POLITECNICO DI TORINO

MS in MECHATRONICS ENGINEERING

Thesis

Indoor autonomous navigation for a sanitizing UGV for Intesa Sanpaolo.

Supervisor:
Chiaberge Marcello

Candidate:
Temurov Zafarjon

TABLE OF CONTENTS

ACKNOWLEDGMENTS

The author wishes to express sincere appreciation to Professor Marcello Chiaberge for their assistance in the preparation of this work. In addition, special thanks to Matteo Nazzario whose experience and ideas was very helpful during the programming phase of this undertaking. Thanks also to the members of the school council for their valuable input.

Abstract

The COVID-19 pandemic keeps spreading across the world and, while national governments concentrate on lockdowns and restrictions to mitigate the disaster, advanced technologies could be employed more widely to fight the pandemic. This thesis describes existing robotic solutions that could be employed for pandemic care and presents a systematized description of desired robot properties based on a particular application area and target users. In the first chapter with brief introduction to robots and description of types of sanitizing robots available in the market. The choice was made by innovation center of Intesa Sanpaolo bank members after thorough research is Aris K2 robot. In the following chapters this robot was described in detail. The next section describes all possibilities of this robot for further improvements and obstacles to achieve some of them. Finally, discussion ends by describing map segmentation and optimal path planning for disinfection of scanned areas.

Chapter 1

ADVANCED TECHNOLOGIES TO FIGHT COVID-19

Introduction:

The sudden cause of global pandemic disease disrupts human beings in disease management. Recently, the whole world is facing the COVID-19 outbreak, which has almost deteriorated global health and especially the economy. The COVID-19 is caused by a deadly coronavirus SARS-CoV-2 (Severe Acute Respiratory Syndrome – Coronavirus-2) the cause of the virus is creating a much negative impact and spreading overseas. As of 4th December 2020, about 44,350,473 cases have been reported globally and 1,494,668 deaths were recorded. Approximately, a gross estimate of 25,000 frontline health workers has been infected with this deadly disease across 52 countries (WHO, 2020). However, disease treatment without any drug or vaccine is an additional headache among health workers, since symptomatic treatment is being carried out. Even, some countries are facing the unavailability of frontline health workers or very least in numbers for treating a huge number of the infected population. Another major issue is the inadequate supply of Quality PPE's to health workers, which might enhance disease spread.

In such a tremendous crisis, it's our mission to serve and protect our frontline health workers. This global pandemic disease has taught us to bring-up innovations, and technologies for disease management, and treatment in healthcare systems. Robotics and Artificial Intelligence are promising tools with vast potential to deal with the present situation. As we mentioned above, the major success of this virus is community transmission, which could be minimized or hampered by the use of robots include Medical robots, Drones, UV-light robots, Automated robots, and Collaborative robots in hospitals and associated labs. Moreover, Artificial intelligence could be implemented to monitor every patient, categorize different stages of infection, and find out severe cases for high-end treatment. These were successfully executed in some countries and found fruitful to succumb viral transmission in hospitals and major metropolis. This present study highlighting the futuristic application of Robotics and Artificial Intelligence in pandemic disease management.

Robots in disease management

Robots have been invented to perform complex task or group of tasks which minimizes workload and labor. However, the current standings are ready to defend the cause and loss. In the present scenario, robotics could also be used for pandemic disease management mainly to prevent contact and spread of viral diseases. The design and structure of medical microrobots minimize the physical interactions with the cells of the immune system. However, the surface-borne design and parameters are also critical in the locomotion and performance of microrobots.

Cleaner robots

Cleaners in healthcare systems are playing a major role in maintaining the regular cleanliness of hospitals and health care centers. During the situation like pandemic disease, the demand for such workers is skyrocketing, due to disease threat. To solve this issue, cleaner robots could be employed in health sectors coupled with human intelligence and machines to enhance cleaning efficiency and employee safety. Even, cleaning labor supplying companies are facing obtuse labor shortage during this pandemic disease situation due to self-quarantine or other illnesses. Therefore, their only solution is efficiently employing commercial cleaning robots. Cleaner robots could be an addition to autonomous cleaning solutions to maintain the quality of cleaning and might reduce the risk of exposure to highly pathogenic infections. In health care centers, cleaning standards were established to acquire the safety of both healthcare workers and patients. However, during the COVID-

19-like pandemic situation, several criteria were included to improve the safety standards day by day. Therefore, cleaner robots designed under certain safety standards and regulations in healthcare centers would be effective and helpful. The machines don't cough, sneeze, or shake hands, so they can't actively spread deadly coronavirus around the hospital, in the addition to cleaning robots are floor scrubbers, vacuum sweepers, and shelf scanners in a major metropolis. Automotive mobile robots (AMR) enable us to serve cities in labor-intensive cleaning tasks during the global pandemic situation. For example, UV-disinfecting autonomous mobile robots (Sunburst UV Bots) were first implemented in the Northpoint City mall, Singapore, and consequently followed by thirteen different malls in the city.

Ultra-violet light robots

The spread of coronavirus is attributed to nasal or mouth droplets by sneezing or coughing of an infected person to a healthy person. However, other modes of transmissions are direct contact with utensils by a healthy individual, already touched by the infected person. It has been reported that the viral particles remain to persist on the surface of copper (4h), cardboard (24h), stainless steel (2–3 days), wood (2 days), paper money (4 days), surgical masks outside (7 days), cloth (2 days), glass (4 days), and polypropylene plastics (3 days). Therefore, robot-controlled ultraviolet light devices could be efficient in the disinfection of materials employed in health care centers. Three types of ultraviolet rays have been categorized UV-A (320–400nm), UV-B (280–320nm), and UV-C; hence, UV-C is high energy radiating UV rays ranges between the frequency wavelength of 280–100nm can cause severe skin and cornea damage and may induce carcinogenic effects in humans. Although, UV-C is one of the effective rays in sterilizing objects, the PX-UV (Pulsed Xenon Ultraviolet) device is employed in disinfecting surface materials and objects in hospitals and health care centers to combat deadly viral pathogens. It has been proved that the cleanliness rating was improved from 50th to 99th percentile in hospitals while applying this device. Mainly, in the UV cleaning methods, the robots are used for micro-killing, floor cleaning and able to patrol rooms and corridors performing deep cleans surfaces with concentrated UV light. The robot is employed with many sensors, and its self-driving, voice-enabled machine disinfects microbes with a high wavelength of ray which is dangerous for humans to be exposed to. The robot developed by UVD Robots; the operator deploys the robot using a computer. The robots scan the environment using its lidar sensor and create a digital map of the hospital's room, the automated map indicated the rooms and points the robot should not disinfect. After that, the robot relies on Simultaneous Localization and Mapping (SLAM)to navigate. The autonomous robot was emitting 20 joules per square meter per second of 254-nm light to eliminate bacterial and other harmful microorganisms. The Xenex robots use the xenon lamp to generate bursts of high-intensity full germicidal spectrum (200–315nm), this Light strike robot destroys the virus in two minutes achieving a four-log (99.9 percent) and also reduction in that time. The application of Ultra-violet Robots are used for sanitizing patients room after they discharge from hospitals, for example, Vanora Robotics Company in Mangalore have developed UV light disinfection robot named "Vanora" which can sterilize the entire room in 4min, and was employed in Tejasvini Hospital in Mangalore, this device would be most helpful for frontline workers and workers engaged in the COVID-19 battle.

