author turns the Θ -enhanced scan into an up scan and the Θ -reduced scan into a down scan. The scanning process, which will complete the scan up and then the scan down, is called completing one scan cycle. That is, the DRF human body forward scan in this project only sets two upper and lower scan cycles. In addition, the researcher or designer can also increase the number of cycles of DRF human body up and down scanning to obtain more detection data, thereby improving the reliability of modeling data. Of course, this will lengthen the time the target person performs the DRF scan and may affect the target person's experience with the robot.

Why do the authors think that two positive scans of the DRF is enough? This is because, in general, the target person will not prohibit the 100% action when the robot performs DRF scanning, that is, the more times the DRF scan is performed, the greater and more the difference in the detected data may be. Of course, some researchers can reduce errors through good algorithms, but they also need to spend computing resources and time of the robot. On the other hand, in order to make the robot have a more comprehensive understanding of the target person and establish a more practical human model, the robot of this project has set up DRF lateral scanning and DRF back scanning in addition to DRF forward scanning. The specific related modeling process will be explained later. The above analysis is based on the premise that the LRF operating speed is general. Of course, if the control accuracy of the LRF is high and the speed is very fast, this report suggests that the number of up and down scan cycles of the DRF can be appropriately increased.

In the discussion above, this report describes the general flow of DRF scanning. The above analysis is mainly based on the change of the scanning angle Θ . That is, the above scanning modeling determines the rotational motion flow of the LRF in a plane perpendicular to the XY plane. Then, another important question worth considering is how to set the scanning angle or scanning range of the lateral scanning of the LRF in the rotational motion with the Z axis as the rotation axis. This also relates to the question of how much DH and DW should be set.

Preliminary analysis of the DRF human body scanning modeling algorithm

1. First, after receiving the following task, the robot performs a pre-position scan on the three rectangular areas to determine the location of the target person. After determining the location of the target character, the robot will call the steering and position alignment algorithm to control the robot to move to the position that is positively aligned with the target character. Or the robot will issue a prompt to make the target person adjust his position or turn. The end result is that the target person is in the opposite direction of the LRF for the DRF modeling scan. After the LRF is positively aligned with the target person, the DRF pre-positioning algorithm is completed and the robot will begin a formal DRF scan. After the DRF pre-positioning algorithm is executed, the robot will obtain the detectable width of the target portion of the two-legged portion of the target character, defining the detectable width as WF. Define the initial width of the human rectangle to be WH. Then the robot will mobilize the LRF towards the target person, and start scanning for the DRF modeling scan of the target person from $\Theta=0$. The initial width of the scan of the DRF is set to WSET. WSET is a preset scan width value that provides a preset average scan width. N.B.: The width DW of the last DRF is not necessarily equal to WSET. The DRF scanning algorithm

will be detailed in the later algorithm part.

2. Then as the Θ gradually increases, as shown in Figure 41, the LRF starts scanning the D0 area + Area1. The reason why it is called "D0 area + Area1" is because the robot does not know the exact width of the D0 area in this process, so the area of the support surface cannot be divided. Only when the robot scans the feet of the target person, the width Dd of Area1 can be determined according to the algorithm. Therefore, the robot controls the LRF to scan the support surface with the horizontal scan width WSET, and scans until the target person's foot is scanned. Once the LRF scans to the foot of the target person, the detection algorithm automatically divides the scan area into the D0 area and the Dd area according to the width of the "D0 area + Dd area". The division algorithm is as follows:

- When the LRF performs the support surface scanning, there is already a preset width Dd of the Area1 area, which can be preset by the engineer or the user, or automatically optimized and updated by the two-foot detection algorithm during the walking of the target person. The specific optimization and update algorithms will be described in the two-foot scan algorithm;
- The LRF scans along the support surface and scans until the LRF detects the target's feet. Once the LRF detects the feet of the target person, the detection algorithm records the distance from the start position of the support surface corresponding to $\Theta=0$ to the position of the target person's feet, which is recorded as Ds. Let the width of the D0 area be D0, that is:

$$D0 = Ds - Dd \quad (77)$$

- As shown in Figure 41, the rectangle in Figure 41 represents the human foot space. Obviously, the target person's feet space has two substantive performances. The first type is the separate feet that can be detected. In these circumstances, the target person wears trousers or shorts, or a short skirt and other clothing that can make the target person's feet exposed, so that the LRF can detect the target person's feet. Generally speaking, the target person's feet are wearing shoes, but whether or not the shoes are worn does not have a great influence on the LRF's scanning detection of the target person's feet. In general, if the target person wears a shoe with a relatively fixed shape, it will help the LRF scan and detect the target's feet. Another substantive manifestation of the target person's feet space is the feet that cannot be directly detected. This is usually the case when the target person wears a long skirt that touches the ground, or other clothing that can cover his feet. For these two cases, it will be specifically described in the pre-positioning algorithm for DRF.
- Although the target person is required to stand in front of the robot before the DRF body modeling scan, due to individual special reasons, sometimes the target person is not strictly facing the DRF body modeling scan opposite and standing on the

back of the robot. In this case, whether the robot is to continue scanning or other corrective actions is what this report will cover next. Obviously, if the robot continues to scan, then the modeling scan of the target person's DRF will lose some scanning accuracy and accurate scan data. Therefore, the author suggests that the robot should perform corrective actions when the target person does not strictly face the LRF group facing the robot. The target person is placed in the forward orientation of the robot's LRF group.

First of all, to deal with this problem, the author must first clarify one thing, that is, let the robot know whether the target person is in the front radial position of the LRF group. To solve this problem, combined with the analysis of Figure 39, Figure 40 and Figure 41, the author defines the following concepts:

- ***DRF Scan Rectangle:*** A projection of the abstract cubic space of the DRF modeling area along the direction of the LRF beam. The projection surface is a plane parallel to the Z axis.
- ***Target Person Rectangle:*** A projection of the abstract cube space of the target person along the direction of the LRF beam. The projection surface is a plane parallel to the Z axis.

For the above rectangles, in view of the characteristics of the LRF scan, the author only focuses on the width of the rectangles in the following discussion. The above two rectangles are two-dimensional projections of their respective cube detection spaces. Therefore, in space, the above two rectangles are coplanar. In order to facilitate the presentation, the author will refer to the DRF rectangle, the target person rectangle and the LRF scan rectangle as the second detection rectangle in the following discussion.

- Dimension line: The intersection boundary line between the plane parallel to the Z axis where the target person rectangle is located and the plane of the support surface parallel to the XY plane where the target person stands. The dimension boundary coincides with the detectable bottom edge of the target person's rectangle.
- Baseline: In the DRF scan rectangle, the height of the target rectangle is clearly defined. The author defines a line in the plane of the two-scan rectangle (I.E.: DRF scan rectangle, target rectangle) that is parallel to the dimension boundary and is used to analyze and contrast the width of the two-scan rectangle, called the baseline. In addition, the baseline has the effect of virtual and contrasting the height of the three rectangles.

Next, this report defines the following parameters:

Ac: Take the position point where the LRF is located as the origin, and establish

the RΦΘ scan spherical coordinate system with the side of the rectangle of the robot where the LRF is located as the reference axis. The angle between the position where the LRF is located and the position of the geometric center of the rectangle of the target character is rotated counterclockwise, and the angle formed by the reference axis of the RΦΘ scan spherical coordinate system is called Ac.

Aradial: Set the RΦΘ scan sphere coordinate system with the position of the LRF as the center of the circle and the side of the rectangle of the robot where the LRF is located. Take the position where the LRF is located as the starting point and make a straight line perpendicular to the plane where the two scanning rectangles are located. When the line is rotated counterclockwise, the angle between the line and the reference axis of the RΦΘ scan spherical coordinate system is called Aradial.

Ahumanr: Take the position of the LRF as the origin, and establish a RΦΘ scan sphere coordinate system with the rectangle of the robot where the LRF is located as the reference axis. The LRF probe beam is tangent to the first boundary line of the target person rectangle (ie, the right boundary line) to a tangent point Pt. Starting from the position point of the LRF and ending with the Pt point, a straight line Lt is formed. The angle between the straight line Lt and the reference axis of the RΦΘ scan spherical coordinate system is called Ahumanr.

Ahumanl: Take the position of the LRF as the origin, and establish a RΦΘ scan sphere coordinate system with the rectangle of the robot where the LRF is located as the reference axis. The LRF probe beam is tangent to the second boundary line of the target person's rectangle (ie, the left boundary line) to a tangent point Pt. Starting from the position point of the LRF and ending with the Pt point, a straight line Lt is formed. The angle between the straight line Lt and the reference axis of the RΦΘ scan spherical coordinate system is called Ahumanl.

Adrfr: Take the position point of the LRF as the origin, and establish a RΦΘ scan sphere coordinate system with the rectangle of the robot where the LRF is located as the reference axis. The LRF probe beam is tangent to the first boundary line of the DRF rectangle (ie, the right boundary line) to a tangent point Pt. Starting from the position point of the LRF and ending with the Pt point, a straight line Lt is formed. The angle between the straight line Lt and the reference axis of the RΦΘ scan spherical coordinate system is called Adrfr.

Adrfl: Take the position of the LRF as the origin, and establish a RΦΘ scan sphere coordinate system with the rectangle of the robot where the LRF is located as the reference axis. The LRF probe beam is tangent to the second boundary line of the DRF rectangle (ie, the left boundary line) to a tangent point Pt. Starting from the position point of the LRF and ending with the Pt point, a straight line Lt is formed. The angle between the straight line Lt and the reference axis of the RΦΘ scan

spherical coordinate system is called Adrlf.

Φ_{da} : (deflection angle):

Dr: The width of the space between the human rectangle on the right side of the human and the DRF scan rectangle.

Dl: The width of the space between the human rectangle to the left of the human and the DRF scanning rectangle.

DRF scan pre-positioning algorithm

Before the robot performs a DRF scan, the robot is "unable to visible" where the target person is. So the robot must use some methods to get the initial position of the target person. This follow section of the report will describe the process by which the robot determines the target person by some means before performing the DRF scan. This is called a three-rectangle scan of the target person. Abbreviation: Scanning pre-position.

In order to solve this problem, the author proposes an algorithm flow. The author sets three scan pre-positioning areas for an LRF, which are represented by three rectangular cube spaces, I.E.: rectangular space AP1DO1, rectangular space P1P2O2O1, rectangular space P2DO2E. For convenience of explanation and discussion, the author defines the rectangular space AP1DO1 as the left rectangle, the rectangle space P1P2O2O1 as the middle rectangle, and the rectangle P2DO2E as the right rectangle.

For these three cube spaces, the authors represent them in a two-dimensional map with their respective topped rectangles. In fact, the LRF scan of the three rectangles are three scans of the corresponding areas of the support surfaces represented by the three rectangles, correspondingly. As shown in Figure 43, Le represents the edge of the robot rectangle where the LRF is located. The author defines that the volume of the rectangular space in the middle is the largest, and the three rectangle spaces extend along the radial direction of the LRF perpendicular to the Le side, correspondingly. The author takes the middle lines of the three spatial rectangles to represent the extension direction of the corresponding space rectangle, correspondingly. I.E.: The extending direction of the rectangular AP1FO1 is OL, the extending direction of the rectangular P1P2O2O1 is OC, and the extending direction of the rectangular P2DO2E is OR. The position and distribution of these three spatial rectangles are specially considered. For example, the rectangle AP1FO1 and the rectangle P1P2O2O1 have the same position points P1 and O1, and the rectangle P2DO2E and the rectangle P1P2O2O1 have the same position points P2 and O2. The purpose of the author's setting is to construct a closed detection area to ensure that no dead spots are detected in the detection area

consisting of three rectangles. N.B.: The detection area here is mainly divided by the support surface area where the target person is located.

Another question worth considering is, why does the author designs the detection rectangle into a three-rectangular merged space? This is because the author believes that in the DRF human body modeling scan, the detection space area merged by the three rectangular spaces can include the position where the target person stands as much as possible. If the target person does not stand in the three rectangular space area, then the robot will think that the target person is not willing to accept DRF body modeling scan, and the system will automatically stop the DRF scanning process. That is to say, the robot set in this report allows the target person to stand in front of the radial direction of the LRF without a very standard when performing a DRF modeling scan. This is because the robot's control algorithm automatically adjusts the positions of the robot and LRF group to meet a more precise DRF modeling scan for the target person. However, if the target person deviates too far from the robot's scanning area, the control algorithm will consider that there is no target person in the detection area, and the robot will automatically stop the subsequent DRF modeling scan operation. The robot is used to determine whether or not the target character is in the three-rectangular spatial region, to define and divide whether there is a target person, and then to determine whether to take the further control algorithms.

The specific algorithm for the three rectangle scan pre-positioning

First, as shown in Figure 44, define the following parameters:

WSETL: the scan width of the left rectangle in the three rectangles;

HL: the maximum extension length of the left rectangle in the three rectangles when scanning by the LRF;

WSETC: the scan width of the middle rectangle in the three rectangles;

HC: the maximum extension length of the middle rectangle in the three rectangles when scanning in the LRF;

WSETR: the scan width of the right rectangle in the three rectangles;

HR: the maximum extension length of the right rectangle in the three rectangles

when scanning in the LRF;

SL: *scan start point of the left rectangle;*

O: *the origin of the RΦΘ scan spherical coordinate system corresponding to the LRF for scanning;*

OL: *the extending direction of the left rectangle in the three rectangles, that is, the radial direction;*

OR: *the extending direction of the right rectangle in the three rectangles, that is, the radial direction;*

Le: *The line corresponding to the polar axis of the RΦΘ scan spherical coordinate system of the LRF is also the line where the edge of the robot rectangle where the LRF is located;*

Or: *in the RΦΘ scan spherical coordinate system corresponding to the LRF, with the O point as the origin, the deflection angle of the OR radial line;*

Ol: *In the RΦΘ scan spherical coordinate system corresponding to the LRF, the deflection angle of the OL radial straight line is taken from the O point as the origin.*

After the relevant parameters are defined, do the analysis in conjunction with FIGURE 44 and FIGURE 45, and the execution process of the pre-positioning algorithm is designed and implemented as follows:

1. First, the LRF group scans the middle rectangle because the target person has the highest probability of appearing in the middle rectangle. During scanning, the mechanical control algorithm of the LRF will control the detection of the light emitted by the LRF to perform a folded round-trip scan of the intermediate rectangle.
2. After the LRF scanning the middle rectangular area, if no target person is found, the LRF mechanical control algorithm will control the detection light emitted by the LRF to perform a folding round-trip scan on the left rectangular area. If the target person is detected, the target person orientation processing algorithm is entered to determine the exact position and orientation of the target person. The robot will then implement a DRF orientation and position correction algorithm to correct the relative position and orientation between the robot and the target person.
3. After the LRF scans the left rectangular area, if no target person is found, the LRF mechanical control algorithm will control the detection light emitted by the LRF to perform a folding round-trip scan on the right

rectangular area. If the target person is detected, the target person orientation processing algorithm is entered to determine the exact position and orientation of the target person. The robot will then implement a DRF orientation and position correction algorithm to correct the relative position and orientation between the robot and the target person.

4. After the LRF scans the right rectangular area, if the target person is not found, the DRF scan modeling algorithm will determine that there is no target person in the three rectangular area, that is, the robot will stop the following person who has already started, and will be in the standby state again. If the target person is detected, the target person positioning processing algorithm is entered to determine the exact position and orientation of the target person. The robot will then implement a DRF orientation and position correction algorithm to correct the relative position and orientation between the robot and the target person.

In the above scanning process, the algorithm used by the LRF to scan the rectangle becomes a rectangular folding progressive scanning algorithm. In the elaboration and analysis of the rectangular folding progressive scanning algorithm, the author also combined with the pre-positioning algorithm for analysis.

The determination of the initial scan point of the pre-positioning scan algorithm

The scan initial point of the pre-positioning scan algorithm is a key parameter of the pre-positioning algorithm. Since the pre-positioning scanning algorithm has three scanning rectangles, three scanning initial points must be determined. Referring to FIGURE 44 analysis, the author sets the initial scan point of the middle scan rectangle to point O(x_0, y_0, z_0), and sets the scan initial point of the left scan rectangle to point SL (x_{sl}, y_{sl}, z_{sl}), The scanning initial point of the scanning rectangle on the right side is set to a point SR (x_{sr}, y_{sr}, z_{sr}). The following is a detailed determination of the XYZ coordinates of the three initial scan points:

Calculation:

The scan point located at any position of the middle rectangle is the scan initial point: among the three scan initial points in the three rectangles, the acquisition of the scan initial point of the middle rectangle is most important. The following authors are

illustrated in conjunction with Figures 44 and 45.2. In fact, the XYZ coordinate method for obtaining the scan initial point of the middle rectangle is very simple, and the specific Implementation process is as follows:

1. By calling the LRF control algorithm, the control LRF points to the side of the rectangle perpendicular to the robot where the LRF is located. That is to say, the deflection angle of the RΦΘ detecting spherical coordinate system of the LRF and the LRF is 90 degrees, so that the LRF and the LRF have a zenith angle of 180 degrees, and the LRF detecting light will vertically illuminate the supporting surface. . At this time, the RΦΘ detection spherical coordinate obtained by the LRF detection is the scan initial point coordinate of the middle rectangle. Then, the author calls the coordinate conversion operator to convert the RΦΘ detection sphere coordinates of the scan initial point of the middle rectangle into XYZ following coordinates. Then, by the algorithm described below, the XYZ coordinates of the initial scan points of the other two scan rectangles can be found.

Calculation:

1. Combining the analysis in Figure 44, introduce the parameter definition of the three-rectangle scan pre-positioning algorithm, and calculate the length of OSL and OSR:

$$\begin{cases} OSL = \frac{WSETC \times \cos}{2} \\ OSR = \frac{WSETC \times \cos\theta l}{2} \end{cases}$$

2. The author will make WSETL=WSETR generally, so there is a triangle P1FO1 similar to the triangle P2EO2. The following conclusions can be drawn:

$$\begin{cases} \theta l = \theta r \\ OSL = OSR \end{cases}$$

3. Let XYZ following coordinates of origin O be O(xo, yo, zo), XYZ following coordinates of point SL be SL(xl, yl, xl), and XYZ following coordinates of point SR be SR(xr, yr, xr). Because point O, point SL, point SR is in the same scanning plane parallel to the XY plane, so there are:

$$zo = xl = xr$$

4. Taking the O point as the origin, the RXRYRZ rotation translation reference coordinate system proposed in the analysis of the rectangular folding progressive scanning algorithm is introduced, and the RXRYRZ rotation corresponding to the SR (xr, yr, xr) is set. The translation coordinate points are point RSL (rxl, ryl, rxl) and point RSR (rxr, ryr, rxr). Then calculate the RXRYRZ rotation translation reference coordinates of point SL and point SR:

$$\begin{cases} rxl = -(OSL \times \cos(\pi - \theta l)) = OSL \times \cos(\pi - \theta l) \\ ryl = OSL \times \sin(\pi - \theta l) \\ rzl = zo \end{cases}$$

$$\begin{cases} rxr = OSR \times \cos\theta r \\ ryr = OSR \times \sin\theta r \\ rzr = zo \end{cases}$$

Then, the coordinate conversion operator of the RXRYRZ rotation translation coordinate system is called, and then the point RSL obtained by the above and the RXRYRZ rotation translation coordinate of the point RSR are respectively converted into the XYZ following coordinates of the point SL and the point SR:

$$\begin{cases} SL = \Gamma Rt(RSL) \\ SR = \Gamma Rt(RSR) \end{cases}$$

The results obtained:

Thus, by the above algorithm, it is possible to determine the XYZ following coordinates of the scanning initial point of each scanning rectangle of the scanning three rectangles.

In the following content authors will analyze the rectangular folding progressive scanning algorithm, the target person positioning processing algorithm and the DRF orientation and position correction algorithm.

Rectangular folding progressive scanning algorithm

The key design idea of this algorithm is to "fold forward until it finds a condition." This algorithm is an algorithm that combines control with data processing. The realization of the algorithm is that the performance of the mechanical and electronic control part of the LRF group of the robot has no problem and can meet the actual control requirements of the algorithm.

Firstly, the author of the previous mechanical and electronic analysis part knows that the control of the LRF group of the project robot is based on the 180x360 degree rotation space control of the spherical coordinates. Based on the previously defined LRF scan control algorithm, the author can control a specific LRF to perform fixed-point scanning and detection of the detected object in the scan area within the effective scan radius. Under the premise that the mechanical and electronic control meets the accuracy requirements of the control algorithm, the author can perform fixed-point

detection and scanning on the detection area responsible for the LRF through the previously defined LRF control algorithm. Therefore, in the following algorithm discussion, the author will not repeat the algorithm analysis of LRF fixed-point scanning, and assume that the LRF fixed-point scanning algorithm can be performed well, and also meet the hardware requirements of the robot. Then the next question is, how does the author plan the scan path of the LRF after the author knows how to control the LRF to scan a specific point in the detection area, that is, the author sets which points to be detected by the LRF scan to make the LRF detection. The data collected is what the author wants. That is to say, the algorithm of LRF fixed-point scanning is the basis, and the algorithm is based on the algorithm of LRF fixed-point scanning.

After knowing how the control controls the LRF to scan a particular point in the detection space, the author can set a specific scan path for the LRF, which is the key to implementing the algorithm.

The following is a description of the process of the specific implementation of the algorithm:

1. Definition and analysis;
2. Introduce a new coordinate system;
3. RZRYRZ rotation translation reference coordinate system.

It is known from the previous algorithm that if the LRF is to be controlled to scan the folded transition polylines in space, it is necessary to obtain the $R\Phi\Theta$ detection sphere coordinates or the XYZ following coordinates of the respective turning points of the spatial folding polyline. Then, a very straightforward idea is to model the scanning rectangle directly through the XYZ following coordinate system or $R\Phi\Theta$ detecting spherical coordinate system, and find the scanning endpoints of each straight line segment, that is, the turning points of the scanning path.

However, in the actual experimental process of constructing the model, the authors found that if the XYZ following coordinate system or $R\Phi\Theta$ is used to detect the spherical coordinate system to model the scanning rectangle, the algorithm is complicated in solving the coordinates of each turning point. In order to simplify the algorithm and speed up the algorithm execution rate, the author introduces a new coordinate system, which is the RXRYRZ rotation translation reference coordinate system.

The author takes the point OR as the origin, sets the RX axis along the right direction of the bottom edge CD of the scanning rectangle, along the extending direction of the scanning rectangle, sets the RY axis perpendicular to the edge CD, crosses the OR point, and is parallel to the XYZ following coordinate system. The axis sets the RZ axis along the positive direction of the Z axis of the XYZ following coordinate system. These operations will form the RXORRY rotation translation reference coordinate system. Obviously, from the above analysis, the RXORRY rotation translation reference coordinate system plays a role in simplifying the coordinate solution process of the turning point of the scanning rectangle. The RXORRY rotation translation reference coordinate system is one of the cartesian

coordinate system types.

Let the scanning folding turning points of the scanning rectangle to be:

$$\left\{ \begin{array}{l} P1(rx_1, ry_1, rz_1) \\ P2(rx_2, ry_2, rz_2) \\ \dots \\ Pi(rx_i, ry_i, rz_i) \\ \dots \\ Ph(rx_h, ry_h, rz_h) \end{array} \right.$$

That is, the author uses the RXRYRZ rotation reference coordinates to construct the coordinates of the scan folding turning point of the scanning rectangle. The purpose of this is to better combine the structural features of the scanning rectangle, and to facilitate the solution of the coordinates of each scanning corner folding point of the scanning rectangle.

Another issue worthy of attention is that although the RZRYRZ rotation translation reference coordinate system defined by the author can easily solve the various turning point points of the scanning rectangle, the author's LRF scanning control is directly designed based on the RΦΘ detecting spherical coordinate system, while the RZRYRZ rotation The translation reference coordinate system has no direct or indirect connection with the RΦΘ detection spherical coordinate system. Therefore, in order for the LRF control program to obtain the coordinate data of the turning point through the RXRYRZ rotation translation reference coordinate system, the author rotates the RXRYRZ rotation reference coordinate system with the XYZ following coordinate system. Combined. Another point to note is that the author constructs the RXRYRZ rotation translation reference coordinate system only to solve the translation and rotation problems of the relative position of the XYZ following coordinate system and the scanning rectangle. In order to enhance the connection, the other parameters of the two coordinate systems should be consistent. So the author defines that the RXRY face of the RXRYRZ rotation translation reference coordinate system coincides with the plane of the scan rectangle. That is to say, the RXRYRZ rotation translation reference coordinate system indicates that the scanning rectangle of the pair and the scanning track corresponding to the scanning rectangle have the RZ coordinates of 0 at each scanning point. In this way, the author can reduce the scanning rectangle from the 3D processing to the 2D processing, further simplifying the modeling of the scanning rectangle.

The author will elaborate on this process in conjunction with Figure 46.

Analysis:

As shown in Figure 46, in order to describe the above process more clearly, the author defines an OXOYOZ translation reference coordinate system obtained by shifting the XYZ following coordinate system. The point $Pr_0(rx_0, ry_0, rz_0)$ is the origin of the RXRYRZ rotation translation reference coordinate system. Point $Po_0(ox_0, oy_0,$

oz_0) is the OXOYOZ translation reference coordinate system. The point $Po(x_0, y_0, z_0)$ is the origin of the XYZ following coordinate system. (N.B.: Here, the author uses the lowercase letters t, o and r respectively to specify that a certain coordinate belongs to the coordinates representing the XYZ following coordinate system, the OXOYOZ translation reference coordinate system and the RXRYRZ rotation translation reference coordinate system). That is to say, the point Pt_0 , the point Po_0 and the point Pr_0 belong to the same position point in the space, but the coordinate type used to indicate the position is different. Similarly, a point on the straight line segment of the scan rectangle of the scan rectangle can also be represented by three coordinate system coordinates at the same time. As shown in Figure 46, the points $Pti(x_i, y_i, z_i)$, the points $Poi(ox_i, oy_i, oz_i)$ and the points $Pri(rx_i, ry_i, rz_i)$ both represent the coordinate positions of the same scanning point on the scanning rectangle. The OXOYOZ translation reference coordinate system is a reference coordinate system obtained by shifting the XYZ following coordinate system along the $PoPt_0$ direction as a whole. That is to say, the RXRYRZ rotation translation reference coordinate system and the XYZ following coordinate system are related to:

*RXRYRZ Rotational Translation Reference Coordinate System \leftrightarrow (Rotation)
 \leftrightarrow OXOYOZ Translation Reference Coordinate System*

OXOYOZ Translation Reference Coordinate System \leftrightarrow (translation) \leftrightarrow XYZ Following Coordinate System

Then there are:

$$\begin{cases} RX \parallel OX \\ RY \parallel OY \\ RZ \parallel OZ \end{cases}$$

The above RX, RY and RZ represent the three coordinate axes of the RXRYRZ rotational translation reference coordinates, respectively. OX, OY and OZ represent the three axes of the OXOYOZ reference coordinate system, respectively. The angle α is the counterclockwise rotation angle of the RXRYRZ coordinate system and the XYZ following coordinate system. The angle β is the angle formed by the counterclockwise direction of $PoPri$ and the RX axis. The angle γ is the angle formed by the $PoPoi$ and the OX axis in a counterclockwise direction. Point P1 and point P2 are the vertical projection points of point Pri on the RY axis and the RX axis, respectively. Point P3 and point P4 are vertical projection points of the point Poi on the OY axis and the OX axis, respectively. The following is the coordinate transformation analysis:

Rotational coordinate analysis:

Calculation:

1.The known point $Pri(rx_i, ry_i, rz_i)$ is a point on the RXRYRZ rotation translation reference coordinate system. Then the distance $LR0i$ of $Pr0Pri$ is:

$$\begin{cases} LR0i = \sqrt{(rxi - rx0)^2 + (ryi - ry0)^2 + (rzi - rz0)^2} \\ rx0 = ry0 = rz0 \end{cases}$$

And because $rzi=rz0=0$, so:

$$LR0i = \sqrt{rxi^2 + ryi^2}$$

2. Similarly, it is known from previous analysis:

The distance LToi of Pt0Pti is:

$$LToi = \sqrt{(xi - x0)^2 + (yi - y0)^2 + (zi - z0)^2}$$

Because $zi=z0$, so:

$$LToi = \sqrt{(xi - x0)^2 + (yi - y0)^2}$$

3. Therefore, the distance LO0i of Po0P0i is

$$\begin{cases} LO0i = \sqrt{(oxi - ox0)^2 + (oyi - oy0)^2 + (ozi - oz0)^2} \\ ox0 = oy0 = oz0 = 0 \end{cases}$$

Because $oz0=0$, so:

$$LO0i = \sqrt{oxi^2 + oyi^2}$$

4. As shown in Figure 47, the straight line segments LR0i, LT0i and LO0i belong to the same straight line segment, so their space lengths are equal, namely:

$$\sqrt{rxi^2 + ryi^2} = \sqrt{(xi - x0)^2 + (yi - y0)^2} = \sqrt{oxi^2 + oyi^2}$$

5. Suppose the point P1 coordinates are P1 ($rx1, ry1, rz1$), the P2 coordinates are P2 ($rx2, ry2, rz2$), the P3 coordinates are P3 ($ox3, oy3, oz3$), and the P4 coordinates are P4 ($ox4, oy4, oz4$). Get:

$$\begin{cases} (rx1, ry1, rz1) = (0, ryi, z) \\ (rx2, ry2, rz2) = (rxi, 0, z) \\ (ox3, oy3, oz3) = (0, oy3, z) \\ (ox4, oy4, oz4) = (ox4, 0, z) \end{cases}$$

Angle rotation conversion relationship:

Combined with Figure 46 and above, there are:

$$\left\{ \begin{array}{l} \beta = \arctan \frac{rxi}{ryi} \\ \gamma = \beta - \alpha \\ ox4 = \frac{LR0i}{\cos \gamma} \\ oy3 = \frac{LR0i}{\sin \gamma} \\ oxi = ox4 \\ oyi = oy3 \\ ozi = 0 \end{array} \right.$$

Thus: the coordinates between the point Pri(rxi, ryi, rzi) on the RXRYRZ rotational translation reference coordinate system and the corresponding point Poi (oxi, oyi, ozi) on the OXOYOZ translation reference coordinate system are converted into:

$$\left\{ \begin{array}{l} oxi = \frac{\sqrt{rxi^2 + ryi^2}}{\cos(\arctan \frac{rxi}{ryi} - \alpha)} \\ oyi = \frac{\sqrt{rxi^2 + ryi^2}}{\sin(\arctan \frac{rxi}{ryi} - \alpha)} \\ ozi = 0 \end{array} \right.$$

Coordinate translation analysis:

Calculation:

The amount of translational variation of the OXOYOZ reference coordinate system and the components of the XYZ following coordinate system:

$$\left\{ \begin{array}{l} \Delta x = x0 - xo \\ \Delta y = y0 - yo \\ \Delta z = z0 - zo \end{array} \right.$$

Coordinate translation conversion relationship:

Combined with the above analysis, the coordinates between a point Pti(xi, yi, zi) on the XYZ following coordinate system and the corresponding point Poi (oxi, oyi, ozi) on the OXOYOZ translation reference coordinate system are converted into:

$$\begin{cases} xi = oxi + \Delta x = oxi + (x0 - xo) \\ yi = oyi + \Delta y = oyi + (y0 - yo) \\ zi = ozi + \Delta z = ozi + (z0 - zo) \end{cases}$$

And because $xo=yo=zo=0$, the above expression can be reduced to:

$$\begin{cases} xi = oxi + \Delta x = oxi + x0 \\ yi = oyi + \Delta y = oyi + y0 \\ zi = ozi + \Delta z = ozi + z0 \end{cases}$$

Comprehensive analysis of angular rotation and coordinate translation:

Combined with the above analysis, the coordinates between a point $Pti(xi, yi, zi)$ on the XYZ following coordinate system and the corresponding point $Pri(rxi, ryi, rzi)$ on the RXRYRZ rotational translation reference coordinate system are converted to:

$$\begin{cases} xi = \frac{\sqrt{rxi^2 + ryi^2}}{\cos(\arctan \frac{rxi}{ryi} - \alpha)} + x0 \\ yi = \frac{\sqrt{rxi^2 + ryi^2}}{\sin(\arctan \frac{rxi}{ryi} - \alpha)} + yo \\ zi = z0 \end{cases}$$

Defining the corresponding coordinate conversion operator:

In order to facilitating the use of the above algorithm in the subsequent algorithmic description, the author defines the following operators:

TRt : RXRYRZ rotation translation reference coordinates

➔ XYZ following coordinates

TRtx : RXRYRZ Rotational translation reference coordinate x coordinate

➔XYZ follows coordinate x coordinate

TRty : RXRYRZ rotation translation reference coordinate y coordinate

➔ XYZ follows coordinate y coordinate

TRtz : RXRYRZ rotates the z coordinate of the reference coordinate

➔ XYZ follows the z coordinate of the coordinate

Analysis and application:

Through the above algorithm, the algorithm user can convert the coordinates of the track points of the scanning rectangle detected on the RXRYRZ rotation translation reference coordinate system into the coordinates represented by the XYZ following coordinate system. The purpose of the RXRYRZ rotation translation reference coordinate system constructed by the author is to more conveniently control the LRF to scan the trajectory polyline corresponding to the scanning rectangle. Therefore, the

algorithm user only needs to use the RXRYRZ rotation translation reference coordinate to convert to XYZ following coordinates, and no reverse conversion is needed. Therefore, the above algorithmic process does not involve the inverse conversion of the corresponding coordinates.

Through analysis, the author knows that the key to the implementation of this algorithm is to determine the coordinates of each point to be detected on the scanning fold back multi-segment line in the scanning rectangle. However, since the author's designed reentry scan trajectory is composed of multiple straight segments connected end to end, the author uses the idea of segmentation in design, and applies the LRF scanning algorithm for the spatial straight line segment of the known starting point coordinates and the ending point coordinates. As a basic algorithm unit, the start and end point coordinates of each segment of the segmented segment are defined in combination with the rewinding periodicity of the scanning ray of the scanning rectangle, and the LRF is controlled to scan the scanning rectangle.

In the following algorithm description, if there is a reference to the rectangular scanning algorithm, the concept of scanning rectangle is generally mentioned. Scanning rectangles not only represent a scanned rectangular area, but also the entire algorithm implementation process of scanning rectangles.

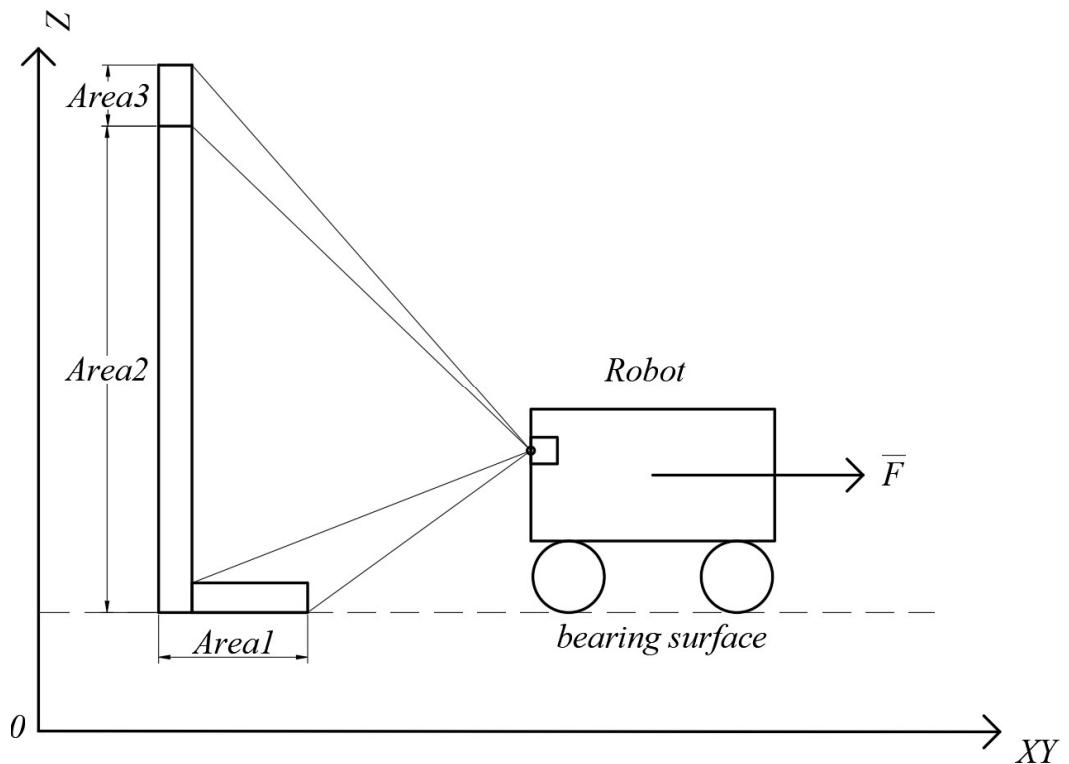


Figure 38.The relative distributions of Area1, Area2 and Area3 of RDF in 3D controls.

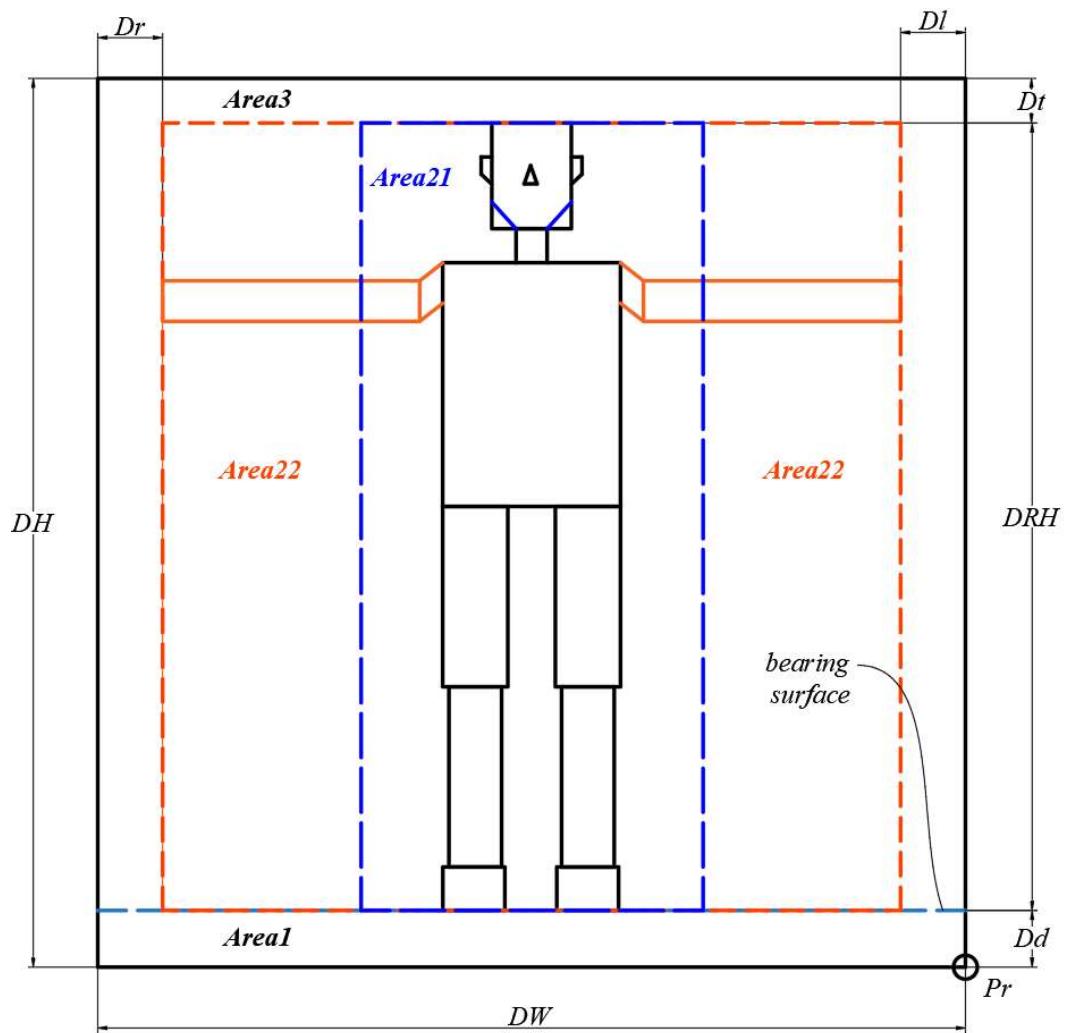


Figure 39. Divisions of the standard scan partition of the DRF.

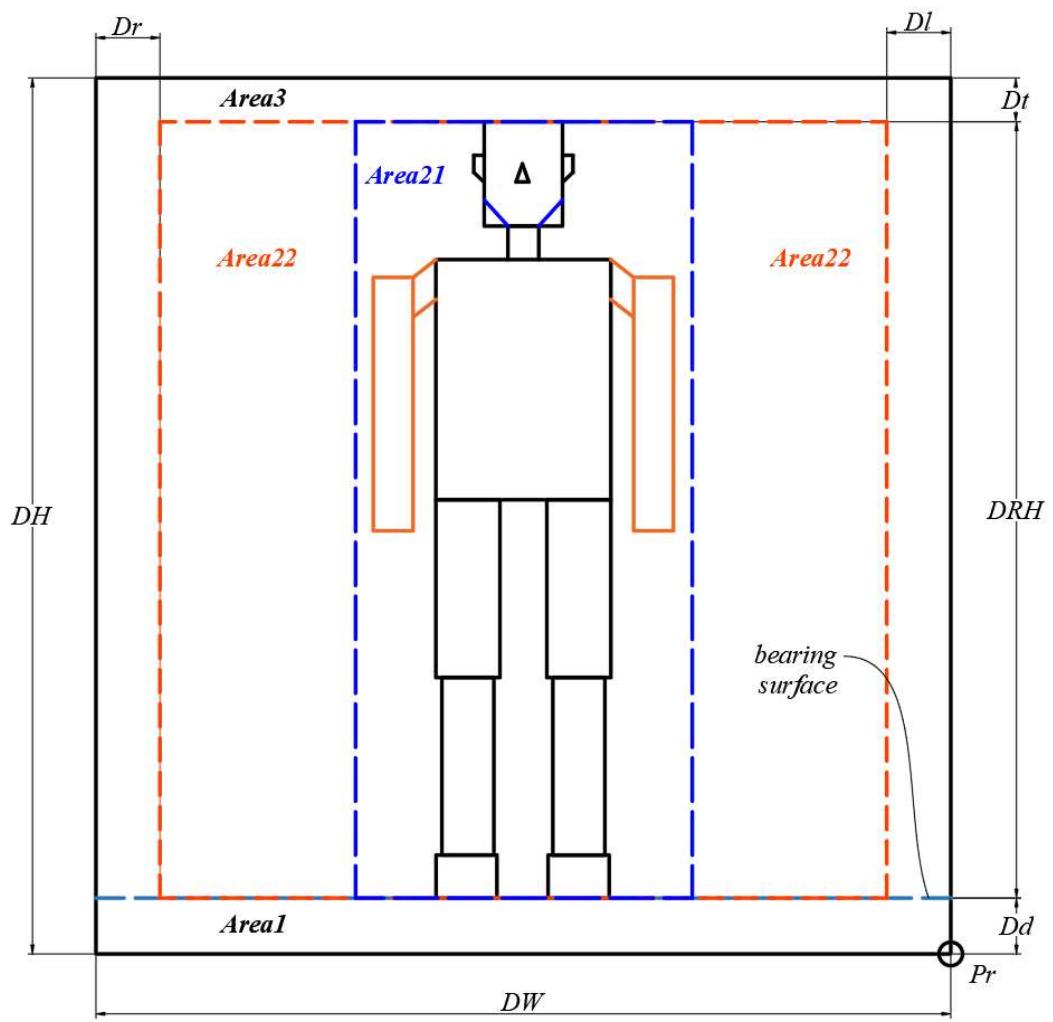


Figure 40. Scan divisions partition of the DRF.

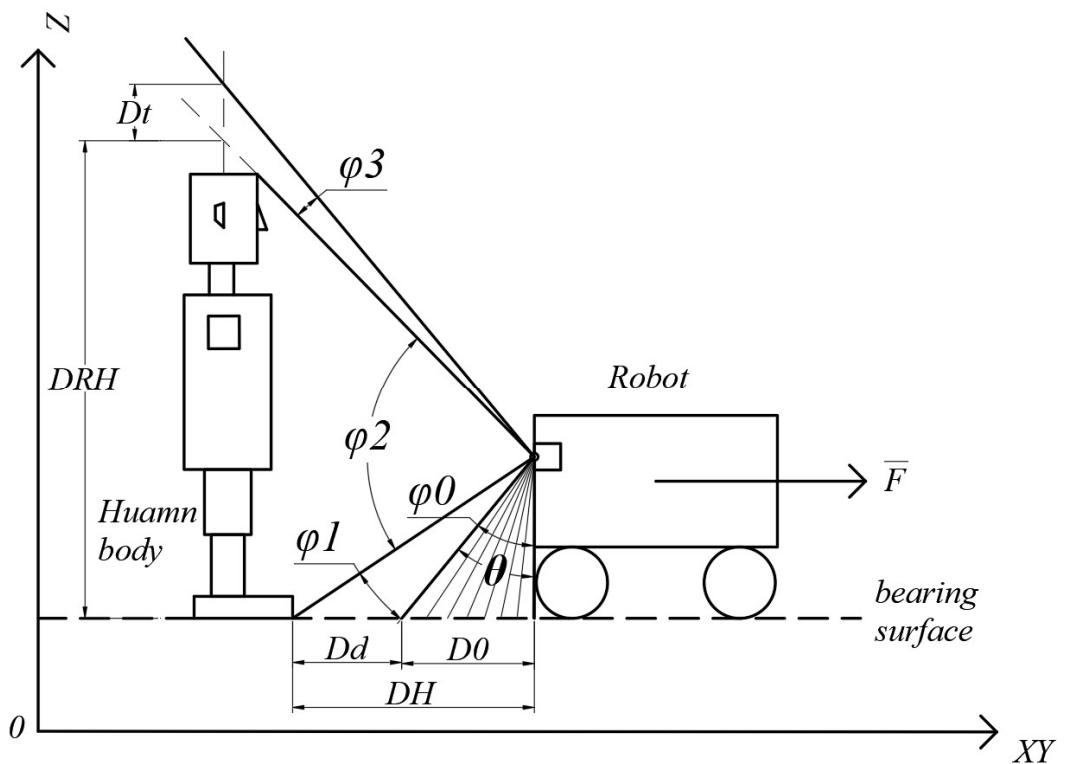


Figure 41. Predicted target person width phase of DRF modeling. Support surface scanning. WDRF<WSET.

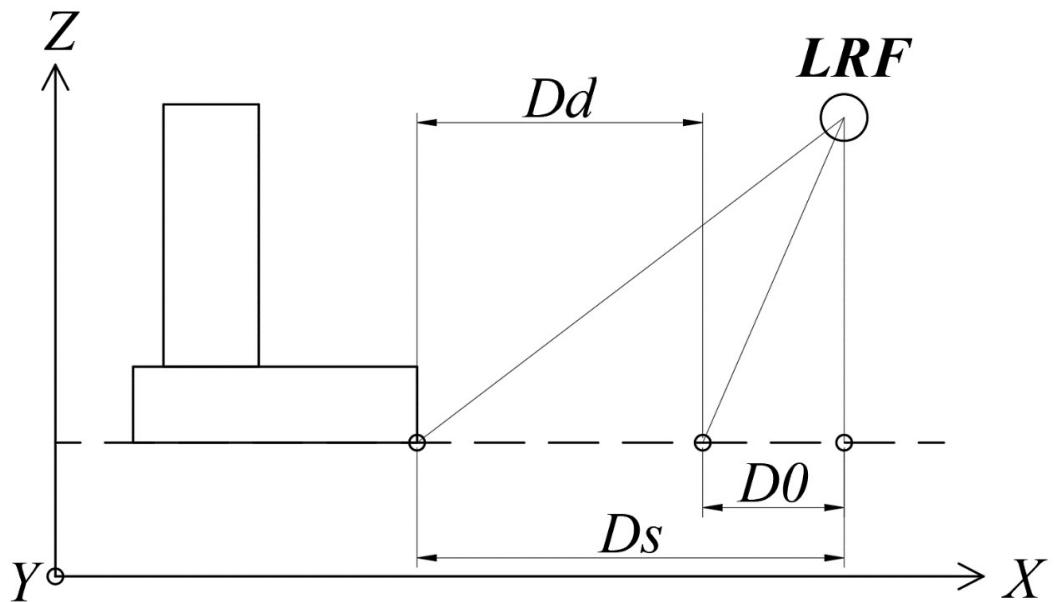


Figure 42. Predicted target person width phase of DRF modeling. Divide the Dd area and the D0 area.

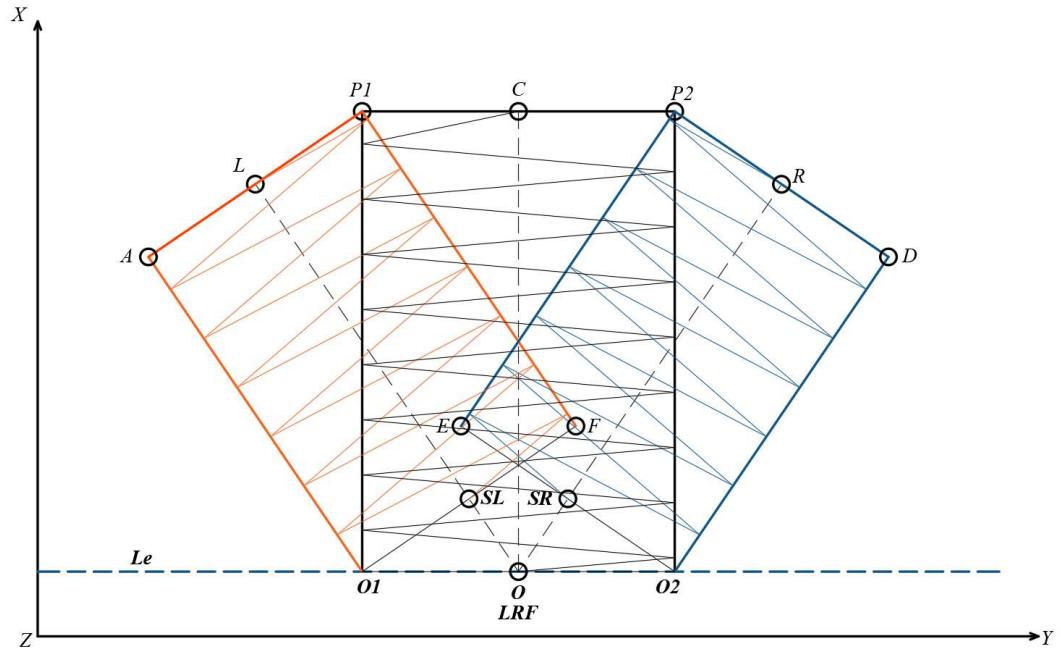


Figure 43. Location and characteristics of the three regions of the scan pre-position.

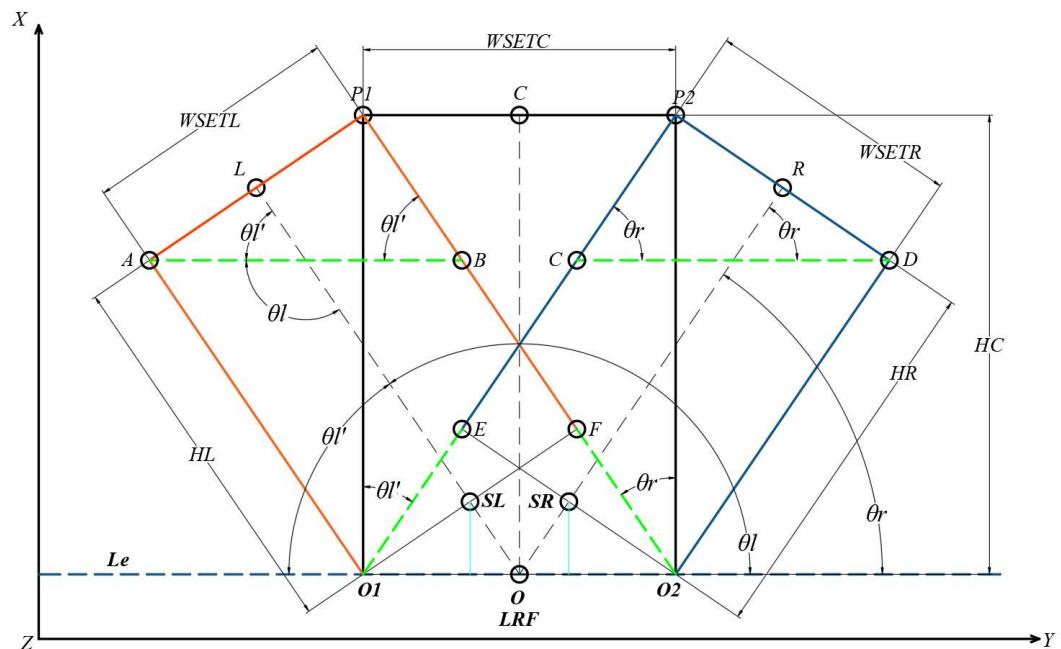


Figure 44. Pre-positioning scan area distribution and parameter settings for the DRF.

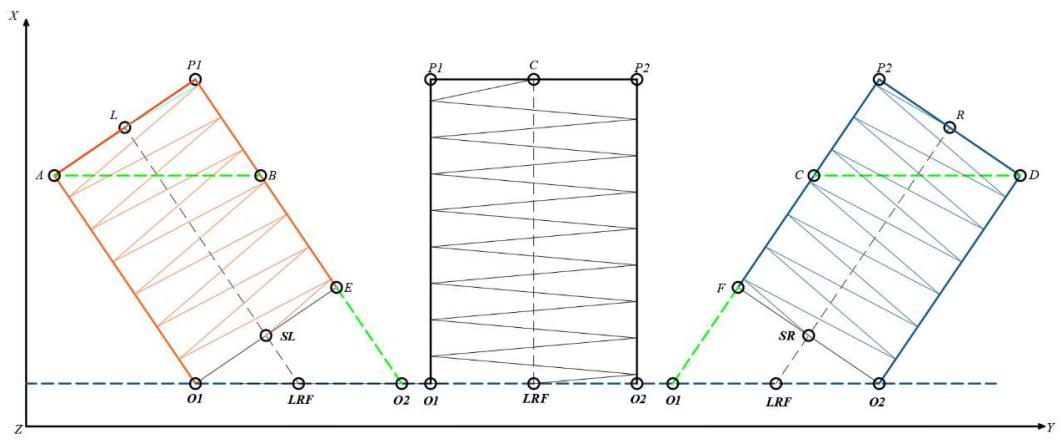


Figure 45. Pre-positioning scan area distribution and parameter settings for the DRF.

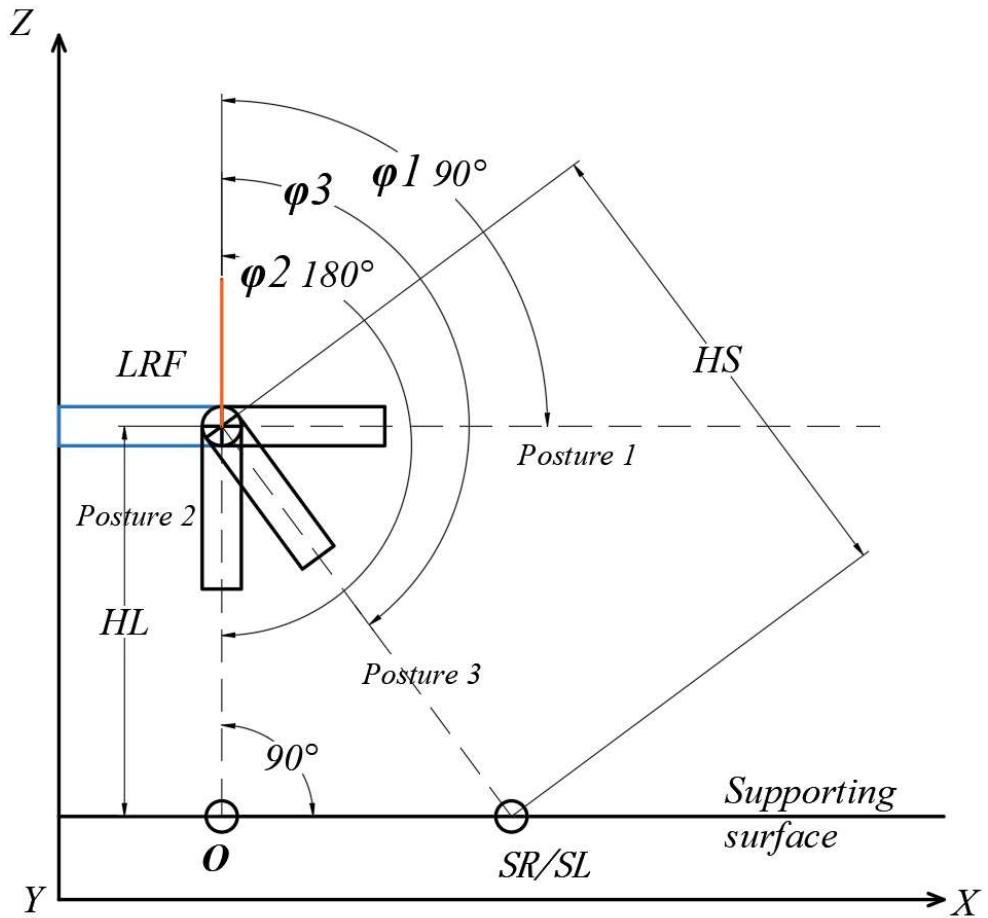


Figure 46. Pre-positioning scan area distribution and parameter settings for the DRF.

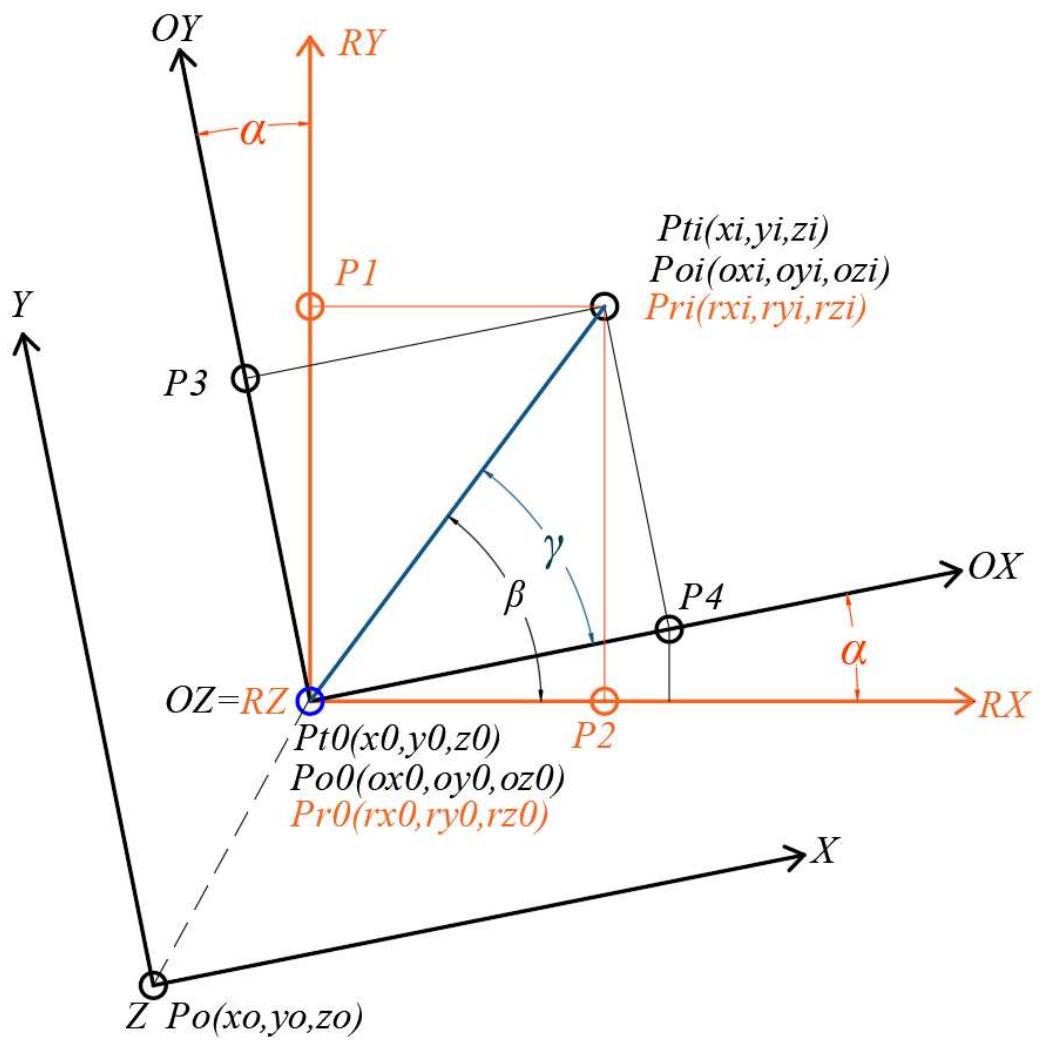


Figure 47. Conversion of the RXRYRZ rotation translation reference coordinate system with the corresponding coordinates of the XYZ following coordinate system.