Mobile robots

Mobile robots are engineered to navigate and transport high-risk patients in any contaminant areas to avoid direct contact with health workers. Mobile robots were equipped with high-end sensors and cameras to record the patient's temperature, blood pressure, and pulse rate, and regular monitoring using a vision algorithm11,12 (Fig. 1). In addition to this, these robots help deliver medicines and food for isolated and quarantined patients in hospitals and health care centers.13

Therefore, we could monitor, study the condition of patients, and categorize the stages of disease under treatment and doctors could attend the patients based on the severity. During COVID-19, mobile robots were most helpful in delivering food and medicines to isolated patients in hospitals to secure the health of frontline workers including doctors.

Artificial intelligence in COVID-19

Till now, the diagnosis of SARS-CoV-2 virus infection is based on RT-PCR (Reverse Transcriptase Polymerase Chain Reaction) from the nasal swab of patients. However, this could not provide the severity of patients infected with the virus, because it takes 6–48h to complete and to get results and also lack of PCR kits to attain samples. Therefore, Computed Tomography (CT) scan is used as a valuable component in assessing the severity of the patients. In this scenario, Artificial Intelligence (AI) could be implemented to assess the stages of infection and severity of cases in a real-time manner. Researchers demonstrated that an AI algorithm could be trained to classify COVID-19 in Computer Tomography (CT) scans with up to 90 percent accuracy. However, the CT is not always recommended as a diagnose tool for COVID-19 because the disease often looks similar to influenza-associated pneumonia on the screens. There are many AI-based algorithms like deep convolutional neural network (CNN) to an initial CT scan, for example, Support Vector Machines (SVM), Random Forest, Multilayer Perceptron (MLP), and Decision-tree Classifier. Moreover, decision-tree classifiers have shown the best performance on the tuning set. Because the accuracies in decision tree classifiers are enormous compared to other models. The Deep-learning-based decision-tree classifier will be designed to integrate the CT scan of the chest, symptoms of patients, exposure history, and laboratory testing to rapidly diagnose and record the different stages of viral infection and inform about the number of cases that need high-end disease treatment. This neural network model is fully comprising of the PyTorch framework, proposed model classifier for detecting COVID-19 form CXR images (Chest X-ray) as normal and abnormal, the normal was a detection coronavirus infection, and abnormal was a sign of tuberculosis. The accuracies of the first and second decision trees are 98% and 80% where the average accuracy of the third decision tree is 95%. Besides, the AI was also used to determine the symptomatic and asymptomatic cases during the disease treatment for their segregated levels of treatment (Fig. 2). Specifically, the Department of Radiology, Shenzhen Second People's Hospital, and University Health Science Center, Shenzhen developed "COVNet" (COVID-19 detection neural network) which was also developed to extract visual features from the volumetric chest CT for the detection of COVID-19. In this model, the collected dataset consisted of 4356 chest CT samples of 3322 patients from 6 medical centers in China between Aug 16, 2016, and Feb 17, 2020. The final dataset has compiles of 1296 (30%) were infected with COVID-19, 1735 (40%) of community acquired pneumonia, and 1325 (30%) of non-pneumonia. All the COVID-19 confirmed cases as positive by RT-PCR were acquired from Dec 31, 2019, to Feb 17, 2020.30 AI has already been used in high throughput medical imaging for deep learning technology. Chest Radiography imaging such as X-ray or computed tomography for diagnosing pneumonia, but also provides a highly sensitive diagnosis with COVID-19. The Automated X-ray Imaging Radiography systems (AXIRs) were used to distinguish patients infected severely with SARS-CoV2. Finally, doctors review the images and their classifications before making decisions further for clinical assessment. In the "COVNet" detection method the imaging system was performed for different standard imaging protocols. Therefore, it could also be applied to study and detect viral pneumonia infection in chest radiographs. The two major limitations of the use of Chest CT. First, the health systems during an epidemic may be overburdened, which may limit the timely interpretation of the CT by radiologists.

Second, the morphology and severity of pathologic findings on CT are variable. In particular, mild cases may have few if any abnormal finding on the chest CT.

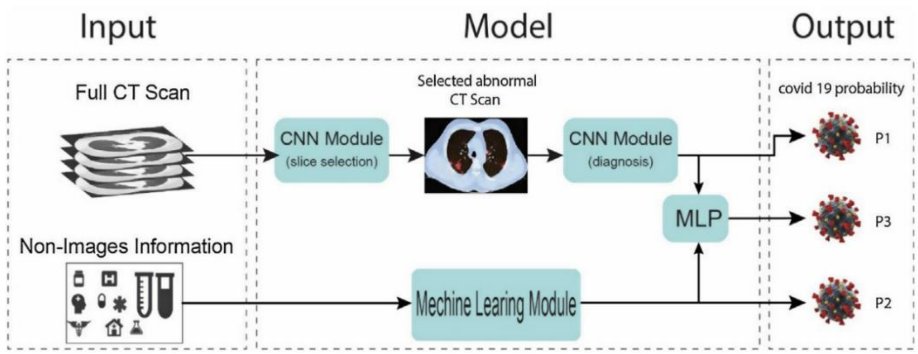


Figure 1: Artificial intelligence is used to enhance the GT scan analysis to detect and report different stages of infection.

Applications of artificial intelligence in COVID-19

The AI-based COVID-19 scan analyzer was installed in Baguio General Hospital in the Philippines to support doctors to segregate COVID-19 patients by analyzing CT scans of patients very quickly.32 Similarly, researchers at Brunel University, London have developed an AI algorithm that scans the images of health reports of patients to identify COVID-19 patients for quick monitoring and treatment. As a powerful tool, AI was implemented to identify highly infected zones and moderately infected zones in New York City by Mount Sinai Hospital. Consequently, in collaboration with Hospitals in China, they have compared the CT scan images of patients with China using AI, which was fruitful to identify COVID-19 patients most efficiently with 84%, which was only 75% by radiology-based evaluation.

Chapter 2

WHY ARISK2 ROBOT?

According to analysis was done by Intesa Sanpaolo innovation center there are startups that focuses to fight against infectious diseases. These companies are listed below:

- MILAGROW
- XENEX disinfection services
- UVDROBOTS infection prevention
- YOUIBOT
- AKARA

Figure 2: List of companies that was analyzed.

Companies of interest that emerged from the analysis of CbInsight's "Infection Disease Collection" that deal with the containment of infectious diseases (ID) and the sanitation of environments through service robotics solutions during periods of epidemics.

YOUIBOT

Youibot Robotics is based in Shenzen and develops autonomous and interactive service robots for a variety of industrial and home environments with intelligent mapping and navigation, voice recognition, facial recognition and robotic arm control based on its ROS (robotic operating system).

The ARIS-K2 robot is equipped with an ultraviolet anti-virus lamp and thermal imaging camera to detect body temperature. ARIS-K2 is able to autonomously disinfect the structure during the evening when the workers are not present. It takes 150 minutes to clean 1,000 square meters.

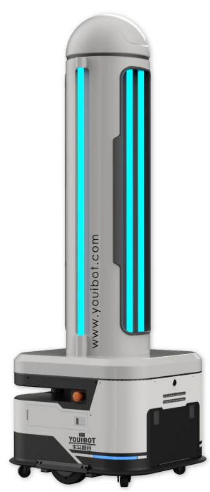


Figure 3: Aris K2 robot from YOUIBOT.

XENEX

Xenex Disinfection Services is a supplier of UV disinfection systems for healthcare facilities. Xenex's pulsed xenon UV environmental disinfection system is a device used for advanced cleaning of healthcare facilities. Thanks to its speed and ease of use, the Xenex system integrates seamlessly into hospital cleaning operations.

The LightStrike Xenex Robot is not a self-contained mobile robot, as the device is mounted on caster wheels and has to be manually placed by hospital staff, however the company describes it as a robot since the disinfection process is automated. The devices cost US $ 125,000 per piece.

The solution was adopted by the Verduno and Alba hospitals to support the fight against Covid-19.

Figure 4: LightStrike Xenex Robot.

UVD ROBOTS

UVD Robots was founded in 2016 by Blue Ocean Robotics with the aim of commercializing robot-based UV disinfection solutions for hospitals worldwide.

It made headlines earlier this year with the sale of more units in China and they are committed to delivering robots as fast as possible.

The UVD robot is used as part of the normal cleaning cycle and aims to prevent and reduce the spread of infectious diseases, virals, bacteria and other types of harmful organic microorganisms in the environment by breaking down their DNA structure.

It is designed to be easy to use and is designed to be managed by the staff who take care of the daily cleaning.

The company indicates it takes 10 to 15 minutes for a UVD robot to disinfect a room.

Figure 5: UVD robot.

AKARA

Akara Robitcs develops Steve the Robot, a socially assisted robot designed to help people stay independent as they get older. The robot keeps people cognitively stimulated with quizzes and games and helps residents stay socially connected.

Akara Robotics has been busy over the past month designing, testing and manufacturing a new AMR designed primarily to sterilize rooms with UV light. The company is a start-up related to Trinity College Dublin.

At this time, the company is awaiting the results of microbiological tests to confirm the effectiveness of the proposed solution.

Figure 6: AMR robot.

To sum up, Aris K2 was chosen for sanitization of Intesa Sanpaolo skyscraper. Because this robot is completely autonomous compared to others competitors and it runs on robot operating system which is open source and could be developed further to use maximum potential of this robot.

CHAPTER 3

ARIS K2 guide walkthrough.

The main power supply to the robot is through air switch, make sure that it is always on **ON** state. And emergency button is not pressed, red lights indicate emergency state. In case it is pressed unscrew it in clockwise direction. Long press green button for 3 seconds to turn this robot on.

Figure 7: Robot EM state, buttons and ports, air switch.

While pressed green button one should hear the relay is turned on and a noise loosening the power button. The robot starts to power up and initializes the system. At this point, the indicator at the bottom of the robot is red.

Initialization process of the system lasts about 50 seconds. After initialization, the robot indicator light is green.

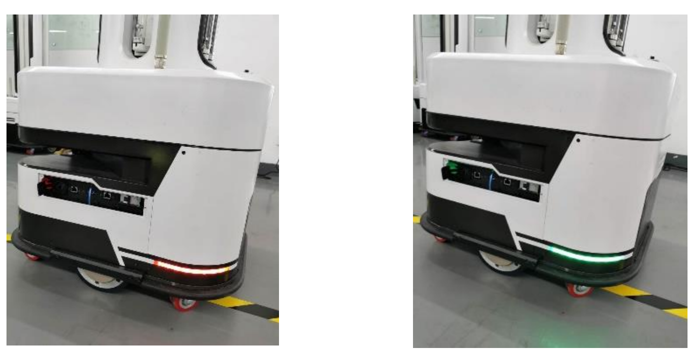


Figure 8: Robot states.

Connection to robot's WIFI

After the robot initialization is completed, the WiFi hot spot is automatically launched. Search for WIFI with name as serial number of a robot and connect to it using default password: "robotrobot".

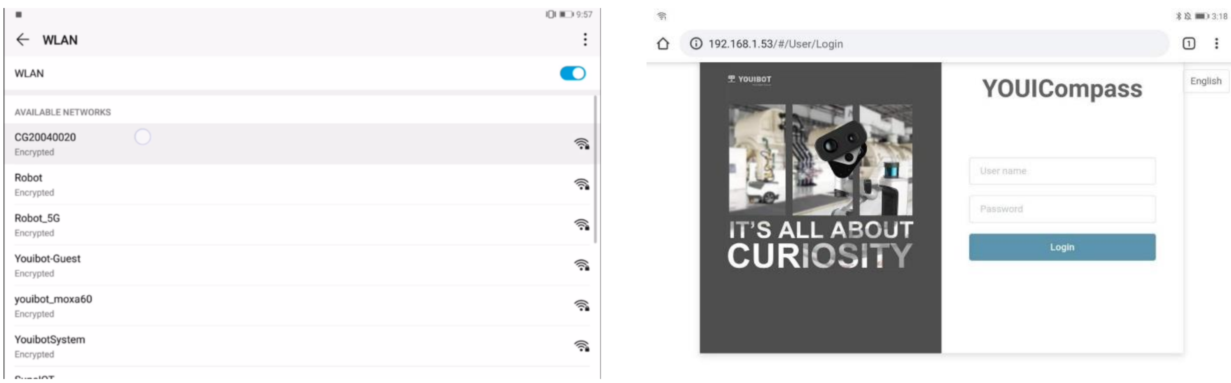


Figure 9: Wi-Fi connection and browser login page.

After successful connection open browser, enter the robot IP address (look to the

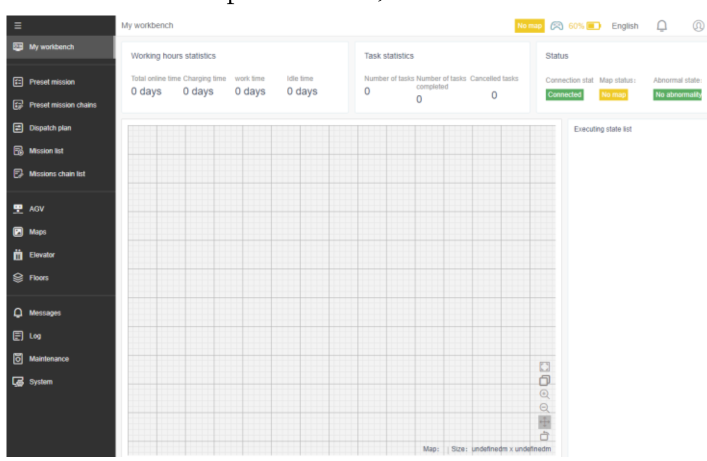


Figure 10: Robot scheduling interface.

sticker in robot) to open the scheduling system login interface.