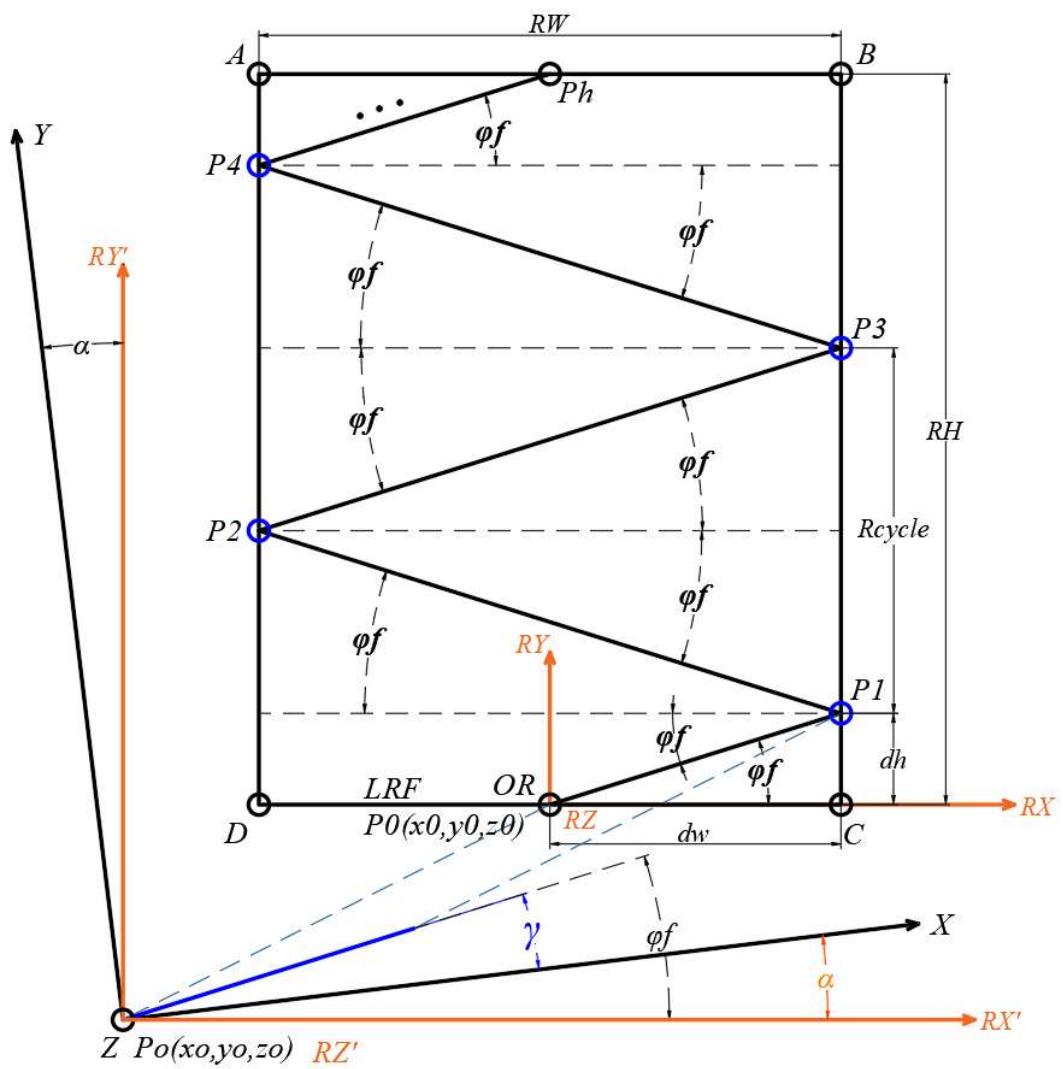


Figure 48. Schematic diagram of a rectangular folding progressive scan algorithm.

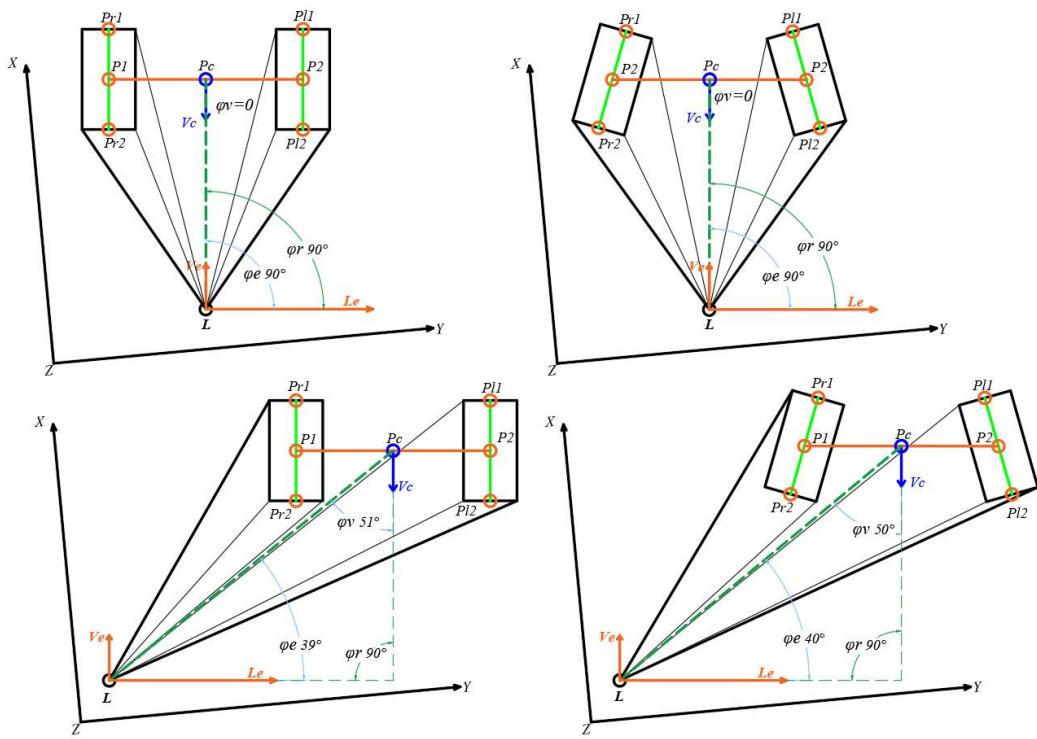


Figure 49. The LRF detects the foot of the target person. The type of possible detection of separate feet. 1.

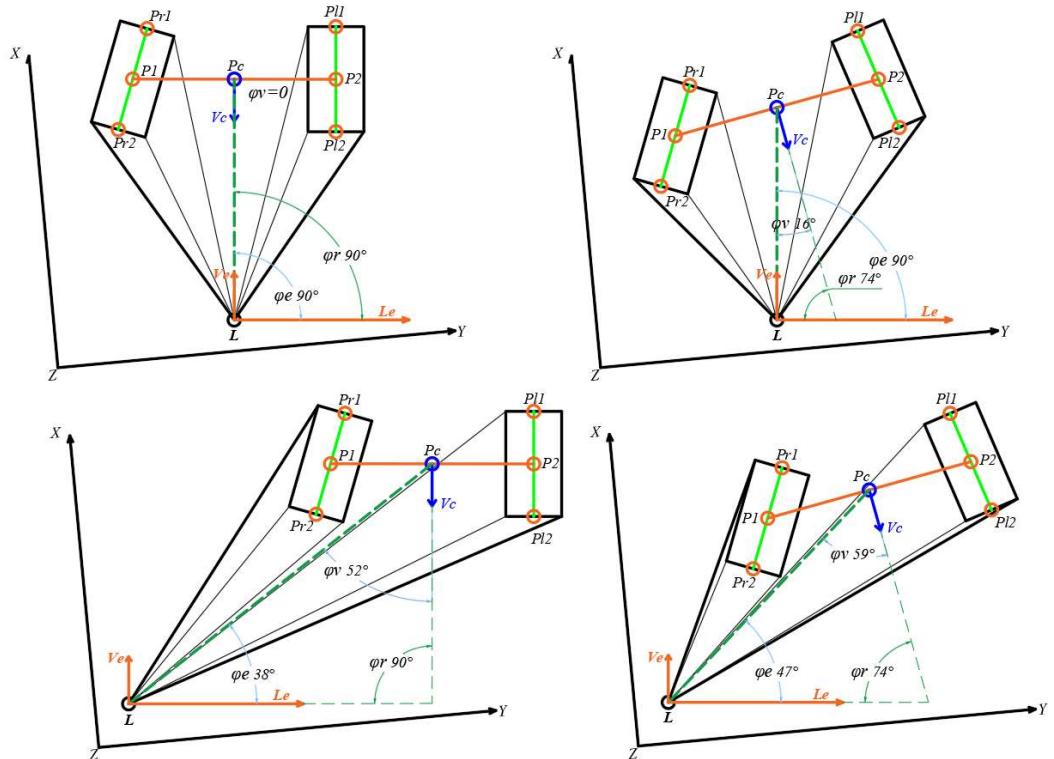


Figure 50. LRF detects the foot of the target person. The type of possible detection of separate feet. 2.

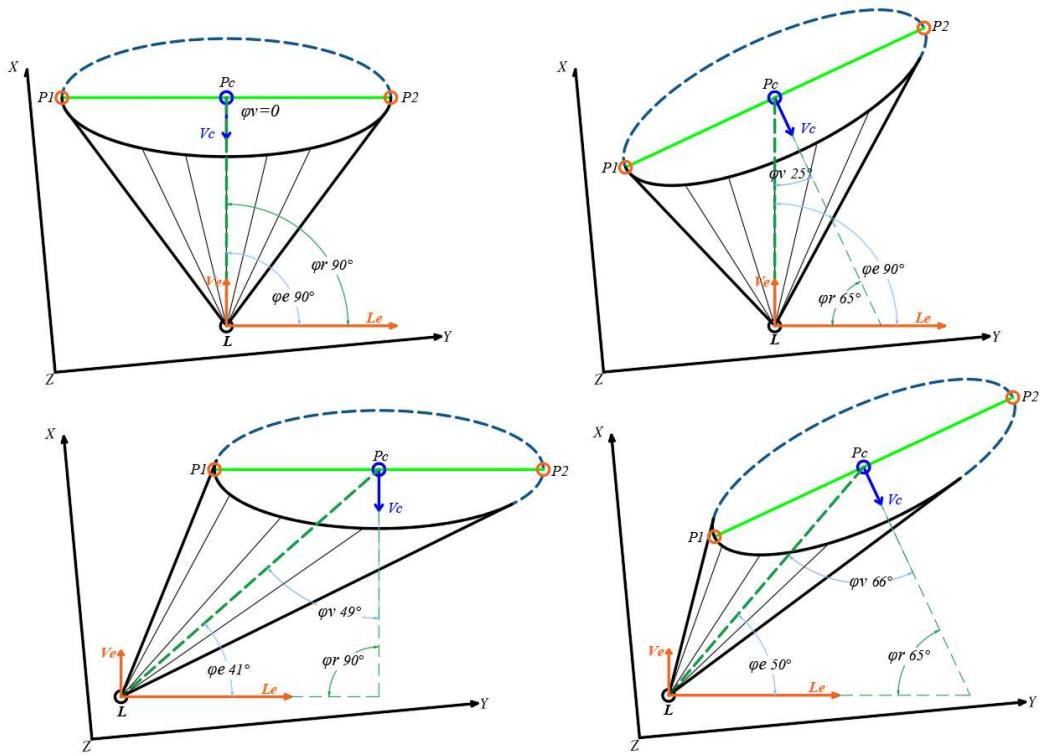


Figure 51. The LRF detects the foot of the target person. The type of possible detection that is obscured by the feet.

For more exciting content, please refer to the second and third articles of this essay:

2. ROBOT DETECTION SYSTEM DESIGN ABOUT FRONT-FOLLOWING TECHNOLOGY PART 2

3. ROBOT DETECTION SYSTEM DESIGN ABOUT FRONT-FOLLOWING TECHNOLOGY PART 3

References:

- Ho, D. M. , Hu, J. S. , & Wang, J. J. . (2012). Behavior control of the mobile robot for accompanying in front of a human.
- Jung, E. J. , Yi, B. J. , & Yuta, S. I. . (2012). Control algorithms for a mobile robot tracking a human in front. IEEE/RSJ International Conference on Intelligent Robots & Systems. IEEE.
- Hu, J. S. , Wang, J. J. , & Ho, D. M. . (2014). Design of sensing system and anticipative behavior for human following of mobile robots. *IEEE Transactions on Industrial Electronics*, 61(4), 1916-1927.
- Moustris, G. P. , & Tzafestas, C. S. . (2017). Intention-based front-following control for an intelligent robotic rollator in indoor environments. *Computational Intelligence*. IEEE.
- Nikdel, P. , Shrestha, R. , & Vaughan, R. . (2018). [IEEE 2018 IEEE International Conference on Robotics and Automation (ICRA) - Brisbane, Australia (2018.5.21-2018.5.25)] 2018 IEEE International Conference on Robotics and Automation (ICRA) - The Hands-Free Push-Cart: Autonomous Following in Front by Predicting User Trajectory Around Obstacles. (pp.1-7).

Project Research Report

ROBOT DETECTION SYSTEM DESIGN ABOUT FRONT-FOLLOWING TECHNOLOGY

PART 2

**Jinwei Lin
(JY Lin, Shenzhen, China)**

December 28, 2018

ID ORCID: <https://orcid.org/0000-0003-0558-6699>

<http://www.ydook.com/>

SPECIFIC IMPLEMENTATION OF THE RECTANGULAR SCANNING ALGORITHM

I. Analysis:

Let the scan rectangle be ABCD, where the length of the AB is determined, and the length of the CD is not determined (IE: the length of the CD is the extension distance when the scan rectangle detects the foot or the foot of the target person.), the bottom edge of the scan rectangle is the CD. . Let the Z-axis height of the XYZ coordinate system in which the scanning rectangle is located be z_0 . The height of the Z-axis of the plane in which the scanning rectangle to be scanned by the LRF is located is defined as the Z-axis height corresponding to the scanning starting point P_0 of the scanning rectangle.

Before constructing the algorithm, first analyze the distribution characteristics of the turning points of the scanning rectangle. The scan rectangle defined by the author is only a representative detection area, and the LRF does not scan all areas within the scan rectangle. The area scanned by the LRF is only a straight line segment in which the start end point and the end point end are respectively distributed on the left and right sides of the scan rectangle. These straight line segments are connected end to end to form a folded scan round-trip polyline. So the scanning algorithm in this part is to control the LRF to scan these folded polylines.

From the analysis of the fixed-point detection scanning algorithm of the spatial straight line segment by the LRF in the previous chapter, we can know that this algorithm is the specific implementation and inheritance and extension analysis of the fixed-point detection scanning algorithm of LRF on the spatial straight line segment. In a nutshell, the function implemented by this algorithm is to combine the LRF fixed-point detection scanning algorithm for the spatial straight line segment into the LRF scanning scan rectangle. The analysis of the fixed-point detection scanning algorithm of the spatial straight line segment by LRF shows that the algorithm designer needs to provide the LRF fixed-point scanning algorithm with the information of the two ends of the straight line segment. In addition, since the scan rectangle is a multi-segment straight line segment that is specific to a position in space, the initial point of the scan is very important. Therefore, one of the main contents of this algorithm is to provide the LRF fixed-point scanning algorithm with the coordinate set of the two ends of the straight line segment and control the LRF to accurately scan from the initial point of the defined scan.

2. Input parameters:

Scan initial point: position coordinates of the OS (xs, ys, zs);

The width of the scan rectangle: RW;

The maximum extension length of the scanning rectangle: RHmax;

Length of the scan cycle: Rcycle;

Analysis:

Therefore, to obtain the coordinate data of the turning point of the scanning rectangle, you only need to know the above four parameters. There is a key point to note here that the acquisition method of the point OS coordinate is obtained by the LRF control algorithm directly controlling the LRF scan to be detected. The specific acquisition method will be explained in the combination of the algorithm and the pre-position scanning algorithm.

2. Calculate the set of turning points:

The author divides the folded round-trip polyline of the scan rectangle into a number of straight segments and names them P0P1, P1P2, ..., P(h-1)Ph. That is, the straight segment of the i-th segment is P(i-1)Pi, (i=1, 2,...,n). Since the scanning rectangle is generally unaware of the length of the extension, the LRF will maintain the scanning state until the scanning detects the object to be scanned. However, in order to make the algorithm run more efficiently, the author defines a maximum extension length RHmax for the scanning rectangle. When the RY coordinate of the turning point is greater than RHmax, if the LRF still does not detect the desired object, the scanning of the scanning rectangle is terminated.

The author uses the array DD[k] in the programming to store the RXRYRZ rotation translation reference coordinates of the two ends of the line segment to be scanned:

$$DD[k] = \begin{cases} DD1 = \left((0,0,z0), \left(\frac{RW}{2}, \frac{1+2\times0}{4}Rcycle, z0 \right) \right) \\ DD2 = \left(\left(\frac{RW}{2}, \frac{1}{4}Rcycle, z0 \right), \left(-\frac{RW}{2}, \frac{1+2\times1}{4}Rcycle, z0 \right) \right) \\ DD3 = \left(\left(-\frac{RW}{2}, \frac{1+2\times1}{4}Rcycle, z0 \right), \left(\frac{RW}{2}, \frac{1+2\times2}{4}Rcycle, z0 \right) \right) \\ DD4 = \left(\left(\frac{RW}{2}, \frac{1+2\times2}{4}Rcycle, z0 \right), \left(-\frac{RW}{2}, \frac{1+2\times3}{4}Rcycle, z0 \right) \right) \\ \dots \\ \dots \\ DDk = \left(\left(\frac{RW}{2}, \frac{1+2(k-2)}{4}Rcycle, z0 \right), \left(-\frac{RW}{2}, \frac{1+2(k-1)}{4}Rcycle, z0 \right) \right) \end{cases}$$

which is:

$$DD[k] = \{DD1, DD2, \dots, DDk\}$$

Where DDi is the i-th element in DD[k], and stores the coordinates of the start point and the end point of the straight line segment after the i-th segment in the scan rectangle. If DDi=(A, B), then A is the starting point coordinate of the i-th straight line segment, and B is the ending point coordinate of the i-th straight line segment. To facilitate the algorithmic explanation, the author defines the following representations:

1. The starting point coordinates and the ending point coordinates of the straight line segment of the i-th space represent:

$$DD[1] = A, DDi[2] = B$$

2. The representation of the coordinates of the starting point of the straight line segment of the i-th space and the coordinates of the end point coordinates:

$$\begin{cases} DDi[1][rx] = A(rx) \\ DD[1][ry] = A(ry) \\ DDi[1][rz] = A(rz) \end{cases}$$

$$\begin{cases} DD[2][rx] = B(rx) \\ DD[2][ry] = B(ry) \\ DDi[2][rz] = B(rz) \end{cases}$$

where A(rx), A(ry) and A(rz) are the RX, RY and RZ coordinate components corresponding to the RXRYRZ rotational translation reference coordinate A, respectively.

3. Since the LRF scan space algorithm needs to provide the parameters of the RΦΘ detection sphere coordinates of the two endpoints of the spatial straight line segment, and DD[k] stores the RXRYRZ rotation translation reference coordinates, the author should perform coordinate transformation.

Transformation:

1. Convert the RXRYRZ rotation translation reference coordinates to XYZ following coordinates:

$$\begin{cases} TDDi[1][x] = \Gamma Rtx(DD[1][rx]) \\ TDDi[1][y] = \Gamma Rtx(DDi[1][ry]) \\ TDDi[1][z] = \Gamma Rtx(DD[1][rz]) \\ TDDi[2][x] = \Gamma Rtx(DDi[2][rx]) \\ TDDi[2][y] = \Gamma Rtx(DD[2][ry]) \\ TDDi[2][z] = \Gamma Rtx(DDi[2][rz]) \end{cases}$$

2. Convert the corresponding XYZ following coordinates to $R\Phi\Theta$ to detect the spherical coordinates:

$$\begin{cases} RDDi[1][r] = \Gamma ttr(TDDi[1]) \\ RDDi[1][\varphi] = \Gamma tt\varphi(TDDi[1]) \\ RDDi[1][\theta] = \Gamma tt\theta(TDDi[1]) \\ RDDi[2][r] = \Gamma ttr(TDDi[2]) \\ RDDi[2][\varphi] = \Gamma tt\varphi(TDDi[2]) \\ RDDi[2][\theta] = \Gamma tt\theta(TDDi[2]) \end{cases}$$

The above $TDDi[1]$ is the XYZ following coordinate of the starting end point of the straight line segment of the i-th space, $TDDi[2]$ is the XYZ following coordinate of the ending end point of the straight line segment of the i-th space, and $RDDi[1]$ is the ith space $R\Phi\Theta$ of the starting end point of the straight line segment detects the spherical coordinates, and $RDDi[2]$ is the $R\Phi\Theta$ detecting spherical coordinate of the starting end point of the straight line segment of the i-th space. Now, the author can input the $R\Phi\Theta$ detection sphere coordinates of the starting point and the end point of the ith space straight line segment into the LRF-to-space straight line segment scanning algorithm, and control the LRF to perform the continuous scanning process on the multi-segment space straight line segment.

Defining new variables:

Define the extension length of the scanned rectangle that has been completed: $RHnow$; Define the number of scan cycles that have been completed so far: Nc ,

$$Nc = NC + 1$$

The initial value of Nc is set to 0. Whenever the LRF completes a scan cycle, it executes:

$$RHnow = Nc \times Rcycl$$

Algorithm execution completed or terminated:

The above algorithm describes a continuous scanning process, but the LRF under the control of this algorithm does not always scan the scanning rectangle. This is because the author ultimately wants to get the result of whether there is a target person

in the scanned area. As described in the previous section, the author defines a maximum extension length RHmax for the scan rectangle to determine if the algorithm has no target person within the scan rectangle. Another reason why the author does not set the LRF to scan the infinitely extended scan rectangle is that in the pre-positioning algorithm, the target person is required to stand in the specified area. If no target person is detected in the three scan rectangles, then it is very likely that the target person gave up this robot to follow the execution of the character. If you do not control the extension of the scan rectangle, it will cause an error that will scan the unrelated human being as the target person. In addition, as the scanning distance increases, the efficiency and accuracy of the LRF detection result will decrease, and the detection error will increase. Therefore, the author does not recommend that the LRF use the long-distance scanning method for DRF model.

Based on the above analysis, the author defines the following termination judgment for the scanning execution of the scanning rectangle by the LRF:

Definition:

Maximum number of scan cycles Ns: Used to define how many scan cycles the LRF performs on the scan rectangle without detecting the target person.

Calculation:

The maximum extension length RHmax is equal to the product of the length Rcycle of the scanning period and the number of maximum scanning periods Ns.

$$RHmax = Ns \times Rcycle$$

Judging:

1. Did you detect the target person:
 1. If yes, terminate the LRF scan of the scan rectangle and return the test data;
 2. If not, perform judgment 2.
2. Whether RHnow < RHmax:
 1. If yes, continue to execute the LRF scanning algorithm;
 2. If not, the LRF scans the current scan rectangle and ends the LRF scan.

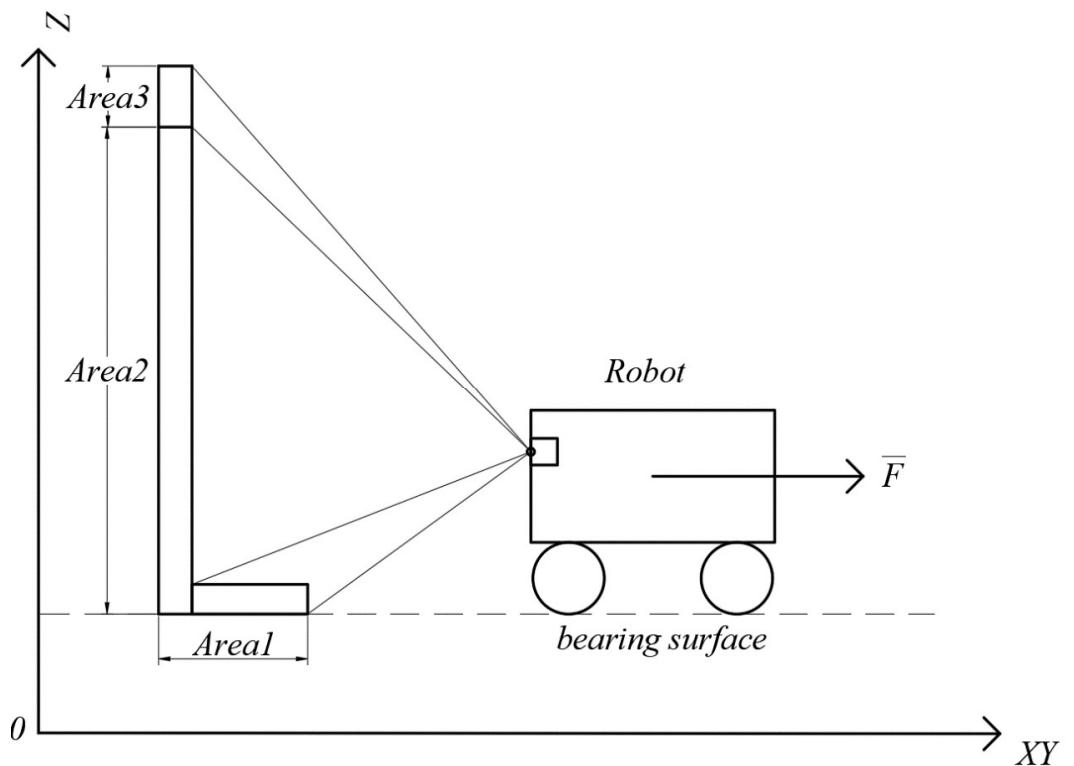


Figure 38. Relative distribution of Area1, Area2 and Area3 of RDF in 3D controls.

TARGET PERSON POSITIONING PROCESSING ALGORITHM

When designing the algorithm, the author determines the area where the target person is located by detecting the foot area of the target person. In this algorithm, a very important algorithm, an object abstraction algorithm, which will be explained later, is needed. The object abstraction algorithm is to abstract the object into a simple geometry by processing the shape data obtained by processing the real three-dimensional object. The object abstraction algorithm is powerful, and it provides strong support for the implementation of the algorithm for the four detection partitions of the target person's body. The author will directly use the results of the object abstraction algorithm in this algorithm, and the specific object abstraction algorithm will be uniformly explained in the following content. The following explains the target person positioning processing algorithm.

Algorithm principle and implementation function:

Because the robot of this project adopts three-dimensional scanning model, and the robot is on the same supporting surface as the target person, when the LRF performs DRF model scan on the target person, the Z-axis position of the LRF is higher than the foot area of the target person. . That is to say, the LRF scans the target person's foot area from high to low. More scanning information can be obtained than scanning in parallel directions.

The author considers the LRF's scanning of the target's feet into two categories:

1. The target person's feet are not obscured, that is, the target person's feet can be detected separately;
2. The target person's feet are covered by a skirt or the like, that is, the target person's feet cannot be detected separately;

Analysis:

For the first case, the object abstraction algorithm abstracts the two feet into two spatial rectangles based on the detected data of the human foot of the target person. The authors refer to a rectangle that is abstracted from human feet as a foot rectangle. Further, a rectangle abstracted by the left foot of the target character is referred to as a left foot rectangle, and a rectangle abstracted by the right foot of the target character is referred to as a right foot rectangle.

Please note that the following algorithm analysis is mainly based on the plane XOY plane of the XYZ coordinate system. If necessary, the author will add the analysis of the components of the Z axis. Why do authors mainly focus on the XOY overhead projection surface? This is because, during the DRF model scan, the target person is required to stand opposite the LRF of the robot. Therefore, in general, the robot is on the same support surface as the target person. Therefore, the author does not focus on the influence of the Z-axis coordinate component.

The author obtains the left and right rectangles of the target character through the object abstraction algorithm. Define the two sides of the foot rectangle, the shorter side is the width side, and the longer side is the length side. The center line along the length direction of the foot rectangle is defined as the direction line of the foot rectangle, that is, the center line connecting the two width sides is the direction line of the foot rectangle, and the direction is from the midpoint of the width edge away from the LRF to the LRF. The midpoint of the width side. As shown in Figure 48 and Figure 49, the direction line of the left foot rectangle of the target person is P11P12, and the midpoint of the direction line is point P2. The direction line of the right foot rectangle of the target character is Pr1Pr2, and the midpoint of the direction line is point P1. Note that the angle of the drawing is based on observing the target person from the outside. The direction of the direction line P1P12 is the point P11 to the point P12, and the direction of the direction line Pr1Pr2 is the point Pr1 to the point Pr2. The author defines that the line connecting the midpoints of the two direction lines of the left foot rectangle and

the right foot rectangle is the width side projection line of the target person. P1P2 in Figure 48 and Figure 49 is the width side projection line of the target person. The width side projection line is used to abstractly reflect the body plane of the target person along the width direction, and can be used to indicate the direction in which the target person's body is turned. In order to more intuitively represent the body of the target person, the author defines the midpoint of the width side projection line as the starting point, and the vector perpendicular to the width side projection line pointing to the LRF is the forward vector of the target person. As shown in Figure 48 and Figure 49, P_c is the midpoint of P1P2, V_c is the forward vector of the target person, V_e is the forward vector of LRF, and V_e is perpendicular to L_e . Since the author is concerned with the direction of V_c and V_e , the author defines that the forward vector V_c of the target person and the forward vector V_e of the LRF are both a unit vector and have a size of one. The author defines the line L_{Pc} of the LRF position point L and the point P_c as the scanning offset line. The angle between L_{Pc} and V_c is a radial deviation angle φ_v . It is obvious that the larger the φ_v , the more the frontal orientation of the target person deviates from the front scanning direction of the LRF. Φ_v is the angle between the L_e extension line and V_c , which is used to reflect the parallelism between V_e and V_c . Φ_v is the angle between the straight line segment P_cL and L_e , which is used to reflect the extent to which the target person deviates from the forward radial direction of the LRF.

Assuming that the angle between V_e and V_c is Ψ_v , then there is:

$$\Psi_v = \left| \frac{\pi}{2} - \varphi_r \right|$$

Assuming that the intersection of the L_e extension line and V_c is point V , the nature of the sum of the internal angles of the triangle $L_{Pc}V$ is easily obtained:

$$\varphi_r = \pi - \varphi_v - \varphi_e$$

According to the scanning characteristics of the LRF, the ideal DRF model scan should be on the front radial extension of the LRF, and in the three rectangular scan area of the LRF, and the target person's feet should be in contact with V_c . Parallel position. That is to say, the upper left diagram in Fig. 48 is the standard target person standing posture. However, in practical applications, the target person may be due to various special reasons, because the target person's own skepticism about the robot may be because the target person encounters an unexpected event while scanning, possibly because the target person scans the DRF. The request is unclear, or it may be because the target person wants to intentionally test the robot, etc. That is to say, when the robot performs DRF scanning, it is likely that the target person cannot match the robot in a standard standing posture. The author reflects the degree of non-standardity of the above situation by Ψ_v and φ_v , φ_e , φ_r .

In order to solve this problem, the author proposes the possible scenarios of Figure 48, Figure 49 and Figure 50:

1. When the LRF scans the foot of the target person, it detects when the two feet are separated:

Figure 48:

Upper left: $\varphi_v=0$, $\varphi_e=90^\circ$, $\varphi_r=90^\circ$;

The feet are radially parallel, the target person's feet are parallel to V_c , and the radial vector of the robot is parallel to the target person's radial vector and collinear scan mode. Point P_c is on the extension line of V_e .

Bottom left: $\varphi_v \neq 0$, $\varphi_e \neq 90^\circ$, $\varphi_r=90^\circ$;

The feet are radially parallel, the target person's feet are parallel to V_c , and the radial vector of the robot is parallel to the target person's radial vector but not collinear. Point P_c is not on the extension line of V_e .

Upper right: $\varphi_v=0$, $\varphi_e=90^\circ$, $\varphi_r=90^\circ$;

The feet are not radially parallel, the target person's feet are not parallel to V_c , and the robot's radial vector is parallel to the target person's radial vector and collinear scan mode. Point P_c is on the extension line of V_e .

Bottom right: $\varphi_v \neq 0$, $\varphi_e \neq 90^\circ$, $\varphi_r=90^\circ$,

The feet are not radially parallel, the target person's feet are not parallel to V_c , and the radial vector of the robot is parallel to the target person's radial vector but not collinear. Point P_c is not on the extension line of V_e .

Figure 49:

Upper left: $\varphi_v=0$, $\varphi_e=90^\circ$, $\varphi_r=90^\circ$;

The feet are not all radially parallel, the left foot of the target person is parallel to V_c , and the radial vector of the robot is parallel to the radial vector of the target person and is in a collinear scan mode. Point P_c is on the extension line of V_e .

Bottom left: $\varphi_v \neq 0$, $\varphi_e \neq 90^\circ$, $\varphi_r=90^\circ$;

The feet are not all radially parallel, the left foot of the target person is parallel to V_c , and the radial vector of the robot is parallel to the radial vector of the target person but not in a collinear scan mode. Point P_c is not on the extension line of V_e .

Upper right: $\varphi_v=0$, $\varphi_e=90^\circ$, $\varphi_r=90^\circ$;

The feet are not radially parallel, the target person's feet are not parallel to V_c , and the radial vector of the robot is not parallel with the target person's radial vector and is not collinear. Point P_c is on the extension line of V_e .

Bottom right: $\varphi_v \neq 0$, $\varphi_e \neq 90^\circ$, $\varphi_r=90^\circ$,

The feet are not radially parallel, the target person's feet are not parallel to V_c , and the radial vector of the robot is parallel to the target person's radial vector but not collinear. Point P_c is not on the extension line of V_e .

Figure 50:

Upper left: $\varphi_v=0$, $\varphi_e=90^\circ$, $\varphi_r=90^\circ$;

A scan mode in which the radial vector of the robot is parallel and collinear with the radial vector of the target person. Point P_c is on the extension line of V_e .

Bottom left: $\varphi_v \neq 0$, $\varphi_e \neq 90^\circ$, $\varphi_r=90^\circ$;

The scan mode in which the radial vector of the robot is parallel to the radial vector of the target person but not collinear. Point P_c is not on the extension line of V_e .

Upper right: $\varphi_v \neq 0^\circ$, $\varphi_e = 90^\circ$, $\varphi_r \neq 90^\circ$;

The scan mode in which the radial vector of the robot is not parallel to the radial vector of the target person and is not collinear. Point P_c is on the extension line of V_e .

Bottom right: $\varphi_v \neq 0^\circ$, $\varphi_e \neq 90^\circ$, $\varphi_r \neq 90^\circ$,

The scan mode in which the radial vector of the robot is parallel to the radial vector of the target person but not collinear. Point P_c is not on the extension line of V_e .

Enter:

The XYZ coordinate of the position where the LRF is located is $L(x_l, y_l, z_l)$.

Spatial analysis based on XOY projection surface. Calculation:

Because the author is most concerned about the relationship of the various components of the XOY projection surface, the author first analyzes each of the above components of the XOY projection surface. In part, the author will explain how to obtain the values of φ_v , φ_e , φ_r . The three coordinate components of XYZ are discussed and analyzed based on the XYZ following coordinate system.

1. Assume that through the object abstraction algorithm, the author obtains the point P_{r1} , the point P_{r2} , the point P_{l1} , and the XYZ coordinates of the point P_{l2} are:

$$\begin{cases} P_{r1}(x_{r1}, y_{r1}) \\ P_{r2}(x_{r2}, y_{r2}) \\ P_{l1}(x_l, y_l) \\ P_{l2}(x_{l2}, y_{l2}) \end{cases}$$

2. From the midpoint coordinate formula, the XYZ coordinates of point P_1 and point P_2 are:

$$\begin{cases} P_1\left(\frac{x_{r1} + x_{r2}}{2}, \frac{y_{r1} + y_{r2}}{2}\right) \\ P_2\left(\frac{x_l + x_{l2}}{2}, \frac{y_l + y_{l2}}{2}\right) \end{cases}$$

3. Vector $\overrightarrow{P_1P_2}$ is:

$$\overrightarrow{P_1P_2} = \left(\frac{x_l + x_{l2}}{2} - \frac{x_{r1} + x_{r2}}{2}, \frac{y_l + y_{l2}}{2} - \frac{y_{r1} + y_{r2}}{2} \right)$$

That is:

$$\overrightarrow{P1P2} = \left(\frac{xl1 + xl2 - (xr1 + xr2)}{2}, \frac{yl1 + yl2 - (yr1 + yr2)}{2} \right)$$

$$\overrightarrow{P1P2} = \begin{bmatrix} \frac{xl1 + xl2 - (xr1 + xr2)}{2} \\ \frac{yl1 + yl2 - (yr1 + yr2)}{2} \end{bmatrix} = \frac{1}{2} \begin{bmatrix} xl1 + xl2 - (xr1 + xr2) \\ yl1 + yl2 - (yr1 + yr2) \end{bmatrix}$$

4. Pc point coordinates are:

$$Pc = \left(\frac{1}{2} \left(\frac{xl1 + xl2}{2} + \frac{xr1 + xr2}{2} \right), \frac{1}{2} \left(\frac{yl1 + yl2}{2} + \frac{yr1 + yr2}{2} \right) \right)$$

which is:

$$Pc = \left(\frac{xl1 + xl2 + xr1 + xr2}{4}, \frac{yl1 + yl2 + yr1 + yr2}{4} \right)$$

5. Let the Vc point coordinates vector be expressed as:

$$\vec{Vc} = \begin{bmatrix} ac \\ bc \end{bmatrix}$$

6. Because Vc is a unit vector, there are:

$$\sqrt{ac^2 + bc^2} = 1$$

7. Since the vector Vc is perpendicular to the vector P1P2, there are:

$$\begin{aligned} \overrightarrow{P1P2}^T \cdot \vec{Vc} &= \frac{1}{2} \begin{bmatrix} xl1 + xl2 - (xr1 + xr2) \\ yl1 + yl2 - (yr1 + yr2) \end{bmatrix}^T \cdot \begin{bmatrix} ac \\ bc \end{bmatrix} \\ &= \frac{1}{2} \{ (xl1 + xl2 - (xr1 + xr2)) \cdot ac + (yl1 + yl2 - (yr1 + yr2)) \cdot bc \} = 0 \end{aligned}$$

Although the author only pays attention to the direction of Vc, since Vc is a unit vector, Vc must be uniquely determined. Calculated by the above equation analysis, the vector expression of Vc can be obtained. Recorded as:

$$\vec{Vc} = \begin{bmatrix} \xi x \\ \xi y \end{bmatrix}$$

8. Let the expression of the Ve vector be Ve(ae, be). Since the vector Ve is perpendicular to Le, let the angle between Le and Y be ψ_{ly} , and the angle between the vector Ve and the X axis be ψ_{ex} . There are:

$$\psi_{ex} = \psi_l$$

9. Let the unit unit vector of the X axis be $Xo(1,0)$, and the value of ψ_l from the angle sensor of the robot, that is, the value of ψ_{ex} . The following equations can be obtained from the two space vector angle formulas:

$$\begin{aligned}\vec{Ve} \cdot \vec{Xo} &= [ae \quad be] \begin{bmatrix} 1 \\ 0 \end{bmatrix} \\ &= |\vec{Ve}| \cdot |\vec{Xo}| \cdot \cos\psi \\ &= \sqrt{ae^2 + be^2} \cdot 1 \cdot \cos\psi_{ex}\end{aligned}$$

Although the author only pays attention to the direction of Ve , since Ve is a unit vector, Ve must be uniquely determined. From the above formula, the vector expression of Ve can be obtained, which is recorded as:

$$\vec{Ve} = \begin{bmatrix} \varepsilon x \\ \varepsilon y \end{bmatrix}$$

10. Calculate and get the angle between the vector Vc and the vector Ve Ψ_v :

$$\begin{aligned}\Psi_v &= \arccos \left(\frac{\vec{Vc} \cdot \vec{Ve}}{|\vec{Vc}| \cdot |\vec{Ve}|} \right) \\ &= \arccos \left(\frac{\xi x \cdot \varepsilon x + \xi y \cdot \varepsilon y}{1 \times 1} \right) \\ &= \arccos(\xi x \cdot \varepsilon x + \xi y \cdot \varepsilon y)\end{aligned}$$

Judgment:

Whether vector Vc is collinear with vector Ve :

Since the vector Vc and the vector Ve are both based on the two-dimensional unit vector of the XOY plane, if the vector Vc is collinear with the vector Ve , there must be:

$$\begin{cases} \Psi_v = 0 \\ \xi x = \varepsilon x \\ \xi y = \varepsilon y \end{cases}$$

Analysis based on three-dimensional space. Calculation:

The analysis algorithm based on three-dimensional space is an upgraded version of the above algorithm. However, it should be noted that the relationship between the vector Vc and Ve obtained by the algorithm is a three-dimensional relationship, that is,

if the vector V_c is collinear with V_e , it is collinear in three-dimensional space. If the vector V_c and V_e are parallel, Parallel in three dimensions. The algorithm takes into account the Z coordinate of the third coordinate component of the XYZ following coordinate system, which is suitable for the confirmation and analysis of key parameters of more accurate DRF scanning. In practical applications, the author suggests combining the two algorithms. This can achieve complementary advantages and obtain more accurate test results.

In this section, the author will explain how to obtain the values of φ_v , φ_e , φ_r in the three-dimensional coordinate space. The XYZ coordinate components are discussed and analyzed based on the XYZ following coordinate system.

1. Assume that through the object abstraction algorithm, the author obtains the point Pr_1 , the point Pr_2 , the point P_{l1} , and the XYZ coordinates of the point P_{l2} are:

$$\begin{cases} Pr_1(xr_1, yr_1, zr_1) \\ Pr_2(xr_2, yr_2, zr_2) \\ P_{l1}(xl_1, yl_1, xl_1) \\ P_{l2}(xl_2, yl_2, xl_2) \end{cases}$$

2. From the midpoint coordinate formula, the XYZ coordinates of point P_1 and point P_2 are:

$$\begin{cases} P_1\left(\frac{xr_1 + xr_2}{2}, \frac{yr_1 + yr_2}{2}, \frac{zr_1 + zr_2}{2}\right) \\ P_2\left(\frac{xl_1 + xl_2}{2}, \frac{yl_1 + yl_2}{2}, \frac{zl_1 + zl_2}{2}\right) \end{cases}$$

3. Vector $\overrightarrow{P_1P_2}$ is:

$$\overrightarrow{P_1P_2} = \left(\frac{xl_1 + xl_2 - (xr_1 + xr_2)}{2}, \frac{yl_1 + yl_2 - (yr_1 + yr_2)}{2}, \frac{zl_1 + zl_2 - (zr_1 + zr_2)}{2} \right)$$

Which is:

$$\overrightarrow{P_1P_2} = \left(\frac{xl_1 + xl_2 - (xr_1 + xr_2)}{2}, \frac{yl_1 + yl_2 - (yr_1 + yr_2)}{2}, \frac{zl_1 + zl_2 - (zr_1 + zr_2)}{2} \right)$$

$$\overrightarrow{P_1P_2} = \left[\begin{array}{c} \frac{xl_1 + xl_2 - (xr_1 + xr_2)}{2} \\ \frac{yl_1 + yl_2 - (yr_1 + yr_2)}{2} \\ \frac{zl_1 + zl_2 - (zr_1 + zr_2)}{2} \end{array} \right] = \frac{1}{2} \begin{bmatrix} xl_1 + xl_2 - (xr_1 + xr_2) \\ yl_1 + yl_2 - (yr_1 + yr_2) \\ zl_1 + zl_2 - (zr_1 + zr_2) \end{bmatrix}$$

4. The coordinates of the P_c point:

$$P_c = \left(\frac{1}{2} \left(\frac{x_{l1} + x_{l2}}{2} + \frac{x_{r1} + x_{r2}}{2} \right), \frac{1}{2} \left(\frac{y_{l1} + y_{l2}}{2} + \frac{y_{r1} + y_{r2}}{2} \right), \frac{1}{2} \left(\frac{z_{l1} + z_{l2}}{2} + \frac{z_{r1} + z_{r2}}{2} \right) \right)$$

Which is:

$$P_c = \left(\frac{x_{l1} + x_{l2} + x_{r1} + x_{r2}}{4}, \frac{y_{l1} + y_{l2} + y_{r1} + y_{r2}}{4}, \frac{z_{l1} + z_{l2} + z_{r1} + z_{r2}}{4} \right)$$

5. Let the V_c vector be expressed as:

$$\vec{V_c} = \begin{bmatrix} ac \\ bc \\ cc \end{bmatrix}$$

Because V_c is a unit vector, there are:

$$\sqrt{ac^2 + bc^2 + cc^2} = 1$$

Since the vector V_c is perpendicular to the vector P_1P_2 , there are:

$$\begin{aligned} \vec{P_1P_2}^T \cdot \vec{V_c} &= \frac{1}{2} \begin{bmatrix} x_{l1} + x_{l2} - (x_{r1} + x_{r2}) \\ y_{l1} + y_{l2} - (y_{r1} + y_{r2}) \\ z_{l1} + z_{l2} - (z_{r1} + z_{r2}) \end{bmatrix}^T \cdot \begin{bmatrix} ac \\ bc \\ cc \end{bmatrix} \\ &= \frac{1}{2} \{ (x_{l1} + x_{l2} - (x_{r1} + x_{r2})) \cdot ac + (y_{l1} + y_{l2} - (y_{r1} + y_{r2})) \cdot bc + (z_{l1} + z_{l2} - (z_{r1} + z_{r2})) \cdot cc \} = 0 \end{aligned}$$

Although the author only pays attention to the direction of V_c , since V_c is a unit vector, V_c must be uniquely determined. From the above formula, the vector expression of V_c can be obtained, which is recorded as:

$$\vec{V_c} = \begin{bmatrix} \xi_x \\ \xi_y \\ \xi_z \end{bmatrix}$$

6. Let the expression of the V_e vector be $V_e(a_e, b_e, c_e)$. Since the vector V_e is perpendicular to L_e , let the angle between L_e and Y be ψ_{ly} , and the angle between the vector V_e and the X axis be ψ_{ex} . There are:

$$\psi_e = \psi_l$$

7. Let the unit vector of the X axis be $Xo(1,0,0)$, and the value of ψ_l from the angle sensor of the robot, that is, the value of ψ_{ex} , obtained follow expressions content by the formula of the angle between the two space vectors:

$$\begin{aligned}\vec{Ve} \cdot \vec{Xo} &= [ae \quad be \quad ce] \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \\ &= |\vec{Ve}| \cdot |\vec{Xo}| \cdot \cos\psi_e \\ &= \sqrt{ae^2 + be^2 + ce^2} \cdot 1 \cdot \cos\end{aligned}$$

8. Although the author only pays attention to the direction of Ve , since Ve is a unit vector, Ve must be uniquely determined. From the above formula, the vector expression of Ve can be obtained, which is recorded as:

$$\vec{Ve} = \begin{bmatrix} \varepsilon_x \\ \varepsilon_y \\ \varepsilon_z \end{bmatrix}$$

9. Calculate the angle between the vector Vc and the vector Ve in the three-dimensional space Ψ_t :

$$\begin{aligned}\Psi_t &= \arccos \left(\frac{\vec{Vc} \cdot \vec{Ve}}{|\vec{Vc}| \cdot |\vec{Ve}|} \right) \\ &= \arccos \left(\frac{\xi_x \cdot \varepsilon_x + \xi_y \cdot \varepsilon_y + \xi_z \cdot \varepsilon_z}{1 \times 1} \right) \\ &= \arccos(\xi_x \cdot \varepsilon_x + \xi_y \cdot \varepsilon_y + \xi_z \cdot \varepsilon_z)\end{aligned}$$

Judgment:

Whether vector Vc is collinear with vector Ve :

Since the vector Vc and the vector Ve are both based on the three-dimensional unit vector of the XYZ following coordinate system space, if the vector Vc is collinear with the vector Ve , there must be:

$$\begin{cases} \Psi_t = 0 \\ \xi_x = \varepsilon_x \\ \xi_y = \varepsilon_y \\ \xi_z = \varepsilon_z \end{cases}$$

Algorithm results and output:

Through the above algorithm, the author can know the position and deflection alignment of the target person relative to the LRF position point. The calculated result data is then transmitted to the DRF orientation and position correction algorithm for processing. The DRF orientation and position correction algorithm automatically controls the robot to adjust its state (EG: adjust the XYZ coordinates and orientation of the LRF of the robot) so that the LRF used for DRF scan model is just positively aligned radially character.

Of course, the author can also carry out the second idea design, which is to prompt the target person to adjust his own standing posture and orientation by sound or LED flashing light to meet the robot to perform DRF scanning. The algorithm designed in this way is much simpler, and it is not even necessary to call the DRF orientation and position correction algorithm. However, this may cause a bad user experience for certain special user groups. So, in order to solve this problem, the author proposes the following strategies:

First, the DRF orientation and position correction algorithm is called by the robot, and the environmental obstacle scan is performed, but the algorithm result is not executed immediately. First perform the following analysis:

1. If the surrounding environment satisfies the adjustment of the robot's own position and orientation, the results of the DRF orientation and position correction algorithm are performed.
2. If the surrounding environment does not meet the adjustment of the robot's own position and orientation, the robot will give a prompt to prompt the target person to correct his position and orientation.

The above strategy can ensure that the target person has a good user experience when performing DRF scanning before using the robot to perform the following characters.

DRF ORIENTATION AND POSITION CORRECTION ALGORITHM

This algorithm belongs to the control class calculation. If it is to be developed in detail, it needs to be combined with the specific mechanical design of the robot and the design of the power system. In a nutshell, it is to control the movement of the robot so that the target person faces the LRF that performs the DRF scan as much as possible.

That is to say, try to make V_c and V_e collinear. Since different robots have the same mechanical structure and power control system, the basic purpose is the same, that is, control the robot to move to a certain position. These algorithms belong to the basic control algorithm and are not the focus of this research report. Due to the limitation of writing time, this algorithm is only a simple analysis. For detailed precision control, please look forward to the second edition of the research report.

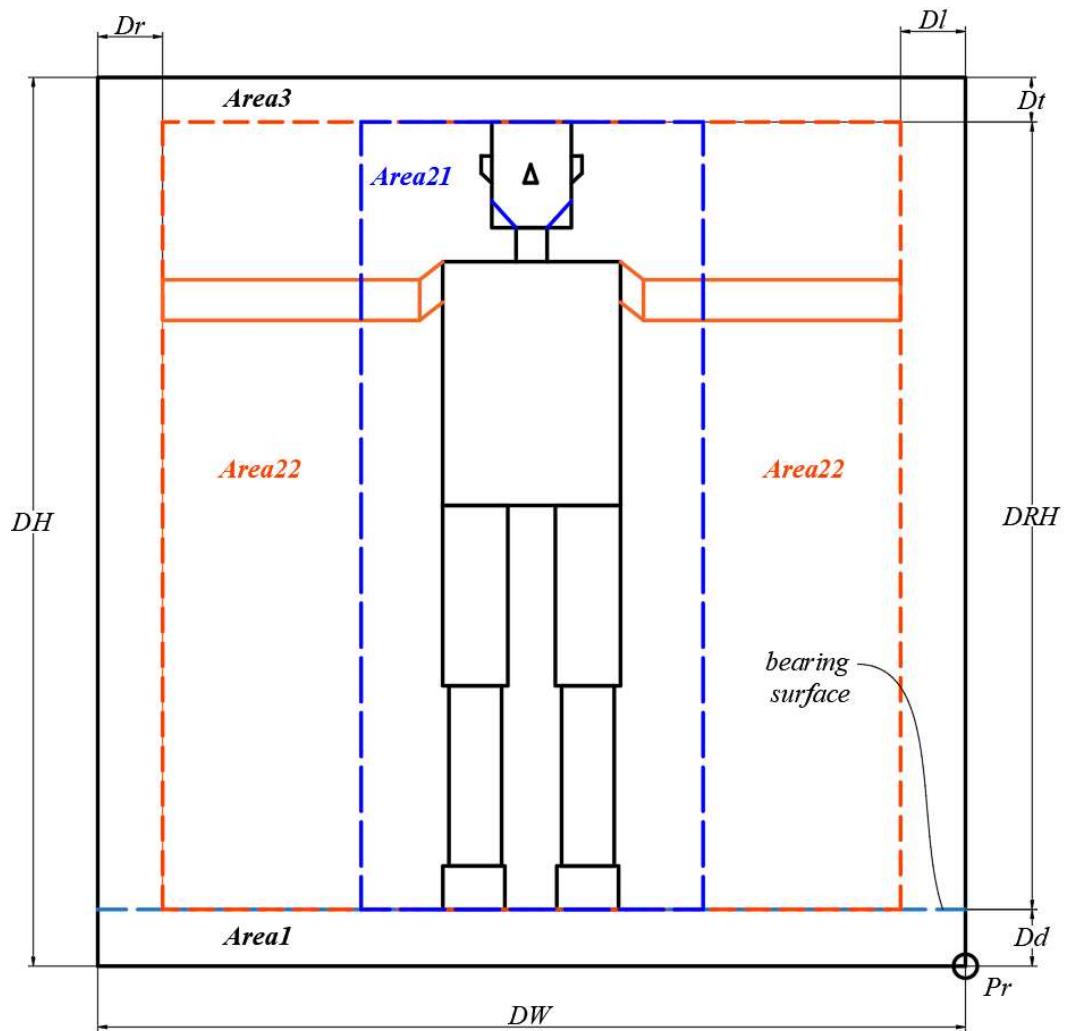


Figure 39. Divisions of the standard scan partition of the DRF.

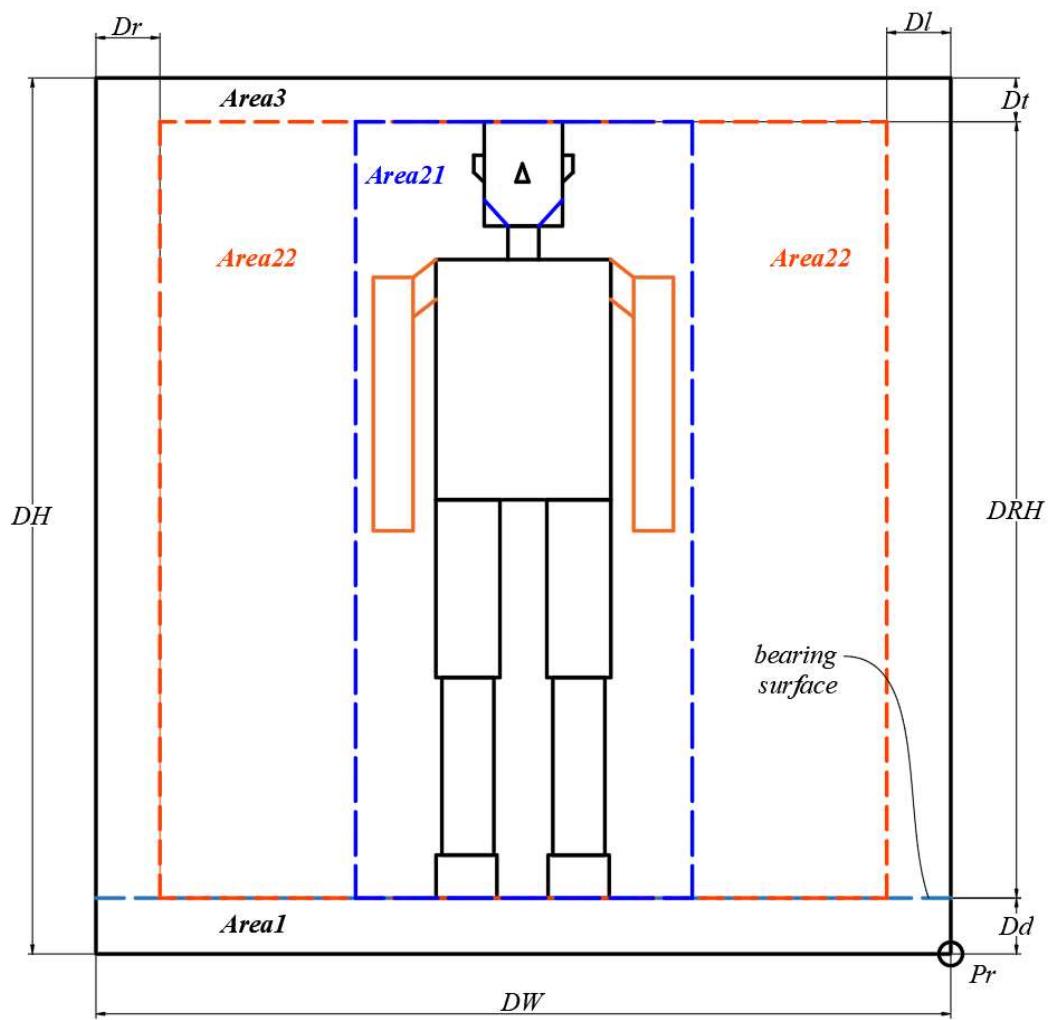


Figure 40. Scan divisions partition of the DRF.

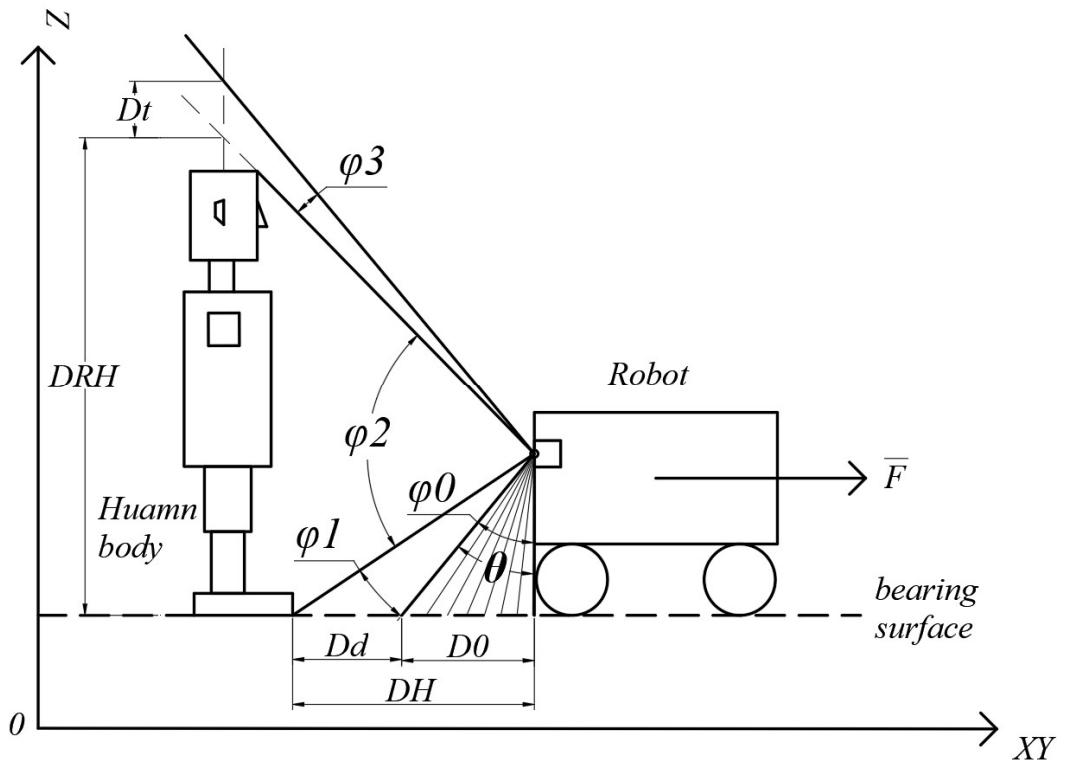


Figure 41. Predicted target person width phase of DRF model. Support surface scanning. WDRF<WSET.