Manual control of the robot

Manual control of the robot has two ways, the first is to use joystick control (F710). The second is controlled by a virtual handle in manual mode in the scheduling system from browser.

Control with F710 handle

Insert the joysticks blue-tooth receiver into the USB port on the robot panel (the joystick's receiver is hidden in the back cover of the joystick or has been inserted on the panel), press the mode button, the handle starts automatically, and the indicator starts to flicker until the handle indicator lights up or out. Indicates that the handle is connected successfully. Ensure switching to D gear.
The maximum forward speed is 0.8 m/s, the maximum turning speed is 0.8 rad/s. using the handle remote control robot

Remote control robot movement method:
And when the Mode lights are on, no need to press and hold "X" button since low speed mode is turned on by default. With your right-hand press and hold RB key. To move around use pad arrows or left analog stick.
To move around in highspeed mode, with your right hand keep holding RB button and red "B" button. With your left-hand using pad arrows or left analog stick move around in higher speed. For checking the movement, short press once any arrow pads with left hand and after that one can notice the movement.
Important to note that both hands left and right should work together for normal functioning otherwise the robot enters the mobile locking state. It was done to prevent accidental movement of a robot.

And when the Mode lights are off,
The left-hand operation can be replaced by the left analog stick with higher degree of freedom.

Press the BACK button to turn on the disinfection lamp. Press the START button and the disinfection lamp is off.
Note: UV lamps are harmful to people's eyes and skin, please protect your eyes and skin!。
When the robot cannot move, it is in an emergency stop state, press LB button + right RB button for two seconds at the same time to release the emergency stop.

Control by manual mode in a dispatch system

1. **AGV →Control Mode →Manual Mode**

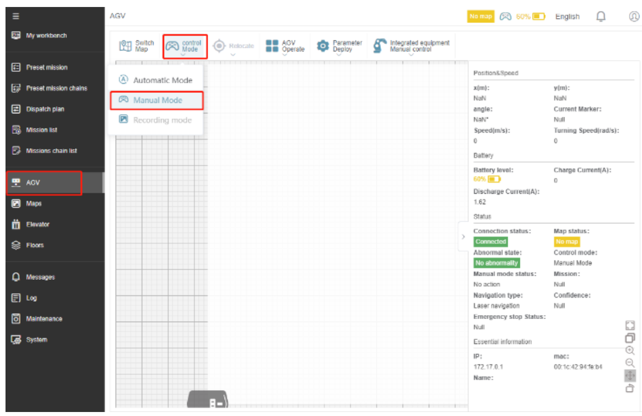


Figure 11: How to access manual mode.

Then select desired speed mode and after using arrows of PC's keyboard control robot.

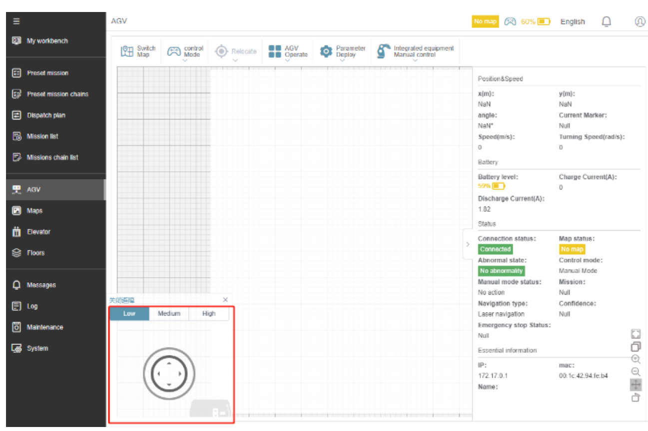


Figure 12: Manual mode interface.

The robot needs a map to run the task. This part mainly introduces how to create the map for efficient working.

New Map

Steps:

•Click Map ⟶ New Map

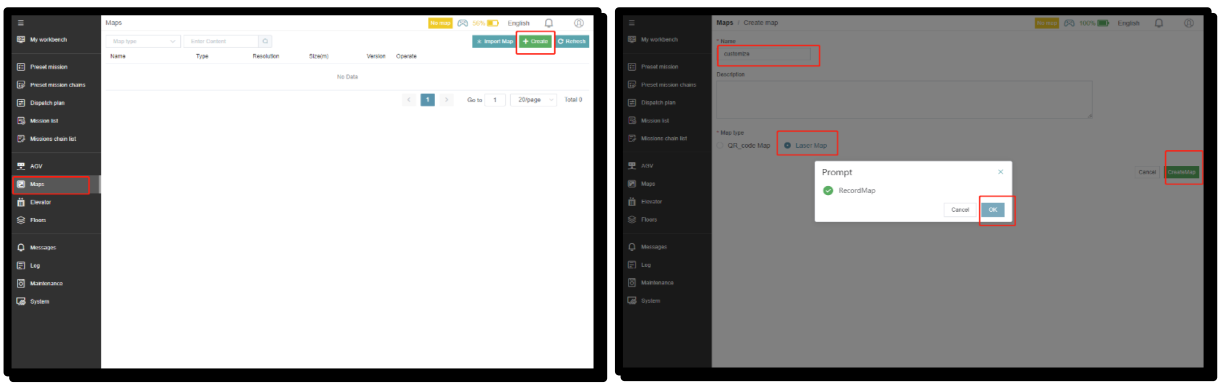


Figure 13: Map creation steps.

There are two methods to scan a map, from LIDAR sensor and QR code, by default Laser_map is selected. In case the map is already available in QR code version printed and set so that the laser can scan, you can choose QR_code_Map.

Map recording

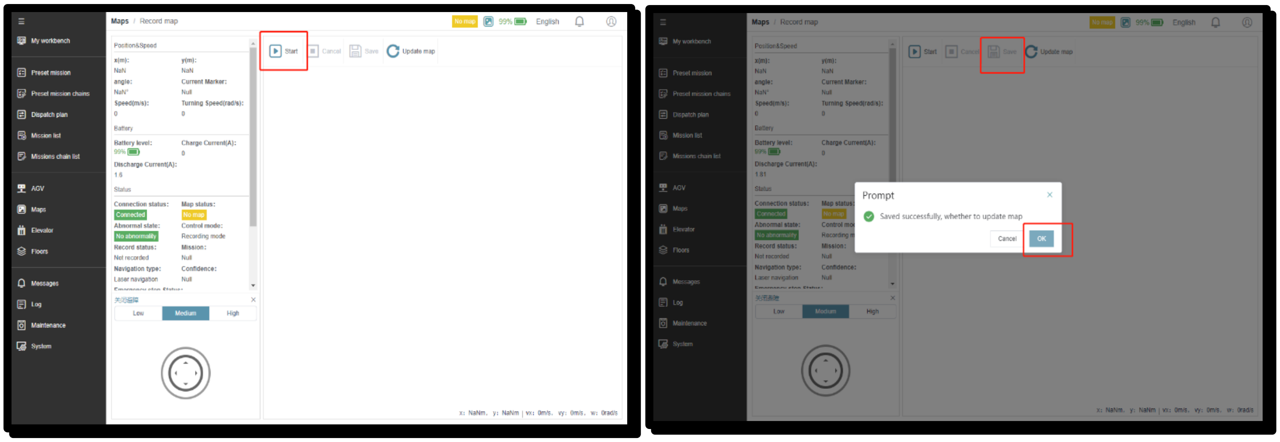


Figure 14: Map recording and saving.

The system will automatically pop up a map recorded by the user. If you have a map in json format it is also possible to upload and use it or save current map.

Environmental requirements

Because of some specific physical properties of materials in nature, the robot with lidar as navigation mode will have some environmental restrictions in use. Here are some common environments that are not suitable for robot operation. Please avoid these situations or deal with them accordingly.

(1) Material in the environment

Materials in the following environments can affect robot mapping and positioning navigation:

● the robot's lidar cannot correctly scan the surface of the mirror reflected objects (such as mirrors, smooth metal surfaces, etc.)

● the robot's lidar cannot correctly scan transparent or translucent objects (e.g. glass, partial glass)

● the robot's lidar cannot correctly identify the black object, black will absorb all the light, so that the effective distance of the lidar can be reduced or even completely invalidated.

Above three kinds of material such as mirror, glass or black object, can be glued on its surface 3 M reflective strip, wool glass film, assist lidar to perceive the environmental profile.

(2) Outline of the environment

The following environment will have an impact on robot mapping and positioning navigation:

● a very open environment, the range of large open areas is larger than the effective distance of lidar (40 meters), that is, the robot can hardly scan the effective environmental characteristics in these open areas.

● very messy environment, the robot cannot get a good environmental profile.
Solution: clean up the environment and remove useless objects.

● there are many areas of extremely similar environment, the robot cannot distinguish the position according to the outline of the environment.
Solution: randomly add some characteristic markers to the environment.

● there is no characteristic corridor environment, there is no reference or reference on both sides of the corridor, so it is impossible to obtain effective features.
Solution: randomly add some characteristic markers to the environment.

● the environment that is easy to change, the environment that will change greatly in a short period of time will make the robot unable to build the normal map or make the constructed map lose its referential property.
Solution: avoid or reduce large changes in the environment.

(3) Topographic requirements

The ground is flat as a whole, without potholes and frequent ups and downs. Laser navigation is not suitable for uneven ground, recommended and office leveling off the ground. cannot run on carpet.

Unsuitable Environment Example

Figure 15: Corridor longer than lidar scanning range, and AGV two sides of the travel without obvious feature reference.

Figure 16: Stainless steel from airport baggage conveyor belt, radar can't correctly scan surface mirror-reflected objects.

A pothole environment. For this kind of environment, when setting the path of robot navigation, we can bypass the road with poor road condition. If the road condition is very dense, laser navigation is not recommended.

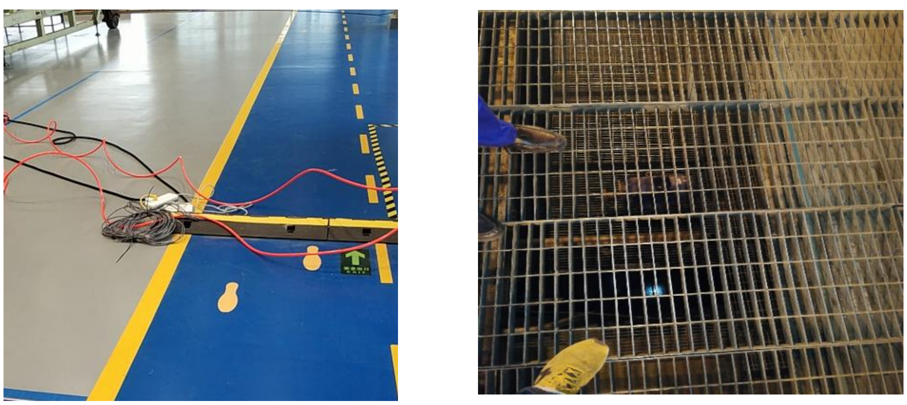


Figure 17: Temporary cables, wire slot or detour.

Suitable environment example

1. The position of the equipment is fixed. The scene changes little.
2. Maintenance of personnel with knowledge of robot operation.
3. The ground flat and clean.
4. Environmental characteristics are obvious and will not be particularly open (larger than the radar effective distance) or particularly narrow (less than 1.5 times the maximum width of the robot fuselage).

Drawing Method of Manual Remote-Control Robot

Handle remote control please refer to the second chapter "connect robot interface" the second section "manual mode control robot ".
Recording Map Interface Manual Remote Control

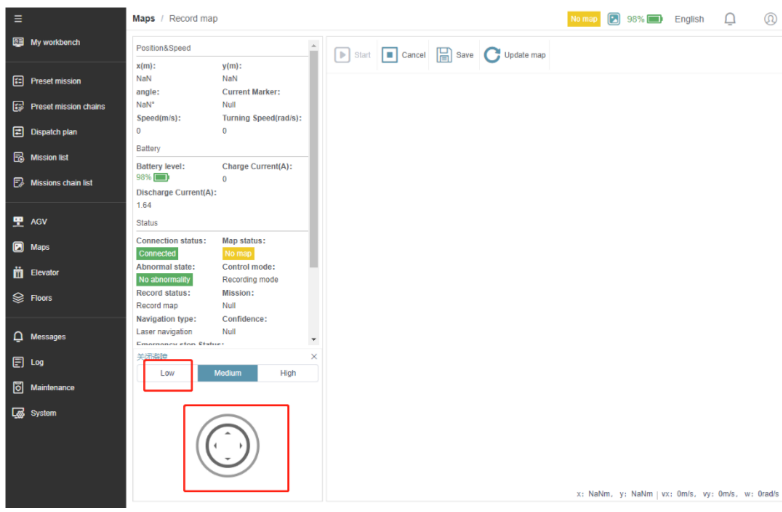


Figure 18: Manual control from PC keyboards.

Try to make the robot's front direction parallel to the wall before building the map, so that the map is more regular. The robot moves smoothly during scanning. Midway let the robot turn in situ, so that the radar can scan the surrounding objects. Do not move back and forth in a small area, which accumulates radar scan errors. At the end of the drawing, we need to go back to the starting point and rotate the robot twice, so that the robot can automatically eliminate the scanning error on the way. Control the robot to move slowly, as far as possible straight line, do not over-adjust the direction during the movement. Click Save and wait to save successfully.

Check maps

Please make sure the outline of the map is clear. There should be no ambiguity in the outline below. A clear map helps the robot to locate and navigate accurately.

If the following situation occurs, build the map should slow down as far as possible, in the fuzzy area more in situ rotation several times.

Editing maps

After the map recording is completed, do some editing.