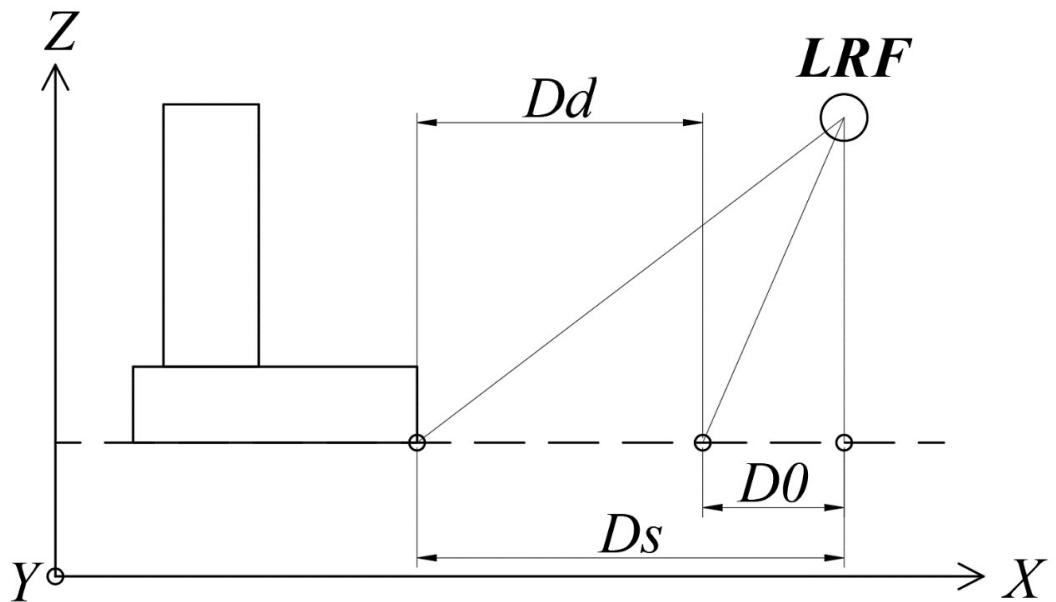


Figure 42. Predicted target person width phase of DRF model. Divide the Dd area and the D0 area.

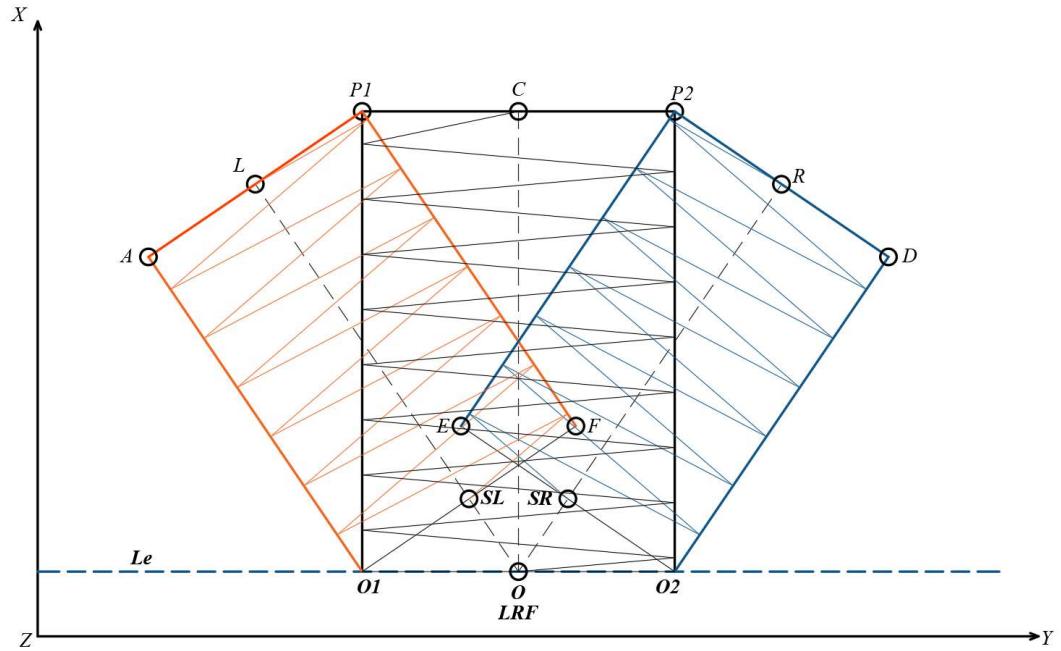


Figure 43. Location and characteristics of the three regions of the scan pre-position.

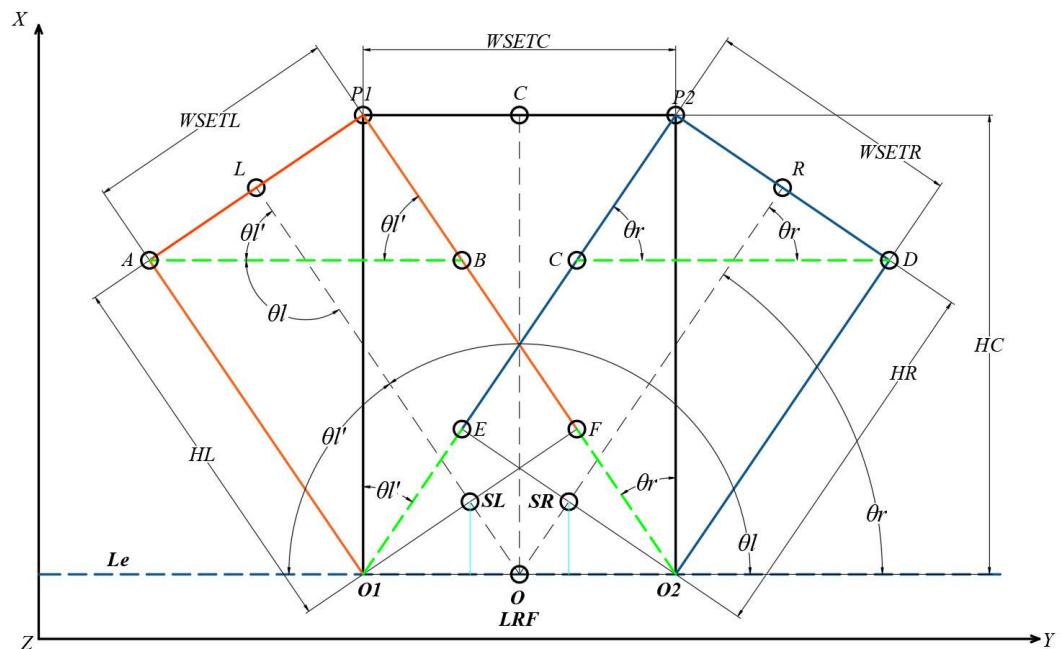


Figure 44. Pre-positioning scan area distribution and parameter settings for the DRF.

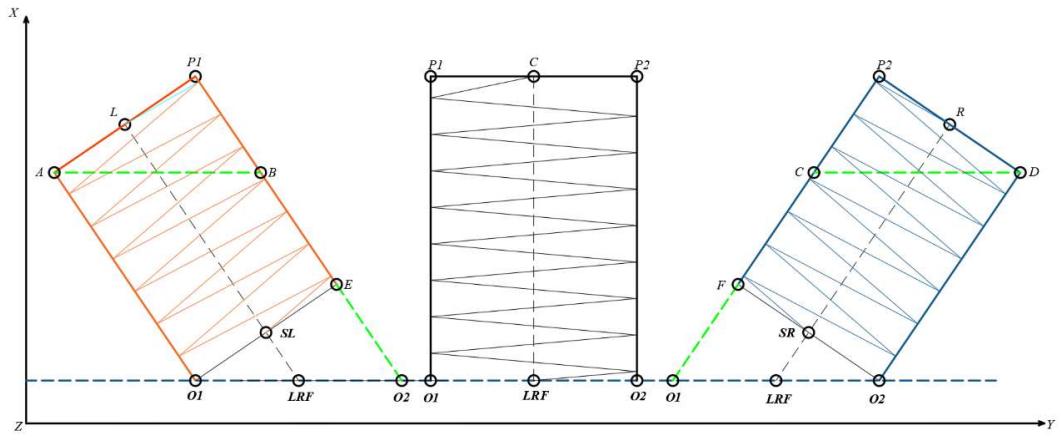


Figure 45. Pre-positioning scan area distribution and parameter settings for the DRF.

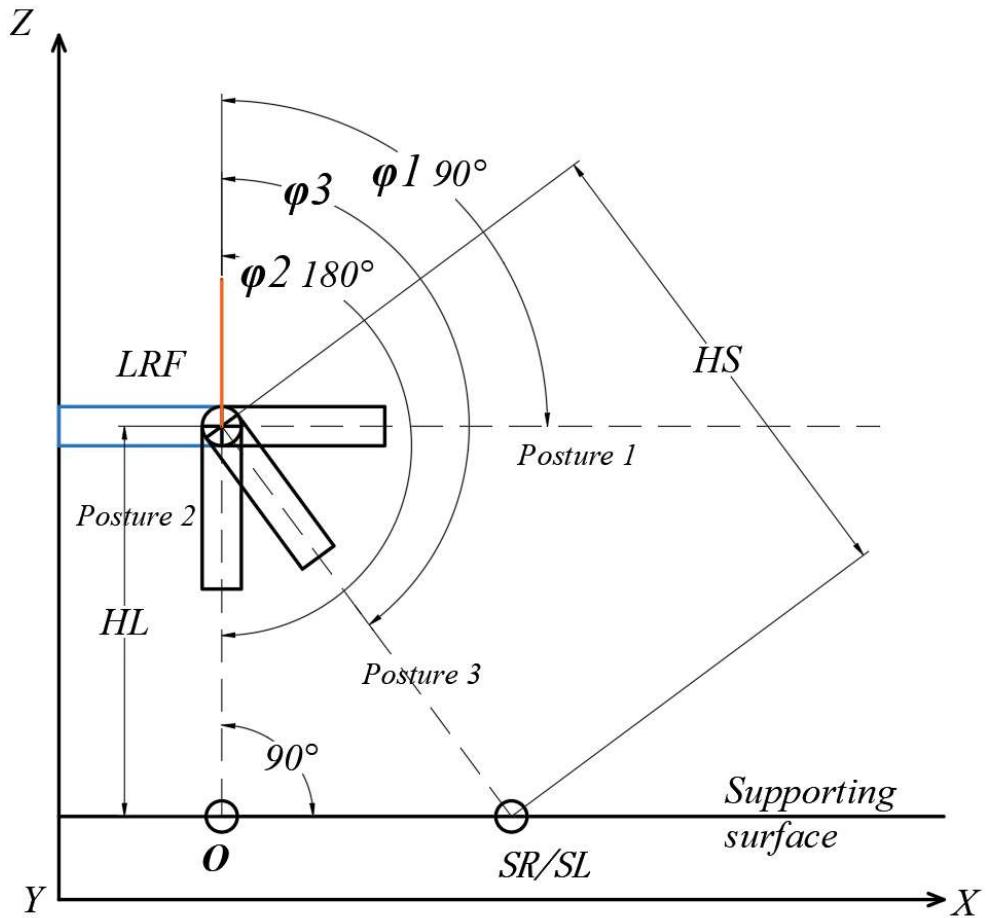


Figure 46. Pre-positioning scan area distribution and parameter settings for the DRF.

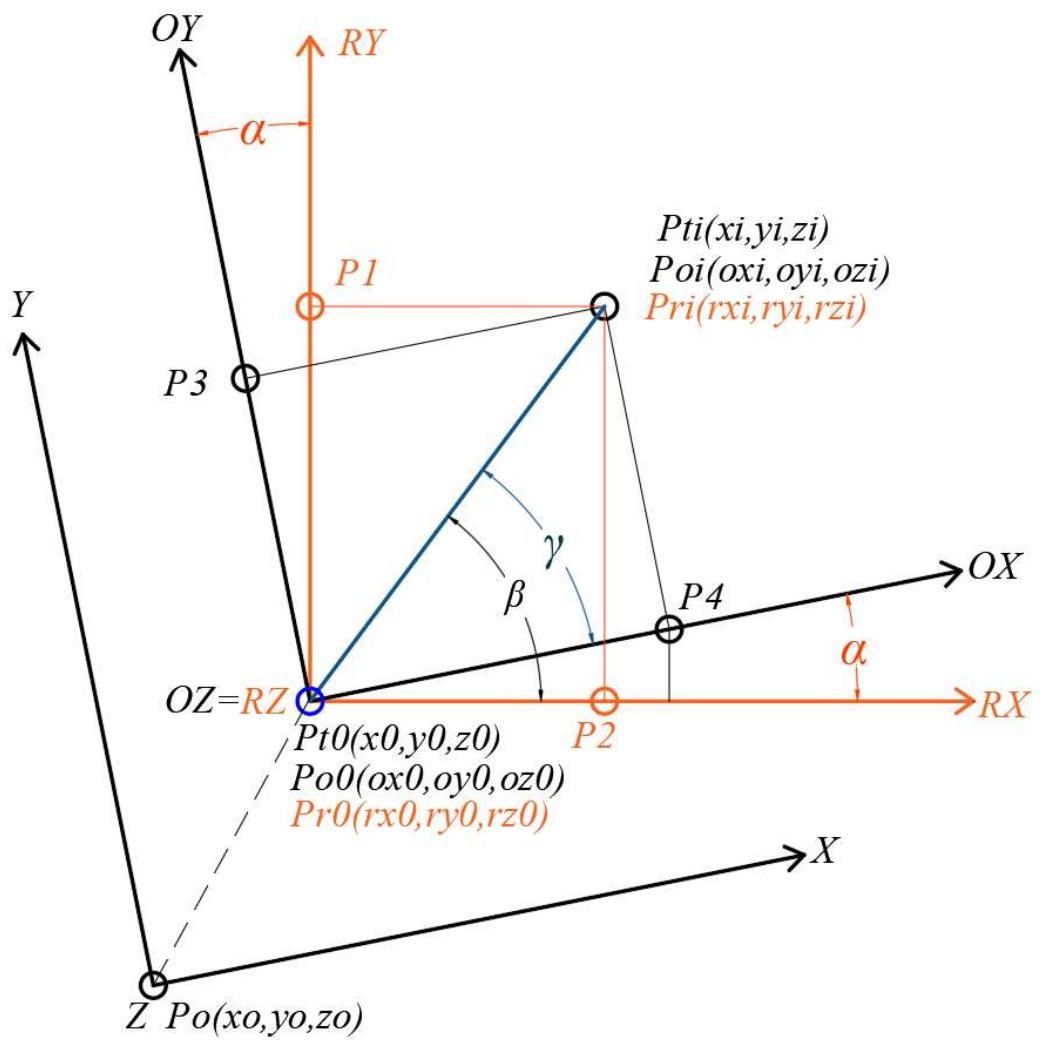


Figure 47. Conversion of the RXRYRZ rotation translation reference coordinate system with the corresponding coordinates of the XYZ following coordinate system.

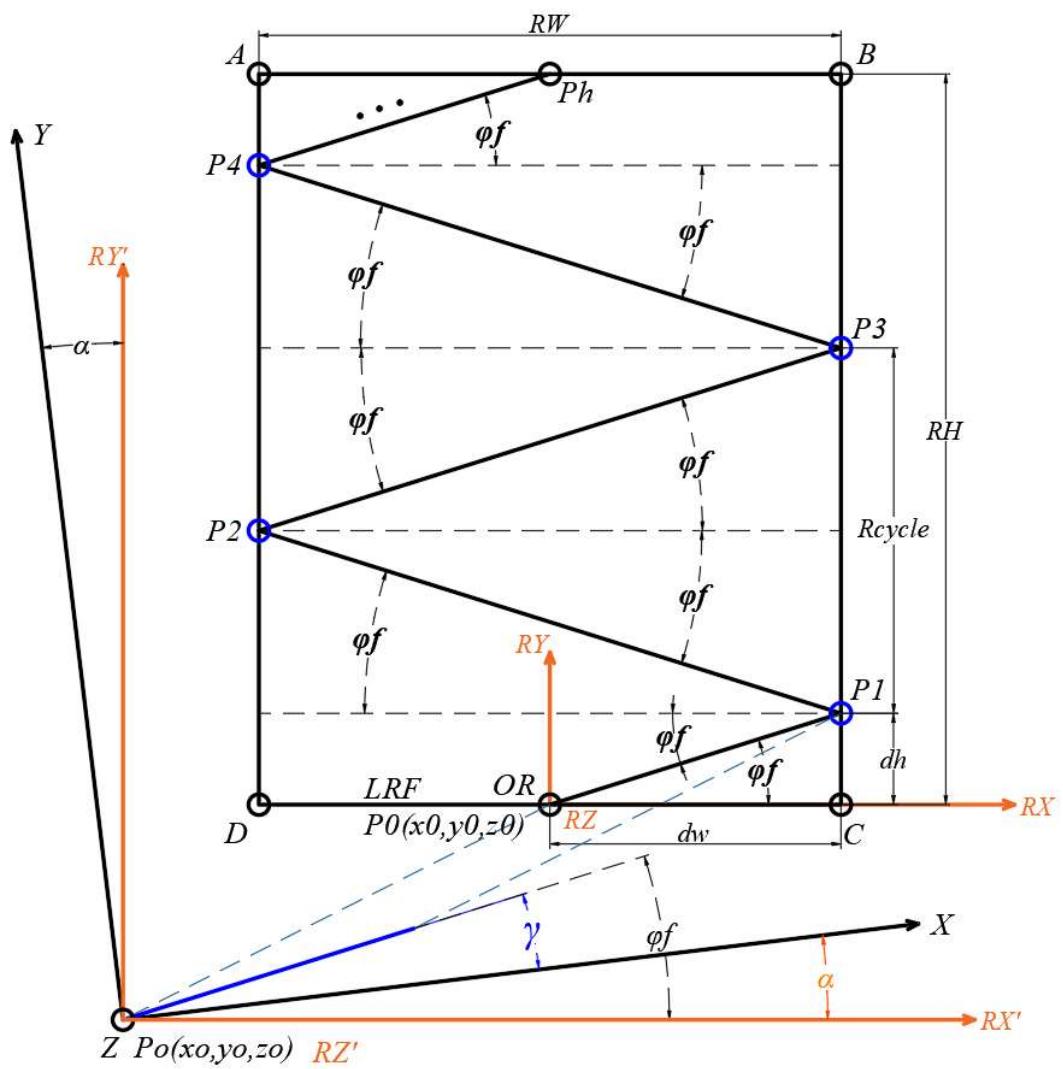


Figure 48. Schematic diagram of a rectangular folding progressive scan algorithm.

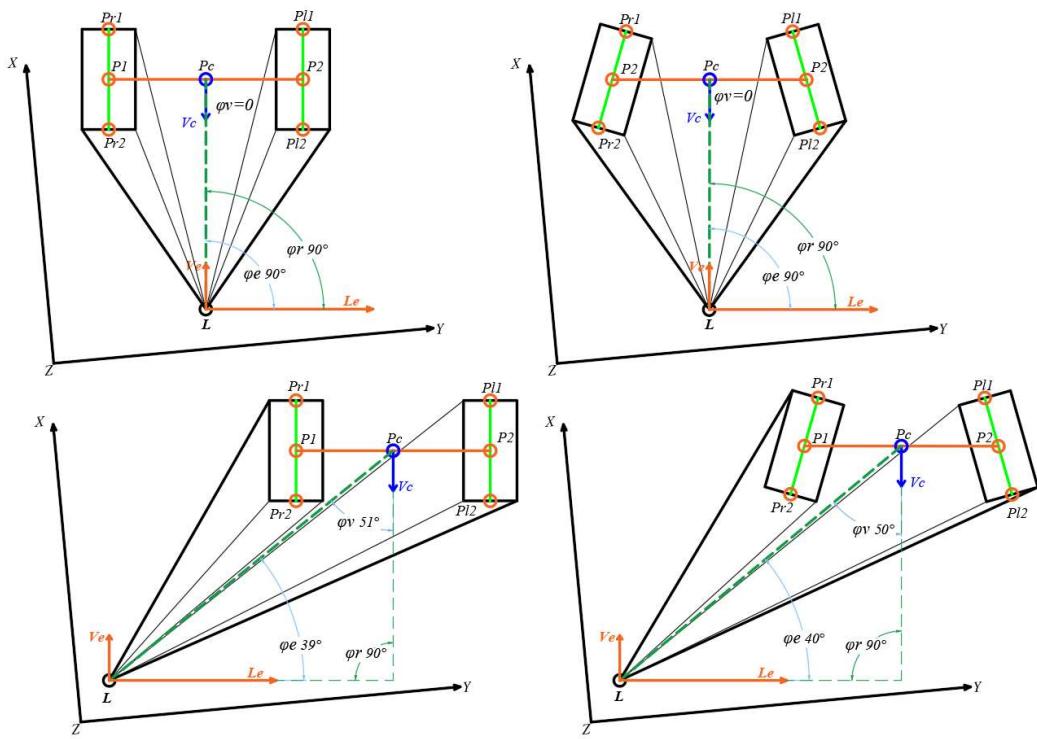


Figure 49. The LRF detects the foot of the target person. The type of possible detection of separate feet. 1.

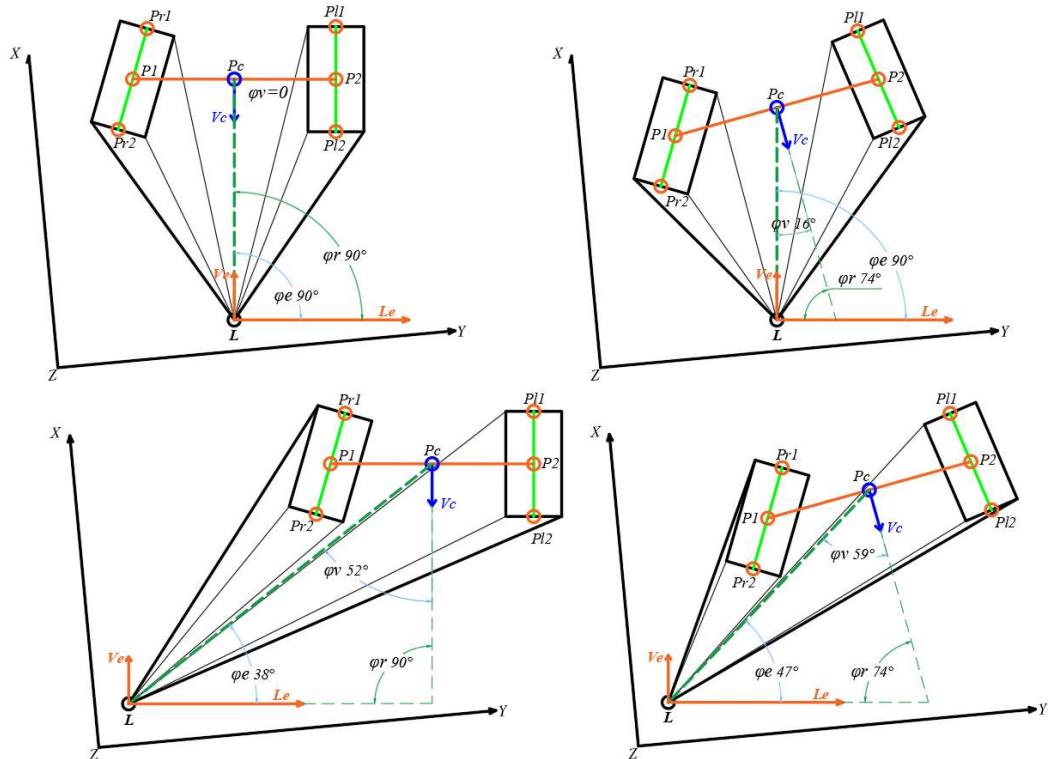


Figure 50. LRF detects the foot of the target person. The type of possible detection of separate feet. 2.

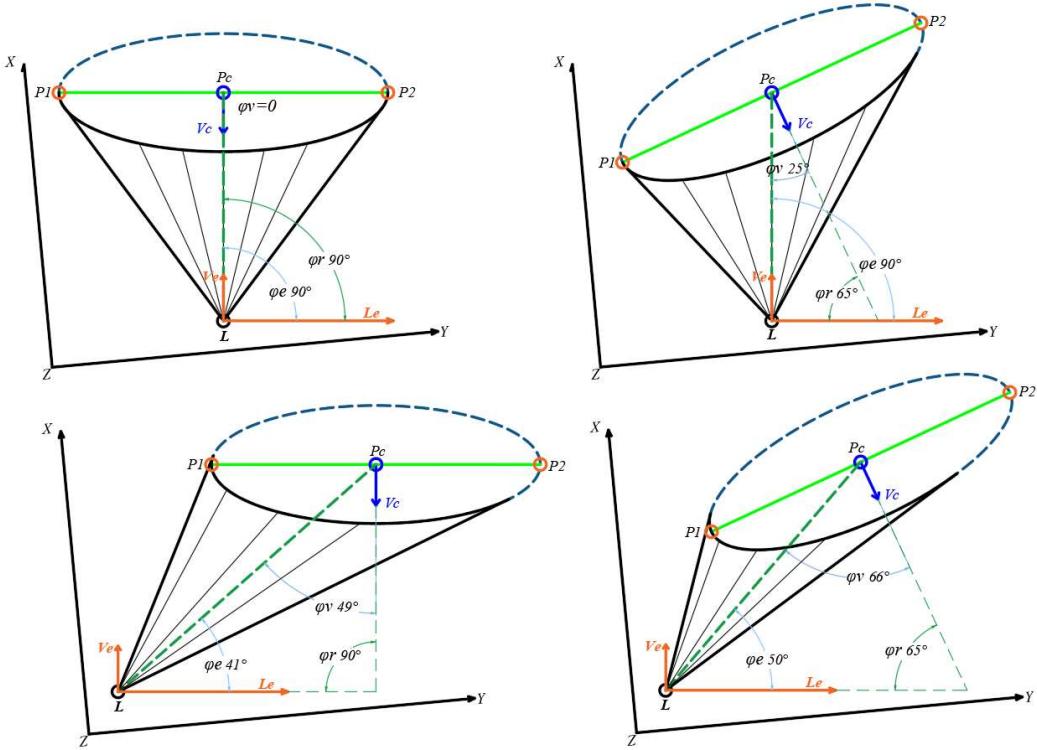


Figure 51. The LRF detects the foot of the target person. The type of possible detection that is obscured by the feet.

DRF OFFICIAL SCANNING ALGORITHM

After execution and processing by the above various algorithms, the robot completes the pre-positioning determination of the target person such that the LRF for scanning faces the detected target person. Thus, a formal DRF scan can be performed. From the preliminary analysis of the previous DRF human body scanning model algorithm, the author knows that when the robot makes the LRF of the DRF model scan radially aligned with the forward direction of the target person, the LRF will start from $\Theta=0$ and along the ground. A scan of the scan rectangle of width WSET is performed.

Combined with the analysis of Fig. 52, the parameters and concepts defined in the preliminary analysis of the DRF algorithm are introduced. The LRF scans the target person directly opposite to the scan width of the scanned DRF with WSET. In Fig. 52, WH is the initial width of the human rectangle. The reason why it is called "initial

"width" is because the width value is an initial value calculated according to the pre-positioning scanning algorithm of the DRF, and is not the final human rectangle width value. It should be noted here that the final width of the DRF scanning rectangle is determined by the object abstraction algorithm after the target person is abstracted into the human rectangle, and the length of the natural horizontal elongation of the two arms corrected by the two-arm posture correction algorithm is determined. This is because the DRF pre-position scans the two-foot area of the target person, and then the author obtains WH according to the following proportional relationship:

The author defines that WF has the following relationship with WH:

$$W_H = \lambda_{fh} \cdot W_F$$

The constant parameter λ_{fh} is an empirical ratio and needs to be obtained after several specific experiments for comparative demonstration. In general, the author suggests that λ should satisfy:

$$4.3 \leq \lambda_{fh} \leq 6.0$$

Then the author gets WSET according to the following proportional relationship:

$$W_{SET} = Dr + W_H + Dl$$

Generally, the author sets Dr to be equal to DL. The author suggests that the setting range of Dr and Dl is:

$$0.5 \cdot W_F \leq Dr = DL \leq 1.0 \cdot W_F$$

According to the preliminary analysis of the previous DRF algorithm, the author knows that after the pre-positioning algorithm in the DRF algorithm is executed, the robot will obtain the detectable width of the part of the foot of the target person, defined as WF. The constant parameter λ_{fh} is an empirical ratio, which is the same as the range of Dr and Dl mentioned above, and needs to be obtained after several specific experiments for comparative demonstration. What the author gives here is only the suggested range and set value. The specific optimum range and set value need to be set according to the actual performance of the robot.

In the DRF, the LRF scans the target person by a rectangular folding progressive scan algorithm. The specific scanning process is consistent with the one mentioned in the preliminary section of the previous DRF algorithm. The authors below will analyze the algorithm turning points that DRF model scans may encounter during implementation.

As shown in FIG. 39, if the author constructs the DRF scan rectangle and the human rectangle by the above-mentioned preset WSET, Dr, Dl and WH, the width

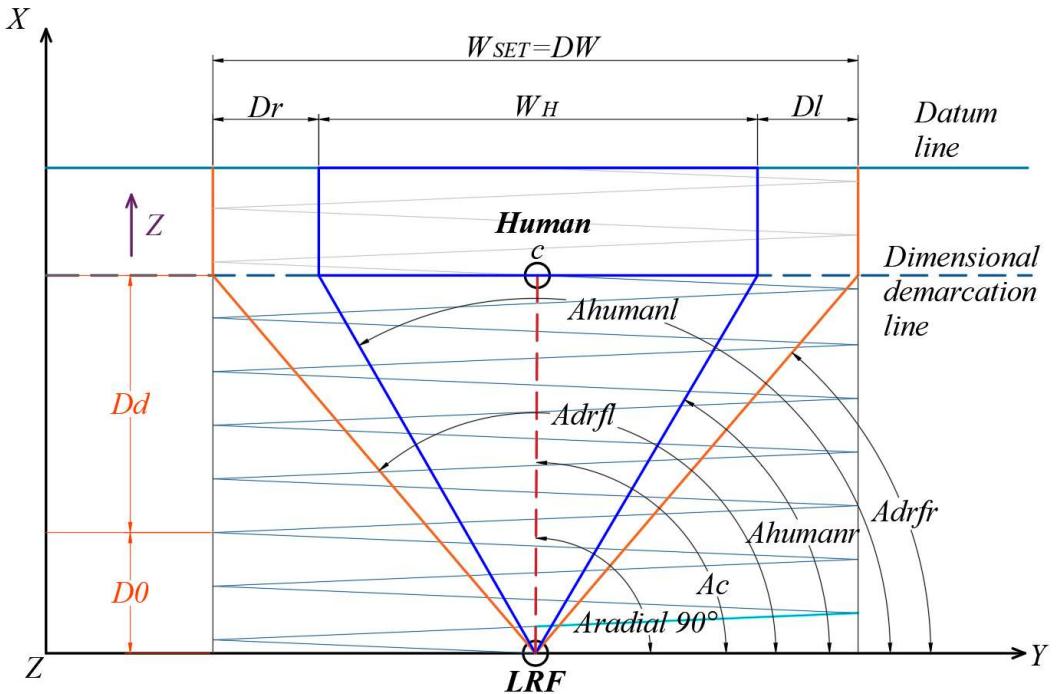


Figure 52. DRF bottom scan. Start with a scan rectangle of width W_{SET} .

setting of the DRF scan can be satisfied, and the width of the DRF scan area is insufficient or the width is unnecessary. Situation, and the target person does not appear in the process of DRF scan execution or the posture is changed or the posture is not standard. Strictly control the arms to be parallel with the support surface, then the robot will smoothly follow the preliminary DRF algorithm. Explain the scanning process mentioned in the description and complete the scanning. Thus, the robot's DRF model scan of the front of the target person is completed. However, in the practical application of the algorithm, the target person is difficult to control his own or her own arms strictly parallel to the support surface. Therefore, it is very necessary to correct the algorithm of the target person's arms posture to obtain the correct width of the DRF scan rectangle.

Below, this article will elaborate on the target person's two-arm posture correction algorithm:

When performing a full-body DRF model scan of the target person, the DRF algorithm combines the object abstraction algorithm to abstract the entire body part of the target person into a simplified human body composed of rectangular blocks as shown in Fig. 39, 35 and Fig. model. The author refers to this simplified model as the human rectangular model. It is obvious that the human rectangle model is a two-dimensional projection simplification of the overall body of the target person. The human rectangle is obtained by a simplified abstraction process of the object abstraction

algorithm. Assume that the scanning algorithm is finally obtained as shown in Fig. 4, the target person's arms posture, and AB is the right arm radial straight line segment of the rectangular mannequin. The CD is a radial straight section of the left arm of the rectangular mannequin with a length of L1. AB is a radial straight line segment of the right arm of the human body rectangle corrected by the algorithm, and the length is Lr. The angle between the straight line segment of the right arm of the human body rectangle before and after correction is φ_r . The following corrections are made:

Judgment and analysis:

First, the two-arm correction algorithm determines if the left arm of the human rectangle is parallel to the support surface. That is, the angle between the radial straight line segment of the rectangle corresponding to the left arm of the human body rectangle and the support surface is determined, and is set to φ_1 (as shown in FIG. 54). But in general, even if the target person is very willing to cooperate with the robot for DRF scanning, they are also difficult to control so that their arms and the support surface are 100% parallel. Therefore, the human body's rectangular two-arm correction algorithm is a very practical and frequently used algorithm. Therefore, the author's minimum expectation for the target person who performs the DRF scan will be reduced to as long as the target person's arms and body are in the same plane, and the arms can be straightened as much as possible, and whether the target person's arms are level with the support surface and Not mandatory.

Due to the symmetry of the human arms, the length of the human left arm is equal to the length of the right arm. By combining the structure of the human body and its symmetry, the author can easily correct the arms of the target person. Since the target person cannot guarantee that the arms are parallel to the support surface during the DRF scan, the author sets a correction algorithm to take the average of the radial lengths of the left and right arms of the target person as the straightening of the real arm of the target person. length.

Enter:

1. Coordinate A of point A (xa, ya, za), coordinate B of point B (xb, yb, zb), coordinate C of point C (xc, yc, zc), coordinate D of point D (xd, yd, zd);
2. The XYZ following coordinates of the above points are positioned and processed by the object abstraction algorithm and provide data.

Calculation:

1. The length of the right arm of the target person:

$$Lr = \sqrt{(xc - xd)^2 + (yc - yd)^2 + (zc - zd)^2}$$

2. The length of the left arm of the target person:

$$Ll = \sqrt{(xa - xb)^2 + (ya - yb)^2 + (za - zb)^2}$$

3. Average length of the left and right arms of the target person:

$$Ld = \frac{Lr + Ll}{2}$$

4. XYZ following coordinates of point A' and point C':

$$\begin{cases} A' = (xb - Ld, yb, zb) \\ C' = (xd + Ld, yd, zd) \end{cases}$$

5. The width of the DRF after the algorithm is corrected:

$$DW = |A' C'| = \sqrt{(xd + Ld - (xb - Ld))^2 + (yd - yb)^2 + (zd - zb)^2}$$

Which is:

$$DW = |A' C'| = \sqrt{(xd + xb + 2Ld)^2 + (yd - yb)^2 + (zd - zb)^2}$$

6. In the true state of the target character, the angle φl between the radial straight line segment of the left arm and the radial straight line segment of the standard horizontal virtual left arm is obtained:

$$\varphi l = \arccos \left(\frac{2Lr^2 - ((xb - Ld - xa)^2 + (yb - ya)^2 + (zb - za)^2)}{2 \cdot Lr^2} \right)$$

7. In the true state of the target character, the angle φr between the radial straight line segment of the right arm and the radial straight line segment of the standard horizontal virtual right arm is obtained:

$$\varphi r = \arccos \left(\frac{2Lr^2 - ((xd + Ld - xc)^2 + (yd - yc)^2 + (zd - zc)^2)}{2 \cdot Ll^2} \right)$$

Update data:

Assign the values of DW to the scan rectangle of the DRF.

Analysis:

Through the above correction algorithm, when the DRF performs a forward scan of the target person, it is not necessary for the target character to strictly keep the arms parallel to the support surface. That is to say, such a double-arm posture as shown in Fig. 54 can also be scanned. However, the author does not recommend that the target person speaks too much about the drooping of the arms when performing a DRF model scan. For example, the situation in which the arms are very close to the torso of the target person or the torso of the target task has not been removed. Of course, if the robot detects that the target person has used a very wrong two-arm motion for DRF scanning, the robot will prompt the target person to correct the wrong two-arm motion by means of voice or LED flashing.

Error action prompt:

If the target person has been informed of the posture that should be maintained during the DRF scan before the DRF model scan, the target person maintains the wrong posture during the DRF scan, and the robot will pose a wrong posture to the target person. Prompt to prompt the target person to perform a DRF scan with the correct two-arm position. On the other hand, the author designed that when the target person performs DRF scanning, the target person's arms should preferably be horizontal with the support surface, in order to reduce scanning errors and obtain more complete and accurate scan data. Moreover, the DRF does not take a long time to scan forward, so it does not cause a lot of interference caused by the user experience of the robot. Unless the target person deliberately conducts motion interference to test the robot, in this case, the robot will issue a maximum of three prompts. After three prompts, the target person will deliberately not cooperate with the robot's DRF scan, and the robot will automatically terminate the DRF scan and follow the person. In order to ensure the best user experience of the target person, the author sets the DRF scanning algorithm to make an error prompt to the target character only when the target person's arms are seriously irregular.

The author judges whether the target person's arms movements are serious or not. The principle is:

1. Determine the degree of parallelism between the left arm of the target person and the support surface by detecting the angle φ_l between the radial straight line segment of the left arm and the radial straight line segment of the virtual left arm of the standard level in the true state of the target person.
2. Determine the degree of parallelism between the right arm of the target person and the support surface by detecting the angle φ_r between the radial straight line segment of the right arm and the virtual horizontal straight line segment of the arm at the true state of the target person.
3. Set the maximum deviation allowable angle χ_{max} , that is, if the angle φ_l and the angle φ_r satisfy the following relationship:

$$\varphi l, \varphi r \in [0, \chi max]$$

It is considered that the posture position of the target person's left arm and right arm is within the allowable range of the DRF scanning algorithm.

If

$$\varphi l > \chi ma$$

It is considered that the target person's left arm posture is wrong, and the robot will give a posture error prompt.

If

$$\varphi r > \chi max$$

It is considered that the target person's right arm posture is wrong, and the robot will give a posture error prompt.

DRF scan rectangle width automatic adjustment algorithm:

After the object abstraction algorithm and the DRF frontal scanning algorithm described above, the author can determine the optimal width DW and the optimal height DH of the DRF front scan rectangle. However, one author cannot ignore the fact that how to make the initially set DRF scan rectangle width WSET must be greater than the width DW of the algorithm-corrected DRF scan rectangle. Although the author provided the range of the proportionality factor λ_{fh} in the initial setting of WSET, it is still not guaranteed 100%. In some cases, for example, the length of the arms of a certain target person has the same width of the foot area as the others. The longest target of WF. At this time, there may occur a case where the initially set DRF scan rectangle width WSET is smaller than the optimum width DW of the finally set DRF front scan rectangle. Although the probability of this occurrence is very low after adjusting the optimal value of the proportional coefficient λ_{fh} in many experiments, in order to make the DRF scanning algorithm designed by the author more robust, the author still needs to consider this problem.

Assuming the WSET obtained by the DRF algorithm scanning algorithm by scanning the pre-positioning algorithm can satisfy the horizontal stretch width calculated by the target person's arms after the correction algorithm, the DRF scanner will determine the DRF scan rectangle according to the DRF rectangle determination algorithm described above. As shown in Figure 55. Point P1 is an LRF scan

When the light scans the arms of the target person from right to left, the folded scanning light first contacts the position of the target person's arms. (NB: The density of the actual scanned folded light is much denser than that shown in Figure 55.) After the scanning contact point P1 of the LRF, the distance detected by the LRF decreases, and then the light extends all the way to the P2 point. After the P2 point, the detection distance of the LRF scanning light increases, and the P2 point is obtained as an edge

point of the right arm of the target person. Then the LRF scanning light is folded and scanned in the opposite direction. After the LRF scanning light reaches the point P3, if the LRF scanning light moves to the right a little, the distance detected by the LRF is reduced, so the algorithm determines that the point P3 is the left of the target person. One edge of the arm. Because the LRF scanning light actually performs multiple round-trip folding scans on the target person's arms, multiple left arm edge points and right arm edge points are obtained. In order to reduce the detection error, the authors represent all the left arm edge points as a multi-point set SDPEL, and all the right arm edge points as a multi-point set SDPER, and there are:

$$\begin{cases} SDPEL = \{Pl1, Pl2, \dots, Pln\} \\ SDPER = \{Pr1, Pr2, \dots, Prm\} \end{cases}$$

In the above formula, the SDPEL is composed of n left arm edge points, and the SDPER is composed of m right arm edge points. Then the author takes the arithmetic mean of the n edge points in the SDPEL as the representative of the left arm edge point of the target character, and sets it as the point Pla, taking the arithmetic mean of the m edge points in the SDPER as the left arm edge point of the target person. The representative, set to point Pra, and there are:

$$\begin{cases} Pla[x] = \frac{\sum_{i=1}^n Pli[x]}{n} \\ Pla[y] = \frac{\sum_{i=1}^n Pli[y]}{n} \\ Pla[z] = \frac{\sum_{i=1}^n Pli[z]}{n} \\ Pra[x] = \frac{\sum_{i=1}^m Pri[x]}{m} \\ Pra[y] = \frac{\sum_{i=1}^m Pri[y]}{m} \\ Pra[z] = \frac{\sum_{i=1}^m Pri[z]}{m} \end{cases}$$

In the above formula, x, y, and z enclosed by "[]" represent the x, y, and z components of the corresponding points. For example, Pli[x] represents the x-coordinate component of point Pli. Thus, through the above algorithm, the author will be able to find the coordinates of the point Pla and the point Pra, so the author can get the optimal width DW of the DRF scanning rectangle:

$$DW = \sqrt{(Pra[x] - Pla[x])^2 + (Pra[y] - Pla[y])^2 + (Pra[z] - Pla[z])^2}$$

However, if a certain target person has a longer arm length than other target persons having the same foot area width WF. At this time, there may occur a case where the

initially set DRF scan rectangle width WSET is smaller than the optimum width DW of the finally set DRF front scan rectangle. At this time, the DRF scanning algorithm is required to automatically adjust the width of the DRF rectangle. As shown in FIG. 56, the width of the initially set DRF scan rectangle is DW1. When the LRF scans the target person with the scan rectangle of width DW1, the two-arm posture correction algorithm actually detects the horizontally extended arms. The length is D2 and there is D2>DW1. Therefore, the original WSET is not suitable for the scanning of the target person, and the automatic adjustment algorithm will automatically use the newly scanned D2 as the target person's arms. That is, the width of the new DRF scan rectangle is updated to:

$$DW2 = Dr + D2 + Dl$$

As shown in Fig. 56, the LRF scanning light determines the Dr from the point P2 to the point P3 by the distance change detected by the LRF, and determines the distance D1 detected by the LRF from the point P4 to the point P5. The scanning trajectory of the LRF is determined by the LRF spatial straight line segmentation scanning algorithm. The entire DRF model scan is finally completed.

The effect of the shadowed portion of the DRF on the front side of the DRF scan:

It can be known from the previous analysis that when the DRF performs the model scan, the foot width WF of the target person is first calculated by the pre-positioning algorithm. In general, the target person's feet are not obscured by the target's clothing, but if the target wears some clothing that will cover the feet, it will make the target's foot area The scan width WF is increased. That is to say, in this case, the WF obtained by the author is larger than the real WF. In another case, the detected target person's feet are separated, but the WF is too large. For this case, the designer can determine whether WF is true by setting the maximum recognition threshold width of WF. But because the author uses an object abstraction model, it is much easier to set up better algorithms to deal with this problem. The DRF model algorithm combined with the object abstraction algorithm will be very simple to deal with this problem. After combining the object abstraction algorithm, the DRF scanning algorithm will deal with this problem as "no processing." That is to say, the DRF scanning algorithm will scan the detected WF directly as a real WF. The most realistic WF is obtained by an automatic adjustment algorithm that scans the width of the rectangle by DRF. With the addition of object abstraction algorithms, the DRF algorithm will become more robust. From another perspective, the designer will also find that the designer ultimately wants to determine the regional distribution of the DRF scan rectangle, which is to determine DW and DH. The DW is determined by the width of the target person's arms extending horizontally plus Dr and Dl, and DH is determined by direct scanning of the LRF scanning rectangle. Therefore, WF only provides a reference WSET width for the DRF scan rectangle,

which does not strictly determine the width and height of the final DRF scan rectangle. If the initial setting of WF is relatively large, the DRF will set a larger WSET value, and finally it will be adjusted to the most reasonable width by the DRF scanning rectangle width automatic adjustment algorithm. As shown in Fig. 57, the DW and DH of the final DRF scanning rectangle do not change regardless of whether the target person's feet are covered by the clothing.

DRF side and back model scanning:

A complete DRF human body model scan should include at least two human scan positions, one on the front of the target person and the other on the side of the target person. Through the scanning positions of the two target characters, the scanning algorithm can approximate the 3D model of the target person and obtain relevant data information. The designer can also add a back scan of the target person for the DRF scan. This will allow the robot to acquire more 3D model information and data of the target person.

That is, by adding a scan of different positions, the robot will acquire more data about the 3D model of the target person, which is obviously more advantageous for establishing a more accurate DRF human scan model. However, the more scan positions that are not scanned for the target person, the better. More scan positions mean more DRF scans, and more scans means that the robot will delay waiting for the target person. time. Of course, the author can reduce the time the target person needs to wait by optimizing the algorithm and upgrading the performance of the LRF. When the robot performs multi-position DRF model scan on the target character, the robot should be rotated as much as possible around the target person to change the scanning position of the robot relative to the target person, thus minimizing the target person's DRF scan time. The increase is caused by an unsatisfactory user experience. To achieve this goal, the robot's motion control algorithm must be combined with the LRF DRF model and scanning algorithm for comprehensive design. Due to the limited writing time of this report, the detailed DRF multi-position scanning algorithm will be presented in subsequent research reports.

As shown in Figure 53., when the robot performs a model scan behind the DRF, the target person is also required to extend the arms to the left and right sides and to be parallel to the support surface. This is to reduce the impact of the target person's arms on the DRF model scan when model the target person. When the DRF scans the target person sideways, the robot mainly wants to obtain information on the side width of the target person. When performing a side scan, the target person is also in the Area 21 area. When the robot performs a back-end DRF model scan of the target character, the target person is also in the Area 21 area. The author's design is that when the robot performs DRF scans of other locations on the target person, the target person should always be in the Area 21 area. When the robot performs a side DRF scan on the target person, the arms of the target person can be unfolded parallel to the support surface, or can be naturally lowered and merged together so as not to affect the scanning. As mentioned

previously, the multi-position DRF scan should include at least a front position scan and a side position scan.

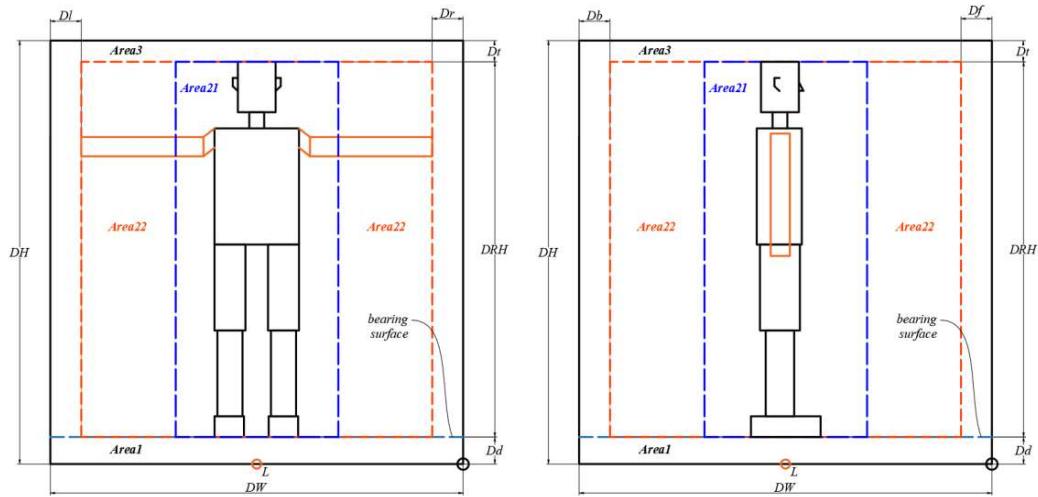


Figure 53. Distribution of scan areas for back and side scans of the DRF.

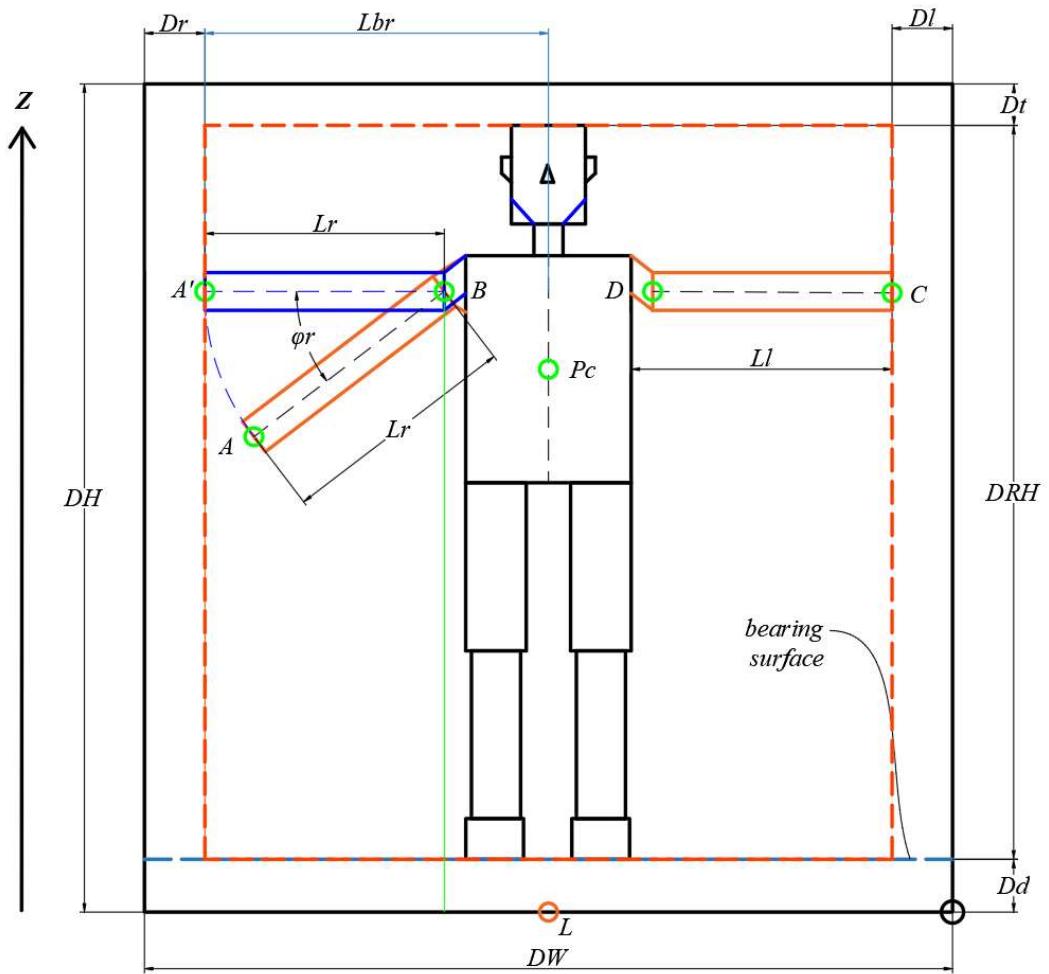


Figure 54. Algorithmic correction of the irregular two-armed gesture of the target person during the DRF scan. The non-standard of the posture of the single arm of the target person is detected.

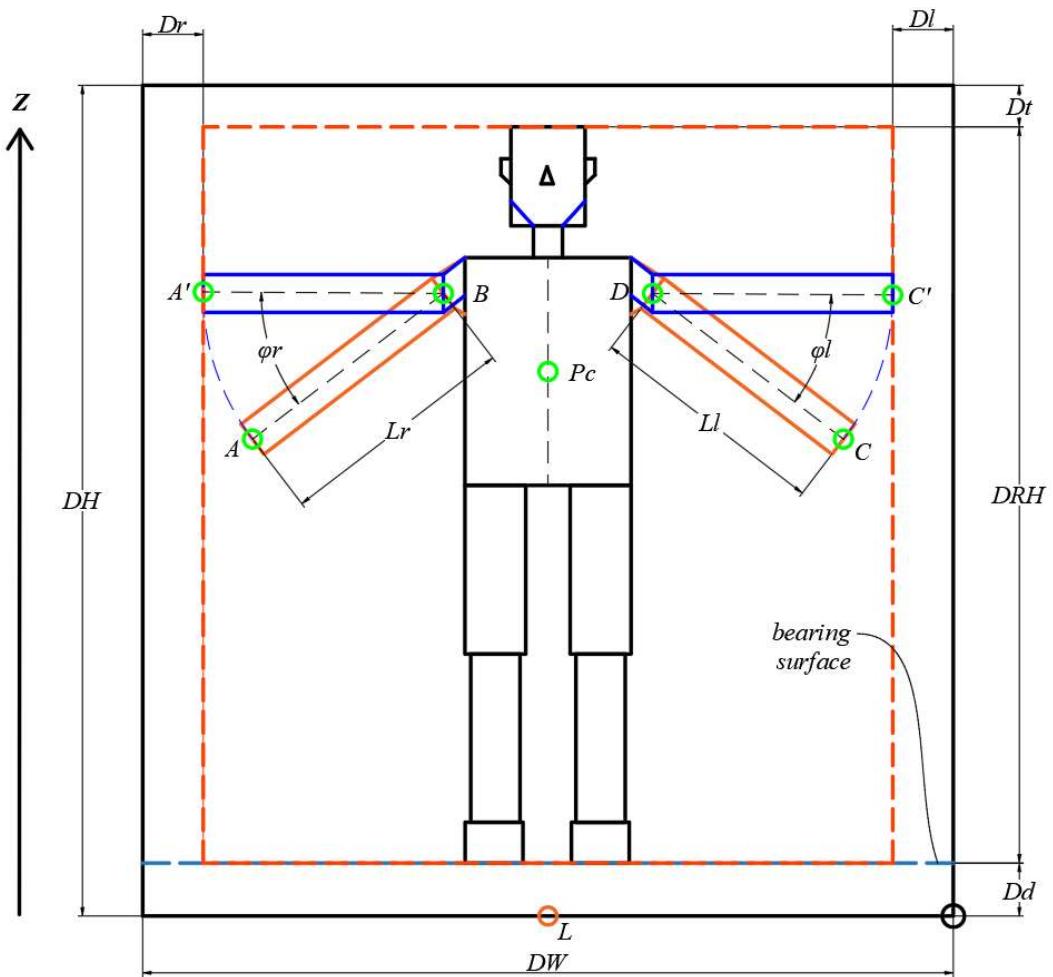


Figure 55. Algorithmic correction of the irregular two-armed gesture of the target person during the DRF scan. The posture of the target person's arms is not standard.

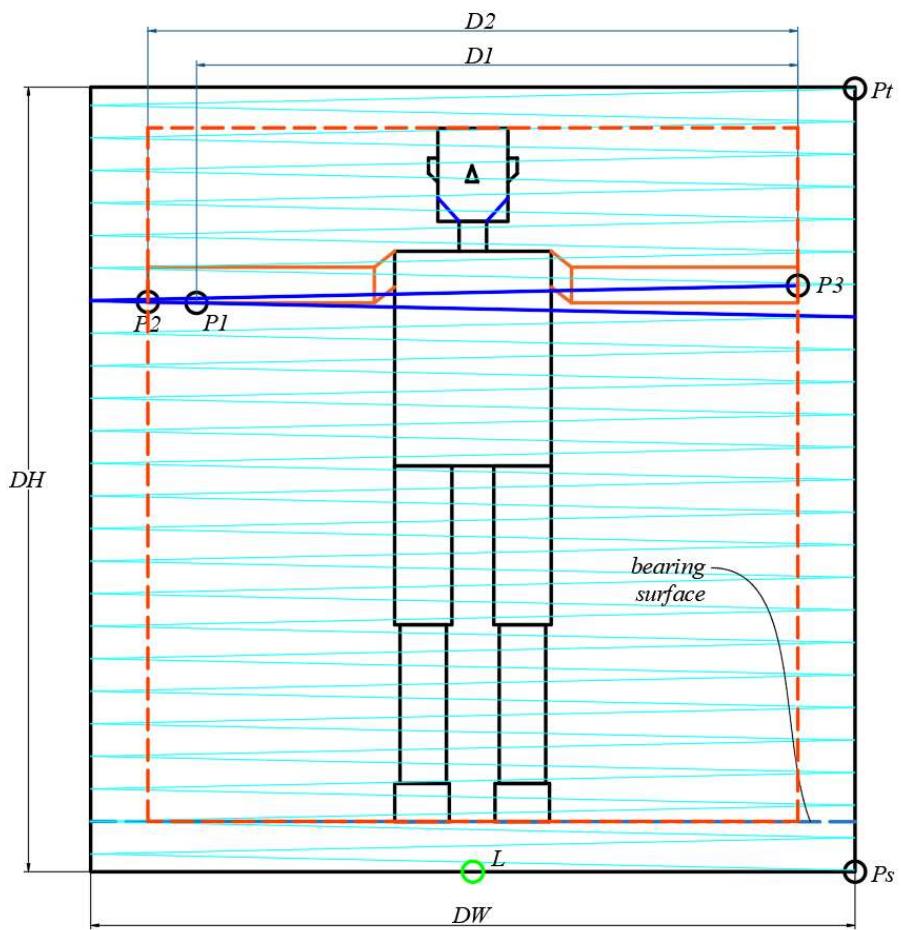


Figure 56. Normal width and height settings for the DRF scan model.