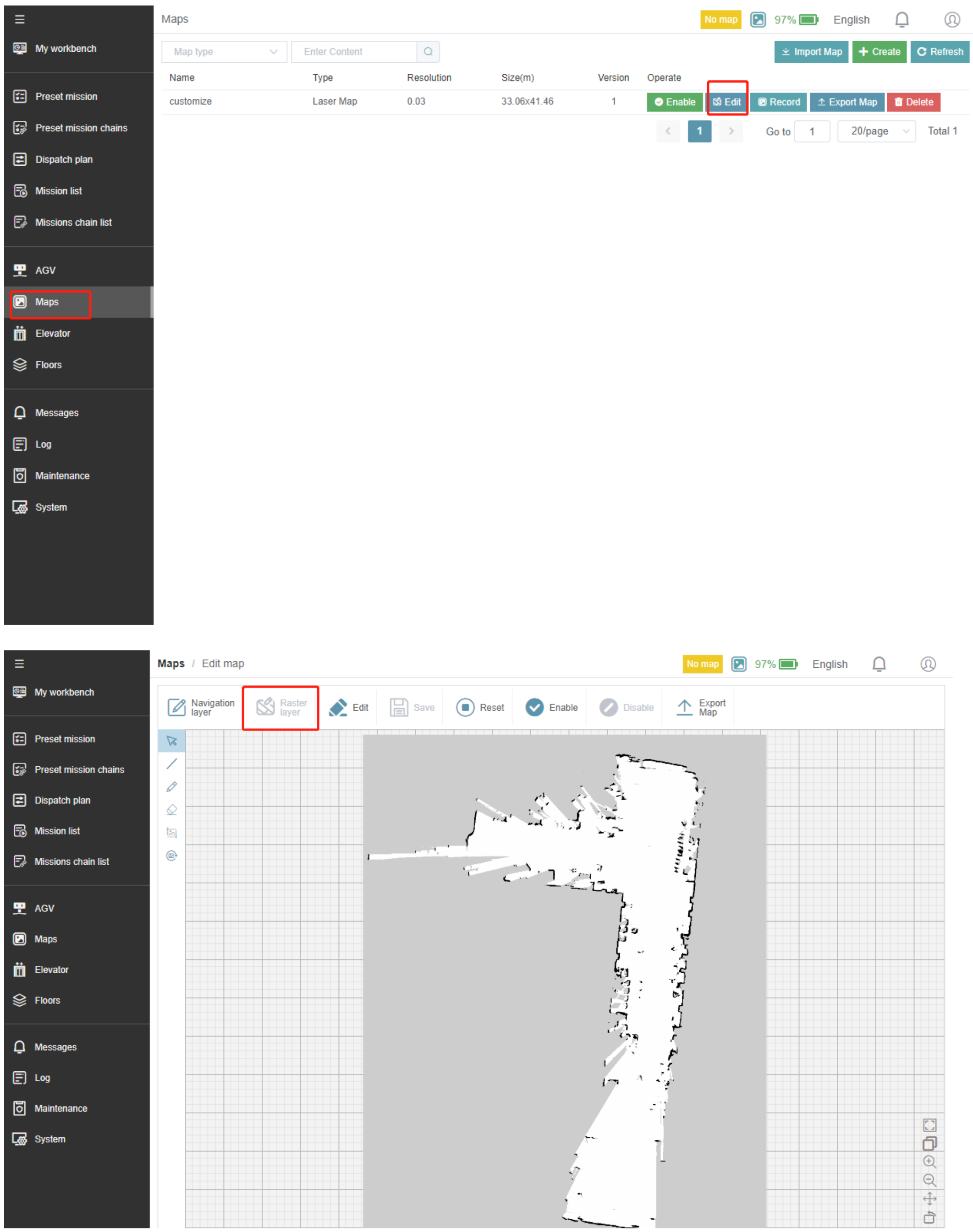


Figure 29: Map editing.

Menu bar description

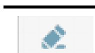
Record a map and edit navigation layer. Plan marker and path resynchronization to robot.

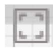
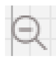
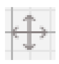
Table 1: Edit Map Page button instructions

Menu	Note
导航图层	Switch to the navigation layer editing interface.
栅格图层	Switch to the grid layer editing interface.
修改名称	Edit the map name.
保存	Save the edit results of the map.
重置	reset the map to the state before editing.
启用	Map enabled.
禁用	Disable maps.
导出地图	Export the map and download the map package.

Canvas menu

Table 2: Illustration menu button operation instructions

Operation button	Note
	Edit the element, select the element, the upper right corner appears the button, click to open the element properties edit page.
	Delete the element, select the element, the upper right corner will appear the button, click delete element.

	Full screen button, switch map to full screen.
	Reset button, map reset to initial state.
	Zoom in button, local enlarge map.
	Shrink button, local shrink map.
	Move button, move map.
	Rotate button, rotate map.

Navigation Layer Menu

Table 3: Navigation Layer Menu Description

Menu	Note
Select element	If you want to select the elements of the layer, you need to switch to element selection mode. Note: drag the element to change the position of the element. Double-click the icon to open the configuration page of the element.
Marking points	Navigation point: the connection point of the robot's navigation path. Work point: robot work site, generally used for robot work site. Docking parameter setting, the robot can docking with other equipment at the working point. Standby point: the standby station of the robot. Charging point: robot charging point, requirements set docking parameters and charging type. Adjustment point: the attitude adjustment point of the robot. The attitude of the robot can be adjusted at this point. Elevator point: when the robot needs to navigate across the floor, it needs to create the elevator point on the elevator, and take the elevator to realize the cross floor
Curve path	Bidirectional curve: the robot can move in both directions on the path.

	One-way curve: determines the direction of the path according to the starting point and end point of the user line. The robot just moves in one direction. Note: Bessel curve, the curvature of the path can be controlled by anchor point. The robot uses curves more smoothly and efficiently when turning.
Linear path	Bidirectional straight line: indicates that the robot can move in both directions on the path. One-way straight line: determines the direction of the path according to the starting point and end point of the user line. The robot just moves in one direction. The junction line of two straight lines, if there is an angle, the robot needs to stop and adjust the direction before moving.
Single machine operating area	There is only one robot in the area at the same time. Other robots that need the area need to wait.
Bulk movement	Select multiple elements to operate at the same time.
Delete element	Click Delete element.

Grid Layer Menu

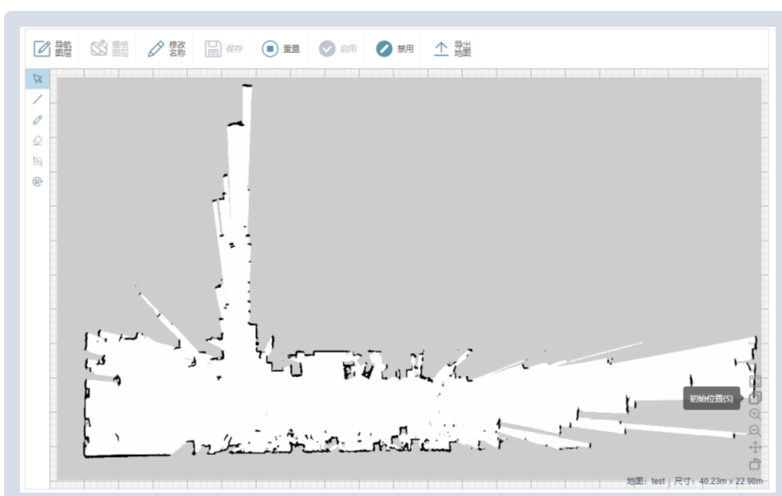


Figure 19: Grid layer edit page.