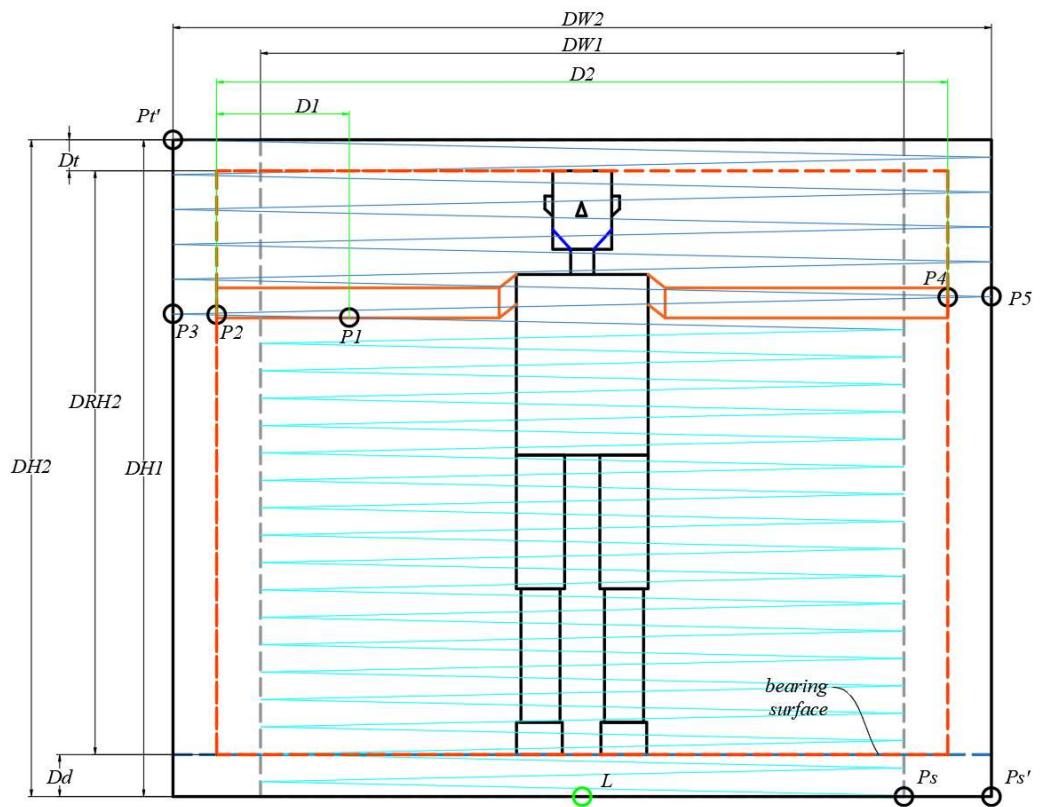


Figure 57. During DRF scanning, WSET is less than the length of the horizontal extension of the arms of the target person.

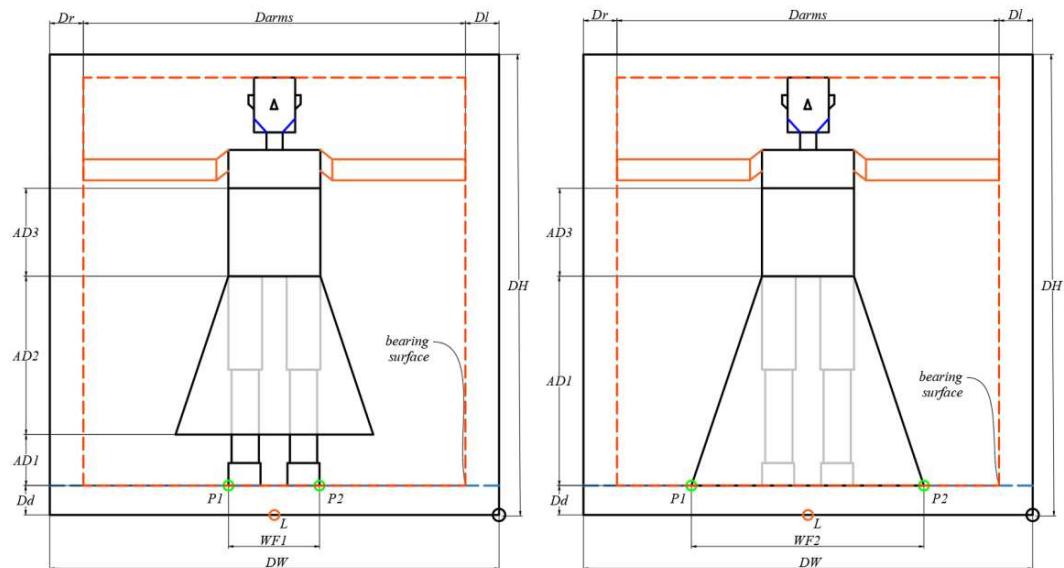


Figure 58. WF has no effect on the width and height of the final DRF scan rectangle.

For more exciting details content, please refer to the other articles of this essay:

1. ROBOT DETECTION SYSTEM DESIGN ABOUT FRONT-FOLLOWING TECHNOLOGY PART 1
2. ROBOT DETECTION SYSTEM DESIGN ABOUT FRONT-FOLLOWING TECHNOLOGY PART 3
3. ROBOT DETECTION SYSTEM DESIGN ABOUT FRONT-FOLLOWING TECHNOLOGY PART 4
4. ROBOT DETECTION SYSTEM DESIGN ABOUT FRONT-FOLLOWING TECHNOLOGY PART 5

References:

- Ho, D. M. , Hu, J. S. , & Wang, J. J. . (2012). Behavior control of the mobile robot for accompanying in front of a human.
- Jung, E. J. , Yi, B. J. , & Yuta, S. I. . (2012). Control algorithms for a mobile robot tracking a human in front. IEEE/RSJ International Conference on Intelligent Robots & Systems. IEEE.
- Hu, J. S. , Wang, J. J. , & Ho, D. M. . (2014). Design of sensing system and anticipative behavior for human following of mobile robots. *IEEE Transactions on Industrial Electronics*, 61(4), 1916-1927.
- Moustris, G. P. , & Tzafestas, C. S. . (2017). Intention-based front-following control for an intelligent robotic rollator in indoor environments. *Computational Intelligence*. IEEE.
- Nikdel, P. , Shrestha, R. , & Vaughan, R. . (2018). [IEEE 2018 IEEE International Conference on Robotics and Automation (ICRA) - Brisbane, Australia (2018.5.21-2018.5.25)] 2018 IEEE International Conference on Robotics and Automation (ICRA) - The Hands-Free Push-Cart: Autonomous Following in Front by Predicting User Trajectory Around Obstacles. (pp.1-7)

Project Research Report

ROBOT DETECTION SYSTEM DESIGN ABOUT FRONT-FOLLOWING TECHNOLOGY

PART 3

**Jinwei Lin
(JY Lin, Shenzhen, China)**

January 19, 2019

ID ORCID: <https://orcid.org/0000-0003-0558-6699>

<http://www.ydook.com/>

OBJECT ABSTRACTION ALGORITHM

I. Analysis

This algorithm is the key algorithm for the implementation of this project, and it is also one of the most important algorithms of the robot of this project. Designers can abstract real objects in 3D space into simple geometry through a variety of algorithms. Due to the time constraints of writing, this version of the research report proposes an object abstraction algorithm that is the easiest to implement and is most suitable for the robot of this project.

Because the personal privacy of the target person needs to be ensured to the greatest extent, this project does not have a camera. That is, the designer's robot cannot obtain the target person and the surrounding environment by analyzing the image data of the pixels arranged in the network format. Information. However, the designer has a very useful hardware that can be used to obtain the detection data of the target person and the environment around the robot, which is the LRF. By the scanning rectangle algorithm described in the previous section of this essay, the designer can design the scanning action of the LRF to scan a certain spatial area in the form of a cube. By scanning the rectangle through the DRF, the designer can scan the target person and the surrounding environment of the robot using the DRF scan rectangle in real time. Through the designer's combined coordinate system and scanning algorithm, the designer can accurately acquire and accurately locate any detectable point in the LRF scanning area. Through the above method, the designer can accurately acquire the information of the detection point of the detectable object in the detection area. But it's not enough to just collect the data, and to the designers, they have to process the data. The previous section covered a lot of data processing methods, and many methods involve this algorithm. It can be said that this statement is a very important algorithm in this project. The following is a detailed algorithm analysis and description.

Discussion and analysis on the characteristics of LRF scan coverage

Due to the design structure and working principle of the current LRF, the current stable and mature LRF in practical applications adopts a single-point emission scanning mode. Combined with the analysis in the previous section, the designer knows that due to the design and distribution of the LRF structure of the project, the LRF in the LRF group can only scan the detected area within a half-circle of 180°. Therefore, in order to fully enable the LRF to perform its important detection performance, the designer must analyze the single-point scanning mode of the LRF.

As shown in Figure 70. Assume that P1P2P3 is the real edge of the detected object. When the LRF detects the environment with a radial probe beam, it will appear because the LRF probe beam can only detect the front and cannot detect the back. In the same

set of scanning points of the LRF, the scanning point at the front will cover the scanning point at the back. This phenomenon is called the scanning coverage phenomenon of the LRF.

So, when the designer knows that the LRF has scan coverage, how can the designer design the algorithm so that the LRF can perform the maximum detection? Because the main function of the robot of this project is to follow the target person and make a reasonable judgment when encountering the situation of making a left turn or a right turn or going straight. Therefore, the designer does not need to be very clear about every detail of the external form of the character. The designer's detection and data analysis processing algorithm is concerned with how to identify the detected object and abstract it when some need to be carefully identified. Simple identification of the geometry. On the other hand, because the LRF scan speed is very fast, the designer's scan data processing algorithm needs to process the real-time scan data, so at least one time should be performed when the LRF completes a 180° half-circle scan period. Processing and analysis. Therefore, when using the current technically mature LRF for scanning work, the designer has no time, and the author does not recommend performing the scan data reconstruction after the overall scanning of the detected object, and then performing key scan data analysis and processing. Obviously, this will cost more computing resources and processing time, and it will not help the final analysis results.

After the designer has determined that only the scan segments of the LRF in a 180° scan period are analyzed, the overall design will enter the most critical discussion: even for the scan segments within a 180° scan period of the LRF, the designer still The impact of the scanning coverage of the LRF on the scanned data needs to be considered. To this end, the author has developed a strategy to deal with LRF scan coverage, which is to analyze and process only the data obtained by real-time scanning. This is a simplified way of handling and the most efficient way to handle it. The LRF scan detection has a relatively fixed phenomenon, that is, at the same moment, the edge of the detected object that the LRF can scan is fixed at the same scanning instant, even in the same scanning period of the LRF, because the scanning speed of the LRF is fast. It can be considered that the edge of the detected object obtained by the LRF scan is almost constant, so the author has fixed the scan data obtained by the LRF in the same scanning period. From the perspective of algorithm design, designers can design more advanced scanning data processing methods to reduce the error caused by LRF scanning coverage. The implementation principles of these methods can be divided into two categories, one is to use the principle of virtual deduction algorithm. Deriving the most probable edge data of the detected object from the scan data that has been obtained so far, this method does not need to re-detect the time and space of the detected object, but since the predicted data is derived from the known data, the simulation is performed. Come out, so the final simulation results are inaccurate with the real results.

Another type of processing method is to enable the robot to re-detect the detected object by replacing the time and space to supplement the scanned data of the detected object. The advantage of this type of processing method is that the data that is replenished is the real data of the detected object, so the detection error will be smaller,

but there is also a need for more time and operations to replace the space and time, thereby re-performing the detected object. Detection. Based on the above analysis, the author does not recommend that the robot scanning algorithm of this project adopt an algorithm which is similar to the above-mentioned scanning coverage reduction phenomenon of LRF.

Description and discussion of the algorithm diagram of this part

In the above discussion, the author explains why data processing is performed directly on the scanned data points of the LRF without using other more advanced algorithms. If the designer stratifies the scanning point of the detected object along the direction of the beam of the LRF, it is clear that the designer obtains the detection data of the first layer of the detected object. Moreover, in the scanning abstract algorithm of this project, the designer only needs the scan detection data of the first layer of the detected object. In order to better explain this part of the algorithm, the author draws some vivid and concise schematic diagrams. In these diagrams, the author uses successive curve segments to represent the true edges of the target object, using small circle key points or scan points. In some specific diagrams for analyzing the scanning line segments, the author stretches and unfolds the original arc-shaped scanned image to display the data information of the three-dimensional scanning line segments in a more intuitive two-dimensional form. Such image result processing allows the author to more directly analyze the specific scan line segment data. And with line spectrum features, it is easier to analyze. In the schematic diagram of such stretched deployment, the author still strictly follows the characteristics of the scanning coverage of the LRF. In some diagrams, the authors use a side-by-side set of straight segments to represent the division of the LRF scan interval. This side-by-side segmentation of the line segment also contributes to the analysis and presentation of the LRF scan coverage characteristics.

Abstract the object to be detected into a polyhedron composed of spatial cubes

Because the appearance of the detected object is various, and the appearance of different detected objects is also very different, and the main function of the robot of this project is to follow the target character forward, the author does not need to In the process of executing the following characters, the robot makes the robot have high recognition ability for different detected objects, but the author cannot completely ignore the appearance characteristic data information of the detected object. Therefore, in order to enable the robot to efficiently perform the characters following the target person, the author proposes an object abstraction algorithm. The implementation principle of the object abstraction algorithm is that, in the case of some very specific

appearance data that does not require the detected object, the detected object with complex appearance characteristic data is geometrically abstracted and simplified into a polyhedron or combination composed of a simple geometric space. Body for subsequent algorithm analysis and processing. From the author of the previous section, the authors determined that the object to be detected is abstracted into a polyhedron composed of a plurality of cubes in space. The author named this algorithm for abstracting a detected object into a polyhedron composed of a spatial cube as an abstract cube transformation algorithm. The following is a detailed analysis and introduction to the abstract cube conversion algorithm:

As shown in Figure 60, L_e is the direction of the abstract edge of the robot rectangle where the LRF is located. The object being detected is represented by a closed spatial curve segment. The P1P2 curve segment is a scan line segment obtained when the LRF scans the detected object at a certain time. The rectangle P1P2P3P4 is a projection of the spatial cube abstracted by the detected object on the XY plane. As can be seen from Fig. 60, at the time of detection, only the edge line segment P1P2 is actually scanned by the LRF. Assume that the entire algorithm discussion takes place in the XYZ following coordinate system. That is to say, in fact, the robot only obtains the P1P2 scanning edge curve segment at the scanning time by the LRF scanning. Another author needs to be clear that the important premise is that if the author wants to abstract the object to be detected into a spatial cube, he must know the length, width and height of the abstracted space cube of the detected object. For the LRF, only the scanning edge corresponding to the length or width of the other detected object can be acquired in one scanning cycle. The height of the detected object requires the LRF to complete the overall scan of a detected object before the relevant data can be obtained. Therefore, the data in the LRF abstract algorithm is actually missing a dimension of data. In the author's robot project, the author can obtain the length, width, and height information of the target character by performing a DRF modeling scan of the target character before the robot performs the following person, so that the target task can be used in the abstract algorithm of this part. The length or width information to help build the abstract rectangle of the current scan bevel of the target character. Since the length and width of the unrelated object are variable and the data is not necessary for the project, the author's analysis focuses on the construction of the spatial abstract cube of the target character. For irrelevant objects to be detected, the author's suggestion here is to construct an abstraction cube that is independent of the detected object and that is known to know the height, and that knows one dimension of the length and width, because of the environment in which the target person is located. The abstract meaning of the irrelevant detected object is not large, so the author will omit the extension of this algorithm analysis in this version of the project research report. The abstract algorithm described below will only be elaborated and analyzed for the target person.

The following author will describe how to abstract the abstract rectangle P1P2P3P4 through the curve segment P1P2:

First, assume that the target character acquired by the robot in the DRF modeling scan performed on the target person before performing the following person is W_p and has a length of L_p .

Then, assume that the coordinates of the point P1 are P1 (x1, y1, z1), P2 (x2, y2, z2), and the angle between the P1P2 and the Y axis is θ_1 . It is assumed that the coordinates of the point P3 are P3 (x3, y3, z3), the coordinates of the point P4 are P4 (x4, y4, z4), and the length of the P2P3 side of the abstracted rectangle is l23.

Then θ_1 is:

$$\theta_1 = \arcsin \frac{x_1 - x_2}{\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}}$$

Then, the straight line distance of P1P2 is:

$$l_{12} = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2}$$

At this point, the author has to judge that the scan curve of the P1P2 scanned by the author corresponds to the length of the target person or the width of the target person. If

$$|Wp - l_{12}| > |Lp - l_{12}|$$

Then the P1P2 curve segment corresponds to the length of the target person, and the following assignments are made:

$$\begin{cases} l_{12} = Lp \\ l_{23} = Wp \end{cases}$$

If

$$|Lp - l_{12}| > |Wp - l_{12}|$$

Then the P1P2 curve segment corresponds to the width of the target person, and the following assignments are made:

$$\begin{cases} l_{12} = Wp \\ l_{23} = Lp \end{cases}$$

If

$$|Wp - l_{12}| = |Lp - l_{12}|$$

It means that the true width and length of the target person are very similar. Although the probability of this happening is very low, it still occurs and exists. At this time, the designer can assign any one of Wp or Lp to l12. That is, the assignment operation of

both cases can be performed.

And because the points P1, P2, P3, and P4 are on the same scanning chamfer plane, in order to simplify the data analysis and processing of the algorithm, the author defines:

$$\begin{cases} z3 = z2 \\ z4 = z1 \end{cases}$$

So there will be:

The coordinates of point P3 are:

$$\begin{cases} x3 = x2 + l23 \cdot \cos\theta1 \\ y3 = y2 - l23 \cdot \sin\theta1 \\ z3 = z3 \end{cases}$$

The coordinates of point P4 are:

$$\begin{cases} x4 = x1 + l23 \cdot \cos\theta1 \\ y4 = y1 - l23 \cdot \sin\theta1 \\ z4 = z1 \end{cases}$$

Thus, by this step, the author can completely determine the abstract rectangle P1P2P3P4 represented by the scan curve segment P1P2.

It can be seen from the analysis that the scanned abstract rectangle shown in Fig. 60 is low on the left side and high on the right side, and the scan abstract rectangle shown in Fig. 61 is the case where the left side is high and the right side is low. Therefore, in Fig. 61, $\theta1$ is a negative value. Therefore, it still has the following relationship:

The coordinates of point P3 are:

$$\begin{cases} x3 = x2 + l23 \cdot \cos\theta1 \\ y3 = y2 - l23 \cdot \sin\theta1 \\ z3 = z3 \end{cases}$$

The coordinates of point P4 are:

$$\begin{cases} x4 = x1 + l23 \cdot \cos\theta1 \\ y4 = y1 - l23 \cdot \sin\theta1 \\ z4 = z1 \end{cases}$$

Assumptions and conclusions are established.

It can be known from the above analysis that when the algorithm of this part performs spatial rectangle abstraction on the detection part of the target character, only

one scanning curve segment of the target character is abstracted. So, does it mean that the designer only needs to analyze and analyze a scanned curve segment of the target person's detected part? In fact, in general, the designer only needs to analyze and analyze a scan curve segment of the detected part of the target character, because the robot has obtained the specific values of W_p and L_p in the DRF modeling scan. In general, the specific values of W_p and L_p are not changed during the robot's execution of the follow-up task for the target person. Even if it changes, it will have little effect on the final scan and positioning results of the target person.

However, in practice, if the absolute value of the angle formed by one side of the abstract rectangle and the Y axis is closer to 90 degrees, as shown in Fig. 61, the algorithm may face a false edge in a specific implementation process. As shown in Fig. 61, the scanning curve segment of the original LRF increase and decrease detection is P2P3, but in the end, the LRF can only detect P2P5, which causes an error in determining the final abstract rectangle. In fact, in the process of scanning the scanning curve segment corresponding to P1P2 by LRF, it is also possible to encounter this total false edge. Therefore, in order to eliminate the error caused by such false edge phenomenon, the author introduces the following judgments and Processing mechanism:

Judgment:

If there are two expressions in the following inequality group:

$$\begin{cases} |L_p - l_{12}| > \tau \\ |W_p - W_{12}| > \tau \end{cases}$$

It is said that the LRF scan encountered a situation where the false edge was scanned. τ in the above formula is an experimental empirical constant determined by multiple experiments.

The false edge phenomenon of the scanning of the LRF is caused by the absolute value of the angle between the representative straight line segment P1P2 and the Y axis of the scanning curve segment being too close to 90 degrees. As shown in FIG. 61, since the purpose of the author design algorithm is to abstract the detected portion of the target person into a corresponding rectangle, if the absolute value of the angle formed by one side of the abstract rectangle and the Y axis is closer to 90 degrees, according to the rectangle The characteristics of the two adjacent edges, then the absolute value of the angle formed by the other abstract edge of the abstract rectangle and the Y axis must be closer to 0 degrees. If, therefore, the absolute value of the angle of an abstract edge of the abstract rectangle with the Y axis is close to 0 degrees, then the scan curve segment corresponding to the edge of the abstract line segment is an excellent curve segment for abstraction. That is to say, in the process of scanning by the LRF, if the edge of a rectangle of the abstract rectangle appears to scan the false edge, it is only necessary to segment the scan curve to be processed into another curve segment to be

scanned.

Discussion and analysis on the edge identification of detected objects

The processing of edge problems is the key to this algorithm. Because the author's amount of detection sensor is LRF. The LRF is a distance detector. Analogous to 3D images, LRF can provide and only provide depth information for 3D images. Since the LRF can only detect the distance information, the algorithm needs to convert the detected distance information into similar image information, and then process similar image information to convert similar image information into basic geometry. Data information. One of the key to achieving this goal is to divide the depth data information represented by different objects, that is, to realize object data analysis. The most important thing to achieve object data separation is to identify the edge of the object and then divide the object by the edge of the object. So the edge identification problem will be one of the keys to this algorithm.

When designing an algorithm, the author divides the contours of objects into two categories, one is the outer contour and the other is the inner contour. The outer contour of the object refers to the contour represented by the outer features of the object, and the inner contour refers to the contour of the inner spatial structure of the object. Obviously, the LRF scan yields the outer contour of the object, and the DRF scan requires the outer contour. The key to the LRF's acquisition of the outer contour of the object is to scan the edge of the object. Combined with the scanning characteristics of LRF, the author will use the detection distance difference of LRF to realize the edge identification.

Since the scanning accuracy of the LRF decreases as the distance of the LRF's detected light increases, assuming that the maximum distance that the LRF can detect is R_{max} , the maximum detectable range of the LRF is the semicircle with the R_{max} as the center and the R_{max} as the radius. Then the maximum detectable area of the LRF is:

$$S = \frac{\pi \cdot R_{max}^2}{2}$$

Obviously, as the detection distance of the LRF increases, the detection error of the LRF increases. And the general LRF R_{max} is very large, but in fact designers and users do not need to detect such a large detection area. For a follower robot, the robot only needs to detect objects within a certain distance around it. Generally, the distance is 6-10 meters. The authors suggest setting the effective detection distance of the LRF to 6-10 meters. Because this distance corresponds to a general mechanically driven robot, it has sufficient reflection range and detection range, and makes data storage resources not wasted. For robots equipped with a more accurate LRF, a larger R_{max} can be set. For robots equipped with a less accurate LRF, a smaller R_{max} can be set. The specific R_{max} setting needs to be specifically set according to the detection accuracy of the LRF of the robot and the data processing capability of the robot and the performance of the hardware device of the robot.

LRF edge detection algorithm design concept

As shown in Figure 62, the author sets the maximum detectable distance for the LRF to D_r , i.e.:

$$D_r = D_{max}$$

Then, the author divides the edge detection light of the length of the LRF into D_r into three segments:

D0: direct detection segment, the distance between the scanned edge point of the object and the LRF point;

D1: The edge mutation allows the length of the segment, the allowable range of the edge when the edge is abrupt;

D2: The detectable edge ignores the segment, and the range distance of the recorded area after the edge is abruptly changed.

There is a close relationship between the three regions, there is no segmentation, as shown in Figure 62, there are:

$$D_r = D_0 + D_1 + D_2$$

The author divides the detectable range of the DRF into the above three regions for the construction of the edge recognition algorithm. The following is a description of the specific characteristics of the above three regions:

1. **D0 Area**: the distance that the LRF light directly illuminates the detectable edge point of the detected object, within which there is no other detection light that blocks the LRF;

2. **D1 Area**: When the LRF scans an edge point of the scanned object, if the next point is scanned along the original LRF scanning direction, if the detection distance of the LRF is abrupt, the author will mutate the detection distance of the LRF. The detection point is called the LRF scan mutation point. When scanning the LRF, if there is a scan mutation point, there will be two cases:

2.1. In the first case, the edge point is indeed the edge point of the detected object. When the scanning light of the LRF moves a certain distance from the point along the original scanning direction, the scanning light will be separated from the detected object, so The LRF's detected light will be transferred from the object to another object. Then the detection distance of the LRF is abrupt.

2.2. In the second case, the point is not the edge point of the detected object, but the edge distribution of the detected object is not smooth, and there is an uneven edge with a large slope angle at the edge of the detected part. Distribution changes. Thus,

when the LRF detection light travels along a fixed scanning direction, from a smooth detectable segment of the detected object, through a different depth of fold, moves to the smooth detectable segment of the next segment. This causes a sudden change in the detection range of the LRF.

The author refers to the edge point of the last edge of the object before the mutation is called the last edge uniform point, and the edge point of the first edge of the object is called the first edge mutation point. The last edge uniform point is opposite to the first edge transition point. As shown in Fig. 63, assuming that the LRF scans in the counterclockwise direction, the point P3 is the uniform edge of the last edge of the Others object, and the point B is the first edge mutation point of the Others object. Point A is the uniform edge of the final edge of Human of the target character, and point A' is the first edge of the object. If the LRF is scanned in a clockwise direction, then point A' is the last edge uniform point of the target character Human, and point A is the first edge transition point of the target person. Point B is the uniform edge of the final edge of Human of the target character, and point P3 is the first edge of the object. That is to say, if the detected object remains relatively prohibited from the LRF, the roles played by the two points acting as the last edge uniform point and the first edge transition point will be relatively changed. However, in general, it is difficult to achieve relative inhibition of the object to be detected and the LRF. Since the scanning angular velocity of the LRF is very fast, for the two adjacent LRF scans, the author believes that the LRF and the detected object will be relatively static. N.B.: Here is the author's analysis of the relative static state of the detected object and the LRF, just to facilitate the implementation of the LRF edge algorithm. In the specific edge algorithm of the LRF, the LRF is not required to remain relatively stationary with the object being detected. Since the LRF is very short in the period of one deflection scan, the relative position of the detected object and the LRF will be so small that the author can ignore this small variation.

The judgment of the scanning edge mutation point and the scanning edge uniform point

For a robot using only LRF, the only tool that can sense the external environment is the LRF. The only external information that can be perceived is the detection distance of the LRF. When the robot mobilizes the LRF for scanning, the robot does not know when an edge transition point will occur, and it is not known what point is the edge mutation point. The algorithm for judging the edge of the edge of the robot for the robot will be described below.

Since the author's edge algorithm makes the analysis and judges for the edge of the detected object, it is the most important key to judge whether the LRF scan point is the scan edge mutation point. Corresponding to the scanning edge abrupt point is the uniform point of the scanning edge. Whether it is scanning edge abrupt points or scanning edge uniform points, it is not alone, but relative to other scanning edge points.

To get the information of the edge point of the scanned object, first, the algorithm designer needs to solve how to determine what edge point is the scanning edge abrupt point, and what point is the scanning edge uniform point. In order to solve this problem, the author introduces the idea of deductive continuous correlation to construct this part of the algorithm. In order to better explain the principle of algorithm implementation, the authors refer to the scanning line segments which are all formed by the uniform points of the scanning edges as the scanning uniform line segments, and the scanning line segments in which the scanning mutation points exist are called scanning mutation line segments.

The detailed algorithm explanation will be given below in conjunction with FIG. 65:

Assumption:

1. The LRF scan direction is along the negative direction of the Y-axis.
2. D1 is the identified edge of the detected object.

Creating new parameters:

Dei : The allowable range of the edge slope change of the front edge of the i-th segment that continuously changes uniformly. If all the scanning points of a certain scanning line segment are all within the Dei range corresponding to the scanning line segment, then the scanning line segment is called a scanning uniform line segment. If even more than one scanning point in a certain scanning line segment is beyond the Dei range corresponding to the scanning line segment, then the scanning line segment is called a scanning abrupt line segment.

Lai : the average distribution straight line segment of each edge point of the edge segment of the i-th segment that continuously changes uniformly on the XY plane. It is called the average distribution straight line segment of the i-th segment edge segment, and is referred to as the average distribution straight line segment of the i-th segment.

Dtij: The linear distance between the straight line segment of the i-th segment and the average straight line segment of the j-th segment along the LRF perpendicular to the radial direction of the robot edge where the LRF is located.

Pk: the scanning edge point in the Dei corresponding to Lai, that is, the point on the scanning uniform line segment.

Ptk: The scanning edge point outside the Dei corresponding to Lai, that is, the point on the scanned abrupt line segment.

Input parameters:

D1.

Calculation and analysis:

1. It is assumed by the assumption that the LRF is scanned from the Y-axis direction. It is assumed that scanning is performed starting from Ps of FIG. It is known from Fig. 65 that the scanning trajectory of the LRF is within the scanning line segment between the point Ps and the point P1, and all the scanning points do not exceed the range of De1. Therefore, the PsP1 scan line segment is evenly scanned. From the point P1 to the point P2, the scanning mutation point Pt1 appears, so the scanning line segment PsP1 is the scanning mutation line segment. As long as there is a scan break point in a scan line segment, the scan line segment can be said to be a scan abrupt line segment. Therefore, by scanning, the scanning uniform line segment and the scanning abrupt line segment and the scanning abrupt edge point and the scanning uniform edge point in FIG. 65 can be obtained:

Scan the uniform line segments: PsP1, P2P3, P4P5, Pt5P6, Pt6P7;

Scanning mutant line segments: P1P2, P3P4, P5Pt5, P5Pt6;

Scan even edge points: Ps, P1, P2, P3, P4, P5, P6, P7;

Scan abrupt edge points: Pt1, Pt2, Pt3, Pt4, Pt5, Pt6.

2. So how do you determine if a scan point is a scan edge abrupt or a scan edge? To solve this problem, the author introduced the concept of Dei. Suppose that most of the scanning edge points of a scanning edge segment are within a certain Dei, that is, if the average straight line of the position of most scanning edge points of a scanning edge segment is a certain Dei corresponding to it, then it is called The scanning edge line segment is a scanning line segment corresponding to Dei, that is, the allowable range of the slope change corresponding to the scanning edge line segment is Dei. Obviously, a scanning edge segment can only correspond to one Dei range, and a Dei range can correspond to multiple scanning edge segments. E.G.: Scan the edge line segment PsP1, and the Dei corresponding to P2P3 and P4P5 are both De1. The Dei corresponding to the scanning edge line segment Pt5P6 is De2, and the Dei corresponding to the scanning edge line segment Pt6P7 is De3. Knowing the correspondence between Dei and the scan line segment, the designer can define a solution to the problem of determining whether a scan edge point is a scan edge abrupt point or a scan edge uniform point. Author's definition: If a scanning edge point is within its corresponding Dei range, the scanning edge point is said to be a scanning edge uniform point. If the scanning edge is not within its corresponding Dei, then the author calls the scanning edge a scanning edge. Discontinuity. Combined with this definition, the following scanning edge mutation points can be found in Figure 65:

The scan edge mutation points corresponding to De1: Pt1, Pt2, Pt3, Pt4.

There is no corresponding scanning edge mutation point between De2 and De3.

The relationship between the larger surface drop and the true edge of the object being detected

After knowing how to judge whether the scanning point on a scanning track line is the scanning edge abrupt point or the scanning edge uniform point, this paper focuses on the scanning abrupt line segment and the scanning uniform line segment, and acquires the relationship of the edge of the detected object. In general, the detected objects will be not scanned by the LRF probe beam at all times. The surface of the scanned portion of the object is often composed of a tortuous multi-curved surface. Different curvature radii and numbers of surfaces that may exist at different locations on the surface of the object. The difference in surface between different positions is generally different. A large surface drop occurs at the edge of the detected object, but the large surface drop is not necessarily the edge of the detected object. When the LRF scans the object being inspected, there are only two cases if there is a large surface drop:

1. The LRF has left the object to be detected, and the detected part is the real edge of the detected object. As shown in Fig. 65, the LRF scanning beam is shifted from the point P5 to the point Pt6, and the process corresponds to the surface difference of the LRF scanning beam. for:

$$Da = Dt13$$

In the above formula, Dt13 represents the radial straight line distance of the scanned LRF beam from the average distribution straight line segment la1 corresponding to the scanning point P5 before the transition to the average distribution straight line segment la3 corresponding to the scanning point Pt6 after the LRF beam transition. It is known from Fig. 65 that there is no true scanning edge line between the scanning point P5 and the scanning point Pt6 (N.B.: the dotted line connection used for convenience in Fig. 65 does not actually exist). This paper assumes that the scan line segment PsP6 belongs to the scanned target person, and the scan line segment Pt6P7 does not belong to the scanned target person. That is to say, Pt6P7 represents the scanning line segment of other objects or scan sites of unrelated humans.

2. The LRF does not leave the detected object, and the detected part is not the true edge of the detected object. As shown in Fig. 65, the LRF scanning beam transitions from the point P5 to the point Pt5, and the surface of the corresponding LRF scanning beam has a curved surface drop. For Dt12. Dt12 represents the radial straight line distance of the scanned LRF beam from the average distribution straight line segment la1 corresponding to the scanning point P5 before the transition to the average distribution straight line segment la2 corresponding to the scanning point Pt5 after the LRF beam transition. It is known from Fig. 65 that there is a real scanning edge line between the scanning point P5 and the scanning point Pt5, that is, the scanning line segment Pt5P6 also belongs to the scanning line

segment PsP6. That is to say, the PsP6 scan line segments are continuous. In this paper, it is assumed that PsP6 is the scanning line segment of the scanned target person, and the scanning portion corresponding to the scanning line segment Pt5P6 still belongs to the scanned target person.

The above two situations in which there is a large gap in scanning may be known. The author plots the two possible scenarios side by side in Figure 65 to illustrate the situation more clearly. Through analysis, we can know that to judge whether the real edge point of the scanned object appears in a certain scanning line segment, an important method is to judge that the LRF scans the scanning point along the original scanning direction and then performs the next scanning point. Whether there is a large scan drop when scanning. This is because, after scanning the real edge point, the LRF scans the next scan point that is not on the currently detected object. Generally, a large scan drop occurs (NB: the target person and other detected are excluded here. The object is very close. If it is the item carried by the target person, it will be considered by the algorithm to be part of the target person.)

Judging different large surface drop cases

The above analysis shows that if the designer wants to scan the uniform scanning point of the scanning edge of the current scanning of the object by the LRF and scan the next scanning point, whether the detection distance of the LRF will undergo a scanning mutation to judge the moment. Whether the scanned scanning edge point is the real edge point of the object to be scanned, the author still needs to deal with another problem, that is, the situation that can cause a large scanning drop when the LRF scans the detected object has the above two Kind of situation. Then, the author must set up an algorithm for judging processing to determine whether the LRF is the first case or the second case when detecting a large surface drop in the scan detection. In combination with the above analysis, combined with the analysis of FIG. 65, since the time taken for the LRF to scan a 180° half-circumferential period scan is very short, there are two processing corresponding to the scan edge data of the detected object obtained by the LRF scan. Method strategy: The first method strategy is to wait for the LRF to complete the scanning of the entire edge scan line segment of the detected object before processing the data; the second method strategy is to tighten the LRF in the process of scanning the scanned object. The scan data of the LRF is then processed. Obviously, the algorithm implementation of the first method strategy is much simpler than the algorithm implementation of the second method strategy. Although the scanning speed of the LRF is very fast, the final efficiency of the two methods for processing the scanned data acquired by the LRF is similar, but in order to ensure the efficient execution of the author algorithm, the author uses the side scan to perform data in this part. Method strategy for handling judgments. In the implementation of the specific algorithm, the author adopts the scanning line edge uniform line segment as the basic unit, and then calculates the position of the lai space line corresponding to

different scanning uniform line segments. The author finally judges the degree of surface drop of the scanning edge of the detected object by comparing the radial distance difference between the different scanning edge uniform segments before and after the LRF scanning point transition.

Analysis of the LRF scan and data processing execution period

After the authors decided to use a more efficient method strategy, another issue caught the attention of the author. This problem was performed immediately after the LRF scanned a scanning edge segment or immediately after the LRF scanned a scanning edge. Data processing? After analysis and consideration, the author chooses the first instant processing method, that is, data processing after the LRF scans a scanning edge segment. Why does the author not use the second way of dealing with it? If the degree of convergence between real-time and time and space is obvious, the second case is more competent (N.B.: the second case will be better). However, when the author designs the algorithm, the real-time performance cannot be emphasized unilaterally. For example, in this algorithm, if the LRF scans a scanning edge point, the data processing is performed, if the scanning edge point is not its corresponding Scanning the last edge of the edge segment to scan the uniform edge point or the different Dei's skip scan endpoint, then the immediate processing of the scan will be redundant computing resources, because the author's purpose is to determine whether the scan face is the last scan The edge is evenly spaced, and if the point is a uniform point of the last scanning edge, the radial transition distance between the lai corresponding to the scanning edge point and the lai corresponding to the next scanning edge point is obtained. Because the author will set an LRF detection distance threshold in the algorithm of this part to compare the radial transition distance, whether the last scanning edge point is the real edge point of the detected object. Therefore, for determining whether a scanning edge point is a real edge point of the detected object, the author is concerned with the uniform point of the last scanning edge of the scanning line segment corresponding to the scanning edge point, and the lai corresponding to the scanning edge point. According to the definition of lai, it can be known that lai is relative to a certain scanning edge segment. If a segment of a scanning edge is longer, there will be more scanning edge points on it. Therefore, for a scanning edge segment, the designer should obtain as much data as possible of the scanning edge point in order to more accurately calculate the lai corresponding to the scanning edge segment. If the designer performs a calculation of lai every time the LRF scans a new scanning edge point, it is obviously a waste of the system computing resources of the robot. However, if the data processing of the scanning edge point of the current scan of the LRF is not performed, the designer cannot know whether the point is a uniform point of the last scanning edge. Therefore, if it is necessary to perform timely scan data processing, an indispensable step is to perform a distance determination procedure every time the LRF performs an action of scanning the edge of the detected object to determine whether the point is a scan. The edge is abrupt. The purpose of the author's design is that if a certain

scanning point detected by the LRF is a scanning edge abrupt point, then the previous point of the scanning point must be a uniform point of the last scanning edge of a certain scanning uniform line segment. If the algorithm has previously recorded the first scan edge uniformity point of the segment of the scan line, the author can determine the scan line segment by determining the scan line segment between the first scan uniform point and the last scan uniform point of the scan line segment. In the algorithm designed by the author, in the process of scanning by LRF, once the algorithm determines the initial point and the end point of a certain segment of the scanning edge, the algorithm will start the lai calculation program, and the corresponding lai line of the uniform line segment will be scanned. Calculated. During this process, the LRF will continue to scan, but the calculation of the lai for the uniform edge of the scanning edge will only be initiated once the algorithm determines that a certain scanning edge segment has been completely scanned. The purpose of this design is to ensure that the computing resources of the robot are utilized efficiently.

About Dei's setup issues

It can be known from the previous analysis that in order to determine whether a scanning edge point is the real edge point of the detected object, the designer needs to perform the relative position of the corresponding lai radial line corresponding to the front and back scanning uniform line segments corresponding to the scanning edge point. Threshold judgment. Therefore, it is important to scan the scanned edge segments and evenly divide the edge segments. To solve this problem, the author must set a reasonable Dei, the actual scan line segment to divide. Therefore, the key problem that the author needs to solve becomes that the Dei can be set to maximize the effectiveness of the segmentation algorithm. Another question is whether Dei's width range is changed or determined from the beginning. Why do designers have to consider this problem? This is because the degree of surface undulation of the scanning line segments on the surface of different detected objects, or the degree of surface undulation of the detected parts at different positions of the same object being detected is generally not the same. Obviously, even if the designer only scans the target person, the curved surface of the scanable edges of different target characters will be different. Therefore, designers cannot use only a single Dei width range standard to measure the degree of surface undulation of the detectable edges of all detected objects. In other words, the designer must dynamically set the width range of Dei. If you return to the discussion of the meaning of the Dei setting, the reader can easily find out that the author's purpose of setting Dei is to determine whether a certain scanning line segment is a uniform line segment of the scanning edge. Since dividing the scanning uniform line segment is very important for judging the true edge point of the detected object, that is, the intelligent division of the Dei range is very important for judging the true edge point of the detected object.

Combined with the previous analysis, the author can not achieve a better Dei range for a certain object to be detected due to the varying degrees of undulations of the edge

of the object being detected. It is just a unilateral targeting of the target human, and the author cannot set a very reasonable Dei range. So the author designed a dynamic range adjustment algorithm for Dei's range setting. The specific algorithm execution flow is as follows:

As shown in Fig. 66, the scanning line segment PsPt is a uniform line segment of the scanning edge. The scanning direction of the LRF is along the negative direction of the Y axis. Define Pt_i as the *i*th scan peak point of the LRF scan curve segment, and define Pd_i as the *i*th scan peak and valley point of the LRF scan curve segment. Since the scanning speed of the LRF is very fast, the author adopts a strategy of waiting for the LRF to complete a data segment after scanning a uniform edge of the scanning edge. The specific data processing method is as follows:

1. Extract each Pt_i recorded in the uniform line segment of the scanning edge and calculate the maximum value Pt_{max}, i.e.:

$$Pt_{max} = \text{Max}\{Pt_1, Pt_2, \dots, Pt_n\}$$

2. Extract each Pt_i recorded in the uniform line segment of the scanning edge, and calculate the minimum value Pt_{min}, i.e.:

$$Pt_{min} = \text{Min}\{Pt_1, Pt_2, \dots, Pt_n\}$$

3. Extract each Pd_i recorded in the uniform line segment of the scanning edge, and calculate the maximum value Pd_{max}, i.e.:

$$Pt_{max} = \text{Max}\{Pt_1, Pt_2, \dots, Pt_n\}$$

4. Extract each Pd_i recorded in the uniform line segment of the scanning edge, and calculate the minimum value Pd_{min}, i.e.:

$$Pt_{min} = \text{Min}\{Pt_1, Pt_2, \dots, Pt_n\}$$

5. Define the range scale factor to be *k*, and define the width of the uniform edge segment Dei of this segment as WDei, i.e.:

$$WDei = k \cdot \frac{(Pt_{max} + Pt_{min} + Pd_{max} + Pd_{min})}{2}$$

The above-mentioned proportional coefficient is referred to as an object-to-scan width ratio coefficient, and the specific value thereof is related to the degree of surface wrinkles and the degree of undulation of the detected portion of the object to be detected. From this, the designer can determine the Dei corresponding to a certain uniform line segment.

The most basic principle for judging the true edge of an object being detected

When judging and recognizing the edge of the detected object, the author only pays attention to the coordinate data of the first and last scanning points scanned and detected along the scanning direction of the LRF of the detected portion of the detected object. The author scans the first scan point scanned along the scan direction of the LRF as the first scan point corresponding to the portion of the detected object. Correspondingly, the last scanning point scanned along the scanning direction of the LRF is referred to as the last scanning point corresponding to the portion of the detected object. As shown in Fig. 64, in the effective detection area corresponding to the LRF, there are a total of three scanned portions of the detected object, which are represented by Part1, Part2, and Part3, respectively. Let Le be the pointer to the right direction of the edge of the robot rectangle where the LRF is located. The overall image is represented in the XYZ following coordinate system. It is assumed that the scanning direction of the LRF is the negative direction along the Y axis. As shown in FIG. 64, the LRF first scans the scanned portion of the detected object represented by Part1, and the obtained point P1(x1, y1, z1) is the first scanning point of Part1, and point P2 (x2, Y2, z2) is the last scan point of Part2. Correspondingly, the point P3 (x3, y3, z3) and the point P4 (x4, y4, z4) are the first search points of the scanned portion of the detected object corresponding to Part 2 and Part 3, respectively. The point P5 (x5, y5, z5) and the point P6 (x6, y6, z6) are the last search points of the scanned portion of the detected object corresponding to Part 2 and Part 3, respectively. During the scanning of Part 1 by the LRF, the scanning algorithm does not care about the specific shape of the specific scanning edge curve segment of the scanned portion of the detected object between the point P1 and the point P2. In this process, the scanning algorithm is concerned that after the LRF continuously scans to the P2 point, it enters the judgment of the real edge of the key detected object. The following is the specific process of judgment implementation:

1. ***Jumping situation 1:*** If the LRF scans the P2 point, the LRF scanning beam immediately jumps to point P4. The reader can find out by enlarging the figure 64 that the point of rotation of the point P2 along the LRF scanning direction on the scanning line segment of the LRF pointing point P4 is the point Pc24, and the point P2 is along the LRF scanning direction at the scanning line segment of the LRF pointing point P3. The rotation projection point on the point is the point Pc23. Then the scanning radial jump distance at which the LRF scanning beam jumps from point P2 to point P4 is:

$$Dp244 = \sqrt{(x4 - x2)^2 + (y4 - y2)^2 + (z4 - z2)^2}$$

2. ***Jumping situation 2:*** If the LRF scans the P2 point, the LRF scanning beam

immediately jumps to point P4. Then the scanning radial jump distance at which the LRF scanning beam jumps from point P2 to point P3 is:

$$D_{p233} = \sqrt{(x_3 - x_2)^2 + (y_3 - y_2)^2 + (z_3 - z_2)^2}$$

3. In order to detect whether the LRF detection beam really jumps away from the detected object, the author sets a true jump judgment threshold Δj :

$$\Delta j = \text{Max}\{W_p, L_p\} + d$$

The above $\text{Max}\{\}$ is a function of taking the maximum of two numbers in braces, and d is an error adjustment balance parameter, which is determined by multiple experiments.

4. Determining the type of jump the jump belongs to:

- a. Let D_{p244} or D_{p233} be equal to D_j , i.e.:

$$D_{p244} = D_j$$

Or:

$$D_{p233} = D_j$$

Judge: If the following formula holds:

$$D_j > \Delta j$$

Then point P2 is indeed the true edge point of Part1;

Judge: If the following formula holds:

$$D_j \leq \Delta j$$

It is not yet possible to determine whether point P2 is the true edge point of Part1.
Go to step 5.

5. Further judgment on the jump of type $D_j \leq \Delta j$:

- a. Let the LRF continue to scan along point P3 until the LRF scans to the second true edge point. (NB: The real edge point at thi time is judged by the judgment method of step 4. As shown in Fig. 64, the second real edge point should be point P5, corresponding to Part 2. However, the robot at this time does not know that point

P5 is If you belong to Part 2, you don't know if Part exists, so you need to make further judgments.)

- b. Calculate the straight line distance of P1P5 and set it to Dp1:

$$Dp1 = \sqrt{(x5 - x1)^2 + (y5 - y1)^2 + (z5 - z1)^2}$$

- c. Enter the LRF scan lateral offset range to determine:

Judge: If the following formula holds:

$$Dp1 > \Delta j$$

Then, the point P2 is indeed the true edge point of the Part 1, determining that the detected area has the detected object corresponding to Part 2, and the point P3 is the first scanning point of Part 2;

Judge: If the following formula holds:

$$Dp1 \leq \Delta j$$

Then, the point P2 is not the true edge point of Part1, and it is determined that the detected area does not have the detected object corresponding to Part2.

Why does the author use a double judgment mechanism in the above algorithm? This is because if only $Dj \leq \Delta j$ holds, then two kinds of scan results may occur: the first one is as shown in Fig. 64, and the point P2 is really the edge point of the detected object (IE: Part1 represents the whole detected) object.). The second case is shown in Figure 63.3. Point P2 is only a tortuous point of the object being detected (I.E.: Part1 represents only a certain detection part of the entire object to be inspected.). Then why does the author compare whether Dp1 and Δj can determine whether Part1 belongs to the entire detected object or belongs to a detected part of the detected object? This is because Δj represents the maximum width range of the detected object (NB: the detected object described in this part of the algorithm is generally the target person if there is no special explanation.), and Dp1 represents The linear scanning distance of the LRF to the detected object. Obviously, if Dp1 is larger than Δj , it indicates that the scanning part represented by Dp1 has exceeded the true range of the detected object, so the P2 obtained by scanning at this time is the display edge of Part1. If Dp1 is smaller than Δj , it indicates that the scanning part represented by Dp1 has not exceeded the true range of the detected object, so the point P2 obtained by scanning at this time is a part of the detected object. Specifically, the point P2 should belong to a certain folded part. End point. It should be noted here that the scanning of the LRF in the scanning area corresponding to Part 1 or Part 2 is continuous. This is because the Part1 and Part2 parts are abstractly simulated at the beginning of the continuous scan edge segment.

Treat the scanned edge segments as straight segments for processing

For the following robots of this project, the primary important responsibility of the LRF on the robot is to detect the other detected objects, obtain the scanned real edges of the detected objects, and acquire the geometric abstract data of the detected objects, so that the detected objects will be detected. Simply separate from other unrelated objects in the environment. In the latter object abstraction algorithm, the abstract algorithm used by the author also abstracts the object into a simple spatial geometric polyhedron. This is because, for the robot designed at this stage, the purpose of the algorithm set by the author is to separate the detected object from other unrelated objects for the actual project requirements and implementation requirements. The author introduces a more advanced algorithm in the next chapter. In more advanced algorithms, the author abstracts the detected object into a geometric polyhedron with a certain degree of complexity. For details of the detailed algorithm and analysis, please refer to the content of the next section. This section will introduce the basic scanning recognition algorithm for the real edge of the detected object. This algorithm is the edge recognition algorithm in the most basic object abstraction algorithm, but it is enough to meet the requirements of the accuracy requirements of this project. The biggest difference between the basic algorithm of this substation and the more advanced algorithm to be described later is that the algorithm in this part treats the scanning edge line segment as a straight line segment, where the straight line segment refers to the 180° semicircular range of the LRF. A straight line segment of the continuous sweepable curve segment of the object detected during scanning. The author will elaborate on this part of the algorithm in conjunction with Figure 65. The following is the specific algorithm implementation steps:

As shown in Fig. 65, the author abstracts the scanning edge curve segment PsP5 into a straight line segment PlsPl5. PlsPl5 belongs to a part of the straight line segment la1. The scanning edge curve segment Pt5P6 is geometrically abstracted and abstracted into a straight line segment Plt5Pl6. Plt5Pl6 belongs to a part of the straight line segment la2. The scanning edge curve segment Pt6P7 is geometrically abstracted and abstracted into a straight line segment Plt6Pl7. Plt6Pl7 belongs to a part of the straight line segment la3. The author defines a straight line segment abstracted from a segment of a real scan curve called a scanned abstract straight line segment. Note that the scanned abstract straight line segment here is different from the scanning straight line span mentioned later.

In an algorithm in which a part of an object scanning edge is abstracted into a straight line segment, an abstraction is performed on a scan curve obtained by scanning a 180° half-circumferential range of the LRF. In the process of processing, the author easily finds that in the whole scan curve PsP5, a small-scale scanning curve segment similar to the scanning edge mutation line segments P1P2 and P3P4 appears beyond the range of Dei corresponding to the scanning curve segment PSP5, and becomes The line segment is abruptly scanned. For these scan edge abrupt segments, the LRF has their complete scan data, but the author only uses the scan data for these scan edge abrupt segments when setting and calculating the corresponding Dei for the entire scan

curve. However, when the average straight line segment lai corresponding to the scanning curve of the latter section is small, if the scanning straight line segment of the scanning edge of the scanning edge span is not large, the author does not use the data of the scanning edge mutation line segment. The purpose of this is to reduce the impact of accidental errors. A new concept is proposed here, which is to scan the straight segment span. Scanning a straight line span means that the scan curve segment

As shown in Fig. 65, the scanning straight line span of the scanning edge abrupt line segment P1P2 is Dp1, the scanning straight line span of the scanning edge abrupt line segment P3P4 is Dp2, and the scanning straight line segment of the scanning edge abrupt line segment P5Pt5 and the scanning edge abrupt line segment P5Pt6 is Dp3, scanning The scanning straight line span of the edge abrupt line segment Pt5P6 is Dp4.

It is known from the previous analysis author that if the scanning straight line segment of the scanning edge of the detected object spans a small span, the algorithm of this part does not use these scanning edge mutations when calculating the average distribution straight line lai of the scanning curve. The data of the line segment. The scanning straight line span is not a straight line segment, but the magnitude of the offset angle of the scanning curve segment in the scanning direction of the LRF lateral deflection. As shown in Fig. 65, the vertically arranged black linear array represents the scanning straight-span recording spectrum. That is, the intuitive form of the original scan result on the surface is expanded into a plane. The authors refer to such a linear array as shown in FIG. 65 as a scan span spectrum. The distance between any two adjacent vertical straight segments in the scan span spectrum is called the LRF scan span interval, referred to as the scan interval. In conjunction with the previous analysis authors, the extraction and analysis of scanned data points will be performed at intervals of one scan interval. So, how much is the scan straight line span? The author assumes that the scan interval set by the LRF is Δd , and the straight line span of the scan line segment of the detected object is Dl, when Dl satisfies the following expression:

$$Dl \leq \lambda \cdot \Delta d$$

it is said that the Dl corresponding to the scanning line segment at this time is not large, that is, when Dl satisfies:

$$Dl > \lambda \cdot \Delta d$$

it is said that the Dl corresponding to the scanning line segment is sufficiently large. Then, when the algorithm calculates the average distribution straight line lai of the scanning curve, the data of the scanning edge mutation line segment is used. If not, the algorithm ignores the scanning line segment. The above λ is an empirical scale factor, which needs to be summarized by many experiments, and is related to the mechanical braking of the specific robot and the hardware operation and reaction parameters of the LRF.

Assume that λ is 5 in the algorithm set by the author, then Dp3, Dp2 and Dp1 in Fig. 65 do not meet the requirements. At this time, the algorithm will calculate these

when calculating the average distribution straight line lai of the scanning curve. Scanning edge abrupt line segments are ignored. From another angle of analysis, if the scanning line span corresponding to a segment of the scanning edge of the scanning edge satisfies a sufficiently large comparison condition, the algorithm considers the scanning line segment as a uniform line segment of the scanning edge. This process is called a scanning edge abrupt line segment. To the uniform line segment conversion of the uniform line segment of the scanning edge, the corresponding scanning straight line span to be sufficient is called the conversion condition that the conversion needs to satisfy. Conversely, as shown in FIG. 65, if Dp4 is not full of a sufficiently large conversion condition, the processing algorithm will treat the scan line segment Pt5P6 corresponding to Dp4 as a scan edge abrupt line segment, that is, do not perform uniform line segment conversion on the scan line segment.

Relative relationship between the scanning edge abrupt line segment and the scanning edge uniform line segment

Before the elaboration of the specific LRF edge recognition algorithm, there is another important issue. What can't be ignored in this paper is how the designer determines that a scan line segment is a uniform scan line segment. It can be known from the concept definition in the previous content that the scanning line segment not in the corresponding Dei is the scanning edge abrupt line segment, and the scanning line segment in the corresponding Dei is the scanning edge uniform line segment. However, the author wants to point out here that scanning a broken line segment is opposite to scanning a uniform line segment. As shown in FIG. 65, if the scanning direction of the LRF is in the negative direction along the Y-axis, when the LRF scans the scanning line segment PsP5, Pt5P6 is the scanning edge abrupt line segment with respect to PSP5. If the scanning direction of the LRF is in the positive direction along the Y-axis, then when the LRF scans the scanning line segment Pt5P6, PSP5 is the scanning edge abrupt line segment with respect to Pt5P6. That is, the actual scanning edge abrupt line segment is opposite to the scanning edge uniform line segment. Therefore, when the designer design algorithm judges a certain scanning edge line segment, it must be judged by combining the actual Dei region.

True edge recognition of the scanned edge segment as a straight line segment

After the division of the uniform line segment transformation of the scanning line segment, the execution judgment portion of the recognition algorithm of the real edge of the detected object can be formally described.

First, the author wants to linearly abstract the scanned uniform segments. The

design idea of the specific abstract processing method adopts the two-differentiation straight line abstract method. The specific algorithm implementation steps are as follows:

1. Assume that the scan curve segment to be processed is Ls1, and there are a total of n scan detection points on P1, P2, ..., Pn. The following analysis:

1.1 Judgment: If n is an odd number, then

$$m = \frac{n - 1}{2}$$

The n-1 scanning points on Ls1 are divided into two parts, which are represented by sets Pa1 {} and Pa2 {} respectively.

$$\begin{cases} Pa1 = \{P1, P2, \dots, Pm\} \\ Pa2 = \{Pm + 1, Pm + 2, \dots, Pn - 1\} \end{cases}$$

1.2 Judgment: If n is even, then let

$$m = \frac{n}{2}$$

The n scan points on Ls1 are divided into two parts, which are represented by the sets Pa1 {} and Pa2 {} respectively:

$$\begin{cases} Pa1 = \{P1, P2, \dots, Pm\} \\ Pa2 = \{Pm + 1, Pm + 2, \dots, Pn\} \end{cases}$$

The three coordinate components defining the XYZ coordinate system of a certain scanning point Pi are (xi, yi, zi). It is assumed that after the above two segments, the Pa1 {} set and the Pa2 set have the same number of scan points of m, and the arithmetic mean points corresponding to the Pa1 {} and Pa2 {} sets are respectively determined below, respectively correspondingly expressed as Point Paal (xaal, yaal, zaal) and point Paa2 (xaal2, yaal2, zaal2). The following is the method for finding the two points of XYZ coordinates:

$$\begin{cases} xaa1 = \frac{1}{m} \sum_{i=1}^m x_{aai} \\ yaal = \frac{1}{m} \sum_{i=1}^m y_{aai} \\ zaal = \frac{1}{m} \sum_{i=1}^m z_{aai} \end{cases}$$

$$\begin{cases} x_{aa2} = \frac{1}{m} \sum_{m+1}^{2m} x_{aa1} \\ y_{aa2} = \frac{1}{m} \sum_{1}^{2m} y_{aa1} \\ z_{aa2} = \frac{1}{m} \sum_{m+1}^{2m} z_{aa1} \end{cases}$$

Thus, the two-point equation of the spatial straight line La1 formed by the point Paal and the point Paa2 is:

$$\frac{x - x_{aa1}}{x_{aa2} - x_{aa1}} = \frac{y - y_{aa1}}{y_{aa2} - y_{aa1}} = \frac{z - z_{aa1}}{z_{aa2} - z_{aa1}}$$

Similarly, assuming that there is another scan curve segment Ls2, the two points of the straight line passing through the arithmetic mean of the binary method of Ls2 are correspondingly represented as points Pbb1(xbb1, ybb1, zbb1) and points Pbb2(xbb2, ybb2, zbb2), respectively. Then the two-point equation corresponding to the spatial line La2 formed by the point Pbb1 and the point Pbb2 is:

$$\frac{x - x_{bb1}}{x_{bb2} - x_{bb1}} = \frac{y - y_{bb1}}{y_{bb2} - y_{bb1}} = \frac{z - z_{bb1}}{z_{bb2} - z_{bb1}}$$

Defined by the vector of the space line, the unit direction vector of the space line La1 is known as:

$$\vec{n}_{r1} = \frac{(x_{aa2} - x_{aa1}, y_{aa2} - y_{aa1}, z_{aa2} - z_{aa1})}{\sqrt{(x_{aa2} - x_{aa1})^2 + (y_{aa2} - y_{aa1})^2 + (z_{aa2} - z_{aa1})^2}}$$

The unit direction vector of the space line La2 is:

$$\vec{n}_{r2} = \frac{(x_{bb2} - x_{bb1}, y_{bb2} - y_{bb1}, z_{bb2} - z_{bb1})}{\sqrt{(x_{bb2} - x_{bb1})^2 + (y_{bb2} - y_{bb1})^2 + (z_{bb2} - z_{bb1})^2}}$$

Let the corresponding vector product of the two scanning line segments La1 and La2 be:

$$\vec{n}_{r1} \times \vec{n}_{r2} = \begin{vmatrix} \vec{i} & \vec{j} & \vec{k} \\ (x_{aa2} - x_{aa1}) & (y_{aa2} - y_{aa1}) & (z_{aa2} - z_{aa1}) \\ (x_{bb2} - x_{bb1}) & (y_{bb2} - y_{bb1}) & (z_{bb2} - z_{bb1}) \end{vmatrix}$$

$$\begin{aligned}
&= (((yaa2 - yaa1)(zbb2 - zbb1) - (zaa2 - zaa1)(ybb2 - ybb1))\vec{i} \\
&\quad + ((zaa2 - zaa1)(xbb2 - xbb1) - (xaa2 - xaa1)(zbb2 - zbb1))\vec{j} \\
&\quad + ((xaa2 - xaa1)(ybb2 - ybb1) - (yaa2 - yaa1)(xbb2 - xbb1))\vec{z})
\end{aligned}$$

Then the common normal vectors of Ls1 and Ls2 are:

$$\vec{n} = \overrightarrow{n_{r1}} \times \overrightarrow{n_{r2}}$$

It is known from the above analysis that the point Paa1 (xaal, yaal, zaa1) is on the spatial scanning line segment abstract straight line La1, and the point Pbb1(xbb1, ybb1, zbb1) is on the spatial scanning line segment abstract straight line La2. Then the vector of the space connecting the point Paa1 and the point Pbb1 is:

$$\vec{m} = (xaal - xbb1, yaal - ybb1, zaa1 - zbb1)$$

Let the distance between the abstract lines La1 and La2 of the two spatial scanning line segments be dls, that is:

$$dls = |\vec{m}| \cdot |\cos\theta| = |\vec{m}| \cdot \frac{|\vec{m} \cdot \vec{n}|}{|\vec{m}| |\vec{n}|} = \frac{|\vec{m} \cdot \vec{n}|}{|\vec{n}|}$$

Thus, by synthesizing the above-mentioned functional formula, it is possible to calculate the linear distance between this two scanning uniform abstract straight line segments La1 and La2 through the linearization abstraction processing in the space. Next, the author defines the LRF edge scan change allowable range D1 to determine whether the scan line segment corresponding to La1 and the scan line segment corresponding to La2 are the true edges of the detected object:

If

$$dls > D1$$

Then, if the dls exceeds the allowable range of the edge change of the detected object of the LRF, the scan curve segments corresponding to La1 and La2 are disconnected, that is, the LRF scan distance between La1 and La2 is abruptly detected. The true edge of the object.

If

$$dls \leq D1$$

Then, it is said that the dls does not exceed the allowable range of the edge change of the detected object of the LRF, and the scanning curve segments corresponding to La1 and La2 are continuous. That is, the abrupt portion of the LRF scanning distance

between La1 and La2 does not belong to the real edge of the detected object.

Obviously, the value of D1 is different for different detected objects. However, for robots, it is very difficult to distinguish and distinguish D1 in complex environments by simply using LRF as a scanning detector. For the follow-up robot of this project, the designer does not need to identify and distinguish the unrelated objects in the environment. The designer only needs to identify and distinguish the target characters to meet the project implementation requirements of this paper. From the above DRF human body modeling scanning algorithm, the author can obtain the side width of the target person, which is called Wpflank. Then, the author introduces the side scale factor λf and defines the value of D1 as:

$$D1 = \lambda f \cdot Wpflank$$

The proportional coefficient λf needs to be obtained through multiple experiments in combination with a specific test experiment, which is an empirical scale factor. In general, this paper suggests defining the range of λf as:

$$1.8 \leq \lambda f \leq 3$$

Now, with the above algorithm, the designer can judge the true scanning edge of the detected object. The following section describes an extended, more accurate algorithm for determining the true edges of an object being detected.

More accurate upgrade algorithm: treat the scanning edge uniform line segment as multi-stage discounting for processing

The reason why the scanning processing algorithm in this part is more advanced is that in this part of the algorithm, the author abstracts the scanning edge of the detected object into a set of line segments connected end to end, that is, the object to be detected. The scanning edge uniform line segment is abstracted into a multi-line straight line segment.

Generally, the scanning edge segment of the object generally presents a stepped type distribution of the line, and the stepped scanning edge curve formed by connecting the plurality of segments of the polygonal line forms a scanning edge of the detected object. As shown in Fig. 69, it is assumed that the scanning direction of the LRF is the negative direction along the Y-axis, and the spatial curve segment line segment P0P3 is the scannable edge of the currently scanned object. The spatial curve segment P2P3 is the scanning edge abrupt line segment of the currently detected object. The spatial curve segment P4P5 is the detectable edge of an unrelated object in the LRF detection region. The author considers the detectable edge of the object to be composed of multiple scanned uniform segments. That is, the algorithm abstracts the detectable edge of the

object into a multi-line line segment composed of a plurality of segments of uniform linear segments. Through analysis, as shown in Fig. 66, the following composition of the abstract polyline of the scan line segment can be known:

P0P1: Sei1, Sei2, Sei3, Sei4, Sei5;

P2P3: Sei6, Sei7, Sei8;

P4P5: Sej1, Sej2.

The above Senm indicates that the m-th strip of the n-th scanning polyline constitutes an abstract line segment of the scanning edge. The author then abstracts the scanned edge curve segment into a polyline line segment consisting of multiple segments of the scanned abstract line segment. Each of the scanned abstract line segments here is abstracted from a uniform line segment of the scanning edge. That is, the scanning edge curve of the entire object to be detected is divided into a plurality of scanning edge uniform line segments. Then, the author separately abstracts the multi-segment scanning edge uniform line segments divided by the entire scanning curve segment of the original detected object into scanning abstract line segments. Then, the original scanning edge of the entire detected object is abstracted into a multi-segment line segment composed of a plurality of scanning abstract line segments. To put it simply, the scanning edge uniform line segment is further subdivided, and then the subdivided scanning and transforming uniform line segment is correspondingly abstracted into a scanning abstract line segment. The purpose of this is to increase the degree of abstraction of the detected object, that is, to increase the abstract details of the detected object, which will be very useful in some detection areas that require high precision and high detail recognition, such as when the author needs to target When the head of the task is simply identified.

Since this part of the algorithm belongs to the extension part of the project research report, due to the writing time limit of this project book, the author will only briefly explain the algorithm of this part. The implementation of the specific math part will be in the next version. Please elaborate on this project research report, so stay tuned and pay attention.

Before the implementation of the specific implementation process of this part of the algorithm, this paper first solves a very critical problem, which is how to re-segment the scanning line segment of the LRF. As shown in Fig. 70, the scanning line segment P0P9 is a partially enlarged portion of a uniform line segment of a segment of the LRF scanning edge. The scanning direction of the LRF is along the negative direction of the Y axis. The purpose of the algorithm designed by the author is to abstract the scanned local line segment P0P9 into a line segment composed of two straight line segments, and the algorithm also finds the intersection point of two adjacent straight line segments, which is represented by the point Pt. The author divides the two scanning edge uniform line segments subdivided by the scanning curve segment P0P9 obtained by the

algorithm as the scanning edge uniformly subdivided line segment. Through analysis, it can be found that the most critical part of this part of the algorithm is to find the specific expression of the scanned abstract subdivision straight line segment of each scanning edge uniformly subdivided line segment, and then according to the intersection of the two scanned abstract subdivision straight line segments. The intersection point P_t that the author finally requested. After acquiring the coordinates of P_t , the original scan line segment can be abstracted into a multi-segment line segment composed of two scanned abstract subdivided straight line segments. As shown in Fig. 70, it is assumed that L_1 is a scanned abstract subdivided straight line segment obtained by scanning the curved line segment P_0P_4 through the scanning abstract straight line segment algorithm, and L_2 is a scanned abstract subdivided straight line segment obtained by scanning the curved line segment P_4P_9 through the scanning abstract straight line segment algorithm. Then, the intersection point P_t where L_1 and L_2 intersect is the intersection point requested by the author.

Then, the problem that needs to be solved in this paper is transformed into how to find the spatial linear equation expression of L_1 and L_2 . By analyzing Fig. 70, it can be known that the designer cannot determine the final end point of the uniformly segmented line segment of the scanning edge by the slope of the line connecting the starting point of the segmented line segment and the other points on the line segment by a certain scanning edge. Similarly, from the upper right drawing of Fig. 70, it can be known that since the slope and its corresponding sine or cosine are both uniquely corresponding to the angle of the connecting straight line with the Y-axis. That is, as long as the angle between the starting point and the other points on the line segment is the same as the angle of the Y-axis, the corresponding sine or cosine value corresponding to the connecting line of the segment is also the same. The problem is that there may be multiple scan points on the connected line corresponding to a certain angle, and some of these scan points are separated from the range used by the author to determine the scanned abstract subdivision line segment, and some are not separated from the author. Used to determine the extent of the scanned abstract subdivision line segment. Therefore, the author cannot judge and determine the end point of a uniformly segmented line segment of a certain scanning edge by the slope of the line connecting the starting point of the segmented line segment and other points on the line segment or its corresponding sine or cosine value. In order to solve the problem of subdividing the scanning edge uniform line segments into scanned abstract subdivision straight lines, the author uses the following algorithm:

1. First find the Dei range straight line in conjunction with the relevant algorithm for Dei's setup problem explained earlier in this report. Note that the range of Dei has become smaller, which also corresponds to the algorithmic idea of the author subdividing the uniform line segment of the scanning edge. As shown in Fig. 70, the equations of the Dei linear segment groups l_{11} and l_{12} of the scanning edge uniform line segment P_0P_4 are obtained, and the equation expression of the straight line corresponding to the intermediate straight line segment P_0P_t between l_{11} and l_{12} is obtained by the idea of arithmetic averaging. Then, the equations of the Dei linear segment groups l_{21} and l_{22} of

the scanning edge uniform line segment P4P9 are obtained, and the equation expression of the straight line corresponding to the intermediate straight line segment P9Pt between l21 and l22 is obtained by the idea of arithmetic averaging.

2. The straight line equation corresponding to the straight line segment P0Pt and the straight line segment P9Pt. Establish a system of linear equations. Find the coordinates of the intersection point Pt.
3. With P9 as the new P0 point, repeat the above steps until all Pt points of the scan line segment corresponding to the LRF are used. At this point, the uniform edge segment of the scanning edge corresponding to the LRF can be completely divided into a plurality of straight segments to form a polygonal segment. The task of the algorithm at this stage is successfully completed.

The following article will discuss what the effects of this part of the algorithm can be in practical applications. In general, the degree of abstraction of object abstraction algorithms is not unique. The higher the degree of abstraction of the object abstraction algorithm, the more complex the morphological structure of the spatial geometry corresponding to the object after abstraction. For the machine of this project, the author only needs to use this part of the algorithm when it needs to identify the subdivision of the target part's feature parts. In other cases, the target person can be abstracted into a simple space cube. All in all, this part of the algorithm belongs to the advanced algorithm of the object abstraction algorithm, which is an optional recognition algorithm for the robot of this project, but when some need to perform subdivision recognition, it abstracts the detected part of the target character into a simple one. In terms of spatial cubes, this part of the algorithm has great advantages and stronger recognition ability.

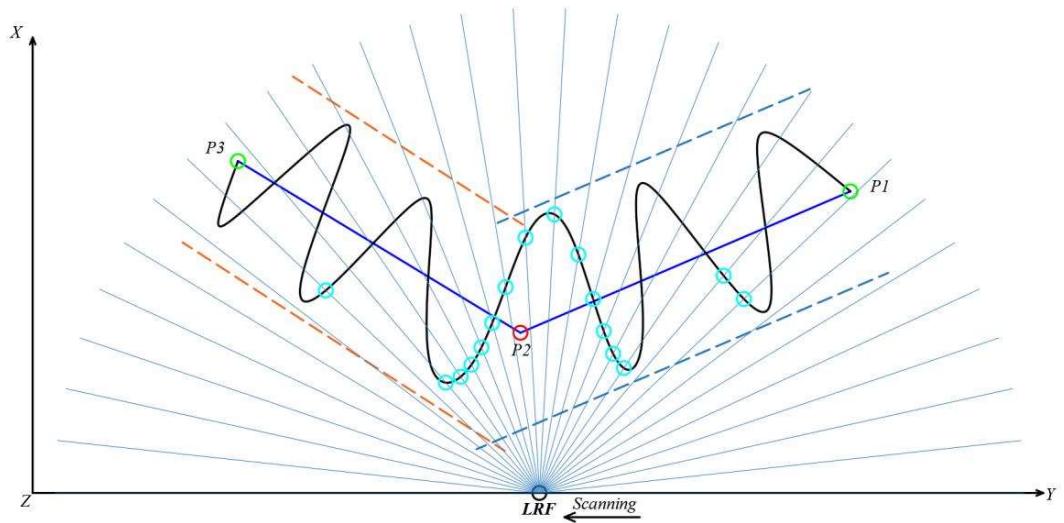


Figure 59. Occlusion when the LRF scans the edge of the detected object.

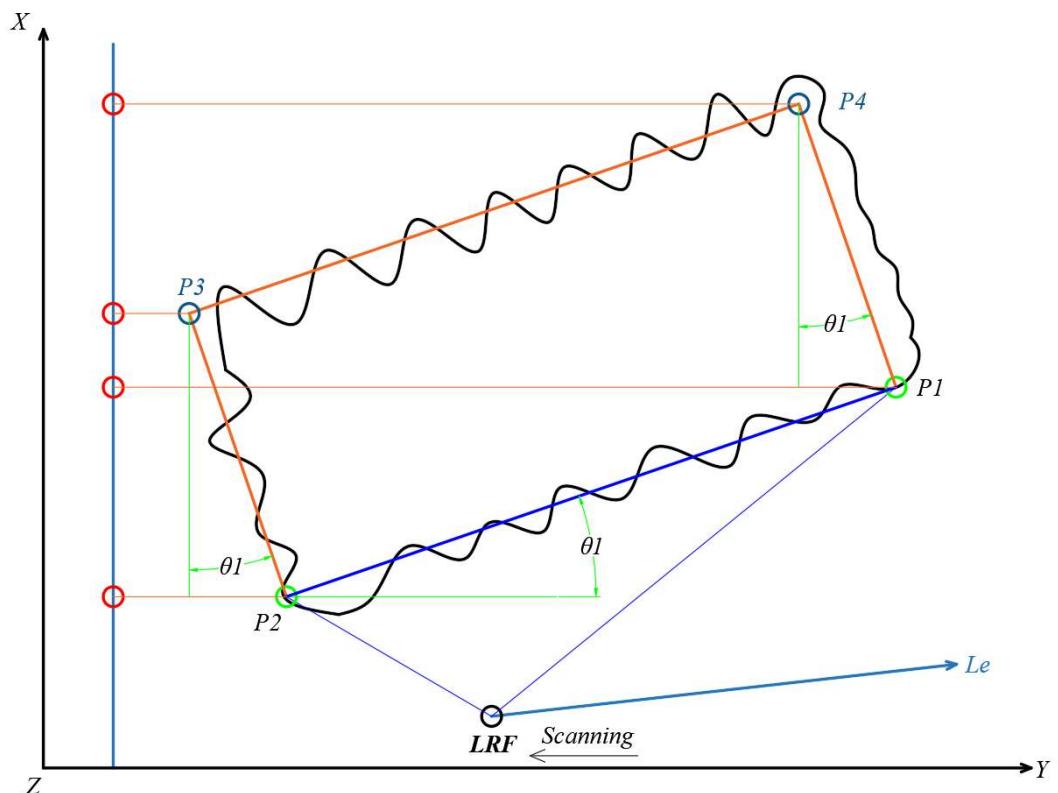


Figure 60. Cube abstract XY plane projection of object abstraction algorithm. I.

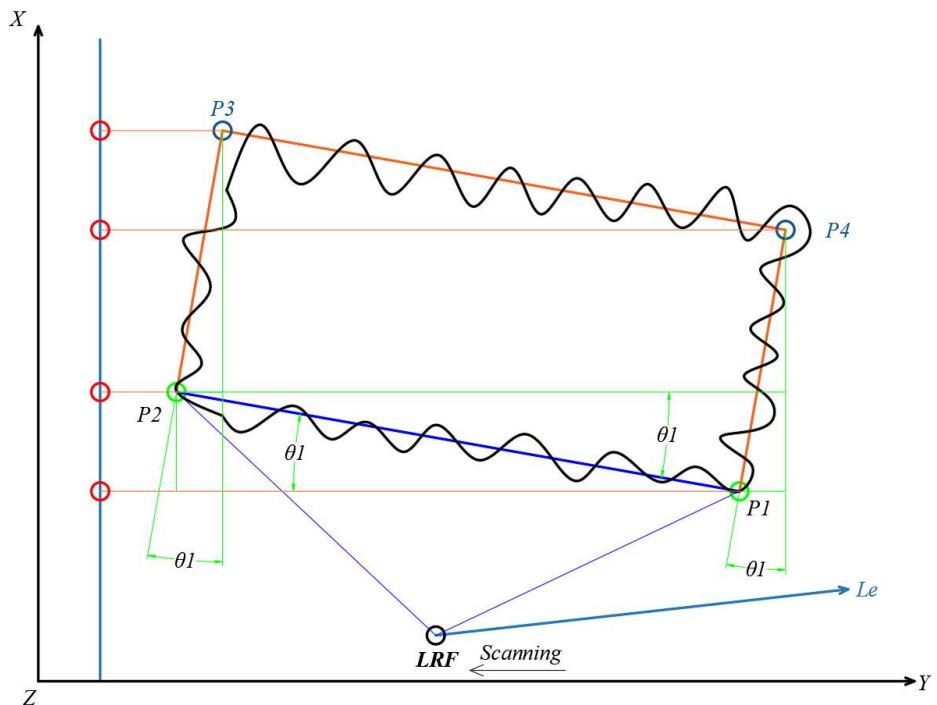


Figure 60. Cube abstract XY plane projection of object abstraction algorithm. 2.

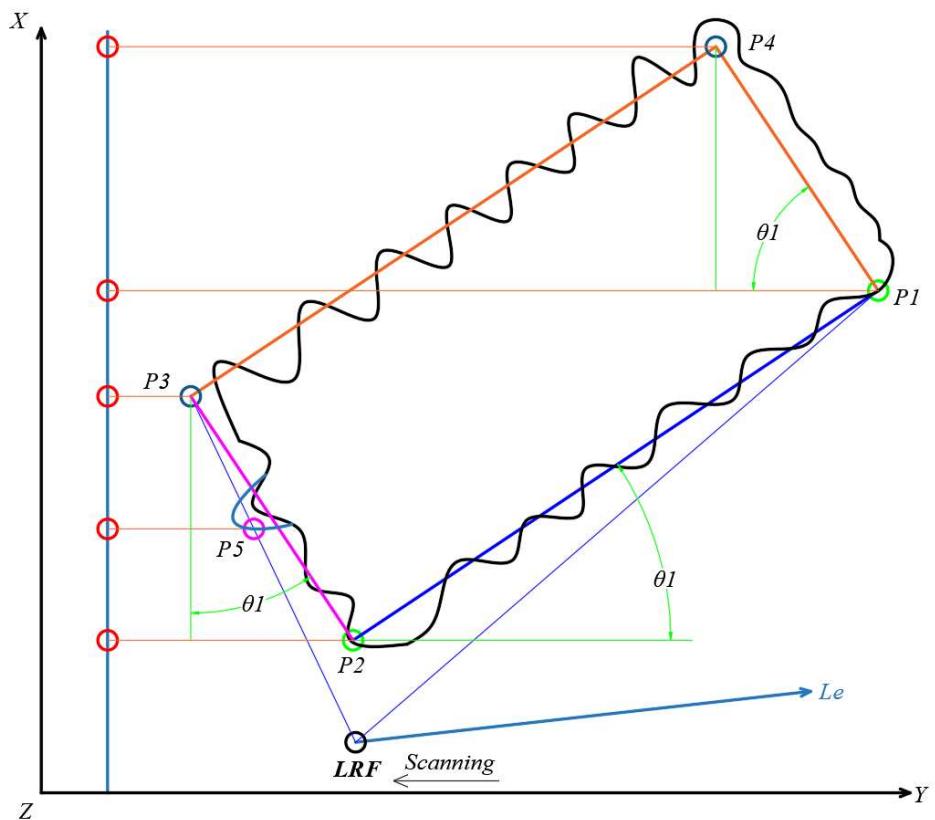


Figure 61. Cube abstract XY plane projection of object abstraction algorithm. 3.

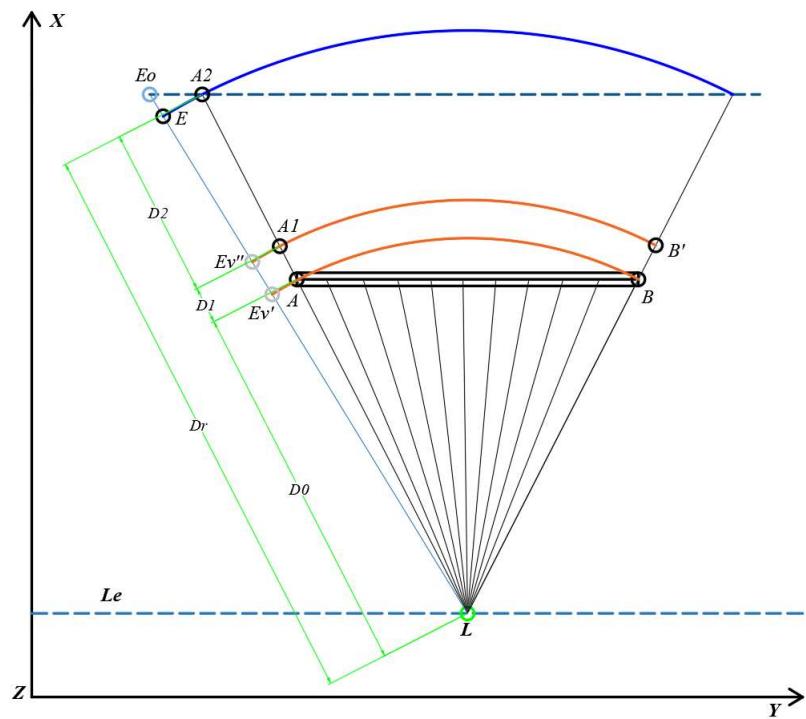


Figure 62. Edge recognition principle for LRF scanning

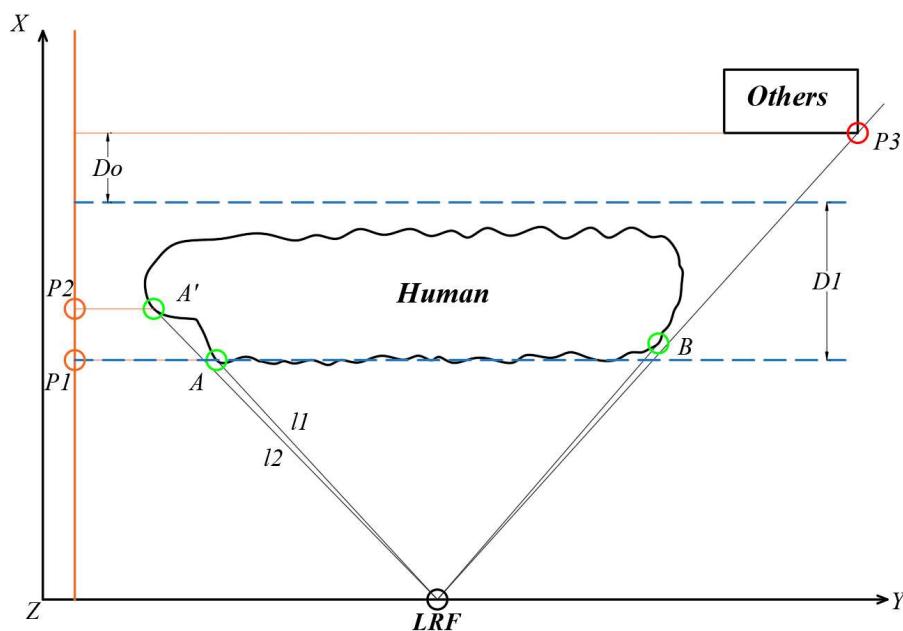


Figure 63. A sudden change in the LRF detection distance due to the large slope of the edge of the object being detected.

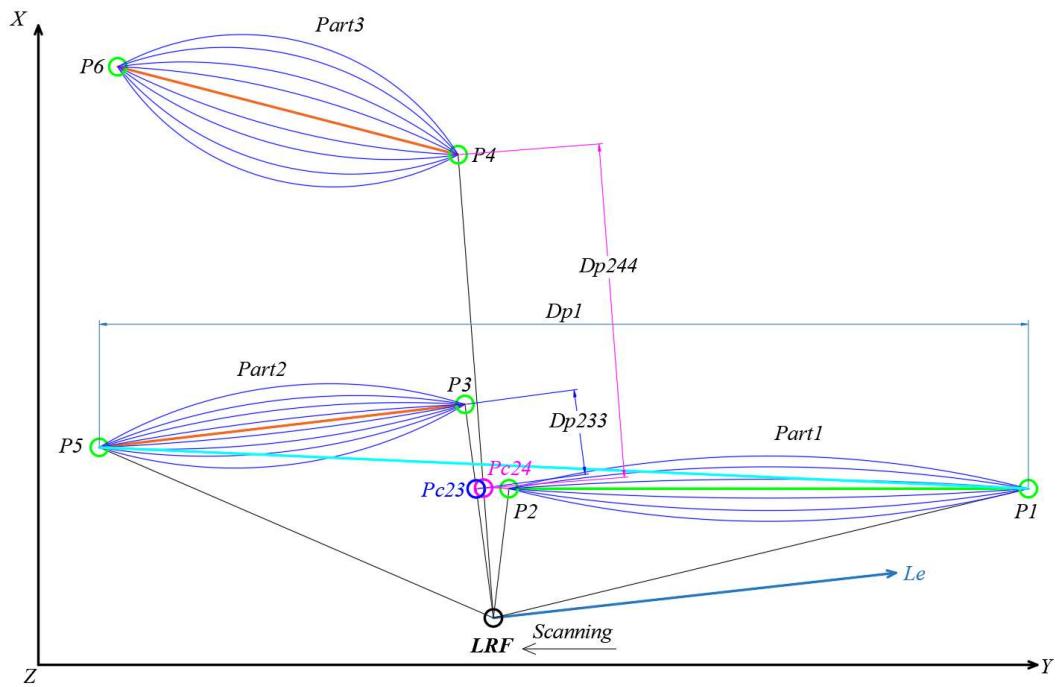


Figure 64. The most basic implementation principle for judging the true edge of the detected object. I.

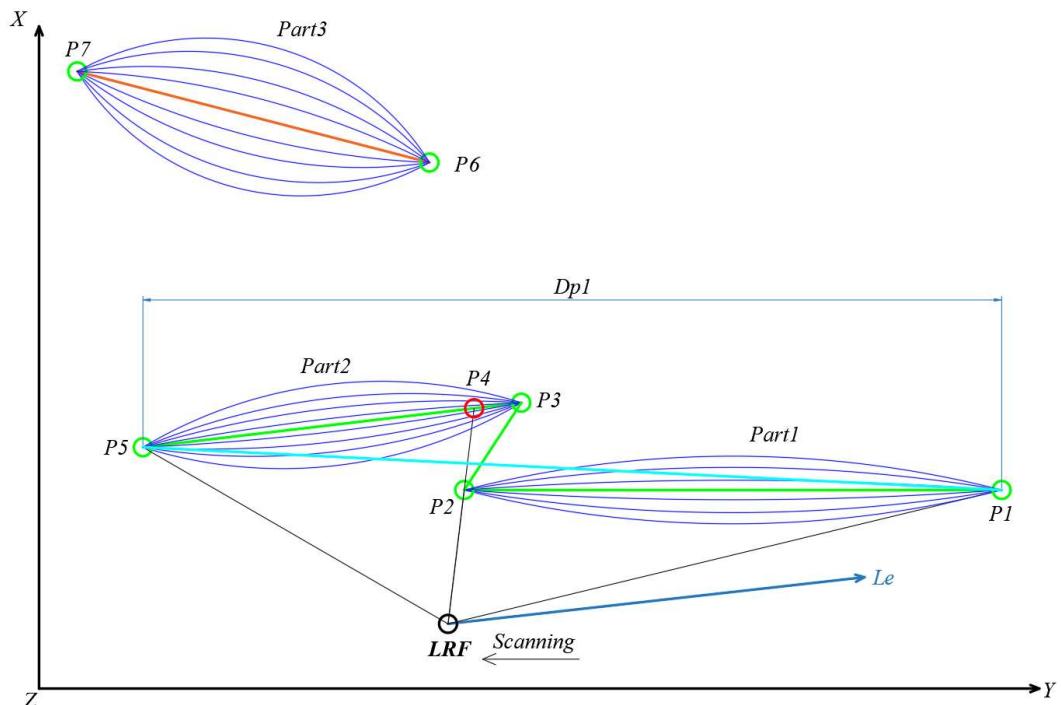


Figure 64. The most basic implementation principle for judging the true edge of the detected object. 2.

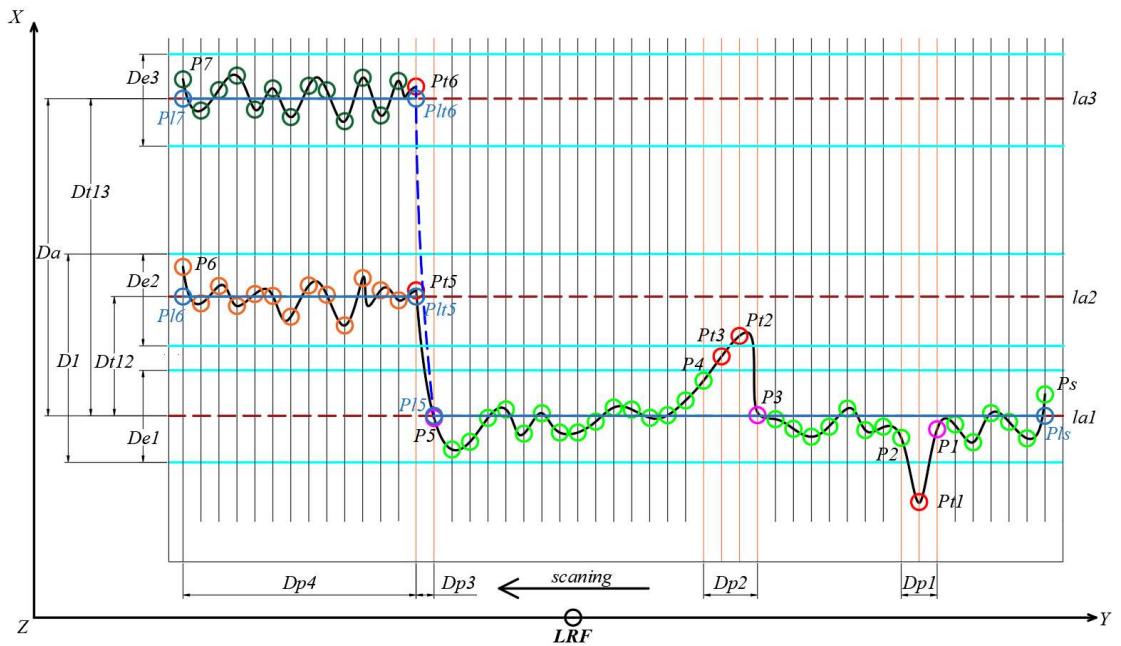


Figure 65. Using deductive continuous correlation to determine whether the edge point is an edge abrupt point or an edge uniform point.

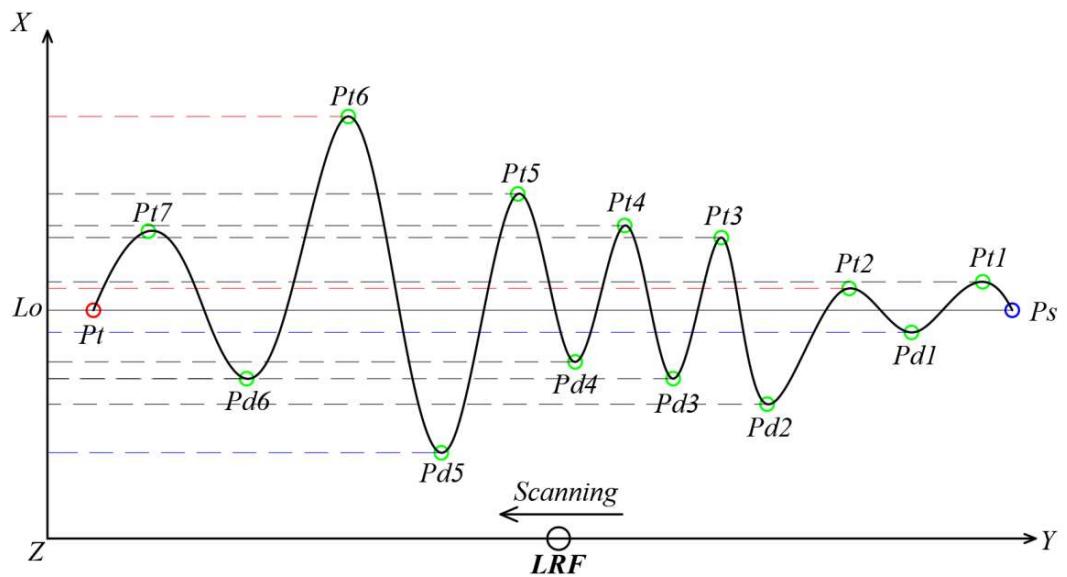


Figure 66. Determination of the range of De_i corresponding to the uniform line segment of the scanning edge.

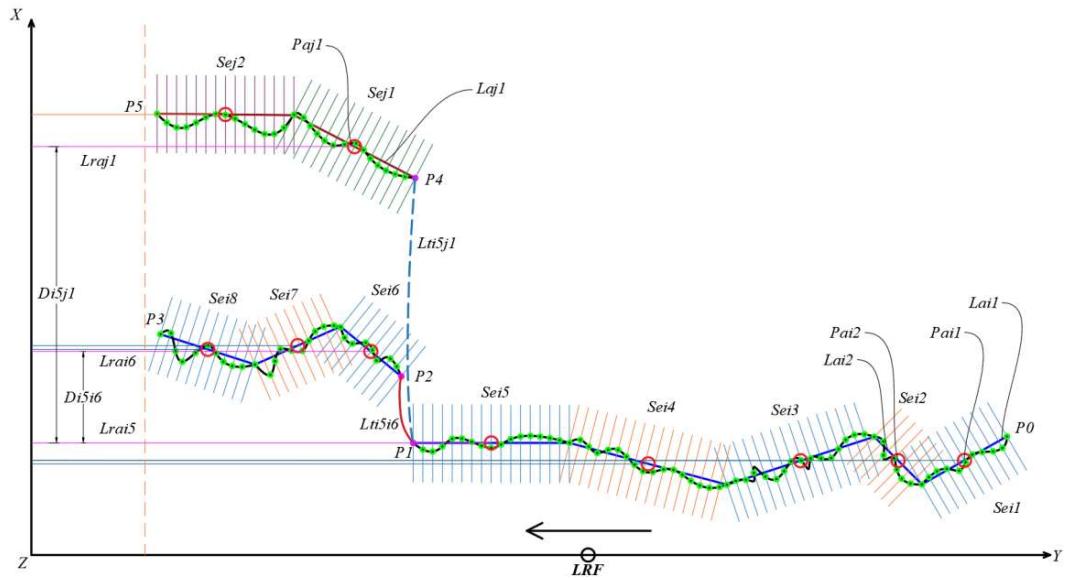


Figure 69. Object abstraction: Abstract the scanned edge segment of the detected object into a multi-segment line segment.

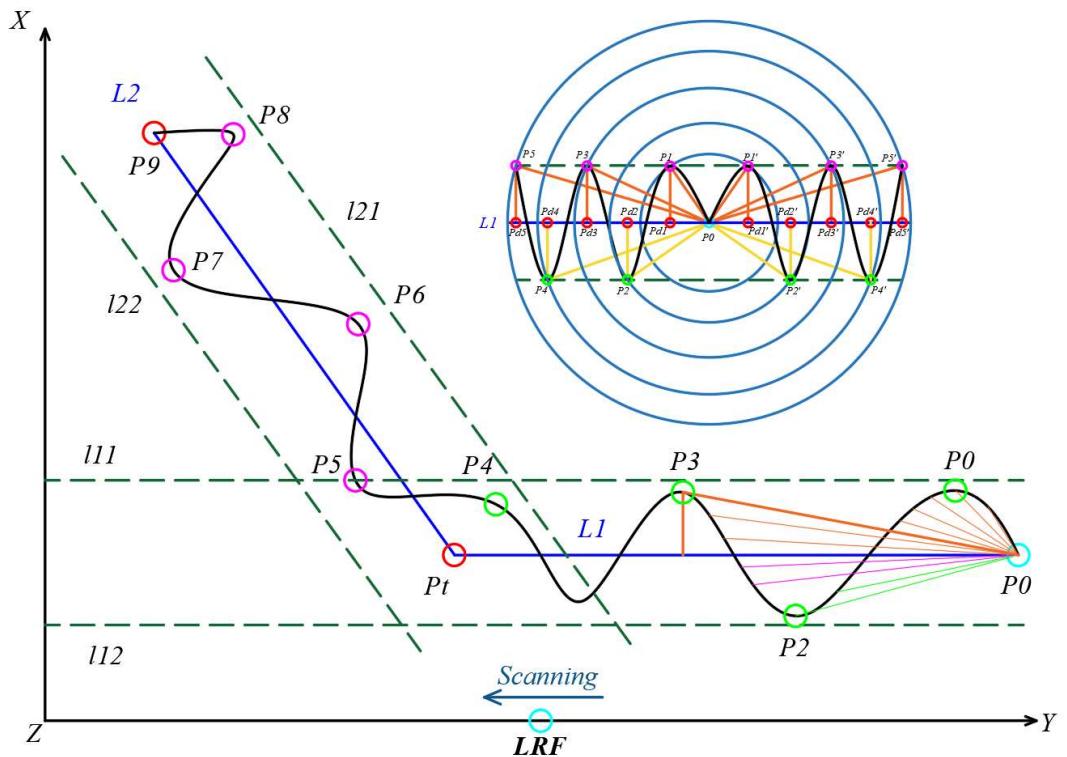


Figure 70. Algorithm principle for re-segmenting the sweeping edge uniform segments.

For more exciting details content, please refer to the other articles of this essay:

1. ROBOT DETECTION SYSTEM DESIGN ABOUT FRONT-FOLLOWING TECHNOLOGY PART 1
2. ROBOT DETECTION SYSTEM DESIGN ABOUT FRONT-FOLLOWING TECHNOLOGY PART 2
3. ROBOT DETECTION SYSTEM DESIGN ABOUT FRONT-FOLLOWING TECHNOLOGY PART 4
4. ROBOT DETECTION SYSTEM DESIGN ABOUT FRONT-FOLLOWING TECHNOLOGY PART 5

References:

- Ho, D. M. , Hu, J. S. , & Wang, J. J. . (2012). Behavior control of the mobile robot for accompanying in front of a human.
- Jung, E. J. , Yi, B. J. , & Yuta, S. I. . (2012). Control algorithms for a mobile robot tracking a human in front. IEEE/RSJ International Conference on Intelligent Robots & Systems. IEEE.
- Hu, J. S. , Wang, J. J. , & Ho, D. M. . (2014). Design of sensing system and anticipative behavior for human following of mobile robots. *IEEE Transactions on Industrial Electronics*, 61(4), 1916-1927.
- Moustris, G. P. , & Tzafestas, C. S. . (2017). Intention-based front-following control for an intelligent robotic rollator in indoor environments. *Computational Intelligence*. IEEE.
- Nikdel, P. , Shrestha, R. , & Vaughan, R. . (2018). [IEEE 2018 IEEE International Conference on Robotics and Automation (ICRA) - Brisbane, Australia (2018.5.21-2018.5.25)] 2018 IEEE International Conference on Robotics and Automation (ICRA) - The Hands-Free Push-Cart: Autonomous Following in Front by Predicting User Trajectory Around Obstacles. (pp.1-7)

Project Research Report

ROBOT DETECTION SYSTEM DESIGN ABOUT FRONT-FOLLOWING TECHNOLOGY

PART 4

**Jinwei Lin
(JY Lin, Shenzhen, China)
January 29, 2019**

ID ORCID: <https://orcid.org/0000-0003-0558-6699>

<http://www.ydook.com/>

ALGORITHM AND MODEL FOR DETECTING THE HEAD

The content of this part of the paper will introduce how to model the head detection of the target person. At the same time, an algorithm for acquiring the detected data and processing the detected data and obtaining good results will be given.

Analysis and discussion on modeling human heads

When a human is walking, the orientation of the head in most cases will follow the direction of the eye and pay attention to the path ahead. However, in the process of human walking, there will also be situations in which humans shift their focus from the path they want to travel to pay attention to other directions. That is to say, in this case, the head posture of the target is unable to provide accurate predicted path data. Therefore, the author sets a filter to filter the head detection data of the target person to extract useful detection data. On the other hand, the author has two important purposes for setting the head detection. One purpose is to obtain certain prediction data about the future path of the human being through the LRF measurement of the human head orientation, and assist the path planner in the future path of the target person. Make more accurate predictions. Another object is to create a human head LRF scan recognition model by scanning the head data of the target person for human head recognition to be described later. Obviously, the technical requirements for building a human head recognition system are more difficult. The main technical difficulty of human head recognition is how to accurately scan the target person's head, (eg, LRF physical structure precision control, efficient design of scanning algorithm), and how to establish high-precision target characters. Identification model.

In the scanning algorithm, the author sets the LRF to scan the entire head of the target person. A locked scan algorithm is used. The specific implementation principle of the lock scan algorithm has been analyzed and discussed in detail at the beginning of this project report. In the implementation of this part of the algorithm, it is necessary to obtain good algorithm results by using a combination of various algorithms described above. The entire scan of the target person will be done in the DRF scan rectangle corresponding to the target person.

The specific implementation process and characteristics of human head modeling scan

As shown in Fig. 70, the LRF performs a ray scan of the target person. It can be seen that the human head is a part that is often exposed outside and is very attractive in appearance relative to other parts of the human body. As shown in Figure 71, the LRF will use the DRF modeling scan algorithm to scan the head of the target person and extract useful scan data information. As shown in Fig. 72, the case where the head area of the target person is scanned by the LRF is shown. From the analysis of Fig. 71 and Fig. 72, it can be known that the head area of the target person has many parts having obvious detection and recognition features, such as the outline of the ear, the nose, and the face. Of course, the target person does not always express these facial contours. Even some parts of the head area of the target person are obscured (EG: the target person puts on his eyes and covers his eyes, puts on a mask, covers his mouth, puts on a hat, and blocks The head of the target person, etc.), those objects used to shield are re-played as a distinguishing feature of the detected part. In general, in the process of the robot performing a follow-up task to the target character, the significant distinguishing feature of the target person's head region does not change too much, so the detected feature of the target person's head region can still function. Very important segmentation recognition.

The role of human head modeling scans in predicting the walking behavior of target people

Part of what is going to be described here is how the author models the scan of the target person's head and how it assists the robot in performing the person following the target character. And when the robot performs the forward-going task, especially when performing the forward-following person, when facing the intersection and the T-junction, if you make the right left or right choice.

1. Use the modeling of the human head to assist the robot in predicting the future path of the target person:

The modeling scan of the head region of the target task using the DRF scan rectangle can play a role in the robot's follow-up task. There are also some related explanations and analysis in the previous part of the project report. Mainly after adding a scan detection area, the author can obtain more information about the target person while walking. This facilitates the author's scanning decision algorithm to make more

reasonable decisions. For example, the author can scan to determine the orientation of the head during the walking process. In general, when a human is walking, the orientation of the human head generally coincides with or is generally in the same direction as the direction of human eye gaze. While human beings are walking, it is impossible for human eyes to never look at the path or direction they are going to walk. In fact, most humans are walking, most of the time, paying attention to the path or direction they are going to walk. This has helped the author to build a recognition prediction algorithm. Of course, there are errors in such ancillary algorithms, and further, the author cannot ask the target person to keep their eyes fixed on the path or direction they are going to walk while using the robot for forward-following tasks. . Because doing so greatly reduces the user experience of the robot, which is why the author does not recommend such enforcement. And such an algorithm is also the meaning and fun of intelligent algorithms.

Although the head of the target person has little effect on the auxiliary help that the author can predict the future path of the target person, and there is a large prediction error. However, the authors reduce this error by jointly reducing the error of the algorithm design idea, and improve the reliable utilization of the detection data of the target person orientation. In the following vector synthesis prediction algorithm, the author will describe in detail the algorithm flow for vector synthesis prediction using the four major partitions of the detected target person. Among them, the detection of the head orientation of the target person also plays a certain role.

2. Use the modeling of the human head to assist the robot in the implementation of the action when waiting for the command intersection:

In fact, when the LRF scans the head region of the target person using the DRF rectangle, the algorithm uses the detection data of obtaining the head orientation of the detected target person to assist in determining the direction of the future path of the target person, not using the target. The greatest benefit of character modeling of the head region. The greatest benefit of using the target person's head region modeling is that the human head region is the most human body part when some further and deeper analysis of the target person is required. One of the parts that can reflect individual characteristics, the author can obtain more detection data by modeling the head of the target person to improve the recognition accuracy of the model designed by the author. This is the author's report in the project. Introduce and design the important purpose of the LRF scanning modeling algorithm for the head region of the target person. As shown in Fig. 72.2, the author uses a rectangle to represent the head detection area of the target person in the overhead XY plane projection view in the abstract algorithm, and uses arrows to represent the forward orientation of the target person's head area.

In order to facilitate the discussion and analysis of the detection algorithm of the detection area of the detected body parts of the following three divided target characters, after the algorithm description and analysis of this part is finished, this paper will enter another research report of this project. Explaining the very important data processing algorithm, this data processing algorithm is a vector synthesis prediction algorithm.

Then the author then briefly describes and analyzes the related algorithms of the remaining three detected parts of the target person in the latter part.

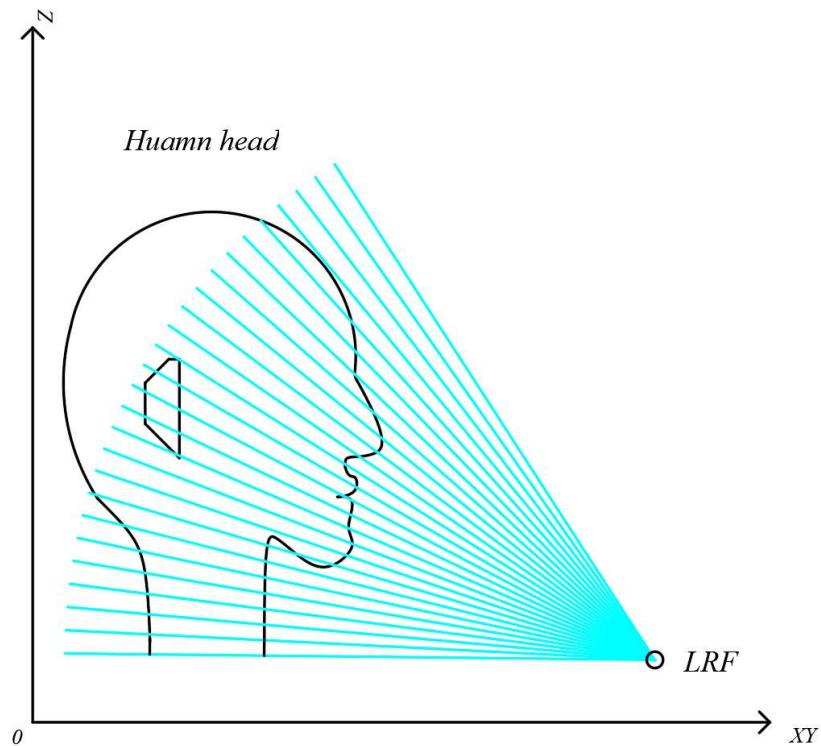


Figure 70. LRF scans the human head. Side view.

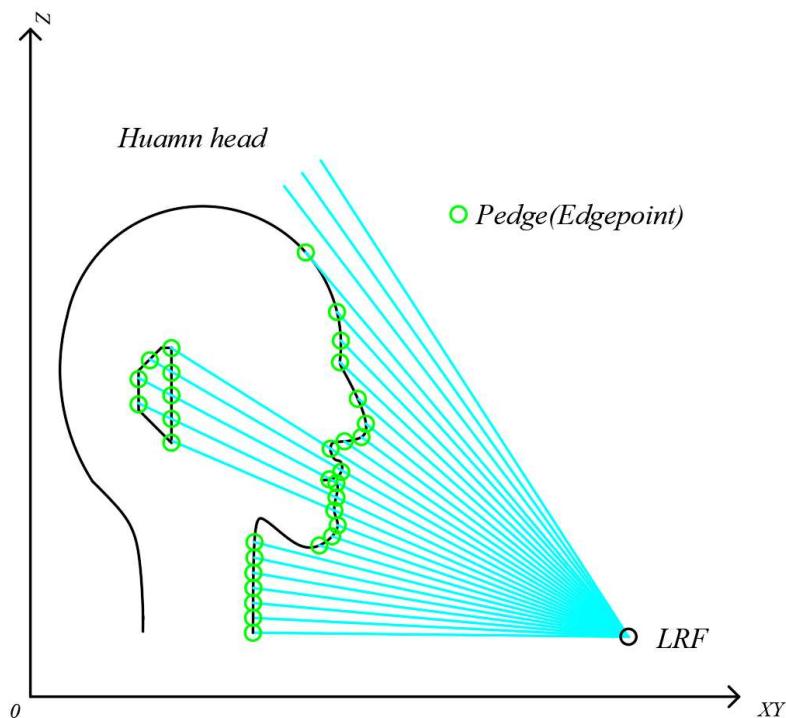


Figure 71. LRF scans the human head. Side view. Scan edge points are displayed.

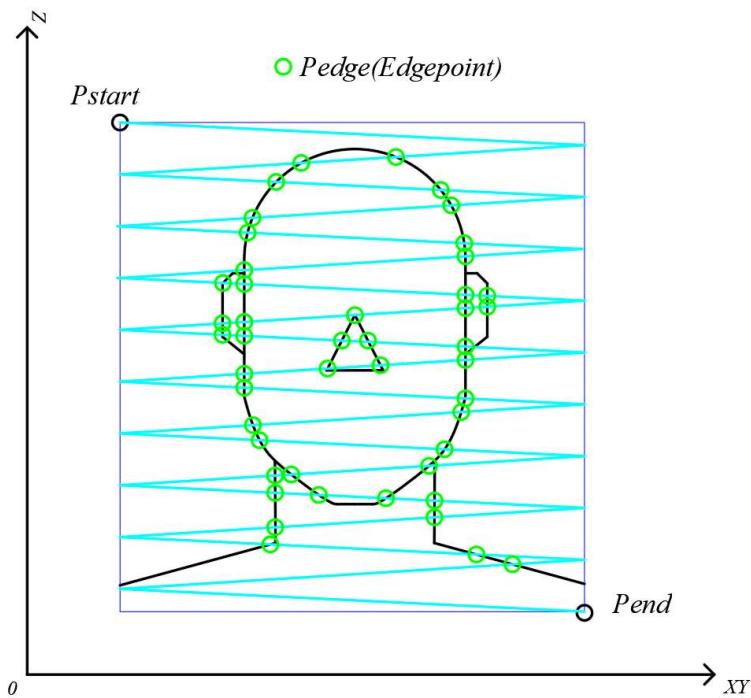


Figure 72. LRF scans the human head. Positive display. Scan edge points are displayed.

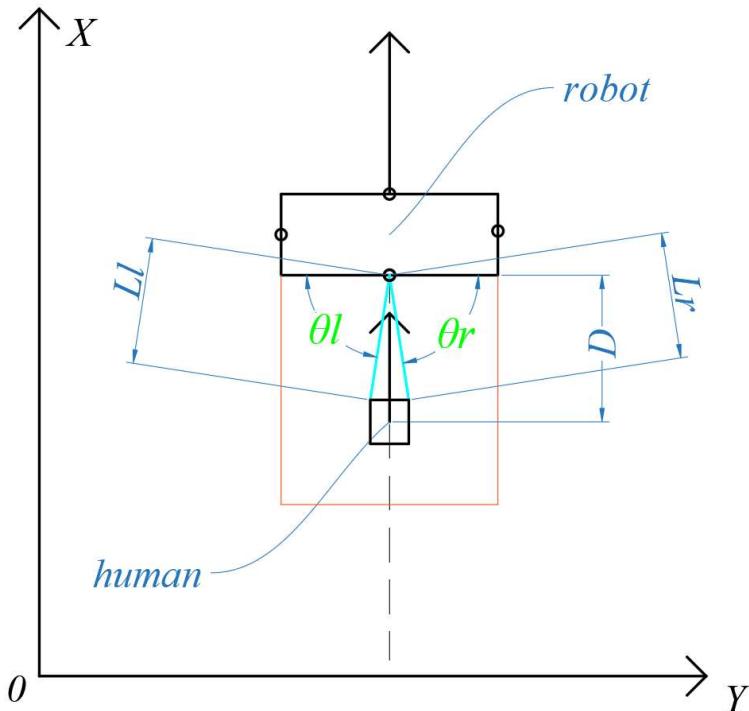


Figure 72.2. LRF scans the human head. An indication of the forward direction of the head detection area of the target person to be scanned is scanned.

VECTOR SYNTHESIS PREDICTION ALGORITHM

Specific algorithm design and implementation principle

From the foregoing analysis and discussion, it can be known that the entire detected body of the target person is divided into four detected parts in the algorithm discussed herein. The four tested parts are: head, torso, legs, and feet. It can be known from the analysis of the previous part that the trustworthiness of the future path of the predicted target person which the scan detection data of the four detected parts can provide is different. If only a certain part of the four major parts of the target person is used for prediction, the predicted future path of the target person has a large error. Because human normal walking is an activity that requires coordination and cooperation in various parts of the human body. For the four detected parts divided in this paper, each part has auxiliary coordination effect on the data prediction of the other three parts, so the author will use this part and the whole combination to coordinate the operation to reduce the calculation error. Design ideas and construct the author's prediction algorithm. Since the detection data of the four detected parts are relatively independent, how can the authors combine the prediction results of the four major parts, and then get the most suitable and best predictive future path orientation for the target person? In order to solve this problem, the author designed a vector synthesis prediction algorithm. The following is a specific algorithm description and analysis:

The design and implementation principle of the vector synthesis prediction algorithm is that by classifying different influencing factors, each influencing factor corresponds to a positive influencing factor, and the range of the positive influencing factor τ is:

$$0 \leq \tau \leq 1$$

The above positive impact factor is used to describe the contribution rate of the influencing factors corresponding to the positive impact factor to the experimental results. That is to say, the larger the positive impact factor of an influencing factor, the more the influencing factor can reflect the correct experimental result information. Therefore, in terms of the positive impact factor of a certain influencing factor, there is also a negative influence factor, or error influence factor, expressed by λ . The positive impact factor has the following relationship with the error impact factor:

$$\tau + \lambda = 1$$

That is, the larger the error influence factor of a certain influencing factor, the smaller the corresponding positive influence factor, and the smaller the error influence factor of a certain influencing factor, the larger the corresponding positive influence factor. When the positive factor of a certain influencing factor is 1, the influencing factor is called the complete correlation influencing factor of the corresponding experimental results, indicating that the influencing factors are error-free and directly correspond to the experimental results. When the positive factor of a certain influencing factor is 0, the influencing factor is called the completely unrelated factor of the corresponding experimental results, indicating that the influencing factors are irrelevant to the experimental results and have no effect.

Through the above discussion, the author understands the concept of influencing factors and their corresponding impact factors. So how does the author relate different influencing factors to calculate the final experimental predictions? To solve this problem, the author introduces the idea of vector synthesis to represent the superposition of different influence factors. Since the prediction algorithm designed in this project is to get the orientation of the future path of the target person, the concept of vector is very suitable for expressing the author to obtain the experimental result. This is one of the reasons why the author designs the vector synthesis algorithm.

In the following, the author will further elaborate and analyze the implementation essence of the vector synthesis prediction algorithm by expounding the vector synthesis prediction algorithm of the four detected parts of the target task described in the project report.

The four test sites represent four different influencing factors. The author sets the reliability level of the prediction data for the corresponding impact factors of the four detection parts:

First level: the predicted future orientation indicated by the target's feet detection data;

Second level: the predicted future orientation indicated by the torso detection data of the target person;

Third level: the predicted future orientation indicated by the target detection data of the target person;

The fourth level: the predicted future orientation indicated by the target detection data of the target person;

Then the author sets the corresponding positive impact factor for the reliability level of the prediction data corresponding to the four detected parts of the above target person:

The first level positive impact factor: τ_1 , the experimentally recommended value is set to 0.9;

The second level positive impact factor: τ_2 , the experimentally recommended value is set to 0.8;

The third level positive impact factor: τ_3 , the experimentally recommended value

is set to 0.6;

The fourth level positive impact factor: τ_4 , the experimentally recommended value is set to 0.2;

In the above algorithm execution, the author assigns a corresponding prediction data reliability level to each influencing factor, and then sets a corresponding positive impact factor for each prediction data reliability level. As shown in Fig. 73, the author uses the percentile system to express the positive influence factors of the four detected parts of the target task, and express them by the influence factor vectors of different colors having the corresponding mode size ratios. The 100% modulus size vector is the reference basic element vector set by the author. Taking the basic element 100% D_p as the benchmark, the corresponding influence factor vectors of corresponding feet, torso, legs, and head are 90% D_f , 80% D_t , 60% D_l , and 20% D_h , respectively. The size of the vector modulus referring to the basic element vector is D_p , and the sizes of the vector modes defining the influence factor vectors D_f , D_t , D_l , D_h are D_f , D_t , D_l , D_h , respectively, and the correspondence between them is:

$$\begin{cases} D_f = 90\% \cdot D_p \\ D_t = 80\% \cdot D_p \\ D_l = 60\% \cdot D_p \\ D_h = 20\% \cdot D_p \end{cases}$$

After defining the influence factor vector corresponding to the impact factor corresponding to the influencing factor, the author can calculate and solve the final predictive vector. The basic step is to add the corresponding influence factor vectors of the influence factors corresponding to the respective influencing factors to the sequential vector, and then the vector of all the affected factor vectors and the represented vector are the final predictions finally requested by the author. Vector. Another notable problem is that due to the order interchangeability of vector additions, the author can simply add all the influence factor vectors to get the final prediction vector. As for the order in which the specific influence factor vectors are added, it is not mandatory. The author adds the order of each influence factor vector by sorting according to the size of the vector's modulus, from large to small, and sequentially added.

Here are the specific steps to elaborate:

1. As shown in Fig. 74, the black arrow vector r_0 represents the reference axis of the space coordinate system, which may be the X axis or the Y axis of the XYZ following coordinate system, or may be the reference axis of $\Phi=0$ of the R Φ Θ detecting spherical coordinate system. The black origin represents the position of the target character in the spatial coordinate system.

2. In the order suggested by the previous author,

Calculate first:

$$\overrightarrow{lc1} = \overrightarrow{lf} + \overrightarrow{lt}$$

Then calculate:

$$\overrightarrow{lc2} = \overrightarrow{lc1} + \overrightarrow{ll}$$

Then calculate:

$$\overrightarrow{lc3} = \overrightarrow{lc2} + \overrightarrow{lh}$$

After the above calculation, the obtained vector lc3 is the final prediction vector. The *lmax* vector in Fig. 74 is a vector obtained by performing the same direction addition of the respective influence factor vectors along the direction of the final prediction vector lc3, and the author becomes a vector perfect vector. The author defines that the closer the modulus size of the final prediction vector is to the size of the maximum perfect vector, the more accurate the corresponding prediction result is, and the higher the corresponding prediction trustworthiness.

It can be known from the above analysis that if the influence factor corresponding to an influencing factor is closer to 1, the influence of the influencing factor on the final overall prediction result will be greater. According to the algorithm of the previous part, the author knows that since the detection data of the human head region can provide a low degree of predictive trust, the author only sets a positive influence of 20% for the prediction result of the detection data of the human head region. factor. As shown in Fig. 75, there is only one difference between Fig. 75 and Fig. 74, that is, the direction of lh is changed, that is, at this time, the orientation of the target character predicted by the lh detection data shown in Fig. 75 has changed, then according to the previous author The execution step of the vector addition prediction algorithm, the final prediction vector obtained by the author is the lc4 vector in Fig. 75. To better compare lc3 with lc4, the author moves the starting points of these two vectors to the same position. Lc3 and lc4 are not much different, the difference between the sizes of the two vectors is small, and the angle between the two vectors is smaller than the angle of r0, but there are certain differences. That is, under the vector synthesis prediction algorithm of the model, the detection data of the head region of the target person has little influence on the final prediction vector, but it also has a certain influence. At the same time, this also shows that in the vector synthesis algorithm of this part, the comprehensive consideration of each influencing factor is fully reflected.