The grid layer is a robot-recorded map consisting of black, white, and gray pixels. Black represents walls or fixed obstacles, white represents areas that can pass through, gray represents unknown areas, and after map recording, there are many noises caused by other devices or people moving. We need to use a skin wipe to clear these noises.

Table 4: Description of Grid Layer Editing Function

Menu	**Note**
Line tool	Draw straight obstacles on the map.
Pencil tools	Draw non-linear obstacles on the map. The user uses the mouse to draw.
Eraser	Can clear black and gray pixels, into white pixels.
Screens	Keep the selected area of the tool and remove other unwanted areas.
Rotation	Rotate the map direction.

Adjusting the Map

If a bit is set on the map, do not adjust the map. Adjust the time of the map, should be before the establishment of the point, after the establishment of the map.
Steps:
1. click the rotation map button, through the mouse click drag map can rotate the map, the map direction.

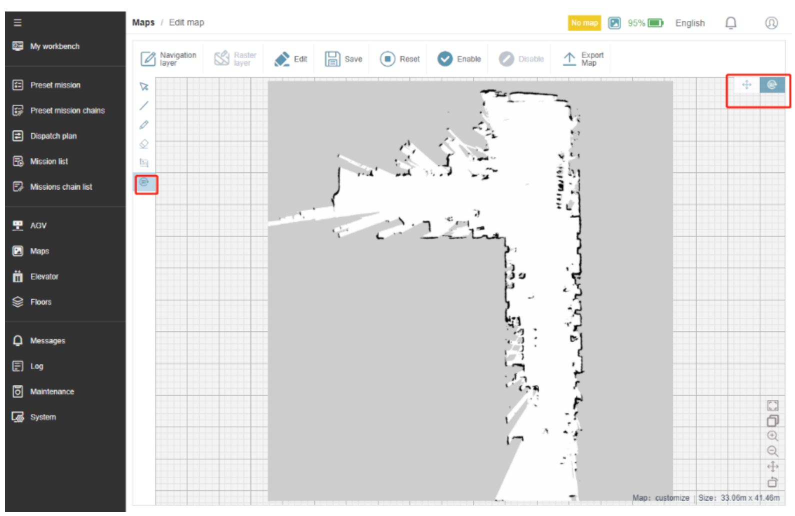


Figure 20: Step 1 illustration.

2. According to the actual demand, if the noise on the map is usually caused by moving objects, use the erase map tool to erase.

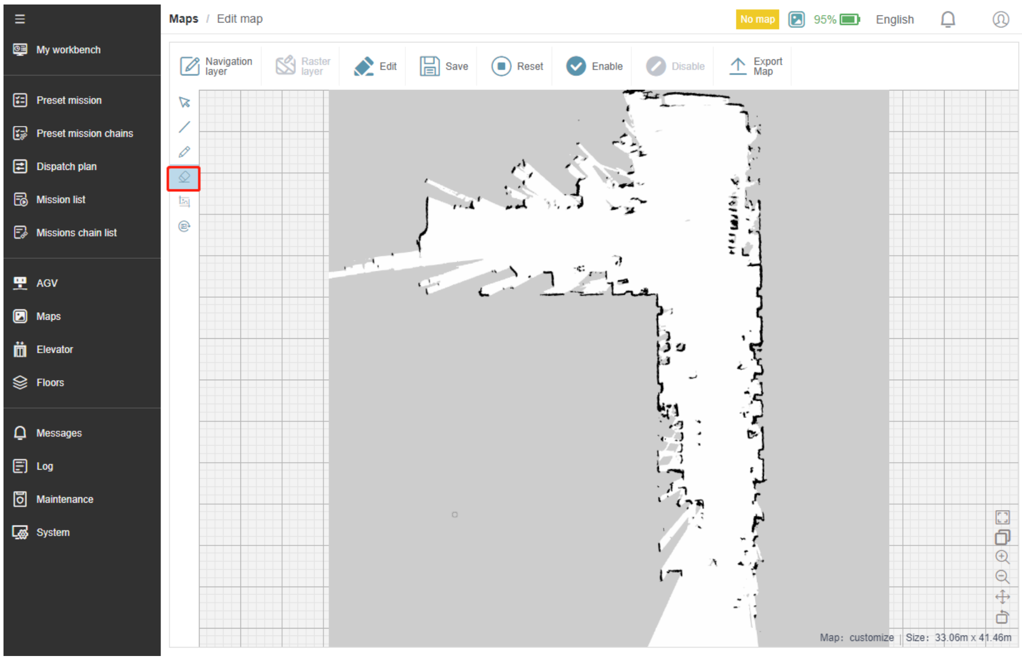


Figure 21: Step 2 illustration.

3. If there are obstacles not scanned to can be drawn with drawing straight walls or hand drawing wall tools.
Screenshots, save only the required part of the map.

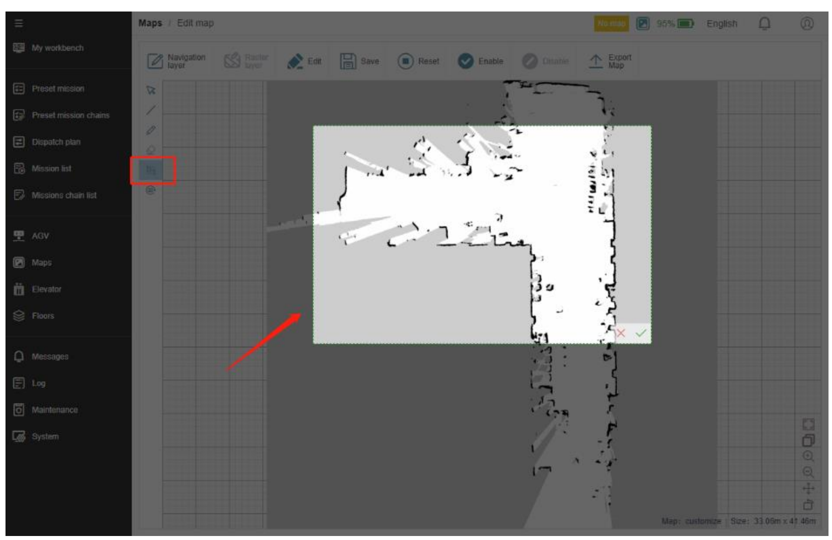


Figure 22: Step 3 illustration.