At this point, the author can obtain the final prediction vector of the target person, that is, the final prediction result of the future path of the target person after

comprehensive analysis of the detection data of the four detected parts of the target person.

Comparison and analysis of prediction performance of vector synthesis algorithms

It can be known from the above that the vector synthesis algorithm can well consider the influence of various influencing factors, and has achieved the effect of reducing the prediction error by using more different types of reference data. Since the prediction accuracy of the four detected parts of the target person is different, in the process of the robot performing the follow-up of the target person, there may be a serious uncoordinated phenomenon of the four detected parts of the target person. That is, the scan data of the target person by the LRF at this time cannot reflect the orientation of the future path of the target person well. So how does the author know when the comprehensive predictions of the test data reflected by the four detected parts are reliable and when are they unreliable? To solve this problem, the author defines that the size of the modulus of the final predictive vector is compared to the size of the largest perfect vector. As shown in FIG. 77, the author defines the sizes of the final prediction vectors $lc3$ and $lc4$ as $Dc3$ and $Dc4$, respectively. The size of the largest perfect vector is defined as $Dmax$. And, introduce the proportional coefficient δ :

$$\delta = \frac{Dc}{Dmax}$$

The above Dci is the size of the modulus of the final prediction vector obtained by the vector synthesis prediction algorithm corresponding to the i -th time. In summary, the author knows that the range of the scale factor δ is:

$$0 \leq \delta \leq 1$$

If the proportional coefficient relationship satisfies δ equal to 1, then the prediction result is said to be a completely successful prediction. If the scale factor δ is equal to 0, then the prediction result is called a complete failure prediction. Generally, since the detection data of the four detected parts of the target person have certain prediction errors, but also have a certain degree of trustworthiness, both the completely successful prediction and the complete failure prediction are two very extreme algorithms. As a result, it does not generally appear. Of course, the author hopes that the δ obtained by the vector synthesis prediction algorithm is equal to 1, but the probability of such prediction is very small. Similarly, the δ obtained by the vector synthesis prediction

algorithm is equal to 0 each time, so the probability of predicting the result is also very small. However, δ equal to 0 is not impossible. As shown in Fig. 78, the result of the vector synthesis algorithm with δ equal to 0 is shown. By analyzing Fig. 78, it can be known that the case where δ is equal to 0 is that the relationship between the posture motion of the four detected parts of the target person and the posture motion of the walking is very small. At this point, the target person can be considered be in a non-walking state, but there are other actions and activities.

From the above analysis, it can be known that the magnitude of δ can reflect the degree of trustworthiness of the prediction result of the vector synthesis prediction algorithm for predicting the future path of the target person. From another point of view, the size of δ can reflect not only the prediction of the direction of the future path of the target person obtained by the vector synthesis prediction algorithm, but also the direction of the target person who is walking toward the predicted future path. The size of the probability (NB: This also contains the magnitude of the prediction error.). This is very useful for the designer to plan the robot to follow the specific path of the target person. Through the vector synthesis algorithm, the author obtains the prediction of the future path of the target person and the corresponding execution possibility. Then, is the designer to design the robot to execute all the predicted results immediately? For this question, the author's suggestion is to perform good predictions and ignore bad predictions. Then, another problem arises again. What kind of prediction result is a good prediction result, and what kind of prediction result is a bad prediction result? To solve this problem, the author introduced a threshold proportional vector lT . The direction of lT is consistent with the direction of $lmax$. The size of the lT module is:

$$0 \leq DlT \leq Dmax$$

The size of the modulus of the threshold vector, DLT , represents the lowest degree of trustworthiness. That is, if the value of δ is satisfied:

$$\delta < DlT$$

Then, it is considered that the corresponding prediction result of the LRF scan data is unreliable, and the processing algorithm automatically discards the scan result of the LRF, and uses the scan result of the next LRF.

If the value of δ is satisfied:

$$\delta > DlT$$

Then, it is considered that the prediction result of the corresponding LRF scan data is reliable, and the processing algorithm automatically adopts the scan result of the LRF and uses the following path driving algorithm of the robot.

The size of the modulus of the threshold vector DLT is an experimental setting coefficient. The larger the DLT , the greater the accuracy of the prediction result of the vector synthesis prediction algorithm, and the more computational resources are

consumed. The smaller the DLT, the smaller the accuracy of the prediction result of the vector synthesis prediction algorithm, and the less the corresponding computational resources are consumed. Vector synthesis prediction algorithm has other more advanced algorithm expansion, specific expansion content, please look forward to the presentation of the second edition of the project report.

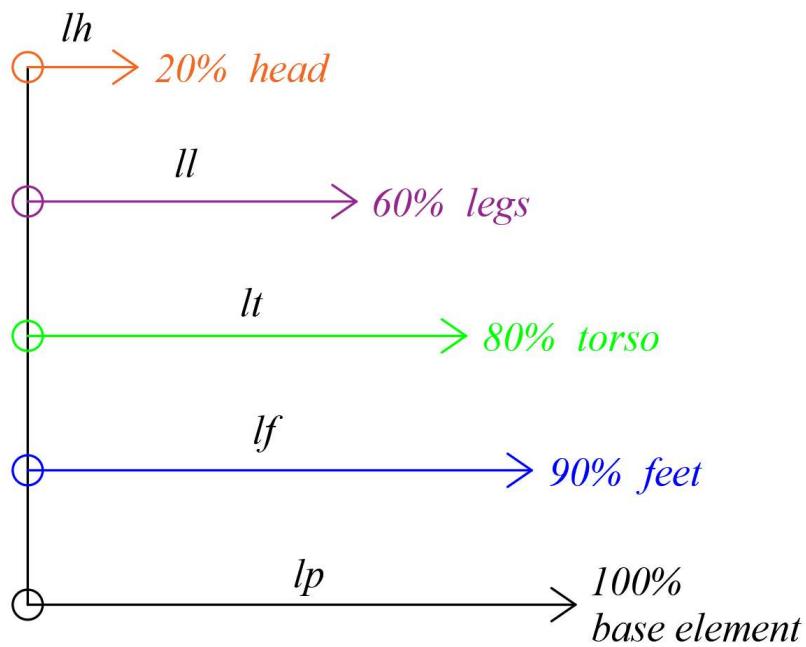


Figure 73. Setting of the influence factor of the vector synthesis prediction algorithm based on this project.

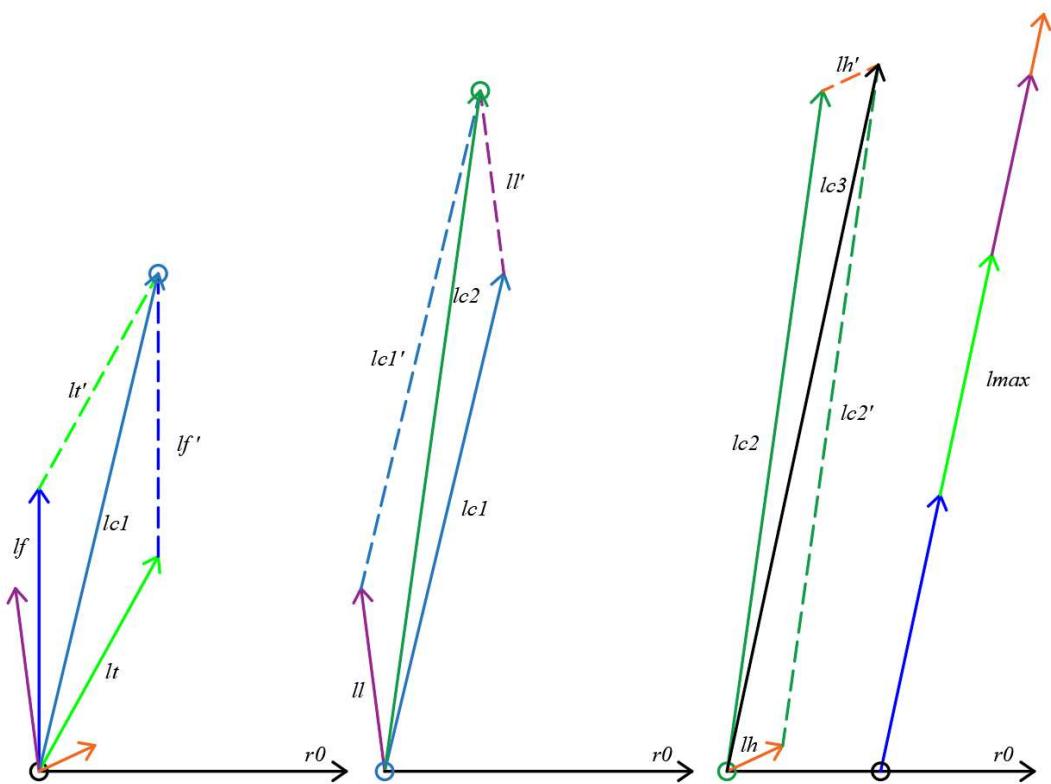


Figure 74. Implementation of the vector synthesis prediction algorithm based on this project. I.

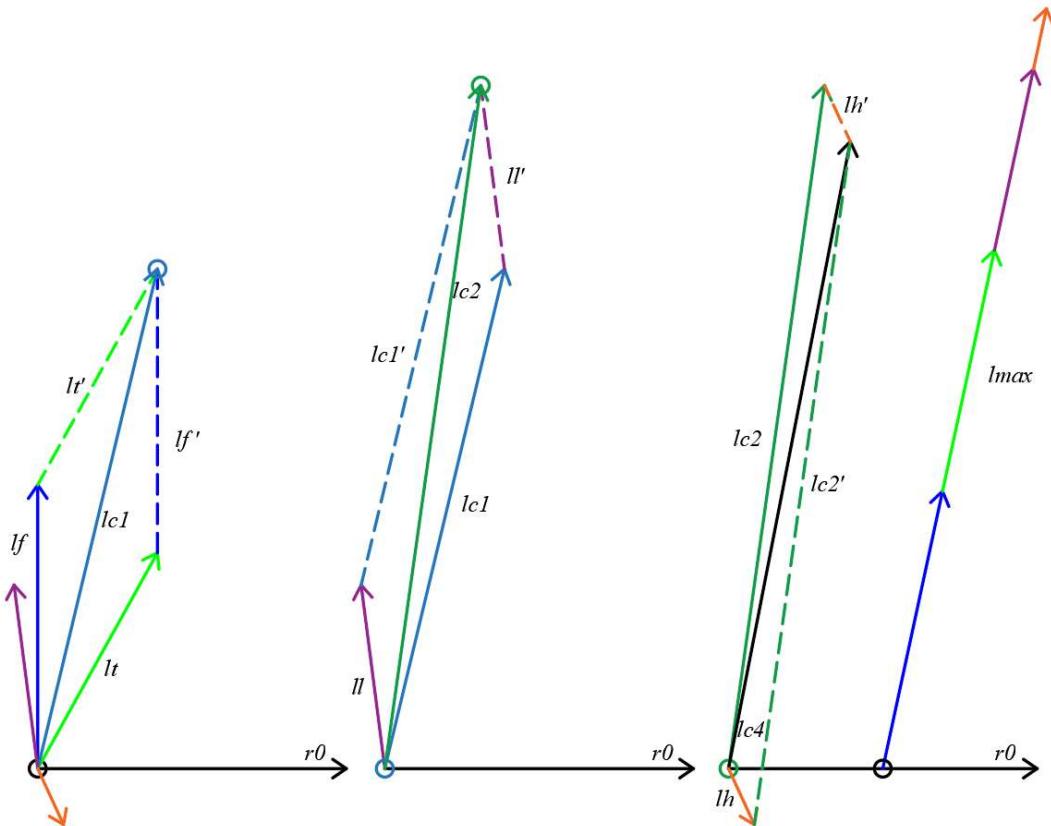


Figure 75. Implementation of a vector synthesis prediction algorithm based on this project. 5.

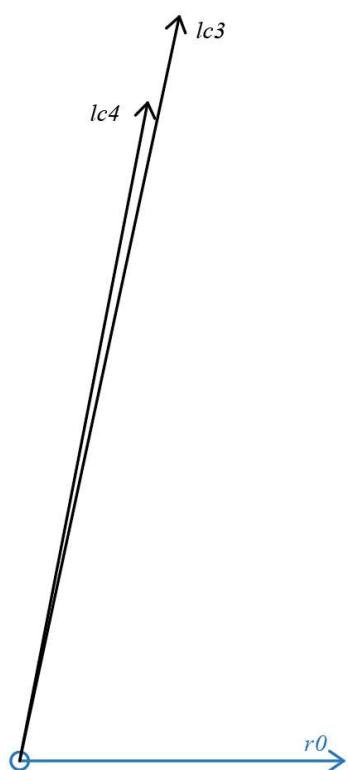


Figure 76. Effect of lh influencing factors on the final experimental predictions.

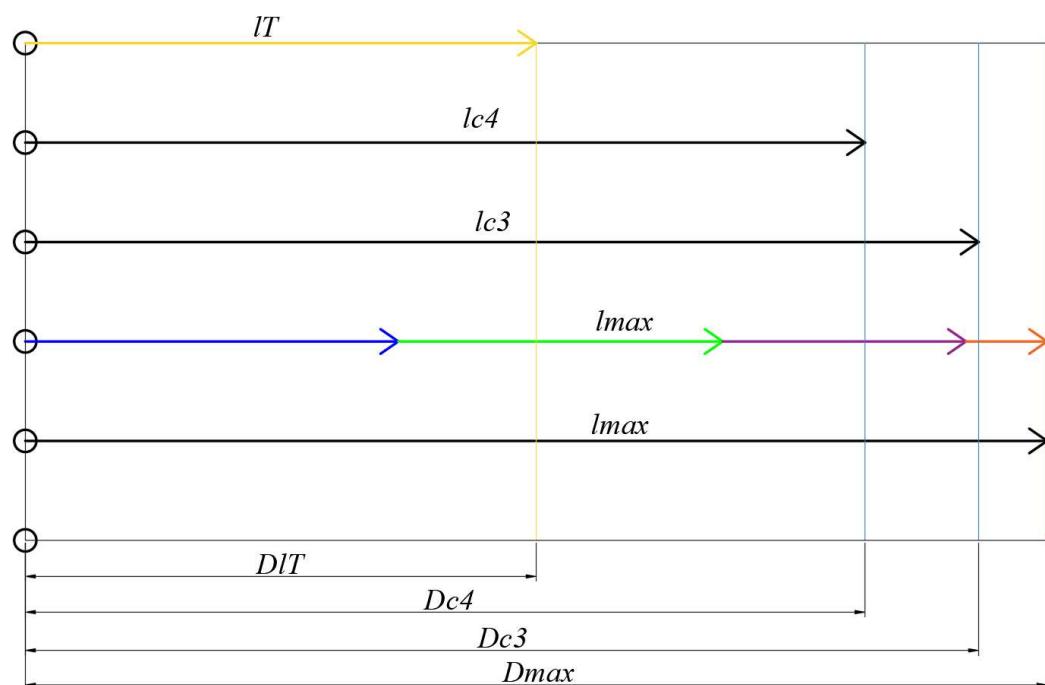


Figure 77. Effect of lh influencing factors on the final experimental predictions.

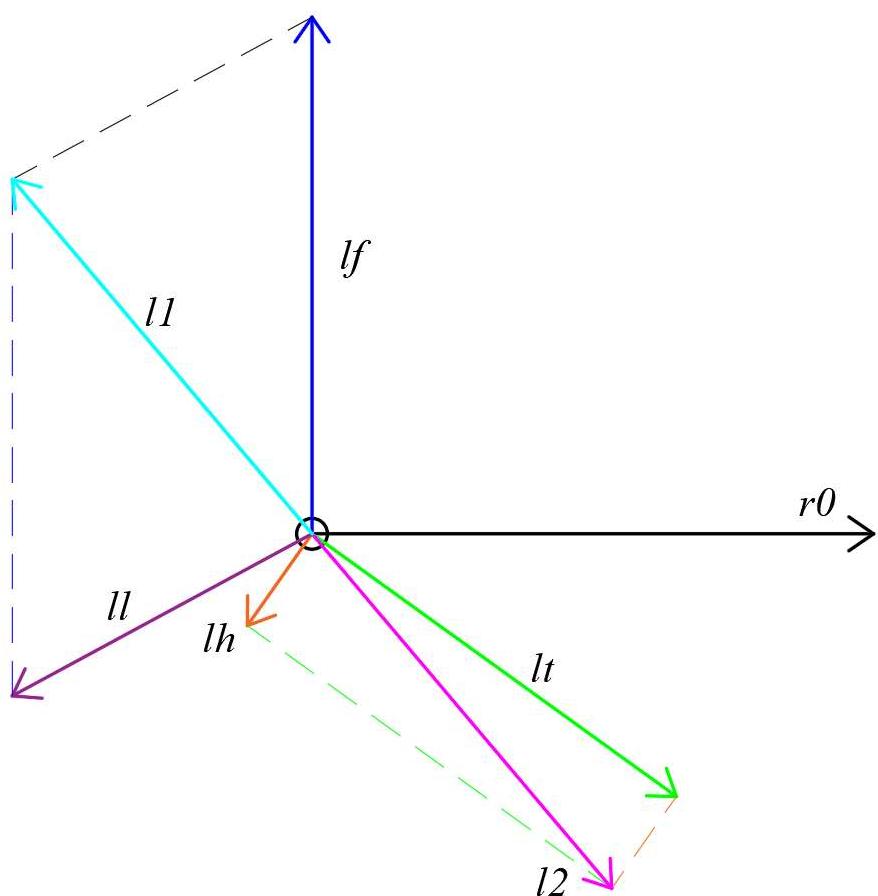


Figure 78. Special case in the vector synthesis prediction algorithm, where the vector sum is zero.

DETECTION MODEL AND ALGORITHM ANALYSIS OF HUMAN TORSO

The advantage of modeling the detection model of the torso of the scanned target person has been described in more detail in the previous section of the research report of this project. Due to the limitation of the writing time of this project report, it will not be repeated or detailed in this section. The following part will directly enter the description and analysis of the relevant algorithm content.

The human torso detection site is the detection factor that the author has set with the second largest impact factor. Therefore, it is very important to model the torso of the target person. As shown in Fig. 79, the robot mobilizes the LRF to scan the torso portion of the target person in a scan form of the DRF rectangle. Since the human arm posture is reflected in the process of human walking, there is little information to predict the future path of human walking, and the degree of correlation between the two is not large. Therefore, the author defines the double-arm detection data of the target person as error detection data. That is, in the algorithm designed by the author, the target person's dual-arm detection data will not be used to predict the future path information of the target person. As shown in Figure 79, the author scans the torso of the target person through the DRF scan rectangle algorithm. The specific scanning process is consistent with that described in the DRF scanning algorithm. What needs to be paid attention to here is that, for the convenience of analysis, the author divides the left and right round-trip scan into left scan and right scan. The *Pscanr* scan code point in Figure 79 represents the right scan point, and the *Pscanl* scan code point represents the left scan point. The *Pedge* point represents the true edge scan point of the torso portion of the target task. The *Perror* point represents the scanning detection points of the target person's arms, called error scanning points, which are ignored by the designer when performing data processing. The designer can obtain the real edge scan points corresponding to the torso of the target person through the DRF modeling and scanning algorithm and the object abstraction algorithm described above, and then abstract into a corresponding simple space cube through the object abstraction algorithm. Figure 80 is an enlarged representation of the actual implementation of the algorithm in this section.

It can be known from the above algorithm that the designer needs to identify the detected data of the target person and distinguish the identified data from the identification data of the target person's torso, so that only the torso of the target person is used in the actual prediction algorithm. Part of the test data is predicted. Fig. 81 is a schematic diagram showing an algorithm for identifying and separating the detection data of the arms of the target person in the algorithm of the present part. Based on the algorithm in the previous section, the author will combine the following two main strategies to implement this function:

1. Analyze from different time and space perspectives, set the space-time recognition interval, and calculate and evaluate the fluctuation frequency of the

detection data collected in the continuous space-time recognition interval. In the continuous recognition time interval, if the spatial transformation frequency of the scanning point is relatively high, the corresponding scanning detection area is likely to belong to the arms of the target person. In contrast, if the spatial transformation frequency of the scanning point is relatively low, the corresponding scanning detection area is likely to belong to the torso of the target person. This is because, in the process of human walking, the possibility that the posture of the human arms changes is much greater than the possibility that the posture of the human torso changes.

2. In addition to detecting and evaluating the spatiotemporal transformation frequency of the scanned points of the detected area, the designer can also perform DRF on the torso of the target person through an algorithm for identifying the edge of the detected object in the object abstraction algorithm. Rectangular scanning acquires the real edge of the torso of the detected target person, in particular, the gap area between the arms area and the torso area of the target person. Then, combined with the true width of the torso of the target person acquired during the DRF modeling scan, further algorithm identification and analysis is performed. (NB: During the DRF modeling process for the target character, the robot has obtained the true torso width of the target character, including at least the front torso width and the lateral trunk width of the target character. This is why the author emphasizes DRF construction. One of the reasons for the analog scan algorithm.)

Figure 82. shows the LRF scanning the human torso. When the LRF scans the target person in front. In the left scan and the right scan, the angle formed by the scan line and the reference line of the XY plane is constant. As shown in FIG. 82, the left scanning angle φl has the following relationship with the right scanning angle φr :

$$\varphi l = \varphi r$$

The corresponding left scan area increment dtl has the following relationship with the right scan area increment dtr :

$$dtl = dtr$$

TT and t2 in Fig. 82 represent two scanning periods, which are continuous. If the designer regards t1 as the previous scan period, then the t2 period is the next scan period immediately following the t1 period. If the designer regards t2 as the previous scan period, then the t1 period is the next scan period immediately following the t2 period. Figure 83 shows the locked scan of the torso portion of the target person using the locked scan mode of the DRF scan rectangle. Specific algorithm implementation sections can be found in the Locked Scanning Algorithm section of the previous section of this report. Figure 83 and Figure 84 show the algorithm implementation of the object

abstraction of the LRF facing the torso of the target person. For specific algorithm implementation, refer to the object abstraction algorithm and the DRF rectangle scan algorithm described in the previous section.

86 and 87 show the straight line distance between the first scan point and the last scan point of the torso portion of the target person detected by the scanning algorithm when the LRF scans the torso of the target person. And the relationship between the angle formed by the line with the point Po. The connected straight line segment between the first scan point and the last scan point of the torso detection scan of the target person's torso is referred to as the detection diameter. Since there is a test diameter, there must be a detection radius. As shown in FIG. 88, r1, r2, r3, and r4 respectively correspond to the scan radius corresponding to the scan diameter when the LRF scans the target person at different spatial positions. Since there is a scan radius, there must be a scan circle. As shown in Fig. 88, c1, c2, c3, and c4 are scan circles corresponding to the scanning radii r1, r2, r2, and r4, respectively. Figure 88 is intended to tell the designer that the designer cannot determine the appearance of the torso of the detected target task simply by the scan diameter or scan radius of the LRF. Because the scan diameter of the LRF scanned from different angles is not necessarily the maximum scan diameter corresponding to the torso portion of the target person. Figure 89 shows the interference of the torso or the relatively strong target of the target task, resulting in the LRF's data on the scanning of such target tasks without directional indication or directional indication. Not big. At this time, the disadvantages of the scanning mode of the single detection area of the target person and the scanning mode of the single-multiple detection area of the target person are displayed. In the four comprehensive detection algorithms of the detected area designed by the author, this extreme situation occurs immediately, and the robot can still obtain the future path of the target with higher accuracy by predicting the detection data of other detected parts. Forecast orientation. Since the author calculates the predicted direction of the future path of the target task indicated by the torso portion of the target person by comprehensively analyzing the vector orientation of the front portion of each segment region of the torso portion of the target person. In this way, the author's prediction algorithm will be more robust than the prediction algorithm that uses only the detection data of a single detection area. (NB: This part of the processing algorithm will use the vector synthesis prediction algorithm described in the previous section. When viewing this project research report, the reader should have a coherent and Huitong thinking, with a partial and overall perspective To analyze the specific application of each part of the algorithm and related thinking calls and thinking connections.) As shown in Figure 90, when the principle of this part of the algorithm is explained, the author defines a rectangle in the XY projection surface to represent the target person. Abstract rectangle of the torso. The orientation of the future path of the target task predicted by the detection data of the torso of the target person is indicated by an arrow perpendicular to the front of the rectangle.

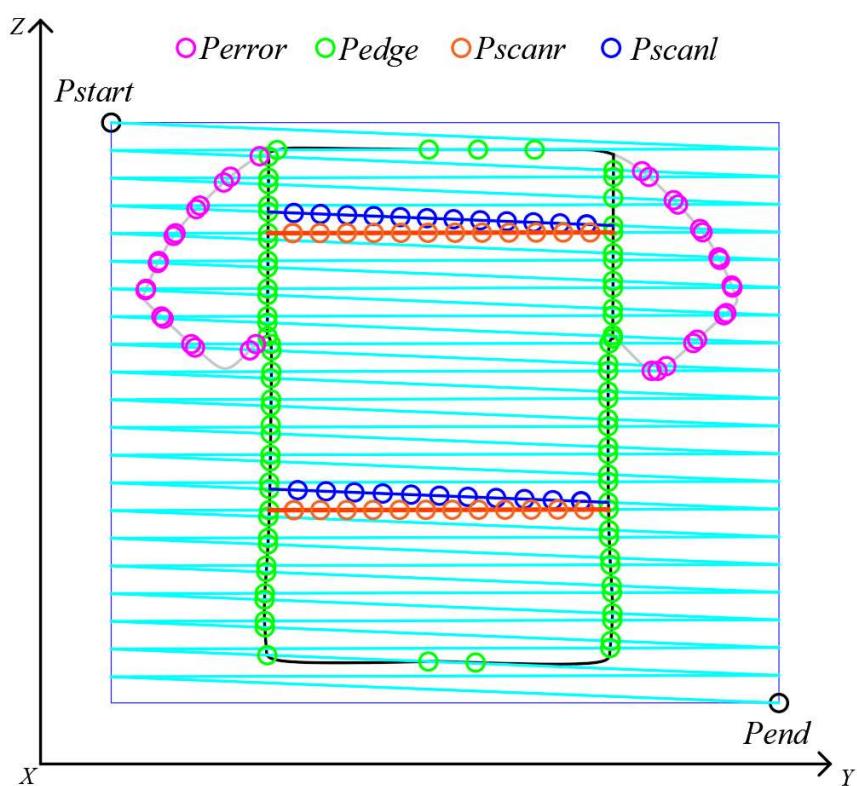


Figure 79. LRF scans the human torso. Positive display. Scan edge points are displayed. I.

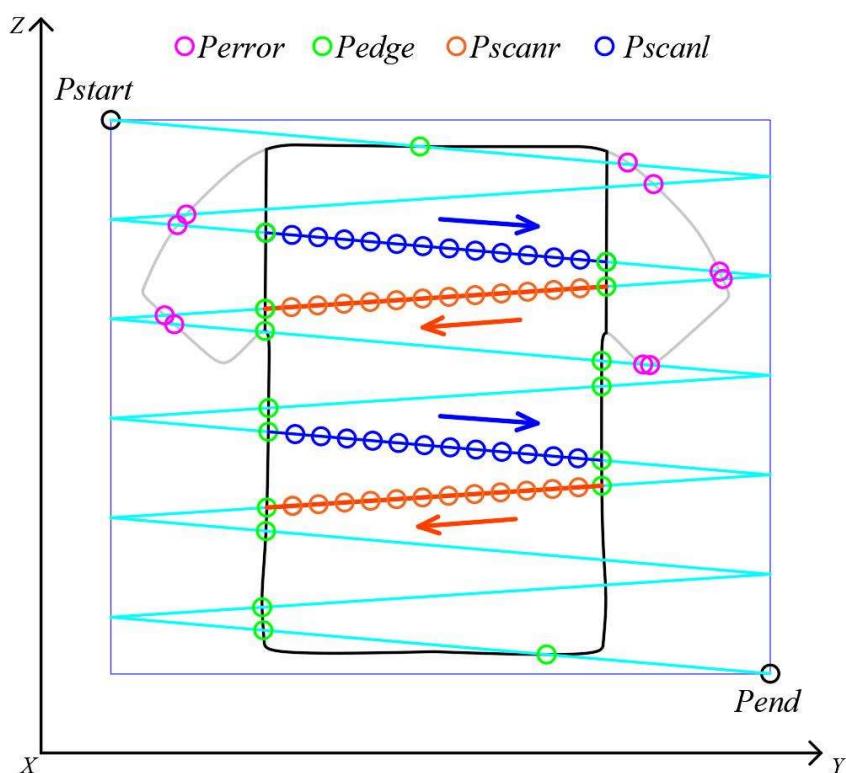


Figure 80. LRF scans the human torso. Positive display. Scan edge points are displayed. 2.

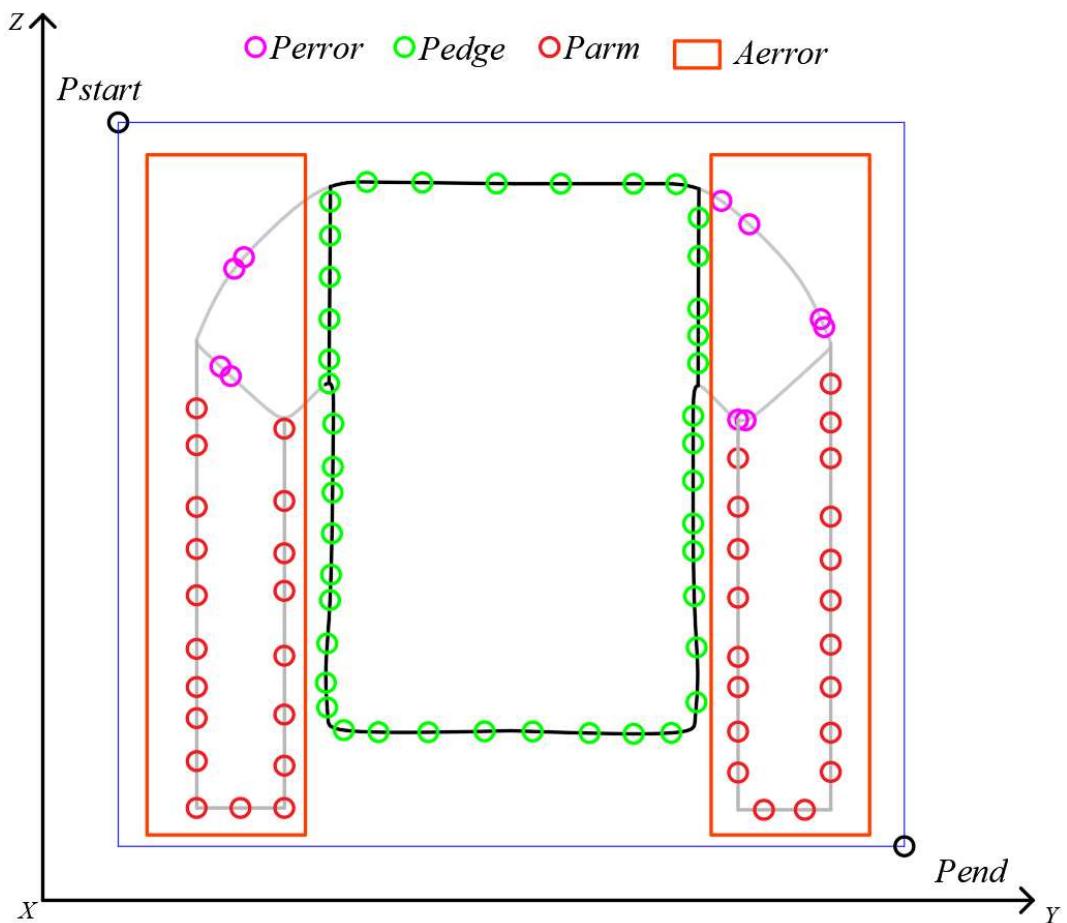


Figure 81. LRF scans the human torso. Positive display. Error interference in the arms.

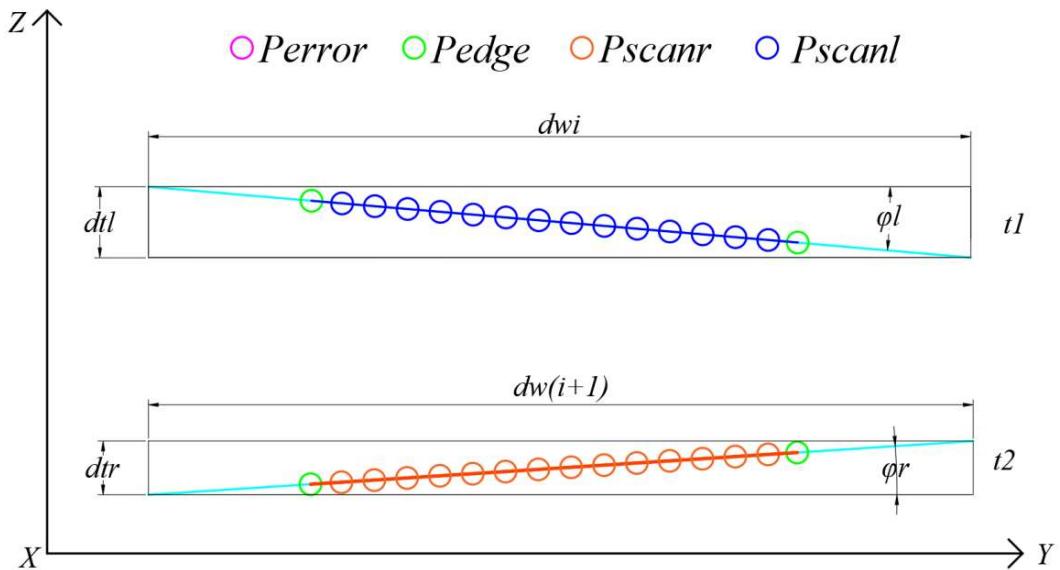


Figure 82. LRF scans the human torso. Frontal partial scan display. Loop the scan mode left and right.

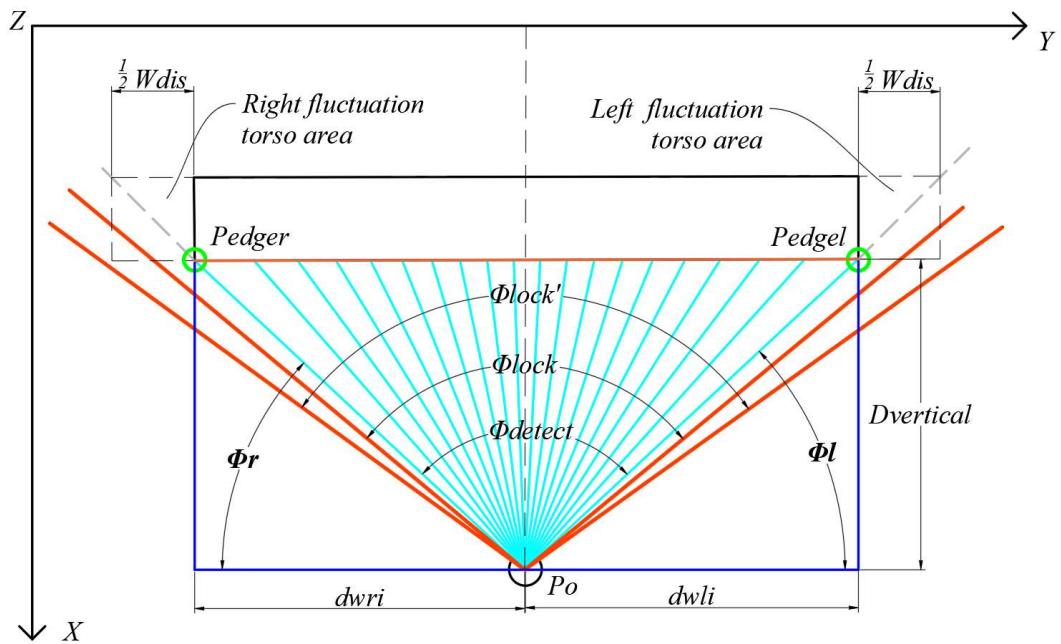


Figure 83. LRF scans the human torso. Frontal partial scan display. Locked scan mode.

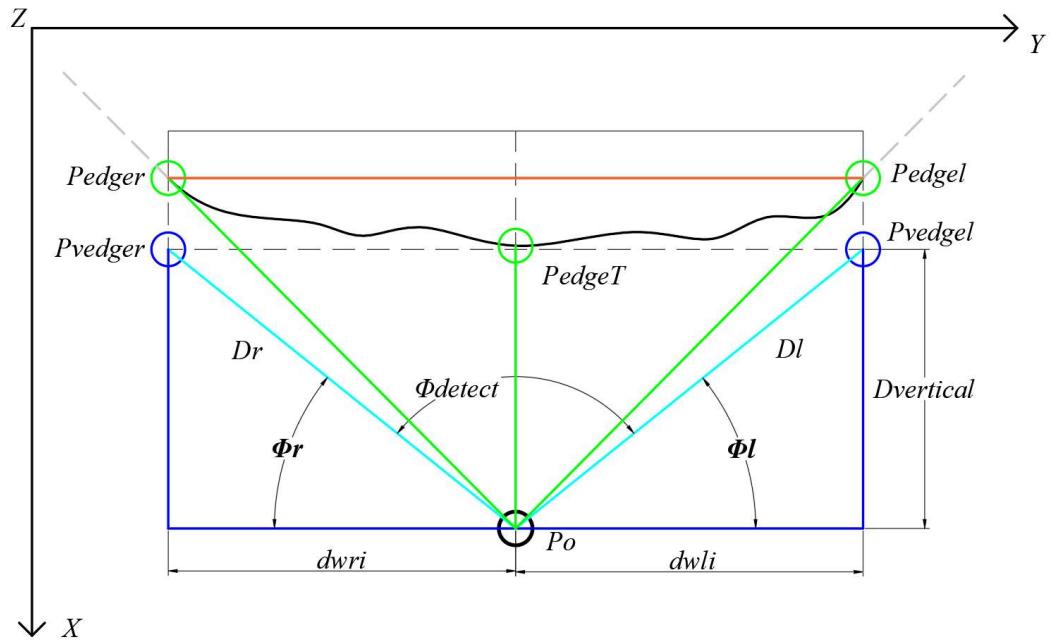


Figure 84. LRF scans the human torso. Positive partial scan display. Algorithmic implementation of a simplified abstract rectangle.

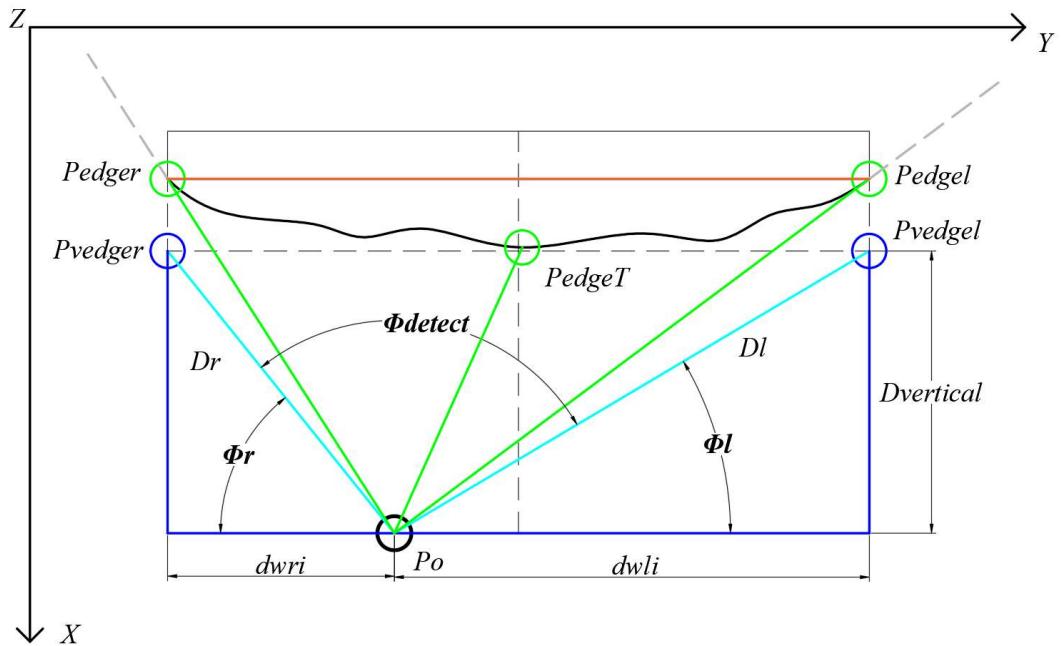


Figure 85. LRF scans the human torso. A partial scan of the front oblique direction. Algorithmic implementation of a simplified abstract rectangle.

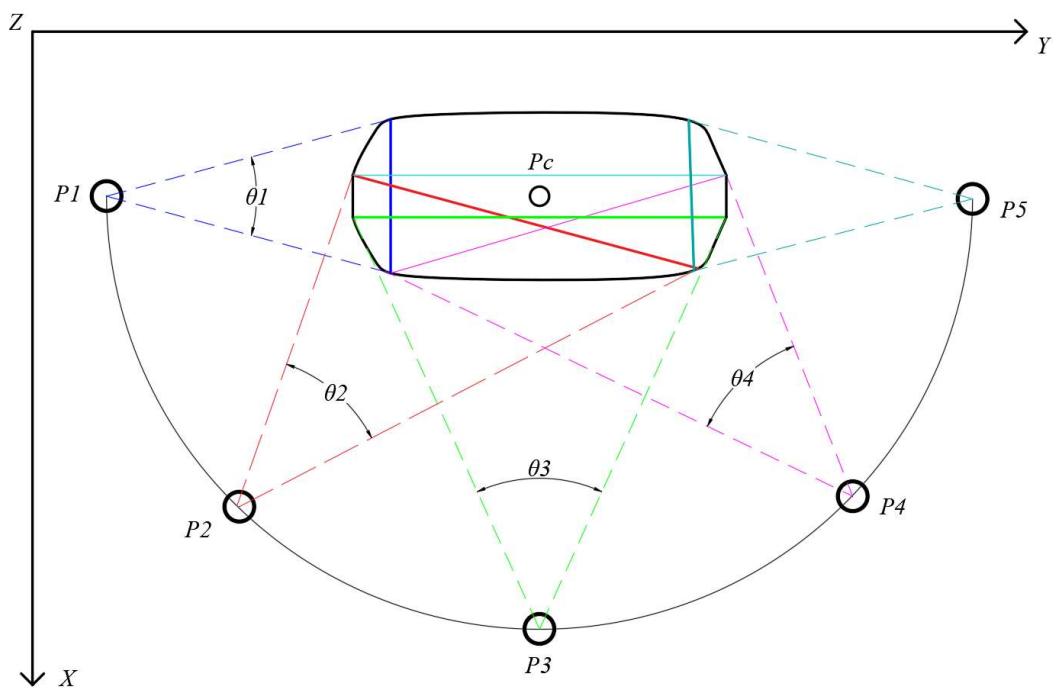


Figure 86. LRF scans the human torso. Display of multi-angle wraparound inspection. I.

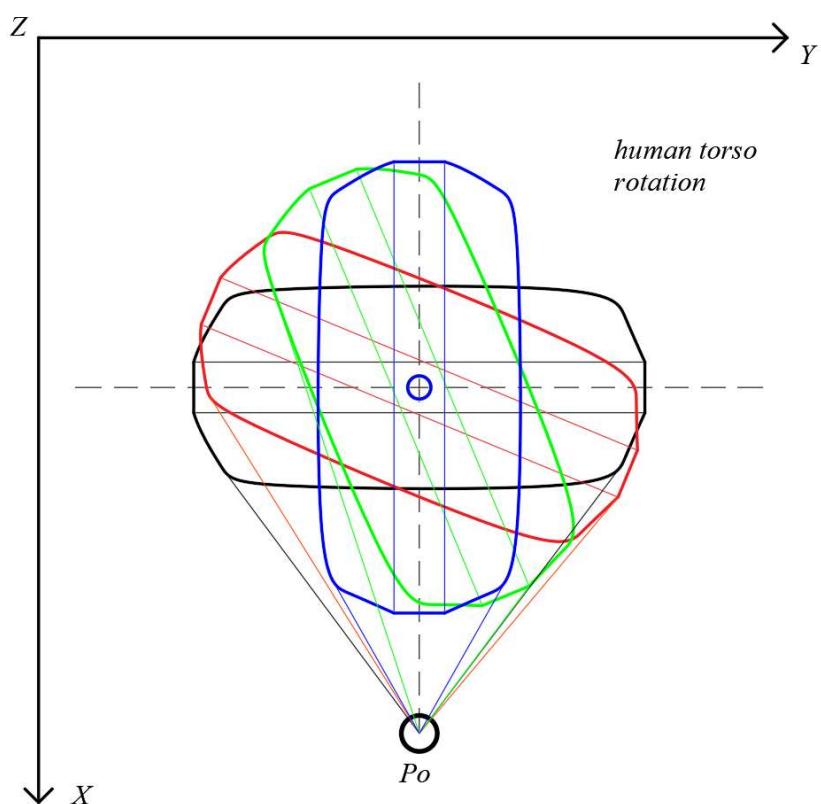


Figure 87. LRF scans the human torso. Display of multi-angle wraparound inspection. 2.

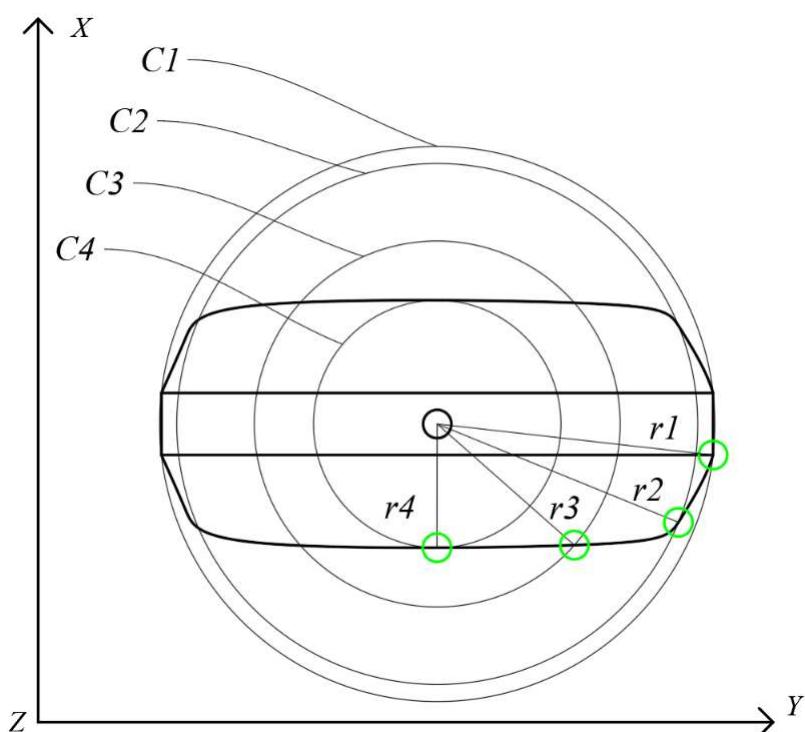


Figure 88. LRF scans the human torso. Display of multi-angle wraparound inspection. The radius of each corresponding analog ring is shown.

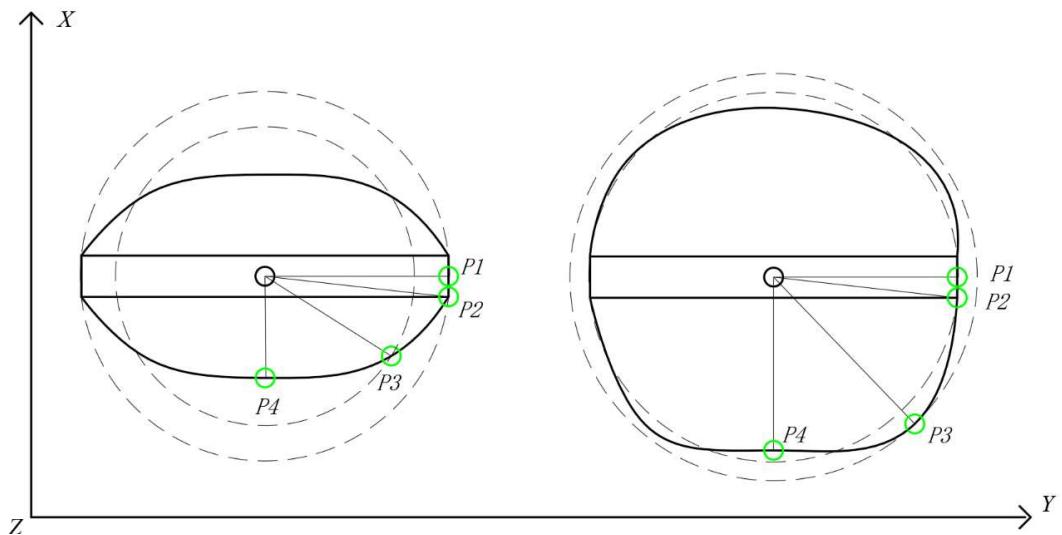


Figure 89. LRF scans the human torso. The error of the predicted data of the trunk scan is very large.

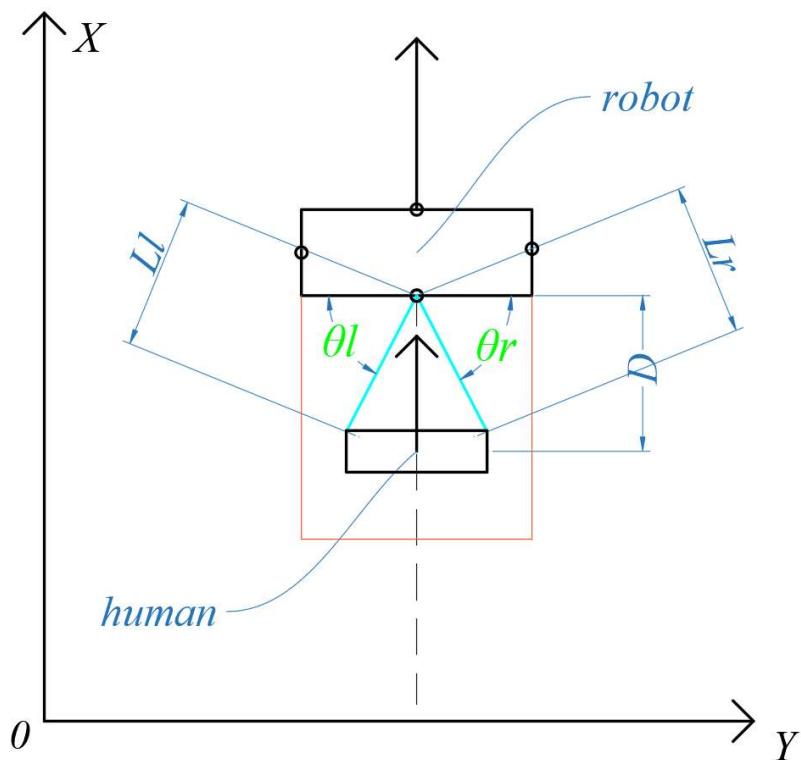


Figure 90. An abstract representation of the LRF scanning of the human torso. The forward direction of the torso detection area of the scanned target person is indicated by an arrow.

DETECTION MODEL AND ALGORITHM ANALYSIS OF HUMAN LEGS

In the previous part of the project research report, the author has already compared the advantages of modeling the detection model using the feet of the scanning target person in more detail. Due to the limitation of the writing time of this project report, it will not be repeated or detailed here. The following will directly enter the description and analysis of the algorithm content.

The human foot detection site is the detection factor of the second largest impact factor set by the author. Therefore, it is very important to model the trunk of the target person. The human foot detection site is the detection factor that the author has set with the third largest impact factor. Due to the physiological structure of human legs, the structure of human legs is relatively fixed, and the range and area in which human legs can move have certain clear limits. According to the joint structure of the human legs, the designer can divide the human legs into four parts for the sake of research. As shown in Figure 91, these four parts are: left thigh, right thigh, left calf, right calf.

As described in the previous algorithm, the author also uses the DRF rectangular scan combined with the object abstraction algorithm in the locked scan mode to scan and detect the legs of the target person. The details of the specific algorithm have been described in detail in the previous section. Due to the time limit for writing this report, the following authors will focus on the key details of this part of the algorithm.

Human legs are the human body parts that are most directly responsive to human walking, in addition to human feet. They are also the body parts that humans play an important role in walking. Therefore, the analysis and modeling of human legs is very useful for predicting the future path of the target person. By combining the DRF rectangle scanning algorithm and the object abstraction algorithm, the author will get the abstract expression of the target's legs and the spatial position and specific direction of the target person's legs in real time, by analyzing the spatial position of the target person's legs. The data and morphology further predict the orientation of the future path of the target person.

In the content of the object abstraction algorithm in the previous section, the author abstracts the four detected parts of the target character into simple cubes. In the latter algorithm, in order to visualize the morphological structure of human legs, the author abstracts the legs and feet of the target person into an ellipse. N.B.: It should be noted here that although in the algorithm diagram of this part, the legs and feet of the target person are represented by an oval. However, in the specific object abstraction algorithm, the author suggests to abstract the legs and feet of the target character into the structure of the space cube, that is, the shape of the interface rectangle appearing on the XY plane. In this case, abstracting the legs and feet of the target person into a rectangle is more conducive to the construction of the object abstraction algorithm. In terms of indicating the directivity, the rectangle is more directional than the ellipse circle. However, since the ellipse is more in line with the abstract image of human legs and feet in reality, the

author uses an oval to display and indicate. All of the abstract cross-sections of the legs and feet used to represent the target person in this report can be represented by rectangles. The following will enter the actual algorithm implementation and explain.

Human legs are important parts of the body when humans walk. At the same time, human legs also play an important role in the study of gait analysis for human walking. If you want to study human walking, the study of human legs will be a very important part. It is assumed that the DRF rectangular scanning algorithm and the object abstraction algorithm discussed in the previous section of this paper can operate normally and obtain reasonable algorithm results. Then the designer can use the DRF rectangle scanning algorithm combined with the object abstraction algorithm to obtain the real-time position distribution and detection data of the detected parts of the target person in the space. The author will use an ellipse to represent the abstract distribution of the real-time space of the target person's legs. The target person's thighs and calves are abstracted in an oval shape.

Since the normal linear walking for humans is the most basic and simple form of human walking, the author begins with the analysis of human walking modeling from the straight walk of human beings. As shown in Fig. 92, a schematic diagram showing the spatial distribution of human legs during normal straight walking is shown. The ellipse in Figure 92 can represent the human thigh or the human calf. Through analysis, it can be found that when humans are walking, the human legs are alternately lifted and landed in these two states. In Fig. 92, the ellipse may represent the thigh of the target person or the calf of the target person. Combined with the actual human walking and gait analysis, the designer can obtain more accurate prediction data about the orientation of the future path of the target person. Since human gait prediction predicts that human legs are related to human feet, the author will unfold a simple analysis of human gait after expounding the scanning model of the human feet in the latter part. As shown in Fig. 92, L_{Ri} refers to the distance of the right leg of the target person from the previous position point at the ith step. L_{Li} refers to the distance from the previous position of the target person's left leg at the ith step. When the author moves the i-th step, the left leg of the target character represents the geometric center point of the ellipse and the right leg represents the elliptical geometric center point, and this segment is called the i-th step of the i-th step. At the same time, the author refers to the midpoint of the center line of the i-th step connection as the ibu center point of the i-th step. As shown in Fig. 92 and Fig. 93, the author constructs the polyline formed by connecting the center lines of the first i-step center points, which are called the target person's trajectory lines, and the author's algorithm aims to pass The known trajectory of the target person to predict the future trajectory of the target person.

The reader may raise a question by analyzing Figure 92 and Figure 93, why the author directly uses the straight line segment to connect the first i-step center point to form a polyline line segment to represent the known trajectory of the target person, instead of passing Is the method of simulating spline segments connecting the first i points with a smooth curve? In response to this problem, the author can have two solutions, one is to simulate the walking curve of the target person by the detected data of the target person's legs, and the other is to pass the legs of the target person. The detected data of

the part simulates the path of the polyline discounted segment of the target person. Relative to the first method, the second method only needs to obtain the position point of the leg when the target person is walking with the corresponding foot off the ground and the leg with the corresponding foot tribe under the ground. The scanning data of the position point at the time can be used. The first approach seems to have a tighter detection interval, making it easier for people to think that the predictions obtained through the first algorithm strategy will be more accurate. However, in fact, the first algorithm strategy does have a larger detection density, but similarly, the second algorithm strategy needs to detect the spatial posture and motion of the target person's legs after leaving the ground, so there will be larger detection errors, and the second algorithm strategy consumes more computational resources of the robot than the first algorithm strategy. Moreover, from another very important point of view, whether the first algorithm strategy or the second algorithm strategy is adopted, the author finally wants to obtain the results that are the trend and direction of the future path of the target person. prediction.

The most important thing to predict an unknown path through a known path is to know the following parameters and factors:

- 1. *The current spatial location point of the target person;***
- 2. *The known path of the target person;***
- 3. *The current speed of the target character;***
- 4. *The walking acceleration of the target person.***

In the advanced development of predictive algorithms, designers need to know the following factors, if possible:

- 1. *Walking habits of target people;***
- 2. *The spatial position of the unrelated object in the environment in which the target character is located and its movement trend.***

Obviously, for calculating and predicting the speed and acceleration of the target person, the linear modeling is simpler than the curve modeling. The designer can pass the trend of the angle between the current step and the last connecting line. Make direct predictions. Predict the future path of the target person by predicting the angle between the next non-occurring step and the current step. This reflects the linear connection characteristics of a human being when walking, from the position of a stepping space to the point of the next stepping space.

By analyzing the gait of human walking, the author finds that in the time interval of human beings, the trajectory of the human leg in the air in the open state has two forms of motion: one is walking straight in humans. At the time, the human leg will remain almost in a plane in space due to the plane of the XY plane. At this time, the author believes that the movement of the human leg in space is along a straight line. The second is that when the human is walking, the human leg will move almost along a curved surface in space (NB: the legs of the target person are treated as a straight line

in the analysis.) The author believes that the movement of human legs in space is along a curve. However, whether the human leg moves away from the ground, whether it moves in a straight line or along a curve in space, the final path of human walking and its orientation are determined by the human feet after landing. That is, the function of the human legs in the process of human walking is to drive the human feet to fall in the designated position. In theory, the designer can predict the future path and orientation of the target person by detecting changes in the posture of the target person during walking, but this is not necessarily accurate because the human legs are off the ground. After that, it is very likely that complex movements will occur in the air, and these complex movements are not indicative of the regularity of human walking. At this time, the designer has to obtain more and more by detecting other detection parts of human beings. The test data of value in order to make a more reasonable prediction judgment. This is one of the key reasons why the author proposes a vector synthesis prediction algorithm. Therefore, in summary, the author suggests to use the straight line segment to connect the first i-step center point to form a multi-line line segment to represent the known trajectory of the target person.

Figure 94 and Figure 95 show the detection of human legs when a person is turning during normal walking. It can be known from the analysis that when a human body makes a turning motion during walking, the movement of the leg near the inner side of the curved curve is small, and the movement of the leg away from the inner side of the curved curve is large. Figure 96 shows a schematic diagram of the author's highly simplified abstraction of the target person's legs. As shown in Figure 96, the author uses two circles to represent the detected legs of the target person.

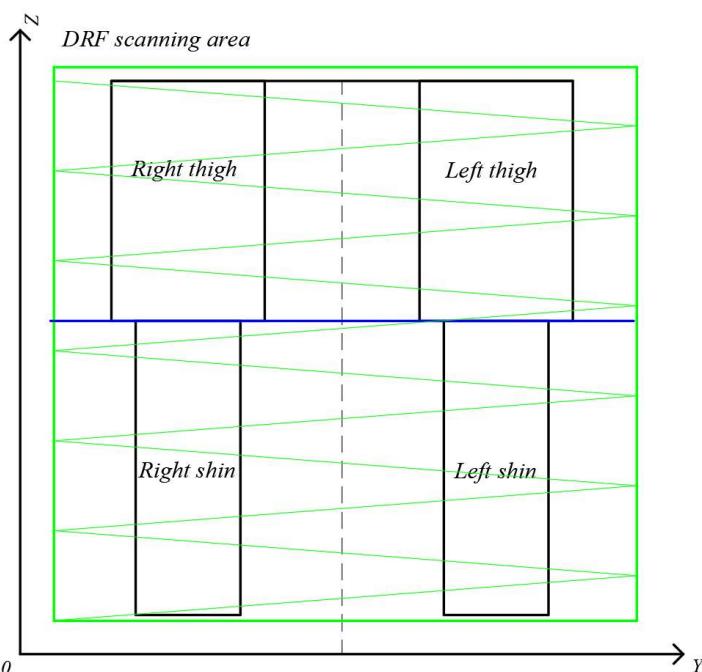


Figure 91. Scanning modeling of the target human legs. The legs of the target task are divided into four detection areas.

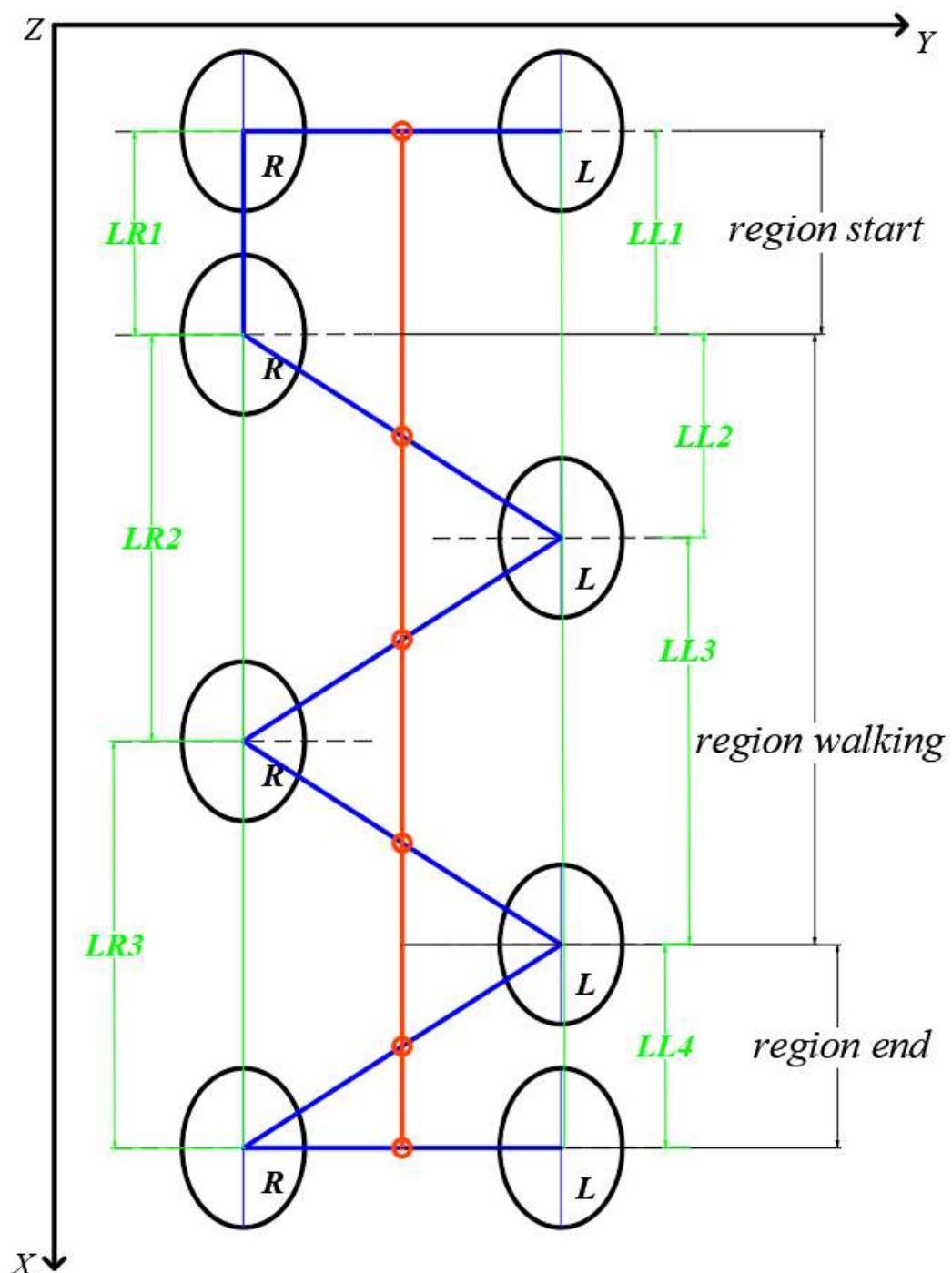


Figure 92. Scanning modeling of the target human legs. Straight walking show. from start to finish. Walk from the right leg.

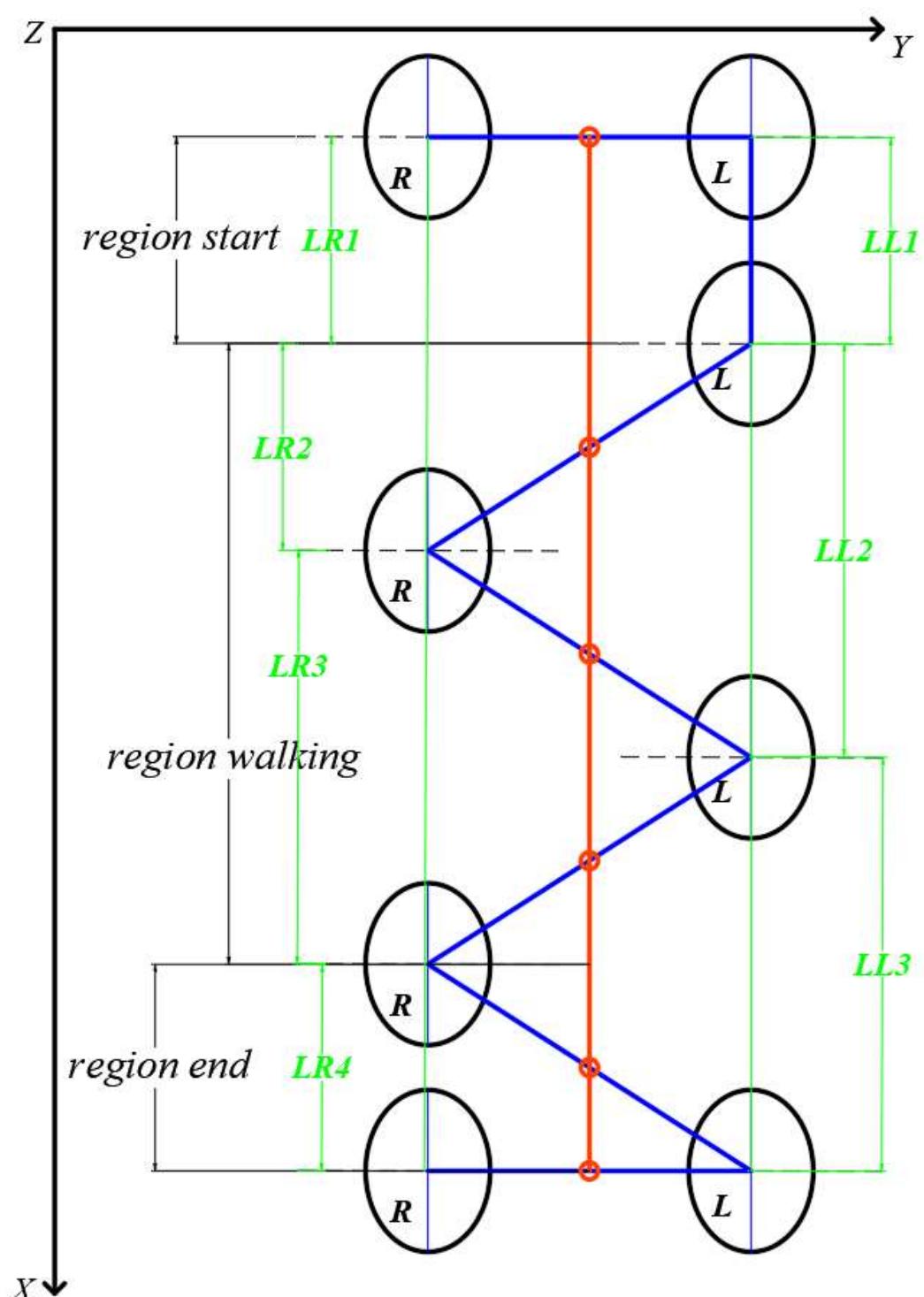


Figure 93. Scanning modeling of the target human legs. Straight walking show. from start to finish. Walk from the left leg.

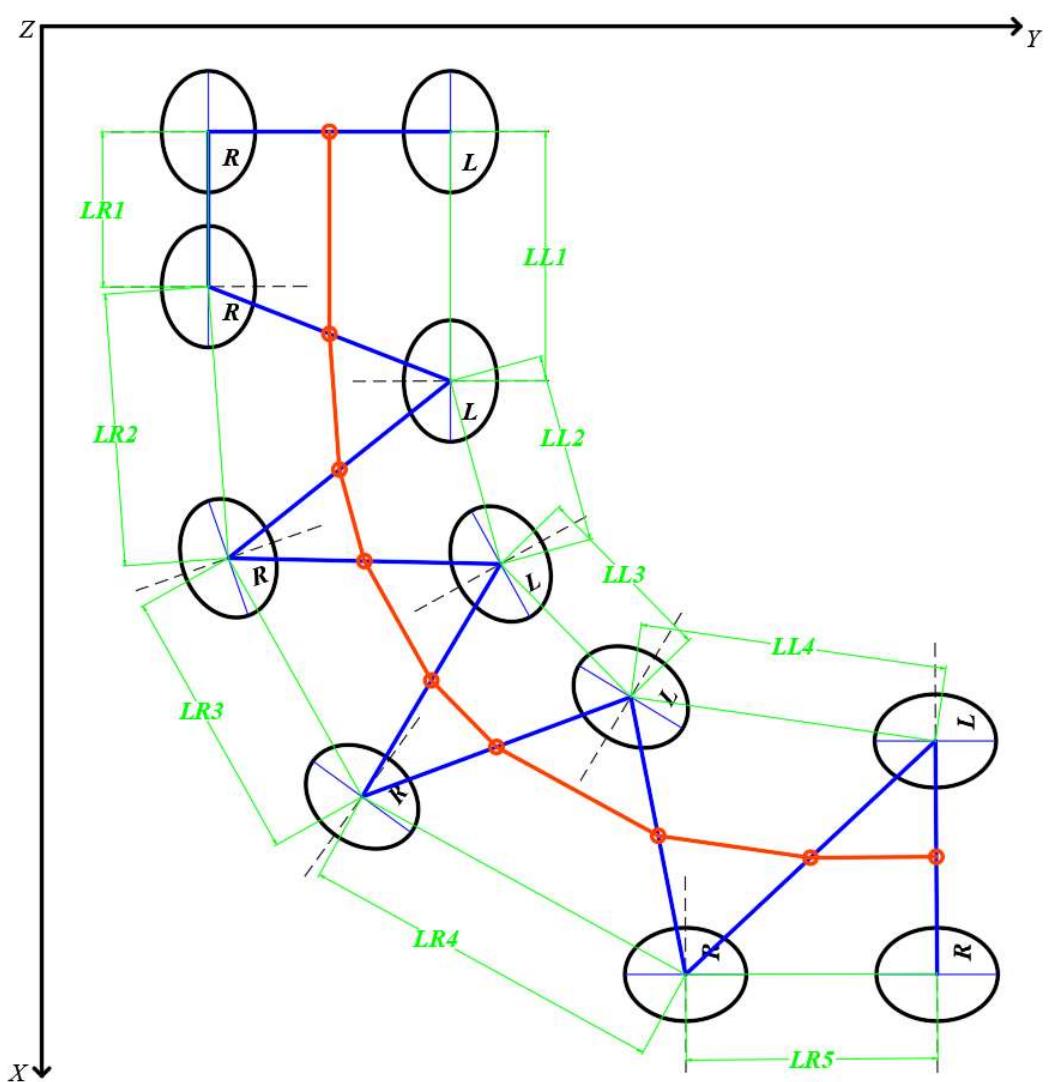


Figure 94. Scanning modeling of the target human legs. The show of turning and walking from start to finish. Walk from the right leg.

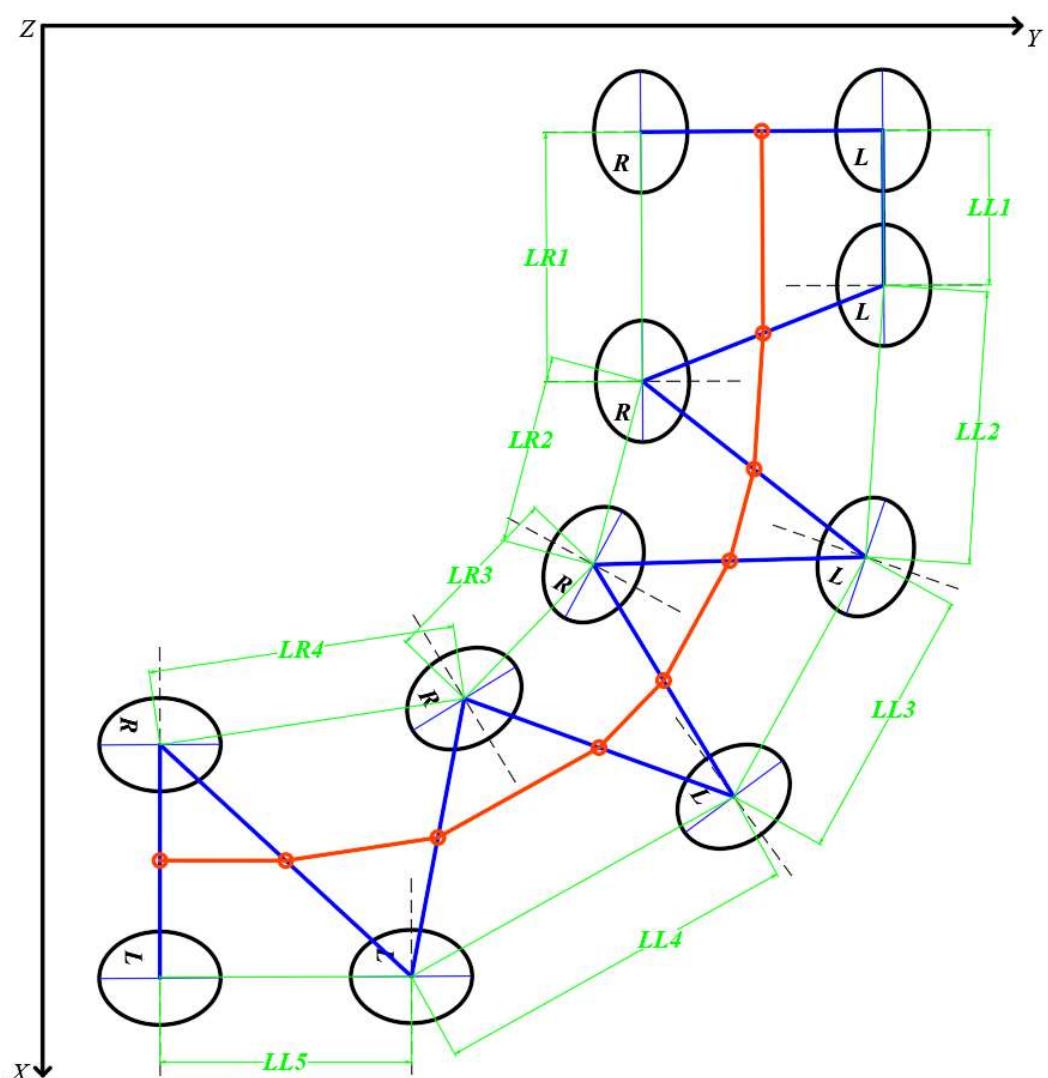


Figure 95. Scanning modeling of the target human legs. The show of turning and walking. from start to finish. Walk from the left leg.

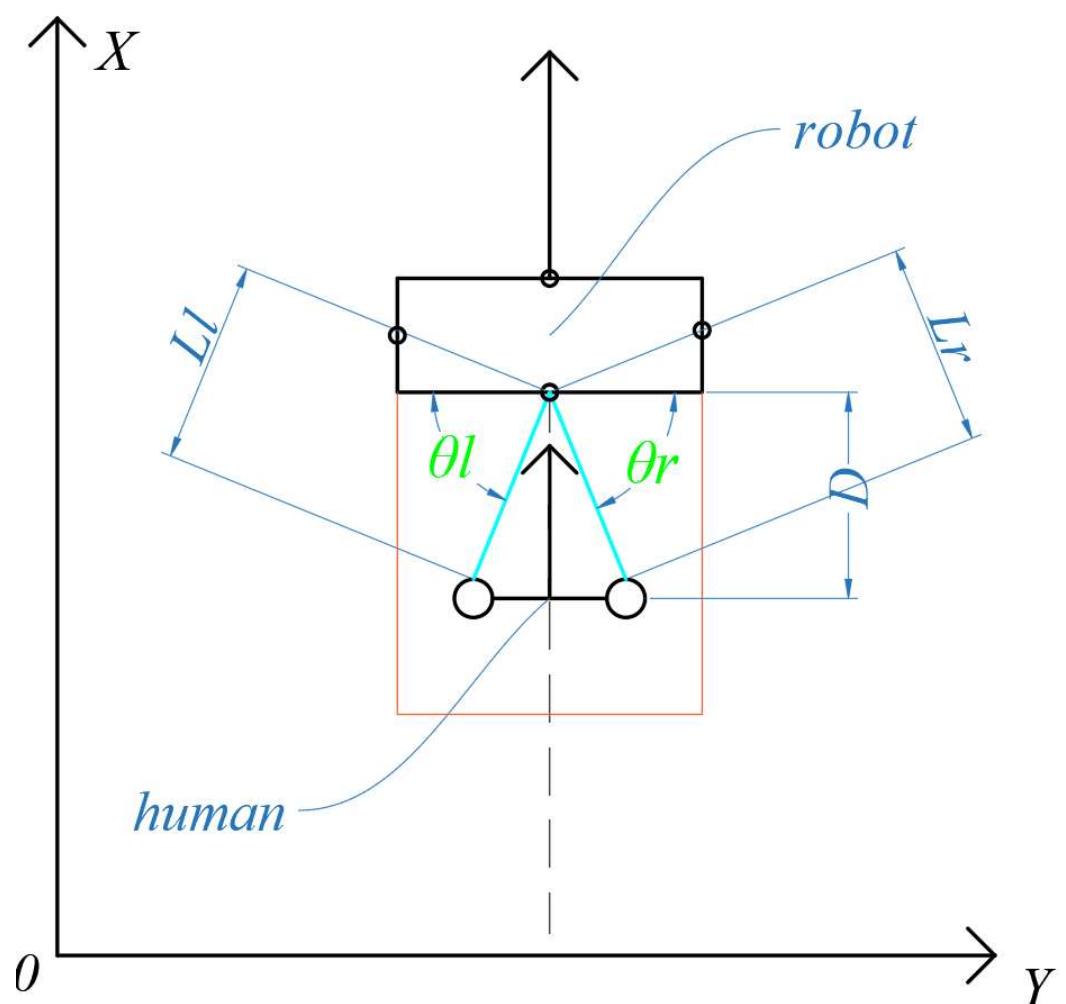


Figure 96. An abstract representation of the LRF scanning of human legs. The forward direction of the leg detection area of the scanned target person is indicated by an arrow.

DETECTION MODEL AND ALGORITHM ANALYSIS OF HUMAN FEET

In the previous part of the project research report, the author has already compared the advantages of modeling the detection model using the feet of the scanning target person in more detail. Due to the limitation of the writing time of this project report, it is not repeated or detailed here. The following directly enters the description and analysis of the algorithm content.

The human foot detection area is the detection factor of the first large influence factor set by the author. Therefore, it is very important to model the trunk of the target person. As described in the previous algorithm, the author also uses the DRF rectangle scan combined with the object abstraction algorithm in the locked scan mode to scan and detect the two feet of the target person. The details of the specific algorithm have been described in detail in the previous section. Due to the time limit for writing this report, the following authors will focus on the key details of this part of the algorithm.

As shown in Fig. 97, the division of the LRF scanning area when the target person stands vertically on the support surface is shown. In order to detect the target's feet better, the author sets three detection areas and one detection angle for the target's feet. The three detection areas and one detection angle are defined as follows:

1. **Area1:** The calf detection area is used to separate and identify the foot and leg of the target person;
2. **Area2:** The foot detection area is used to detect the foot area of the target person;
3. **Area3:** The front foot detection area is used to detect the front area of the target person's foot and to distinguish the target person from the support surface.

When the author designs and detects the recognition algorithm, the author separates and identifies each region through the above-mentioned three detection areas, combined with their appearance characteristics and detection characteristics, and then comprehensively analyzes the detection data of each detection area to obtain the target. The real-time posture and state of the detected foot parts of the character. The following are the detection characteristics of each detection area:

1. **Area1:** Perform object abstraction algorithm analysis on the detection data of this area, and compare it with Area2. If it is found that the vertices of the double-line segment are shown, the area is Area1, and the corresponding fold of the line segment is The boundary between Area1 and Area2.
2. **Area2:** Perform object abstraction algorithm analysis on the detection data of this area, and compare it with Area1. If it is found that the vertices area of the double-line segment of the double-line segment appears, the area is Area2, and

the corresponding break point of the line segment is The boundary between Area1 and Area2.

3. **Area3:** Perform object abstraction algorithm analysis on the detection data of this area. If the Z coordinate of the detection data of the LRF in this area appears and the Z coordinate corresponding to the support surface appears to be similar, it is similar in most detection areas, but In the detection of the spatial position at a certain location, a sudden change of the Z coordinate appears, indicating that the area is likely to be the Area3 area, and the spatial position where the Z coordinate is abrupt is likely to be the boundary position between the Area 3 area and the Area 2 area. In order to improve the accuracy of the detection algorithm and reduce the judgment error, the designer can also scan the following features in combination with the DRF rectangular scanning algorithm and the object abstraction algorithm described above:

1. Does the spatial position corresponding to the Z coordinate mutation of Area3 have two locations;

Reason: The target person's feet are generally in pairs, so if the area is really Area3, then the corresponding Z coordinate change spatial position should be two places.

2. Whether the height of the mutation corresponding to the sudden change of the Z coordinate occurred in the Area 3 area meets the set actual judgment criteria;

Reason: Generally, when using the follow-up service provided by the robot, the target person's feet are not completely exposed in the space, and the target person's feet are likely to be wrapped by foreign objects, and the degree of being wrapped may be very high. Big or small. The foreign objects in the legs of the package target are generally shoes. The shoes that are worn by humans in general are generally of a certain height range, and the designer can determine whether the detected object is the foot of the target person by setting a reasonable height range.

3. Whether the distance corresponding to the spatial position of the two Z coordinate mutations detected in the Area3 area meets the set actual judgment standard;

Reason: This is because the distance between the feet of the human body has a certain distance range. If the distance between the detected spatial positions where the two Z coordinates are abrupt is beyond the range of possible distances in the common sense, then the designer It can be considered that the detected object is not the foot of the target person.

4. Whether the edge width of the corresponding detected object and the distance between the two widths corresponding to the spatial position where the two Z coordinates are abruptly detected in the Area 3 region meet the set actual judgment standard;

Reason: This is because the edge width of the human foot part and the distance between the two widths have a certain distance range, if the detected two detected Z coordinates have a sudden change in the spatial position of the corresponding detected object The width of the edge of the feet and the distance between the two widths are beyond the range of distances that may occur in the common sense, so the designer can think that the detected object is not the foot of the target person.

There is also an important detection feature in the Area 3 area. As shown in Figure 99, if the detected foot part of the target person leaves the ground, another LRF detection distance will appear in the detection area between the Area 3 area and the Area 2 area. Mutation, the author called this phenomenon as a foot-to-ground mutation. A sudden change in the foot from the ground corresponds to an important parameter, Keyline. Keyline refers to the sudden change of the LRF detection distance between Area 3 and Area 2, and the corresponding KL is the vertical height of the mutation. By analyzing the range relationship between Keyline and KL, the designer can determine whether the target part of the target person has left the support surface at a certain moment, and if the target person's feet part leaves the ground, it corresponds to the position in the space.

In addition to the detection characteristics of the three detection areas of the above target characters, this algorithm also has a very important detection parameter, which is *Aankle*. *Aankle* is called the ankle angle and represents the angle between the abstract area of the target person's calf and the abstract area of the foot. It is also an important parameter for the author to study the walking characteristics of the target person. Due to the physiological structure of the human ankle joint, the angle also has a certain range of areas, which can be used to design a judgment algorithm for performing a foot region of whether the currently detected object is a target task.

In addition to the above detection features, Area1, Area2, Area3 and Aankle, there are many corresponding detection features and judgment processing algorithms. More specific related algorithm content will be included in the second version of this report. Expand in detail. The following authors will describe the specific algorithmic implementation of predicting the future path of a target person through the detected two-footed portion of the target person.

As shown in FIG. 100, the author abstracts the feet of the target character into an ellipse, the ellipse corresponding to the left foot of the target character is called the left foot abstract ellipse, and the ellipse corresponding to the right foot of the target character is called the right foot abstract ellipse. The direction of the corresponding foot is indicated by an orange arrow, and the line along the direction of advancement of the abstract ellipticity is referred to as an elliptical direction line. Similarly, using the

construction idea of the prediction algorithm for the two-legged part of the target person similar to the previous part, two geometric centers of the ellipse representing the leg of the target person are connected, which is called the foot center connection. The author then defines three key angle parameters:

- 1. φl** : the angle between the elliptical direction line of the abstract ellipse of the target person's left foot and the center line of the foot;
- 2. φr** : the angle between the elliptical direction line of the abstract ellipse of the target person's right foot and the center line of the foot;
- 3. φf** : The angle between the elliptical direction line of the abstract ellipse of the target person's left foot and the extension line of the elliptical direction line of the abstract ellipse of the right foot of the target person.

In order to design a simple and efficient algorithm, the author abstracts the important characteristics of the human walking foot into three angles. It should be noted that these three key angles are detected when the target person's feet are landing. Because through the analysis, the author believes that the foot posture of the target person's landing is more certain and predictive stability with respect to the foot posture of the target person leaving the ground. Due to the continuity of gait changes in the process of human walking, the designer only needs to study the accuracy of the experimental settings after landing. If you want to add a detection of the foot area of the target person after leaving the ground, then adding more uncertainty will increase the uncertainty.

Below, the author will detail how the three key angles reflect the characteristics of the target when walking, and how the author predicts the future path and orientation of the target through these three key angle changes.

As shown in FIG. 100, if the target person is standing, in general, the abstract direction line of the target person's foot is perpendicular to the frontal orientation of the target person. If it is a standard standing posture, then φl and φr should be the same. A more standard standing posture is:

$$\varphi l = \varphi r = \frac{\pi}{2}$$

In order to better explain the design and implementation principle of this part of the algorithm, the author defines a foot triangle, the three corners of the foot triangle are: φl , φr , φf . An existing foot triangle should have the following three key angle relationships:

$$\varphi l + \varphi r + \varphi f = \pi$$

However, the designer needs to be aware that the angle triangle does not necessarily exist during the walking of the target person. This is generally caused by φ being greater than or equal to π . As shown in Figure 100, the case where the angular triangle does not exist is shown. The left image in Figure 100 shows the abstraction of the foot when a normal human walks straight. The right image in Fig. 100 can also be regarded as the abstraction of the foot when a normal human walks straight, which is caused by the incorrect walking habit of the target person or the imperfection of the physiological structure of the feet. Generally, this situation has been found in the DRF modeling scan of the target character by the robot. In the face of this wrong habit of the target person, the author has two strategies for processing. The first processing strategy is to use a reasonable correction algorithm. The abstraction of the error is converted into an abstraction with the correct pose, that is, the abstract pose of the right graph is converted into the abstract pose of the left graph, although in reality the target person's feet are still in the wrong posture. The purpose of this is to better detect and predict in the next turn prediction algorithm. The second processing strategy is to maintain this erroneous abstract representation of the target person, not correcting the abstract representation, and treating the abstract representation as a reasonable abstract representation of the target person. The author suggests using the second processing strategy because it is simpler and more efficient, and more in line with the author's actual algorithm requirements. The author will proceed from the foot triangle to further clarify and analyze the changes in the posture of the target person's feet during walking and their corresponding modeling.

Since the target person's walking process is dynamic, the designer cannot judge or predict the future path of the target person through a simple posture of the feet at rest. The best way to handle any dynamic prediction is to predict a limited amount of future data based on a large amount of reliable known data. The reliable known data mentioned here has two main points, one is reliable and the other is known, that is, if the designer wants to make predictions, it should be known from the related and reliable. Sexual data starts to make predictions. This implies that the designer cannot simply pursue the size of the known data set, but rather select a reasonable, reliable data set that does not affect the data volume of the computing resources of the robot. Moreover, designers can only predict limited future data, and cannot expect to predict an infinite or a location data set with a larger amount of data than a known data set through a limited known data set. Even if such a data set is predicted, its reliability is not high. In summary, the author has developed the following algorithmic execution strategy for predicting the future path of a target person through the known foot-to-body detection data of the target person:

1. Set the prediction principle according to a certain number of steps as the processing unit, assuming every m ($m > 1$) step as a prediction unit;
2. Starting from the current number of steps, backward m steps backward, and select a total of m steps of detection data including the current number of steps as the prediction data set. If the current known number of steps is less than m steps, the detection data of all the previous steps is selected as the current prediction data set.

3. Analyze the relationship between φ_l , φ_r and φ_f in the detection data of the current prediction data set. Get the trend of the target's next two-legged posture. The future path of the target person and its changing trend are predicted by the trend of the target person's feet. The following are specific analysis and predictions:

1. As shown in Fig. 101, if the changes of φ_l and φ_r are cyclically alternated during the walking of the target person, and φ_f is always greater than or equal to π , then the target person is likely to be straight. Walking exercise. When the author walks straight with the target character, when the left foot is taken, the corresponding φ_l is called normal φ_l , and when the left foot is taken, the corresponding φ_r is called normal φ_r .
2. As shown in the three figures in the upper part of Fig. 102, if the target character takes the left foot and the left foot falls when the target person is walking, the value of φ_l is smaller than the normal φ_l . When the target character takes the right foot and the right foot falls, the value of φ_r is almost equal to the normal φ_r . The author said that the target character is now making a left turn. Moreover, the faster φ_l is reduced, the greater the degree of left turn.
3. As shown in the three figures in the lower part of Figure 102, if the target character takes the right foot and the right foot falls when the target person is walking, the value of φ_r is smaller than the normal φ_r . When the target character takes the left foot and the left foot falls, the value of φ_l is almost equal to the normal φ_l . The author said that the target character is now making a right turn. Moreover, the faster φ_r is reduced, the greater the degree of left turn.

There are many algorithms for thinking about other aspects from the processing of the detection data of the human foot parts and the prediction of the orientation of the future path of the target person. The author just provides one of the simpler and more efficient prediction algorithms. This algorithm needs to be used in conjunction with the vector synthesis prediction algorithm. More exciting, more detailed prediction algorithms Stay tuned for the second version of this project report.

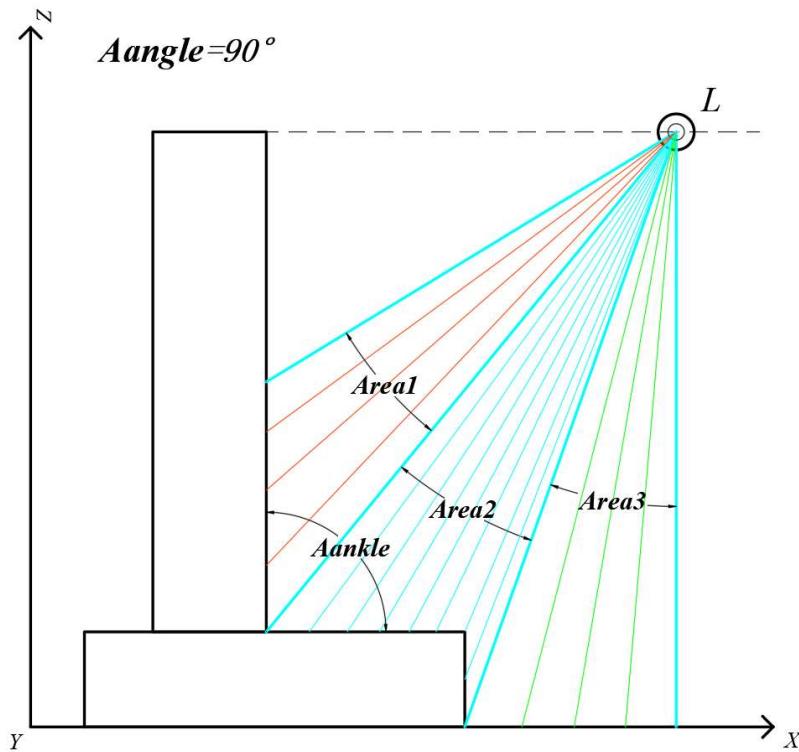


Figure 97. LRF scans the scan of the target person's foot area. The division of the scan area. Side view. 1. The foot of the target person touches the support surface, and the entire foot is perpendicular to the support surface.

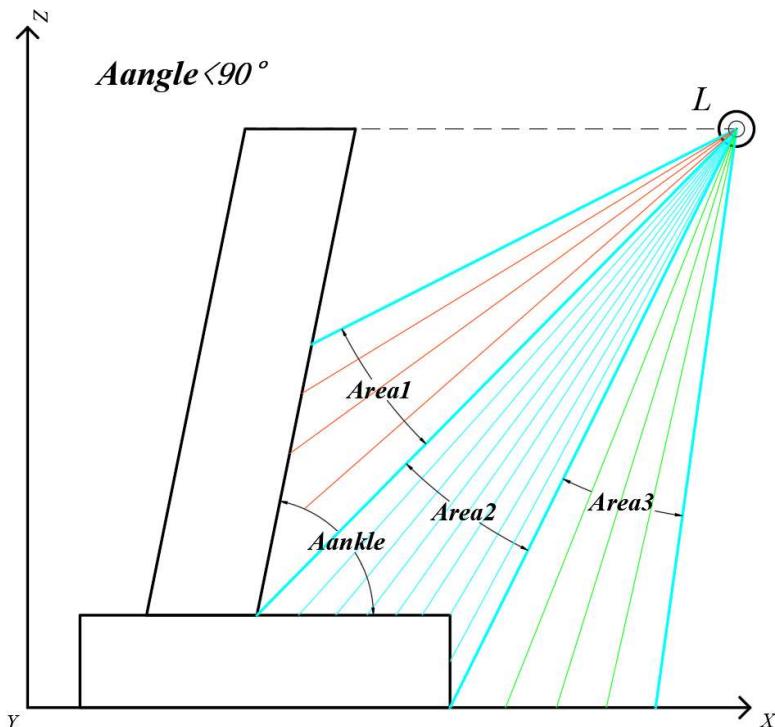


Figure 98. LRF scans the scan of the target person's foot area. The division of the scan area. Side view. 2. The target person's foot touches the support surface, and the overall foot is rearward.

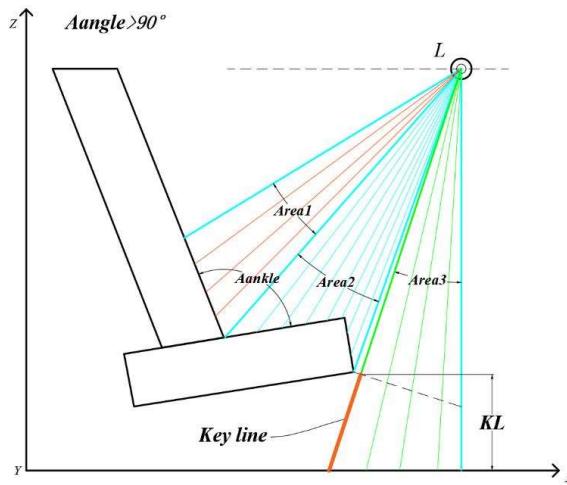


Figure 99. LRF scans the footprint of the target person's foot area. The division of the scan area. Side view. The target person's foot area is off the ground.

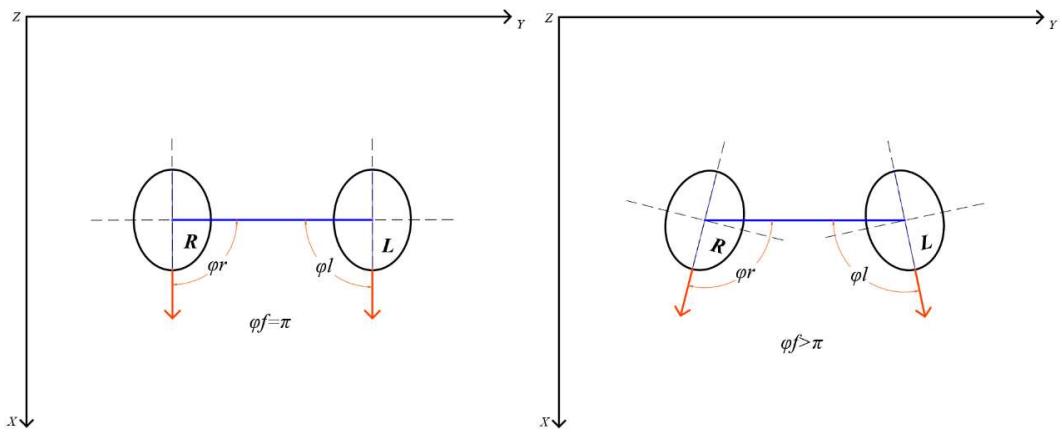


Figure 100. Simulated abstraction of the foot tribe's posture when the target character is standing.

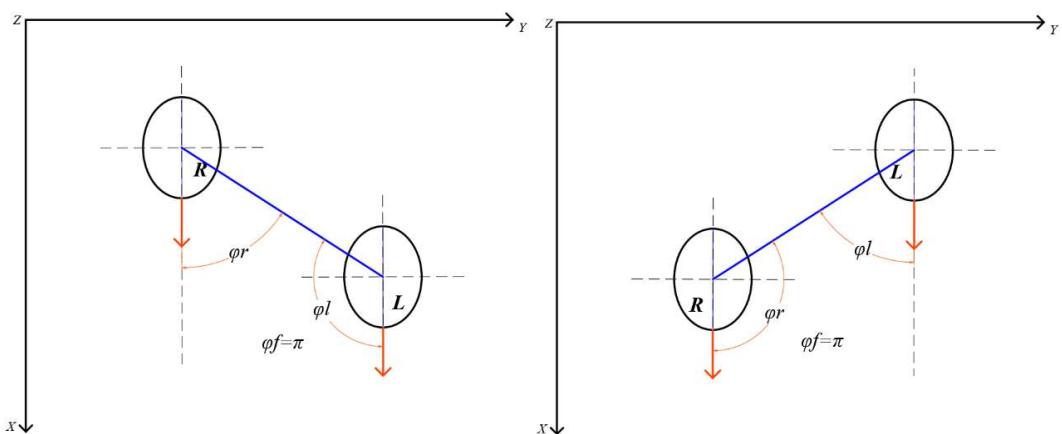


Figure 101. Simulated abstraction of the foot tribe's posture when the target character is walking.

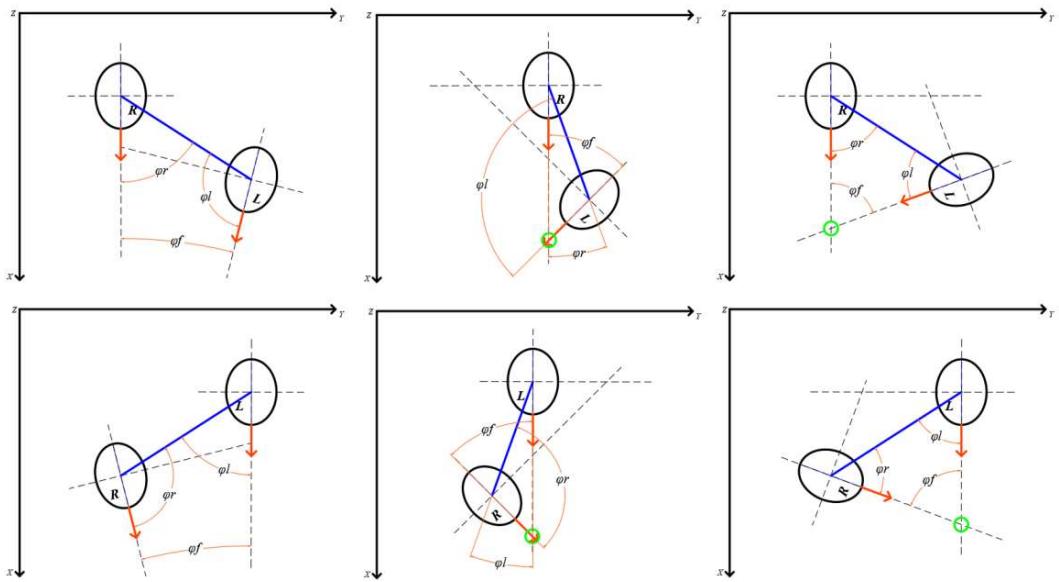


Figure 102. Simulated abstraction of the foot tribe's posture when the target character walks and bends.

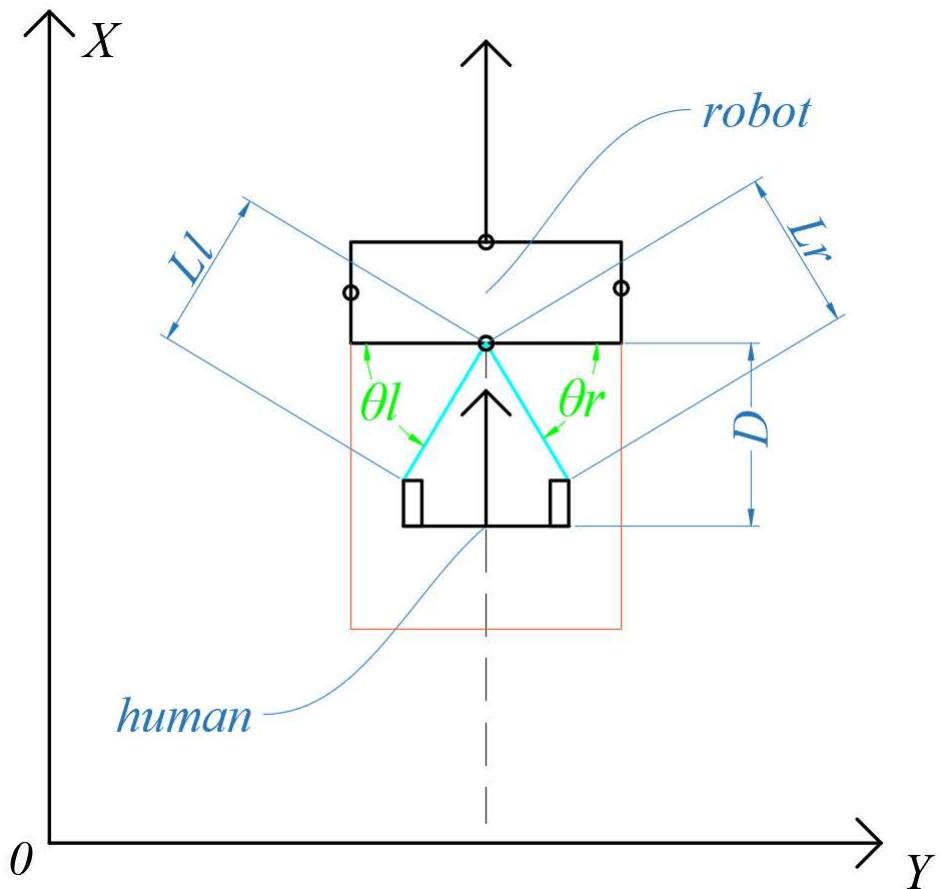


Figure 103. An abstract representation of the LRF scanning of human feet. The forward direction of the two-legged detection area of the scanned target person is indicated by an arrow.

CORRESPONDING TO THE APPLICATION OF GAIT ANALYSIS TO THE DISCUSSION AND ANALYSIS OF THIS PROJECT

In this section, the author does not discuss and analyze whether the gait analysis should be applied to the algorithms involved in the project report. Careful and serious readers should have found that the authors in the above sections In the algorithm description and analysis, gait analysis has been used unconsciously. And in the description of many algorithms, this article also directly describes this as a key explanation. Therefore, the importance of gait analysis for the algorithm design and construction of this project is self-evident.

Since gait analysis is a discipline with a wide range of design knowledge, the author here only does some elaboration and analysis of the specific implementation of the algorithm combined with the project. In theory, through the comprehensive application of the DRF rectangular scanning algorithm and the object abstraction algorithm and vector synthesis algorithm in the previous part of this report, the designer can capture any gait of the target person. In other words, the important algorithm designed by the author before is the important guarantee premise that the author can perform gait analysis on the target person. The following discussion has been reasonably and correctly applied and implemented based on the processing algorithms designed by the author.

In gait analysis, human gait is divided into normal gait and abnormal gait. The normal gait has its corresponding gait characteristics, and the abnormal gait also has its corresponding gait characteristics. By analyzing and comparing these features, the designer can judge whether the target person has a normal gait, thereby inferring the physical condition of the target person, and combining it with the corresponding prompting algorithm and control processing algorithm to improve the service intelligence level of the robot. The relevant domain knowledge of gait analysis is very useful for designers to improve the quality and efficiency of the designed algorithms. For gait analysis, the designer can first study the characteristics of normal gait, study the characteristics of abnormal gait, and derive the normal walking law from normal gait, and apply it to the processing and prediction algorithm of human normal walking. The abnormal walking trait is derived from the abnormal gait, applied to the abnormal phenomena and analysis processing in the processing and prediction algorithms of human normal walking. When performing gait analysis, analysis can be performed from the perspective of quantitative analysis or qualitative analysis.

There is a lot of important knowledge in gait analysis that designers can refer to when designing algorithms. Due to the limited writing time of this project book, the

author does not elaborate on this version of the report. For more details and analysis, please look forward to the second version of the report.

For more exciting details content, please refer to the other articles of this essay:

1. ROBOT DETECTION SYSTEM DESIGN ABOUT FRONT-FOLLOWING TECHNOLOGY PART 1
2. ROBOT DETECTION SYSTEM DESIGN ABOUT FRONT-FOLLOWING TECHNOLOGY PART 2
3. ROBOT DETECTION SYSTEM DESIGN ABOUT FRONT-FOLLOWING TECHNOLOGY PART 3
4. ROBOT DETECTION SYSTEM DESIGN ABOUT FRONT-FOLLOWING TECHNOLOGY PART 5

References:

- Ho, D. M. , Hu, J. S. , & Wang, J. J. . (2012). Behavior control of the mobile robot for accompanying in front of a human.
- Jung, E. J. , Yi, B. J. , & Yuta, S. I. . (2012). Control algorithms for a mobile robot tracking a human in front. IEEE/RSJ International Conference on Intelligent Robots & Systems. IEEE.
- Hu, J. S. , Wang, J. J. , & Ho, D. M. . (2014). Design of sensing system and anticipative behavior for human following of mobile robots. *IEEE Transactions on Industrial Electronics*, 61(4), 1916-1927.
- Moustris, G. P. , & Tzafestas, C. S. . (2017). Intention-based front-following control for an intelligent robotic rollator in indoor environments. *Computational Intelligence*. IEEE.
- Nikdel, P. , Shrestha, R. , & Vaughan, R. . (2018). [IEEE 2018 IEEE International Conference on Robotics and Automation (ICRA) - Brisbane, Australia (2018.5.21-2018.5.25)] 2018 IEEE International Conference on Robotics and Automation (ICRA) - The Hands-Free Push-Cart: Autonomous Following in Front by Predicting User Trajectory Around Obstacles. (pp.1-7)

Project Research Report

ROBOT DETECTION SYSTEM DESIGN ABOUT FRONT-FOLLOWING TECHNOLOGY

PART 5

**Jinwei Lin
(JY Lin, Shenzhen, China)
February 02, 2019**

ID ORCID: <https://orcid.org/0000-0003-0558-6699>

<http://www.ydook.com/>

DRIVE FOLLOWING ALGORITHMS FOR ROBOTS: DOVE OF PEACE ALGORITHMS

The algorithm in this part is the last important algorithm of this version of the project report, and it is also the direct execution motion algorithm that the robot finally performs the following task. Because the execution path of this algorithm is similar to the outer edge shape of the dove of peace, in order to explain the algorithm conveniently, it is also called the dove algorithm. In this part, the author will first analyze the problem background of the algorithm, and then elaborate on the design principle of the algorithm and the specific implementation process.

Thinking about the background of the dove of peace doctrine

This section will tell you what the Dove of Peace algorithm is for, and to illustrate and analyze the algorithmic features of the Dove of Peace.

First, from the various algorithms described in the previous section, the designer can perform corresponding scan detection on the four detected parts of the target person, and acquire data information of the known path of the target person. Moreover, the designer can also make certain predictions about the future path of the target person. The implementation premise of the algorithm is a follow-up algorithm in the case of the normal operation of the above various scan recognition and prediction algorithms. That is, the algorithm is the most direct algorithm for the robot to finally perform the forward-going target person. Of course, the algorithm needs to finally drive the hardware implementation through the robot's mechanical control and drive algorithm. Since the control algorithm of the mechanical hardware part is not the focus of this version of the project report, and the writing time of this report is limited, this report will omit it. For details, please look forward to the second edition of the project research. Writing and publishing of the report.

The dove of peace algorithm is essentially the author's proposed algorithm for the robot to follow the direct execution path of the target person. That is, the algorithm is used to calculate the following path of the robot. The calculated path here is the path of the robot, and belongs to the future path. That is, the dove of peace is used to tell the robot how to forward to the target person, so the importance of this algorithm is self-evident.

The following authors analyze the following background of the problem of forward-following questions. As shown in FIG. 104, in the process of the robot performing the forward-following task, it is assumed that the target person is out of the restricted area of the robot rectangle, and the movement is performed in the direction of the velocity v_h , the acceleration is a_h , and the direction is the P_r vector. The direction is v_r , the acceleration is a_r , and the direction is the direction of the P_v vector. Through analysis, we can know that if the robot does not take any measures after learning that the direction of the target person has changed, and continues to maintain the original motion in the original direction and speed and acceleration, then the robot and the target human will happen very quickly. Separation, that is, the followers of the robot will soon fail. Obviously the designer does not want this to happen. So the designer must design the algorithm so that the target person

can return to the restricted area of the robot rectangle at some point in the next moment. That is, the designer must design a new future path for the robot, so that the robot can make the target character return to the restricted area of the robot rectangle again at some point in the future through this new future path. In order to facilitate the explanation of the principle of algorithm implementation, the author shifts the movement of the target person from the restricted area of the corresponding robot into the deviation movement of the target person, and the corresponding motion trajectory is called the deviation trajectory of the target person. In order to facilitate the further explanation of the principle of the algorithm, the author's movement of the target character to the left side of the corresponding robot's restricted area is called the left deviation movement, and the target character is shifted to the right side of the corresponding robot's restricted area. Deviate from the right movement. Accordingly, the author refers to the motion trajectory corresponding to the left deviation motion of the target person as the left deviation trajectory, and the motion trajectory corresponding to the right deviation motion of the target person as the right deviation trajectory. The left off track and the right off track are combined into an off track. The process of re-entering the robot from its left side to the target character, causing the target character to return to the restricted area of the robot rectangle is called the left-bias homing motion. At the same time, the author re-directs the robot from its right side to the target character, so that the process of returning the target character to the restricted area of the robot rectangle is called right-right homing motion. Then, in order to facilitate the centralized expression, the author collectively refers to the left partial homing motion and the right partial homing motion as the corrective motion. The authors collectively refer to the future motion trajectory corresponding to the corrective motion as the corrected trajectory. Accordingly, the corrected trajectory corresponding to the left-biased homing motion is referred to as a left-biased corrected trajectory, and the corrected trajectory corresponding to the right-biased homing motion is referred to as a right-biased corrected trajectory. The left-biased correction track and the right-biased correction track are collectively referred to as a corrected trajectory.

As shown in Fig. 104, the author uses the symmetrical drawing method to demonstrate the algorithm execution process of the left partial homing and the algorithm execution process of the right partial homing. In Fig. 104, l1, l2 and l3 are three spatial trajectory segments to be selected. After analysis, it can be known that the robot can reach the Pr point through any spatial trajectory line between the point pr0 and the point Pr. That is to say, the three selectable trajectories listed by the author are only used as reference trajectories. As for the trajectory that the robot finally selects, the author will elaborate and analyze later.

As shown in FIG. 105 and FIG. 106, in the process of the robot performing the forward-following task, if the target character suddenly appears to turn left or right, the robot also performs the big-turn correction motion strategy. Through analysis, we can know that the future path of such a robot has the following disadvantages:

1. The turning angle that the robot needs to perform is much larger than the turning angle of the target person, which puts higher requirements on the motion of the robot's mechanical braking algorithm;
2. The turning radius of the big turn that the robot needs to make is much larger than the turning radius of the target person. In addition to the need to consume more power, the larger turning radius means a larger, interference-free environment. Because the robots used in the actual scene of the project cannot stay in an empty laboratory forever or in

other experimental sites that are not interfered by environmental factors. In this project, the robot designed by the author is to face the actual needs and actual use. It is to be put into the market and enter the homes of enterprises or enterprises or other institutions of thousands of households, and can provide reliable and safe. And efficient follow-up services. Therefore, the designer must combine the actual environment to enhance the designer's algorithm adaptability. Since the designer does not know the complexity of the final working site of each finished robot, the designer should try to make the designer's robot following algorithm adapt to as many complex environments as possible, that is, try to improve the designer. Designed to follow the algorithm's environmental adaptability. Since the actual application environment is likely to have various obstacles, and because the curve motion of the large motion radius also means a large range of motion, the range of motion is large, and the chance of encountering obstacles is correspondingly increased, so designers should try to avoid large correction motion radii. That is, the two corrective motion modes of Fig. 105 and Fig. 106 are not suggested by the author.

After determining the premise of the designer's design algorithm, the author can then conduct a preliminary exploration of the implementation of the algorithm. The next problem to be solved is that the designer should design a future path for the robot to correct the trajectory, so that the robot can smoothly re-move to the front of the target character at a certain moment, that is, the target Is the character relocated within the restricted area of the robot rectangle? To solve this problem, the author can introduce two deeper thinking questions:

1. Should the corrected trajectory be straight or curved?
2. Does the corrected trajectory have a fixed pattern of change or change?

The two major component algorithms of the dove of peace doctrine

With these two questions in mind, the author will enter into the key construction and implementation principles of the Dove of Peace. The author analyzes FIG. 107 and FIG. 108 first. As shown in FIG. 107 or FIG. 108, the mutation trajectory of the target character is a curved segment, so the author suggests that the corrected trajectory of the robot should also be a curved segment. Then. Does it mean that the corrective trajectory designed by the author must be designed as a curved segment? With this thinking, the author analyzes Fig. 109 and Fig. 110 again. As shown in Fig. 109 or Fig. 110, the mutation trajectory of the target character is a straight line segment in the pr2pr3 segment, so the author suggests that the correct trajectory of the robot should also be the line segment. . That is, the correction trajectory may be a straight line segment or a curved segment. The specific form of the corrected trajectory needs to be combined with the deviation trajectory of the target character to perform corresponding design. It can also be found through analysis that the best correction trajectory of the robot should be as consistent as possible with the deviation type of the target person. As shown in FIG. 107 and FIG. 108, the radius of the trajectory curve of the corrected trajectory of the robot is too large, which is easy to cause waste of resources, occupying a large

number of environmental sites, and being susceptible to external interference. If the trajectory curve radius of the robot's corrected trajectory is too small, there will be insufficient turning radius, which is easy for the robot to lose the target person. In Fig. 107 and Fig. 108, the radius of the circumference of the curved curve represented by the center of the circle Or3 is obviously too small, and the radius of the circumference of the curved corner represented by the center Or1 is acceptable, but the circular path of the curved corner represented by the center Or2 is not suitable. This is because the circumferential radius of the circumference of the curved corner of the target character with the point Ohc as the center is the same as the circumferential radius of the circumference of the curved corner of the robot with the point Or2 as the center. This means that the correct path of the robot is kept in the same direction as the sudden change of the target person's abrupt path.

The author follows the same path follower algorithm that the direction of change of the correct path of the robot is closely maintained with the direction of change of the target person's abrupt path. The co-directional closely following algorithm is one of the two important parts of the dove of peace. As shown in Figures 107, 98 and 99, if the robot uses a correct path algorithm that is closely following in the same direction. Since the robot is already in the same direction as the target character at the beginning, the robot changes its direction of motion change at the same time as the target task changes direction at any time during the completion of the entire correction path. However, it should be noted here that only the direction of motion remains the same, not the direction of motion remains the same. Therefore, in the process of the robot performing the correcting path motion, the motion of the robot is likely to cause the target person to be separated from the restricted area of the robot. So will the robot eventually replace the target person in its restricted area? This is a question worth considering. In the robot correction path shown in Figures 107, 98 and 99, the robot finally returns the target character to the restricted area of the robot. Does the robot's corrective path achieve this effect? Through analysis, it can be found that in the robot correction path shown in Figs. 107, 98 and 99, the relative straight line of the robot and the direction of the target person before performing the correction path, and the robot and the target person after performing the correction path. The relative straight lines of the directions in which they are located are parallel. That is, if the robot and the target person maintain the same parallel direction before the correction path is executed and after the correction path is performed, the robot can still return the target person to the restricted area of the robot after performing the correction path.

It can be known from the above algorithm analysis that if the robot performs the corrective motion corresponding to the correction path only by the co-directional closely following algorithm, the target person is returned to the restricted area of the robot with conditional restrictions. This limitation is that the relative straight line of the direction in which the robot is in the direction of the target person is paralleled before the correction path is executed and after the correction path is executed. But in practical applications, this limitation is not always met. As shown in FIGS. 110, 101, 102 and 103, the relative straight lines of the robot and the direction in which the target person is located are not kept parallel before the correction path is executed and after the

correction path is executed. It can be analyzed by analysis that in the cases shown in Figs. 110, 101, 102 and 103, the author's algorithm requirements cannot be satisfied only by the co-directional closely following algorithm. In order to solve this problem, the author proposed another important algorithm among the two algorithms of the dove of the peace, namely the variable direction approach algorithm. In the dove of the dove algorithm, the variable-turning approach algorithm is not required from time to time, and is only used when the co-directional closely following algorithm cannot meet the requirements of the algorithm. That is, the redirecting curve approach algorithm is employed only when the relative straight line of the direction in which the robot and the target person are located is not parallel before the correction path is executed and after the correction path is executed. By analyzing the situation shown in Figures 110, 101, 102 and 103, the author knows that the trajectory calculated by the variable curve approach algorithm is an elongated S-shaped center-symmetric offset curve, which is intended to cause the robot to re-regress. The restricted area to the robot can re-include the relative position status of the target person.

The timing of the two sub-algorithms of the dove of peace doctrine

When does the designer use the same-direction tight-following algorithm and when do designers use the variable-turn curve approach algorithm? The following authors consider and answer the above questions by analyzing the graphs 110 and 111. As shown in FIGS. 110 and 111, before the corrective motion is performed, the target person is located at point ph0 in the space, and the robot is located at point pr0 in the space. Then the target person begins to deviate. First the target character moves from point ph0 to point ph1. When the target person reaches the point ph1, it is located on the boundary line of the restricted area of the robot rectangle. If the target character continues to move along the curve, once the point ph1 is removed, the author claims that the target person has actually deviated. At this point the robot will start the correction algorithm. Firstly, the same direction closely following algorithm is started. The robot calculates the specific trajectory of the known path arc ph0ph2 of the target person based on the previously known target path information of the target person, and then the robot will quickly complete the correction path pr0pr1 and execute it. Corresponding to the corrected path arc pr0pr1 of the target person's known deviation path arc ph0ph2. Then, corresponding to the deviation path ph2ph3 of the target person, the robot performs the correction path pr1pr2 accordingly. When the target character moves to point ph3, the robot will move to pr2. Through analysis, it can be known that the speed direction of the robot and the target character is again parallel. However, the target person at this time is not in the restricted rectangular area of the robot. That is, the same-direction close following algorithm can only guarantee that the original speed direction is parallel to the robot and the target person, and after the corrected path, the speed directions of the two are still parallel. However, the same-direction closely

following algorithm does not guarantee that after the corrected path, the target character is still inside the restricted rectangle of the robot. Therefore, at this time, the author has to use the second important algorithm of the dove of peace to deal with this problem. The second important algorithm is the variable corner approach algorithm. As shown in FIGS. 110 and 111, when the robot reaches the point pr2, the forward direction of the robot has been parallel to the forward direction of the target person. At this point, the robot does not immediately adopt the variable direction approaching algorithm to approach the direction of travel of the target person, but continues to use the same-direction close following algorithm to keep straight forward. The variable corner approach algorithm of the dove of the peace is not activated until the robot moves to point pr3. The goal is to minimize the time difference and speed difference when the robot performs the corrective path. Because the target person's deviation movement is ahead and the robot's corrective movement is behind, the robot must speed up to make up for this difference. But the algorithm designed in this way has the following disadvantages:

1. The complexity of the algorithm is relatively large, and there are many parameters that need to be calculated and judged, which is easy to cause motion time delay.
2. Finally, the space left for the robot to make the turning direction approaching algorithm is insufficient, which causes the turning of the robot to be large, which makes it difficult for the robot to drive mechanically.

The authors of the above two questions propose the following strategies for implementing the Peace Dove Algorithm:

1. Once it is detected that the target person is out of the boundary of the robot rectangle, immediately start the same-direction close following algorithm. In the subsequent corrective motion, the same direction closely following algorithm is maintained until the forward direction of the robot and the forward direction of the target person are again in parallel.
2. Once the forward direction of the robot is parallel to the forward direction of the target character, the execution of the same-direction close following algorithm is immediately stopped, and then the variable-turn curve approach algorithm is immediately started. At this point, the robot will move along the correcting trajectory of the changing curve until the forward direction of the robot is parallel with the forward direction of the target person. At this point, the target character will return to the restricted area of the robot rectangle.

The above is the basic implementation process of the entire dove of pigeons.

Detailed and complete implementation of the Dove of Peace algorithm

The following authors will explain in detail the implementation of the dove algorithm in conjunction with Figure 112 and Figure 113.

As shown in Figure 112 and Figure 113:

1. First, the target character is located in the restricted area of the robot rectangle before the target person performs the deviation movement. At this point the target character is at point ph0 and the robot rectangle is at point pr0.
2. The target character then starts the deviation movement along the arc Ch1. However, when the target person has not deviated from the restricted area of the robot rectangle, the robot will not start the correction algorithm, which is the dove of peace. The purpose of this design is also described in the previous algorithmic description. The purpose of this design is to reduce the LRF detection misjudgment of the target person.
3. Then, once the target character continues to move beyond the restricted area of the robot, the robot will determine that the target character is performing a deviation movement. The robot then initiates the same-direction close following algorithm in the dove of peace. After that, the robot will closely follow the target character and follow the movement in the same direction.
4. Since the robot waits until the target person leaves the restricted area of the robot, the same-direction close following algorithm is started. Therefore, in time and space, when the in-timing close following algorithm is started, the robot is a circular arc behind the target person. The author refers to the arc behind the robot as the starting supplemental curvature. Therefore, when the robot knows that the target person has left the restricted area, the robot immediately starts the same-direction close following algorithm, and calculates and simulates a starting supplementary path arc starting from the center of the restricted area of the robot for the target person. It should be noted here that when the target person just deviates from the movement, the position of the target person is not necessarily at the geometric center point of the restricted area of the robot. Generally speaking, the target person is started by the boundary edge region of the restricted area of the robot and is separated from the restricted area of the robot. So why doesn't the author set the starting point of the starting path to the path where the target person leaves the restricted area of the robot? This is because, if the deviation of the target person is calculated from the area near the boundary line of the restricted area of the robot rectangle from the beginning of the target character, the target character is returned to the restricted area of the robot rectangle after the peace dove algorithm is still A boundary point located near the rectangular boundary of the robot. Although

such a result is also accepted, it is easy to start the second consecutive Peace Dove algorithm (IE: after returning to the restricted area of the robot rectangle, the collision-following algorithm of the dove algorithm is accidentally triggered again.) So the author will target The starting point of the character is virtualized at the center point of the robot rectangle, so that more error tolerance space can be provided for the target character returning to the robot rectangle. After determining the true starting point of the target person, the robot will calculate and simulate a moving arc track connecting the virtual starting point of the target person and its real starting point. This trajectory corresponds to the starting replenishment trajectory of the robot, that is, Cr1 corresponds to Ch1.

5. Then, the target character sequentially deviates from the arc through two segments: Ch2, Ch3 and then moves to point ph1. Correspondingly, the robot also moves to the point pr1 after two stages of correcting the arc Cr2 and Cr3. At this time, the forward direction of the robot is parallel to the forward direction of the target person again. Thus, the robot immediately stops the implementation of the co-directional closely following algorithm and simultaneously initiates the implementation of the redirecting curve approach algorithm.
6. Then, the robot will calculate the spatial position point that the target person has reached after the time T has continued to move linearly according to the current speed. The time T here is the approaching task execution time set by the variable direction curve approach algorithm. That is, the algorithm needs to start at the moment when the turning corner is approached by the algorithm, and after the time interval T, it is accurately moved to the calculation destination. As shown in FIG. 112 and FIG. 113, it is assumed that the moment when the robot starts to start the approaching curve approaching algorithm, the system time is T0, the space position point where the robot is located is point pr1, and the space position point of the target character is point ph1. . Assume that after the time interval T, the target character will move along the straight line to the spatial position point ph2, and the robot will move to the spatial position point pr2, at which time the forward direction of the robot is again in parallel with the forward direction of the target character, and The target character at this time is at the center of the restricted area of the robot rectangle. Thus, the robot completes the entire execution of the algorithm of the redirecting curve approach.
7. If the robot detects that the target person is not in a linear motion during the process of changing the approaching curve approach, but uses a curved motion, the robot will immediately stop the implementation of the variable direction approach algorithm and immediately start the same direction. Follow the algorithm closely until the target person makes the next linear motion.

8. The final implementation end of the dove of the dove algorithm is that the target person returns to the restricted area of the robot rectangle and is in the spatial position of the center point of the restricted area. At this point, the robot will assume that the entire dove of pigeon algorithm is completely over and obtain reasonable algorithmic execution results.

Based on the above analysis, it can be known that the execution principle of the same-follower algorithm of the dove of the dove is relatively simple and reasonable. The key complex algorithm of the dove of the dove is the variable-turning approach algorithm. Specifically, it is how the corrected trajectory of the algorithm is calculated. The following authors will focus on the interpretation and analysis of the design and implementation principles of the variable corner approach algorithm in the Dove of Peace.

The core of the dove of peace and the origin of the name "Dove of Peace"

The most critical algorithm in the two sub-algorithms of the dove of the dove is: the variable corner approach algorithm. The initial design of the variable-turn curve approach algorithm is that the robot moves toward the target human in a straight line in the form of a curve by means of the deviation of the traveling track. Then, the key question is what kind of curve trajectory the robot is close to the target person. In response to this question, the author puts forward the following thoughts:

1. *What is the approaching curve with what trajectory characteristics?*
2. *What is the specific implementation principle of the approaching curve?*

After comparing the motion characteristics of many corner trajectories, the author finally selected a symmetrical curve trajectory. This is because the symmetrical curve can well achieve the approaching motion of progressively spanning across different parallel straight lines. Relative to a straight or asymmetrical approach to a curve, a more symmetrical approach to the curve will result in a smoother approach to the motion. So the author eventually adopted a symmetrical approach to the corner. Moreover, the symmetrical approach to the corner has a feature that the trajectory is more beautiful and more in line with the positioning of the intelligent robot.

After determining the symmetrical proximity to the corner, the author solves the problem raised by the second thought. In order to facilitate the author to divide the symmetrical approaching curve into two segments, the straight line along the curve, if it is a convex curve close to the center line, it is called a convex curve, if it is a concave curve away from the center Straight lines are called recessed corners. It can be known

from the analysis that there are two main manifestations of the symmetric approach to the curve. The first form is the close to the curve of the symmetrical curve radius shown in Fig. 114. In this approaching curve, the convex curve on both sides of the central straight line has the same radius as the curved curve corresponding to the concave curve. The author takes an example of a near-curve with a radius of curvature R5. As shown in Fig. 114, the straight line Vline is the center straight line, and the point P_{c5} is the center point of the curve. The author uses the curve center point P_{c5} as the observation angle, and it is easy to find that the entire curve takes the point P_{c5} as the symmetry center and presents a centrally symmetrical image. The curved line segment R5P_{c5} is a convex curved line segment, and the curved line segment P_{c5}P₅ is a concave curved line segment. By comparing and analyzing the curve corners of R₁, R₂, R₃, R₄, R₅ close to the curve, it can be found that the trajectory end point of the symmetrical curve radius near the curve has two changeable parameter properties of horizontal and vertical. It is not suitable for the variable corner approach algorithm designed by the author. Although the schematic diagram of the overall algorithm implementation looks like a beautiful feather, it is not the author's ideal final algorithm.

The second symmetrical curve is expressed in the form of a symmetrical curve span close to the curve. The second curve approach algorithm also has a central straight line and a convex curve and a concave curve. The biggest difference between the second curve approach algorithm and the first curve approach algorithm is that the second curve approaching algorithm is symmetrically the lateral span of the curve on both sides of the center line, rather than the convex and concave curves. The radius. As shown in Fig. 115, it is assumed that the robot is at the point R₁ just after the start of the variable direction approaching algorithm. At this time, the direction of the forward direction of the robot is L₁, and the straight line of the target person is L₂, that is, the target person This is now in the direction of the line L₂. When the robot just reaches the R₁ point, that is, when the robot just stops the in-situ tight follow algorithm and just starts the change direction approach algorithm, the system time is set to T₀. Assume that after the time period t_i, the robot will move to the point P_i by approaching the curve, the target person will move to the point P_{hi}, and the relative distance between the point P_i and the point P_{hi} is exactly the center point of the restricted area of the robot rectangle. The distance from the geometric center point of the robot rectangle. Assume that the distance from the center point of the restricted area of the robot rectangle to the geometric center point of the robot rectangle is D_r.

The specific implementation of the variable cornering algorithm of the dove of peace

The following is a description of the specific algorithm implementation principle:

Assume that the XYZ following coordinates of point R1 are (x_{r1} , y_{r1} , z_{r1}), the computing system of the robot is calculated according to the speed direction and size of the target person and the direction and magnitude of the acceleration. At T_0+ti , the spatial position of the target character is Phi (x_{hi} , y_{hi} , z_{hi}). As shown in FIG. 115, assuming that the angle between the direction of the center line and the X axis of the XYZ following coordinate system is θ_d , the coordinates of the point $P_i(x_i, y_i, z_i)$ are estimated according to the point $\Phi(x_{hi}, y_{hi}, z_{hi})$:

$$\begin{cases} x_i = x_{hi} - Dr \cdot \cos\theta_d \\ y_i = y_{hi} + Dr \cdot \sin\theta_d \\ z_i = z_{hi} \end{cases}$$

Knowing the coordinates of the initial point R_1 (x_{r1} , y_{r1} , z_{r1}) of the robot's changing curve and the end point $P_i(x_i, y_i, z_i)$, the designer can calculate the trajectory of the robot near the curve.

First, the author analyzes some of the curve characteristics of the symmetrical curve span near the curve. It can be known from the above analysis and summary that the close curve of the symmetrical curve span near the curve has a symmetrical curve span, that is, in the approaching curve mode of the symmetrical curve span, the convex curve and the concave are convex. The corners of the curve are the same. So, is there any other curve property near the curve of the symmetrical curve span? By analyzing Fig. 115, it can be found that the near-curve of the symmetric curve span near the curve mode is also in a central symmetry state, and the center of symmetry is located on the center line. In order to further demonstrate that the near-curve of the symmetrical curve span near the curve mode is symmetrically distributed, the author has designed Figure 116. The image in Figure 116 is called a two-winged flying pigeon with a symbolic meaning of peace and soaring, development and endeavor. By analyzing Fig. 116, it is easy to find that the near-curve of the symmetrical curve span near the curve mode is symmetrically distributed.

When the near-curve near the curve mode with the symmetrical curve span is known to be symmetrically distributed, the designer can calculate the trajectory curve corresponding to the curve corresponding to the symmetrical curve span.

Combined with the algorithm principle shown in FIG. 115, from the previous algorithm content, the designer knows the coordinates of the initial point R_1 (x_{r1} , y_{r1} , z_{r1}) of the robot's changing curve and the end point $P_i(x_i, y_i, z_i)$. Assuming that the symmetry center of the curve trajectory R_1P_i is $P_{ci}(x_{ci}, y_{ci}, z_{ci})$, then:

$$\begin{cases} x_{ci} = \frac{x_{ri} + x_i}{2} \\ y_{ci} = \frac{y_{ri} + y_i}{2} \\ z_{ci} = \frac{z_{ri} + z_i}{2} \end{cases}$$

Since the author uses two central symmetrical arc segments to represent the convex curve and the concave curve edge of the robot, the key of the algorithm designed by the author is to first find the center corresponding to the convex curve arc. Then, according to the central symmetry property of the curve edge, the center corresponding to the arc of the concave curve edge is obtained.

Referring to FIG. 115, FIG. 116 and FIG. 117, it is assumed that when the robot is at the point R1, the angle between the forward direction of the robot and the X-axis is θ_d . For the above-mentioned near-curve span, the horizontal span, that is, W_h , is obtained according to the attribute defined earlier in the algorithm, and the horizontal span W_h corresponding to a certain curve end point $P_i(x_i, y_i, z_i)$ is:

$$W_{hi} = \frac{y_i - y_{r1}}{\cos \theta_d}$$

For this part of the algorithm, y_i is a constant, so W_{hi} is also a constant value.

Correspondingly, the radial span W_{vi} corresponding to a certain curve end point $P_i(x_i, y_i, z_i)$ is:

$$W_{vi} = \frac{x_i - x_{r1}}{\cos \theta_d}$$

Next, the author starts from the two-dimensional data of the coordinate points corresponding to the XY plane, and obtains the coordinate point $R1'(x_{r1}', y_{r1}', z_{r1}')$ of the robot starting point $R1(x_{r1}, y_{r1}, z_{r1})$ symmetric about the center line $L0.$, the symmetry of the horizontal line near the curve of the curve, it can be known that $R1'$ must be on the $L2$ line. The following authors will find the coordinates of the point $R1'(x_{r1}', y_{r1}', z_{r1}')$ from the coordinates $P_i(x_i, y_i, z_i)$ of the point P_i and the radial span W_{vi} corresponding to the point P_i and the point P_i :

$$\begin{cases} x_{r1}' = x_i - W_{vi} \cdot \cos \theta_d \\ y_{r1}' = y_i + W_{vi} \cdot \sin \theta_d \\ z_{r1}' = z_{r1} \end{cases}$$

In the above expression formula, in order to simplify the algorithm, considering the magnitude of the width of the support surface and the lateral span near the curve, the designer regards the Z coordinates of the spatial position points of the same lateral span straight line as the same. Or the designer can directly use the LRF to directly detect the Z coordinate of the corresponding $R1'$ point on the XY plane. Then the above formula becomes:

$$\begin{cases} x_{r1}' = x_i - W_{vi} \cdot \cos \theta_d \\ y_{r1}' = y_i + W_{vi} \cdot \sin \theta_d \\ z_{r1}' = z_{r1}'(LRF) \end{cases}$$

After obtaining the XYZ following coordinates of the point R1', it returns to the discussion and analysis of the XY plane. From the principle of the planar two-point linear equation, the two-point expression of the straight line of the straight line R1R1' is obtained as:

$$\frac{x - xr1}{xr1' - xr1} = \frac{y - yr1}{yr1' - yr1}$$

Then, from the central symmetry point Pci(xci, yci, zci) which is close to the curve, and the initial point R1 (xr1, yr1, zr1) which is close to the curve, the midpoint of the straight line segment connecting them is obtained. The XYZ following coordinates of Pci(xtci, ytci, ztci) are:

$$\begin{cases} xtci = \frac{xci + xr1}{2} \\ ytci = \frac{yci + yr1}{2} \\ ztci = \frac{zci + zr1}{2} \end{cases}$$

Then, from the relationship between the slopes of the two straight lines perpendicular to each other, the slope kti of the straight line perpendicular to the straight line segment R1Pci is:

$$kti = \frac{-1}{ki} = -\frac{xci - xr1}{yci - yr1}$$

Then, the point-oblique equation for the straight line perpendicular to the straight line segment R1Pci of the point Pci (xtci, ytci, ztci) is:

$$y = kti(x - xtci) + ytci$$

Which is:

$$\frac{y - ytci}{kti} = x - xtci$$

Finally, the simultaneous equations will be got as:

$$\begin{cases} \frac{x - xr1}{xr1' - xr1} = \frac{y - yr1}{yr1' - yr1} \\ \frac{y - ytci}{kti} = x - xtci \end{cases}$$

Find the projection coordinates (xvci, yvci) of the center point Pvci of the convex

curve arc on the XY plane. At the same time, it is known from the previous analysis that the Z-axis coordinate zvci of the point Pvci is set to coincide with the point R1, or by LRF is directly detected. At this point, the coordinates of the two end points of the convex curve arc (IE: corner starting point R1 (xr1, yr1, zr1) and the curve ending point Pci (xci, yci, zci)) are obtained. The center point Pvci coordinates Pvci (xvci, yvci, zvci). Thus, a complete arc curve is completely determined, that is, the contour curve of the dove of the Dove algorithm close to the algorithm is completely determined.

Referring to FIG. 117, through careful analysis, it can be known that the center of the arc of the convex curve has a central symmetry relationship with the center of the arc of the concave curve, and the center of symmetry Pci of the entire turning direction is the center of symmetry. of. Then, after obtaining the center Pvci (xvci, yvci, zvci) of the convex arc of a certain curved curve of the turning curve, the concave corresponding to the curved arc of the segment is obtained by the following formula. The XYZ following coordinates of the center coordinate point Pcci(xcci, ycci, zcci) of the arc of the curve are expressed as:

$$\begin{cases} xcci = 2 \cdot xci - xvci \\ ycci = 2 \cdot yci - yvci \\ zcci = zi \end{cases}$$

Or the Z coordinate of the point Pcci is directly measured by the LRF, i.e.:

$$\begin{cases} xcci = 2 \cdot xci - xvci \\ ycci = 2 \cdot yci - yvci \\ zcci = zcci(LRF) \end{cases}$$

At this point, the coordinates of the two end points of the concave curve arc (IE: corner starting point Pci (xci, yci, zci) and curve end point Pi (xi, yi, zi)) and the arc are obtained. The coordinates of the center point Pcci are Pcci(xcci, ycci, zcci). Thus, a complete arc curve is completely determined, that is, the concave curve of the dove of the dove algorithm close to the algorithm is completely determined.

Combine with the trajectory of the convex curve of the convex curve obtained in the previous part, the entire segment of the curved curve is calculated near the trajectory curve. That is, the operation to this step, the turn-around curve approach of the dove of the dove algorithm is successfully completed.

In addition, to make the dove of the peace more robust, it is necessary to add obstacle avoidance mechanisms to the dove of peace. Due to the limited writing time of this project report, the obstacle avoidance part will be explained in detail in the second edition of this report. At this point, the dove algorithm described in this version of the project report has been fully described.

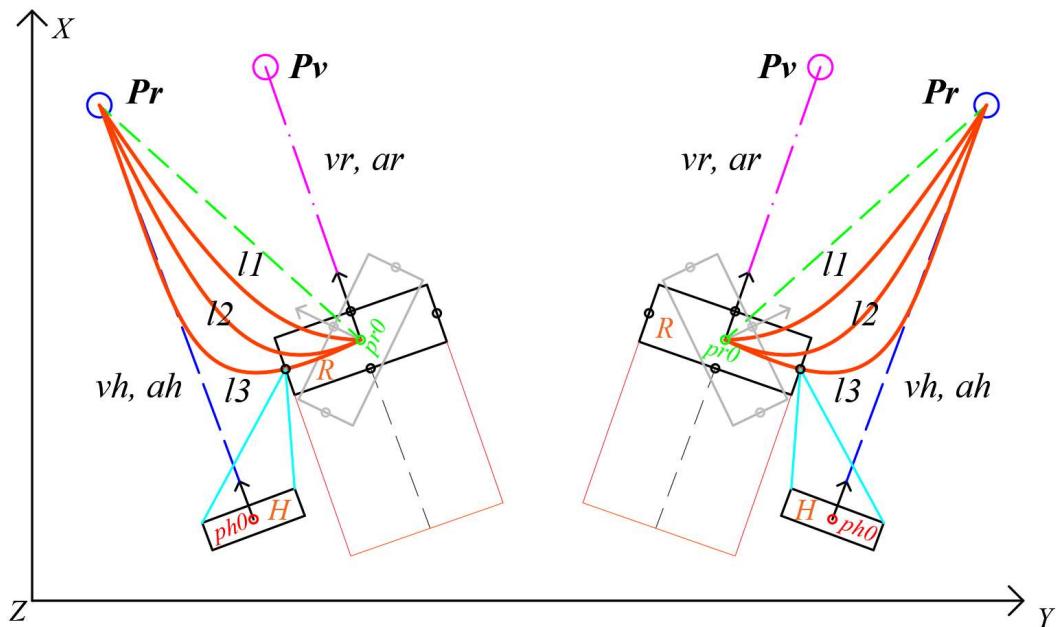


Figure 104. The re-correction algorithm strategy executed by the robot when the robot implements the forward-following, the robot knows that the target person has left the limit of the robot rectangle.

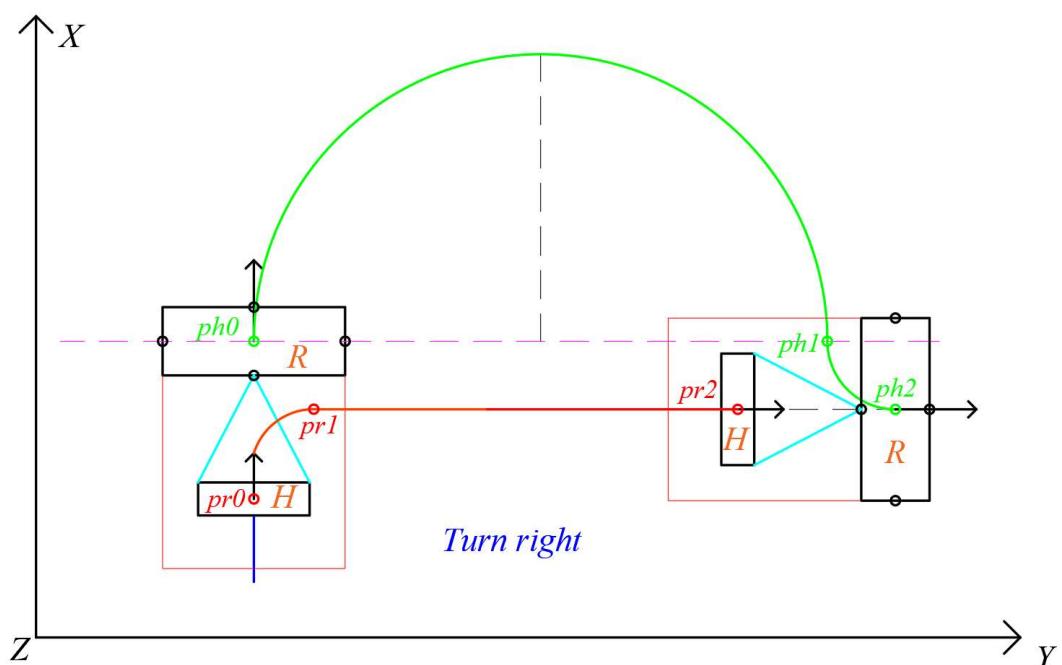


Figure 105. When the target character turns to the right, the robot turns to the right and turns the future path.

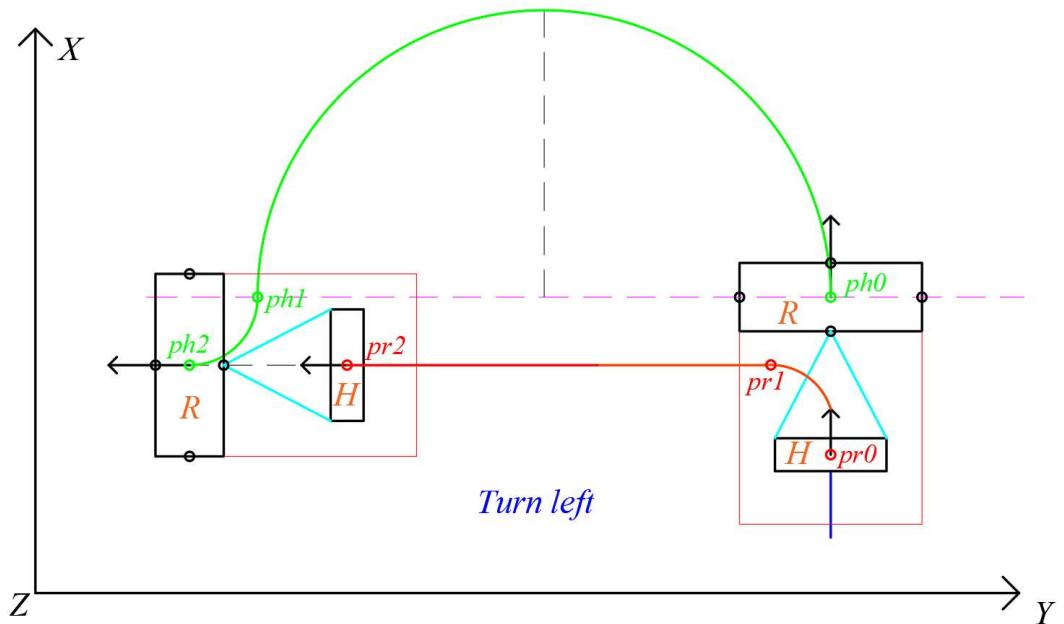


Figure 106. When the target character turns to the left, the robot turns to the left and turns the future path.

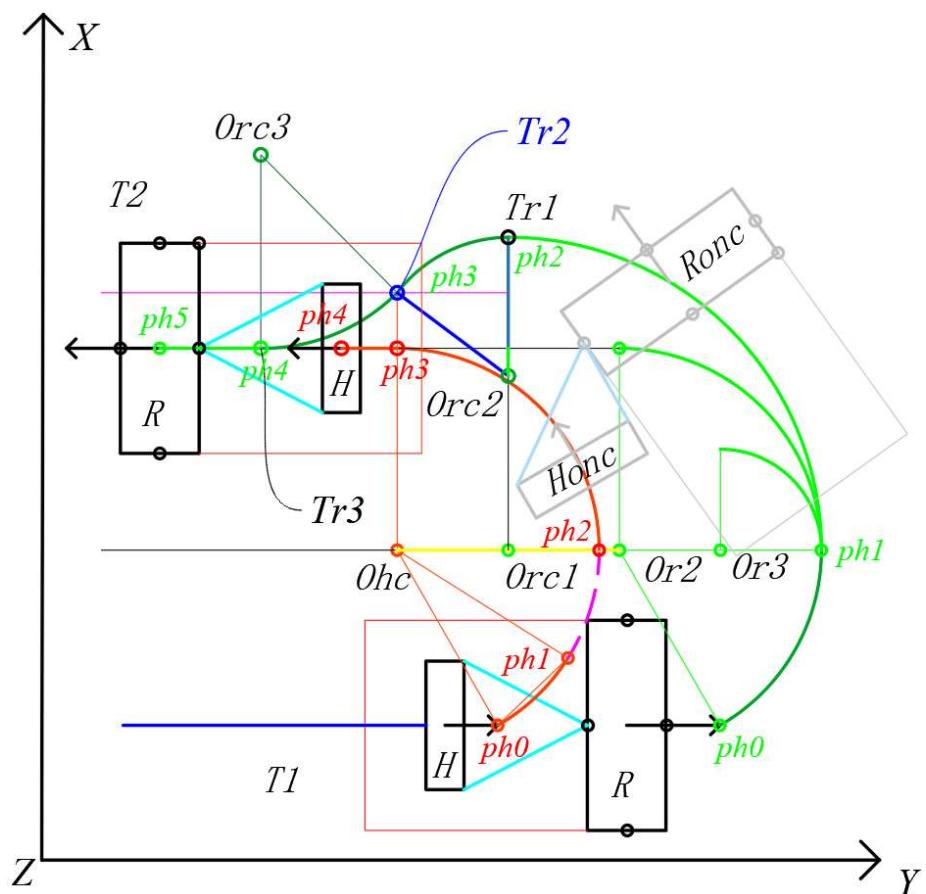


Figure 107. Reflections on the future path of the robot's corrected trajectory. Left bias corrects the trajectory.

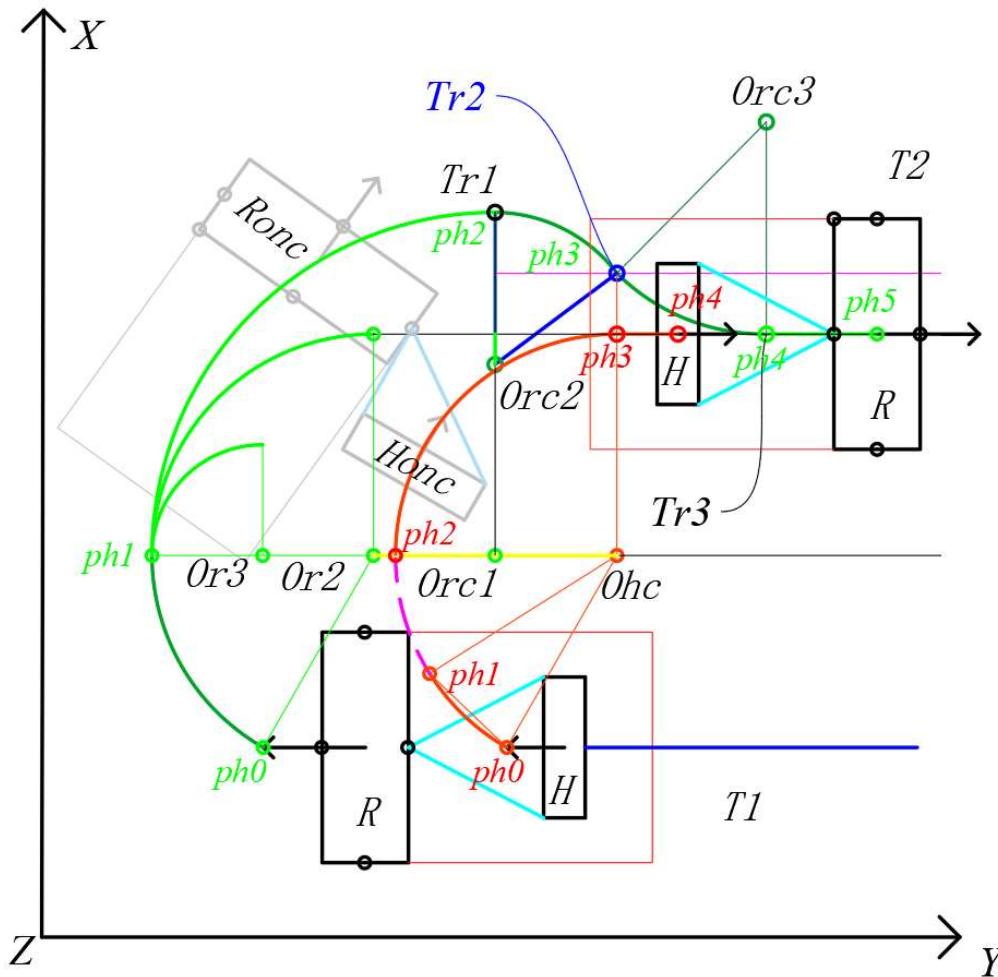


Figure 108. Reflections on the future path of the robot's corrected trajectory. Correct the trajectory to the right.

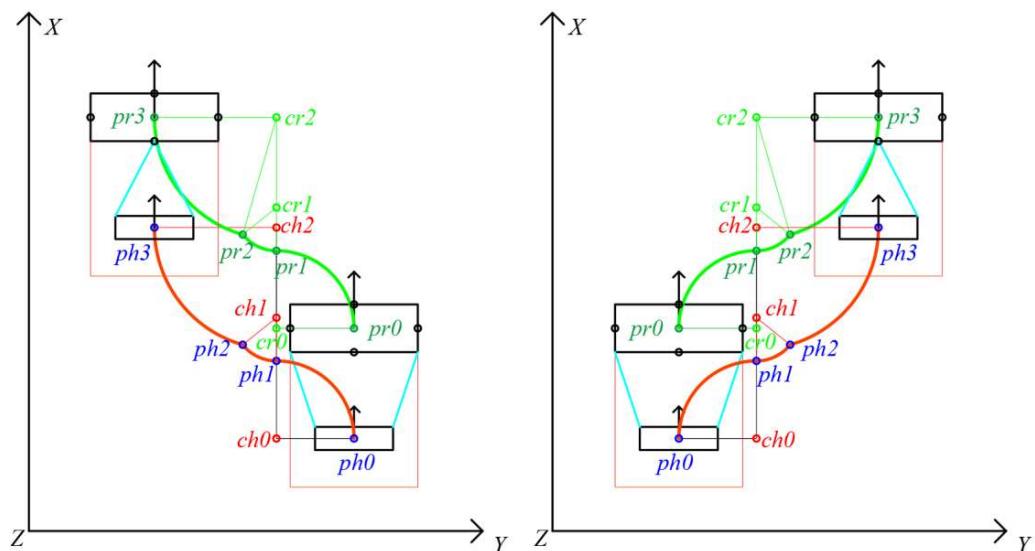


Figure 109. Features of the same-going closely following algorithm. Left-right correction exercise and right-right correction exercise display.

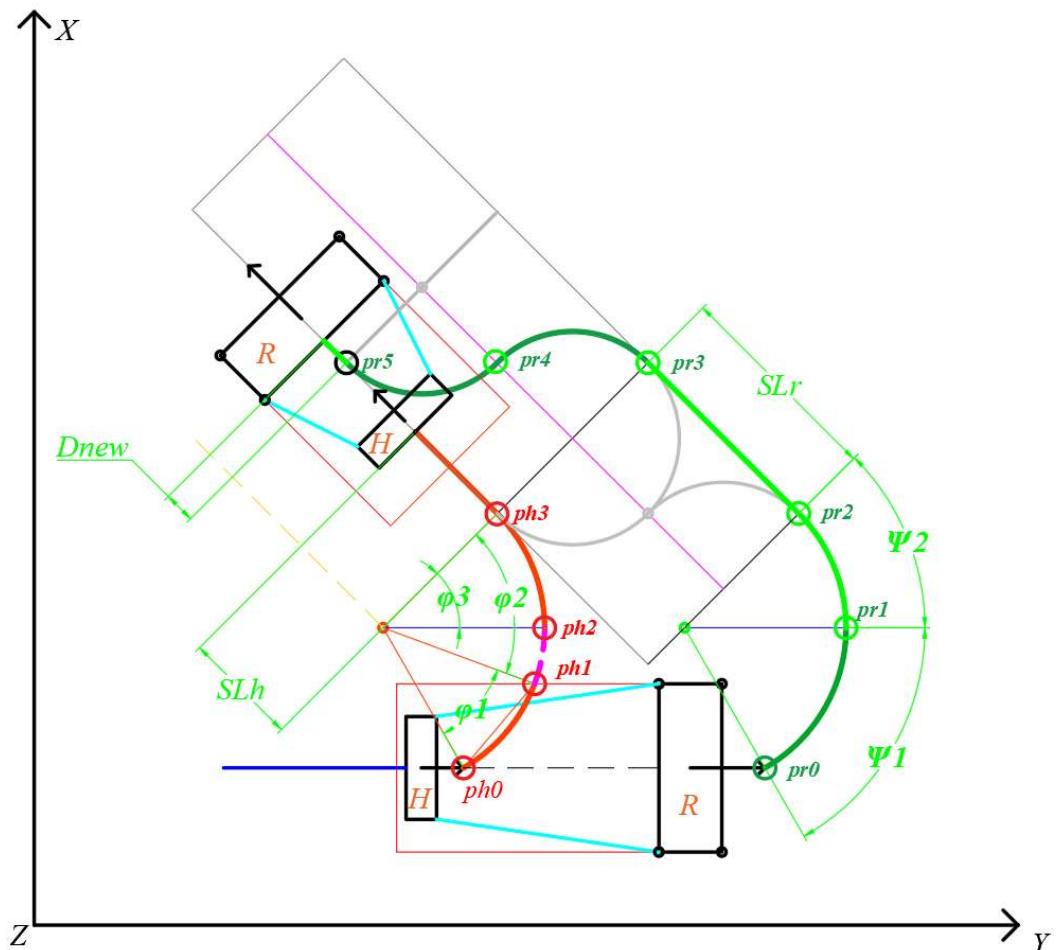


Figure 110. Reflections on the future path of the robot's corrected trajectory. Thinking about the difference in speed when correcting, the left side corrects the trajectory. The turning curve is close to the algorithm.

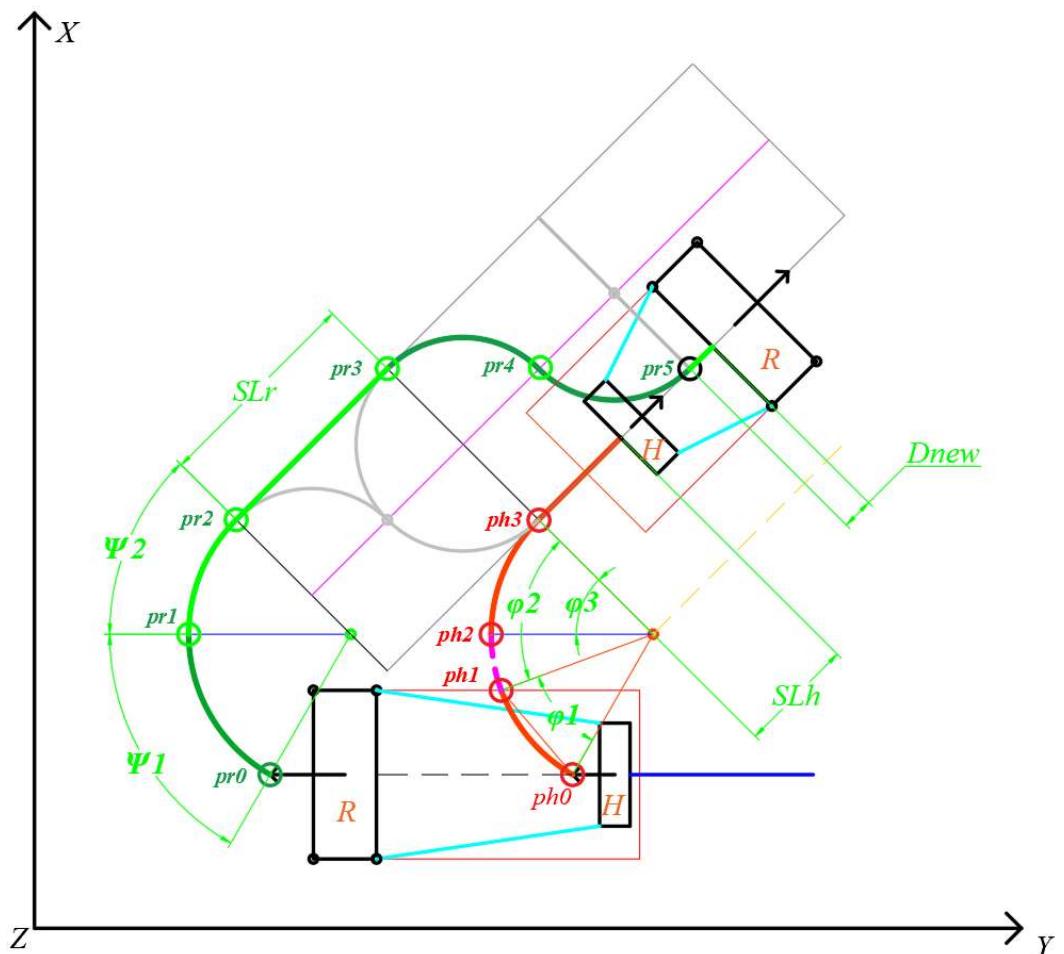


Figure 111. Reflections on the future path of the robot's corrected trajectory. Think about the difference in speed when correcting, right-correcting the trajectory. The turning curve is close to the algorithm.

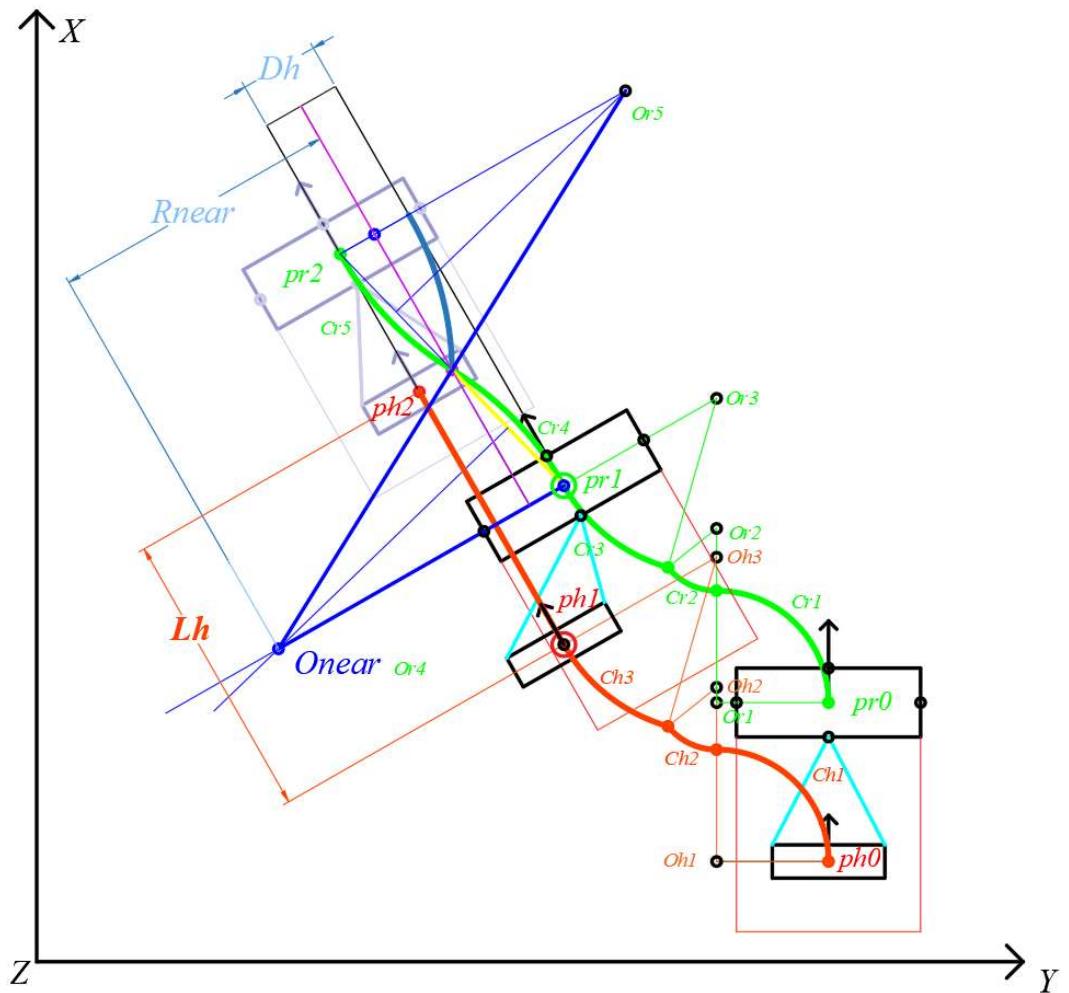


Figure 112. Combination of the same-directional close following algorithm and the variable-turn curve approach algorithm. Implementation of the dove of peace. Left bias corrects the trajectory. Left to correct the movement.

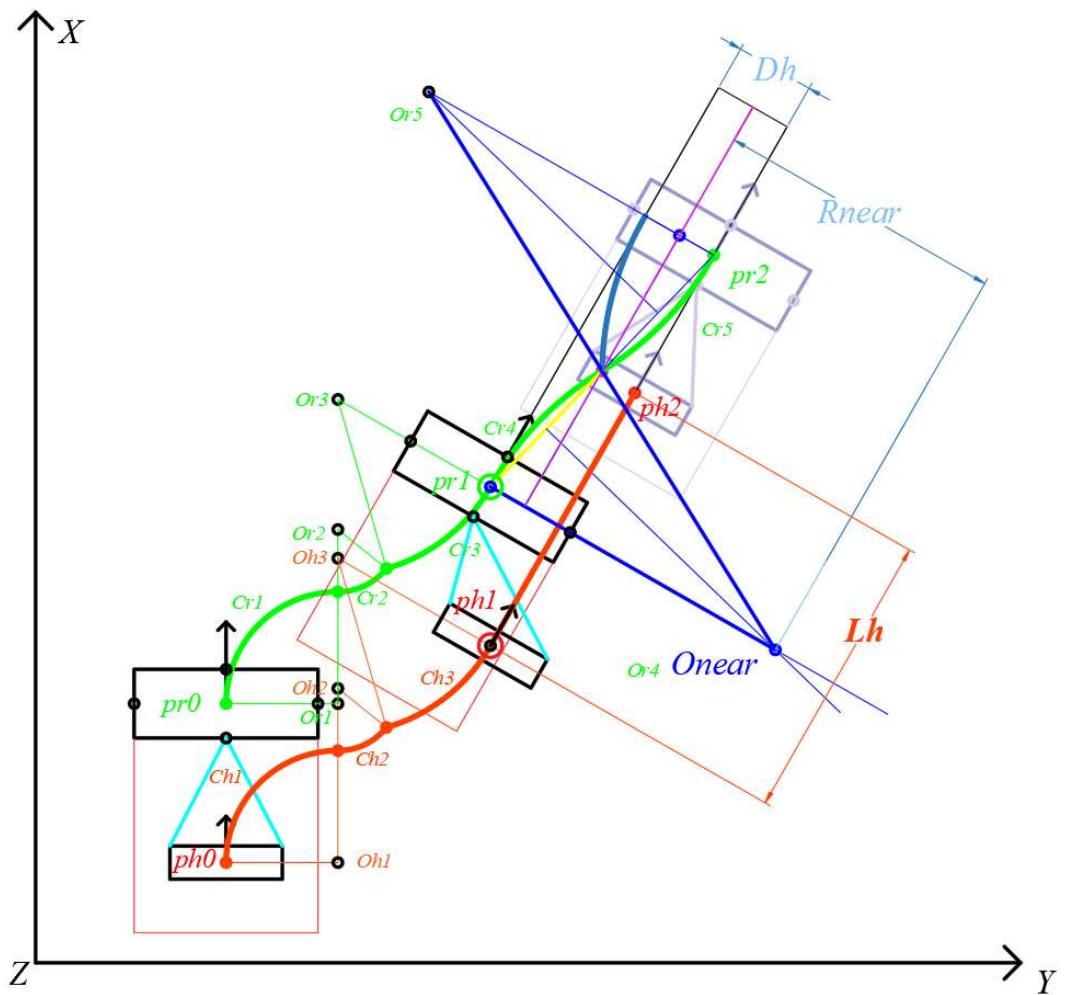


Figure 113. Combination of the same-direction close following algorithm and the variable-turn curve approach algorithm. Implementation of the dove of peace. Correct the trajectory to the right. Right deviation corrects the movement.

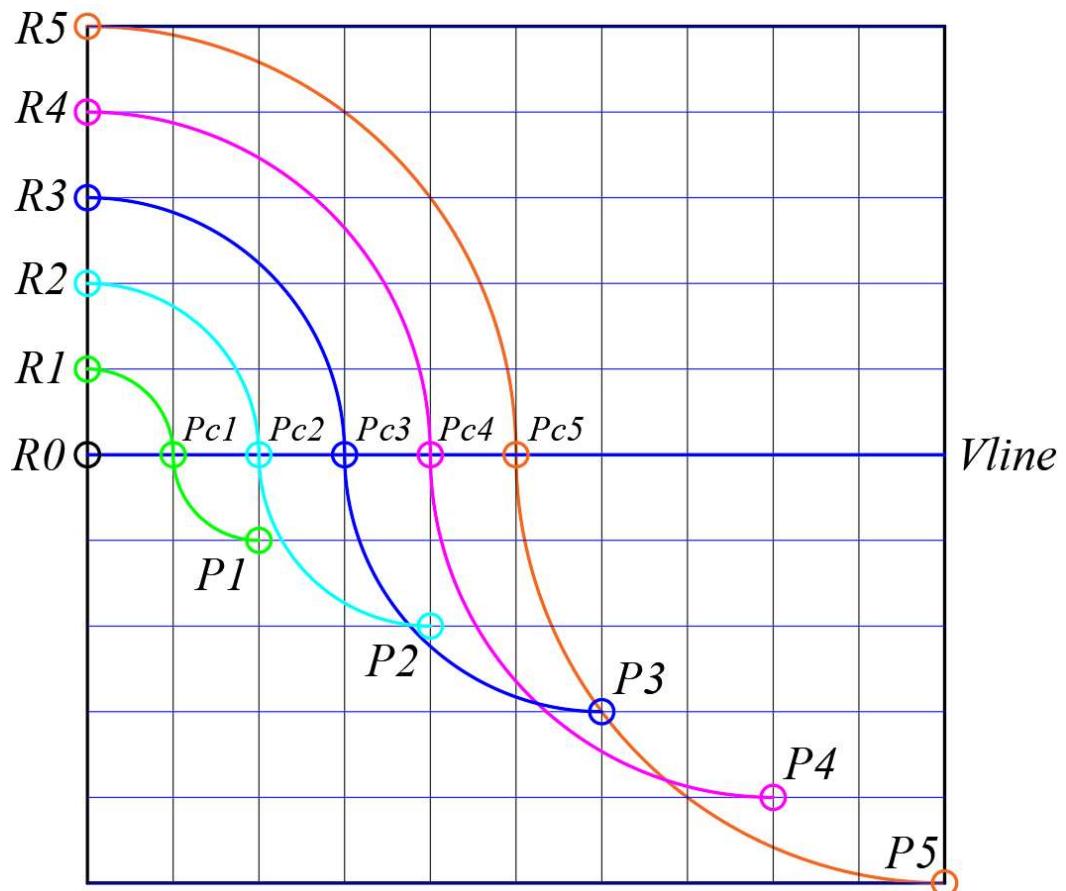


Figure 114. The turning curve algorithm for the dove of peace. Curved radius symmetrical mode.

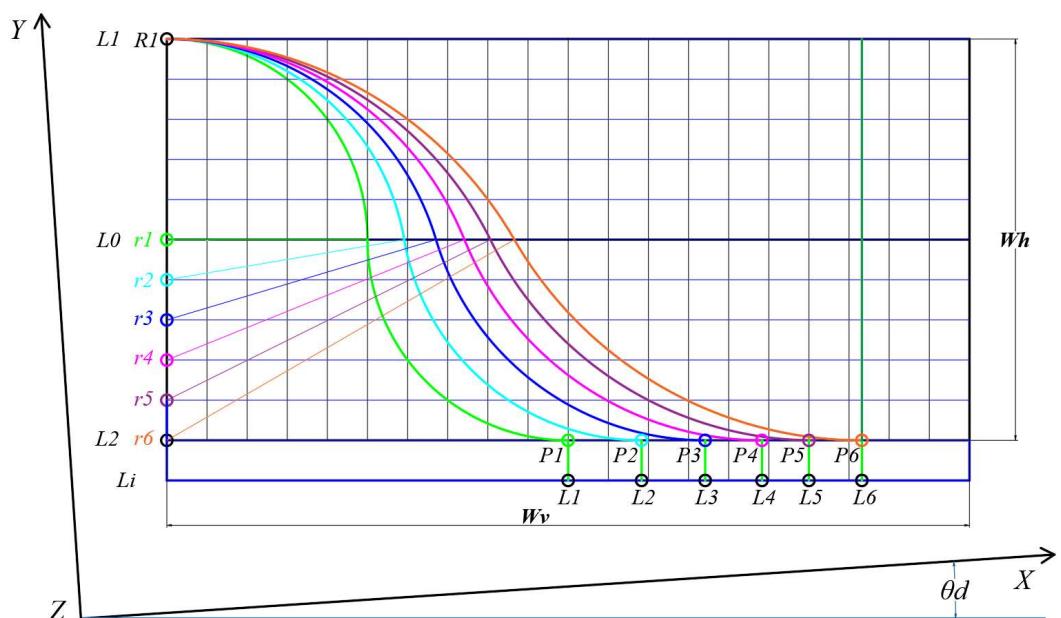


Figure 115. The turning curve algorithm for the dove of peace. Curved span symmetrical mode.

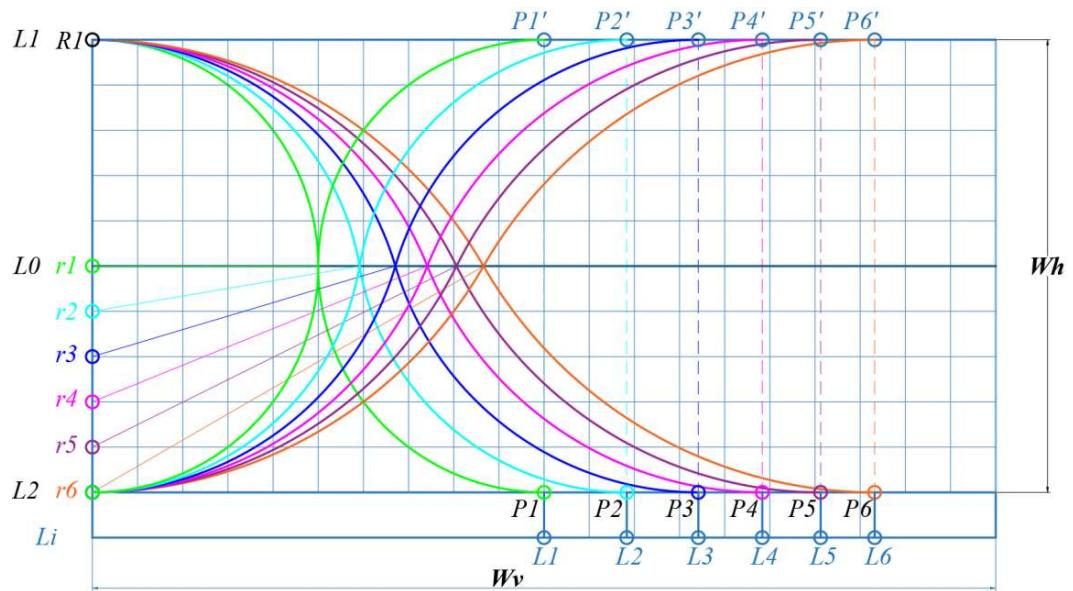


Figure 116. Symmetrical display of the Peace Dove algorithm near the curve.

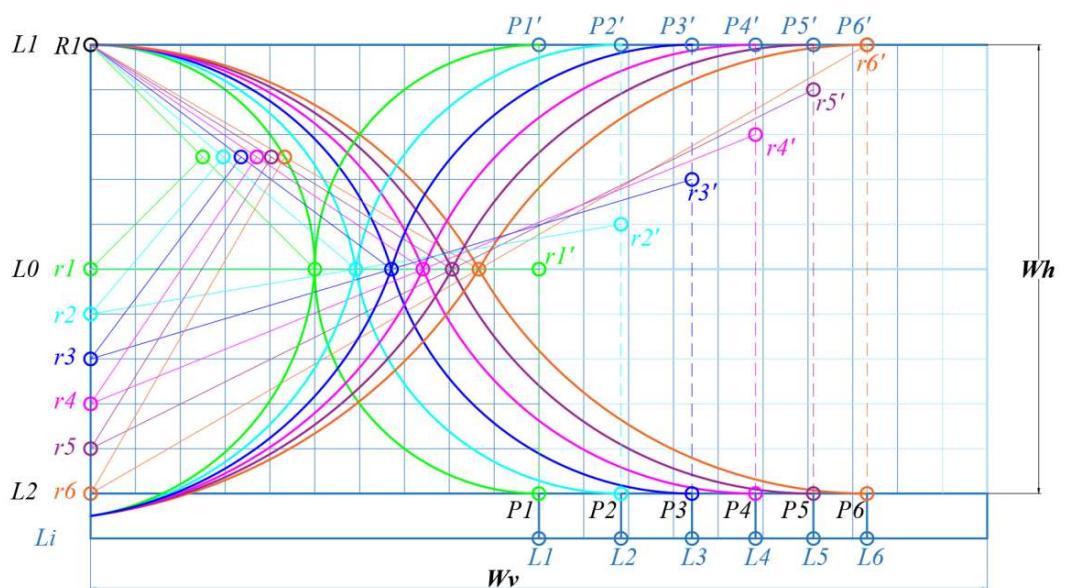


Figure 117. Schematic diagram of the algorithm for solving the curve of the dove of Peace.

IMPORTANT

Due to the limited writing time of this project report, more exciting content will be arranged in the second edition of the project research report. The new version of the second version of this project research report will have:

1. Introduction of gait analysis
2. Filter analysis
3. Exception handling model
4. Functional and technical development.

Here again, please look forward to your expectations. Thank you!

For more exciting details content, please refer to the other articles of this essay:

1. ROBOT DETECTION SYSTEM DESIGN ABOUT FRONT-FOLLOWING TECHNOLOGY PART 1
2. ROBOT DETECTION SYSTEM DESIGN ABOUT FRONT-FOLLOWING TECHNOLOGY PART 2
3. ROBOT DETECTION SYSTEM DESIGN ABOUT FRONT-FOLLOWING TECHNOLOGY PART 3
4. ROBOT DETECTION SYSTEM DESIGN ABOUT FRONT-FOLLOWING TECHNOLOGY PART 4

References:

- Ho, D. M. , Hu, J. S. , & Wang, J. J. . (2012). Behavior control of the mobile robot for accompanying in front of a human.
- Jung, E. J. , Yi, B. J. , & Yuta, S. I. . (2012). Control algorithms for a mobile robot tracking a human in front. IEEE/RSJ International Conference on Intelligent Robots & Systems. IEEE.
- Hu, J. S. , Wang, J. J. , & Ho, D. M. . (2014). Design of sensing system and anticipative behavior for human following of mobile robots. *IEEE Transactions on Industrial Electronics*, 61(4), 1916-1927.
- Moustris, G. P. , & Tzafestas, C. S. . (2017). Intention-based front-following control for an intelligent robotic rollator in indoor environments. *Computational Intelligence*. IEEE.
- Nikdel, P. , Shrestha, R. , & Vaughan, R. . (2018). [IEEE 2018 IEEE International Conference on Robotics and Automation (ICRA) - Brisbane, Australia (2018.5.21-2018.5.25)] 2018 IEEE International Conference on Robotics and Automation (ICRA) - The Hands-Free Push-Cart: Autonomous Following in Front by Predicting User Trajectory Around Obstacles. (pp.1-7)