5. Remember to click Save to take effect.

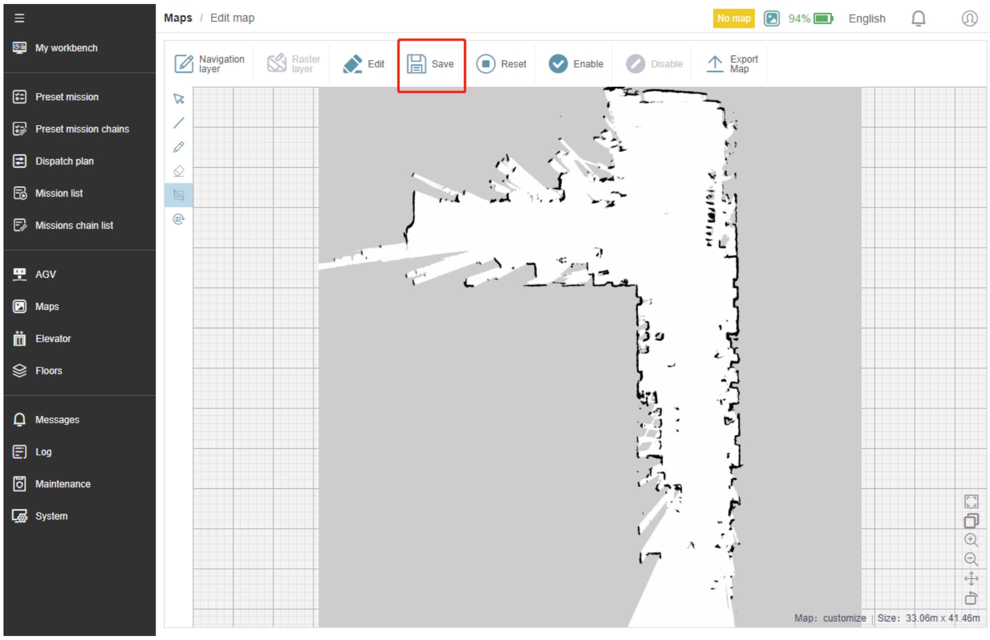


Figure 23: Saving changes made.

Map enabled

Map list

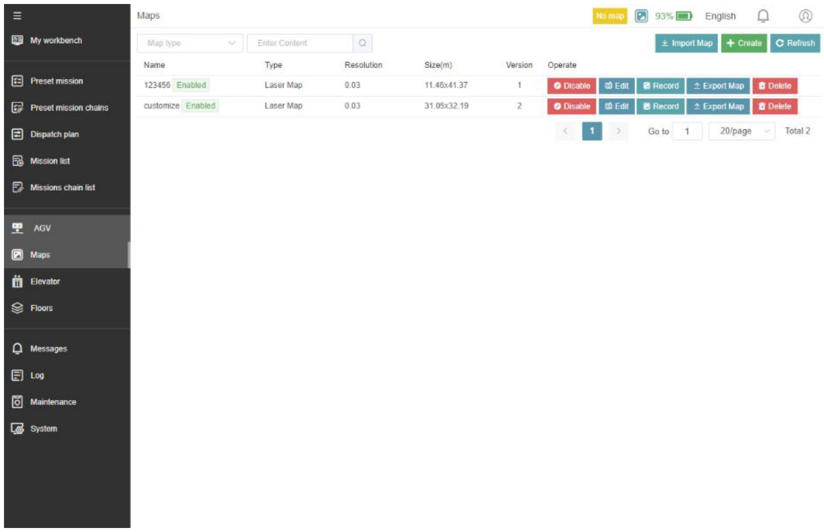


Figure 24: Map lists and control.

Display the map information saved by the current system, can be switched according to different venues.

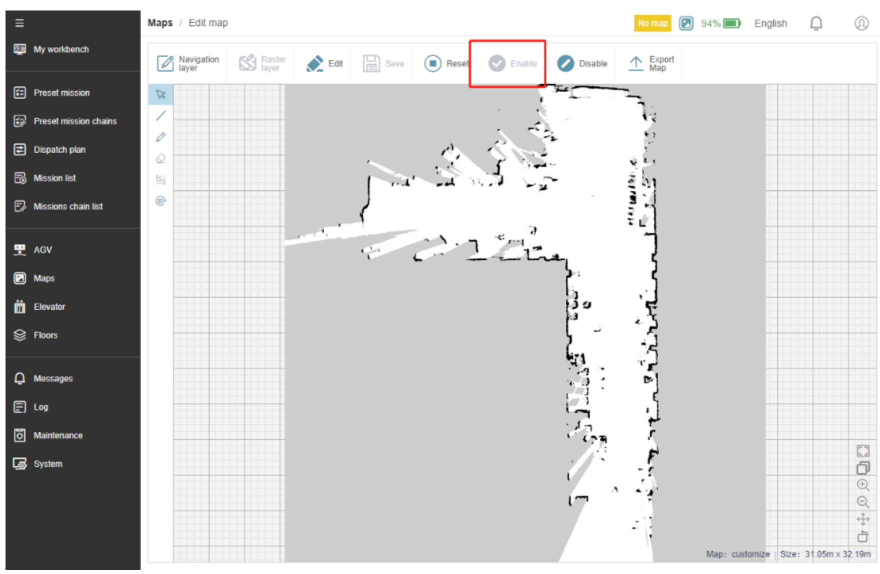


Figure 25: Enabling a map.

Relocation

Introduction

Before relocation, it is necessary to ensure that the robot has established the map of the area and synchronized the map. The purpose of relocation is to give the robot an accurate initial position, both manual and automatic.

Manual relocation

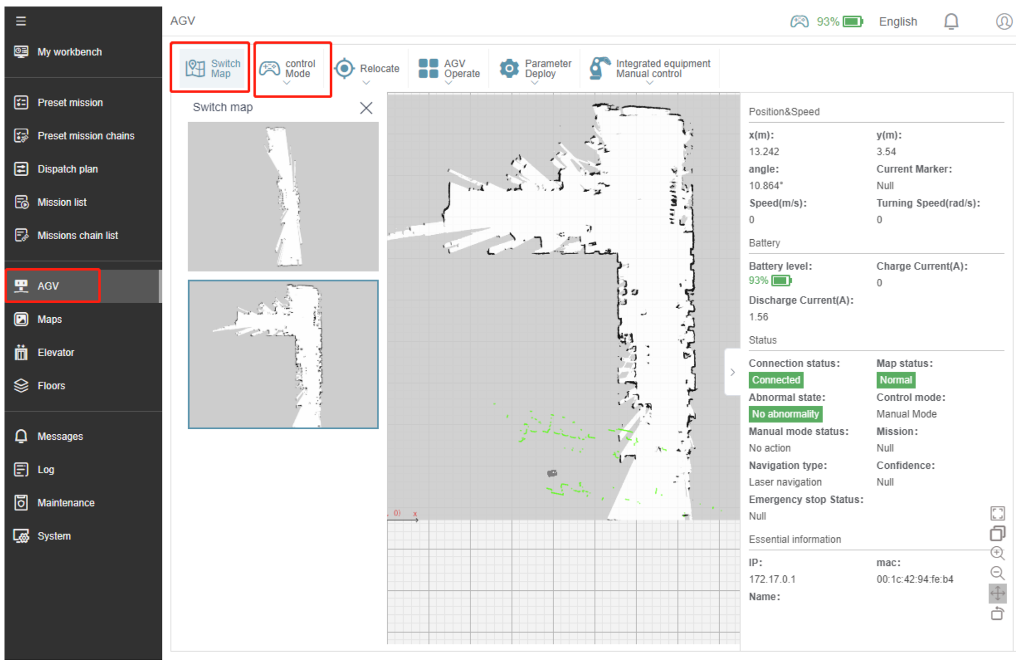


Figure 36: Manual relocation illustration.

1. Click AGV Interface
2. Turn the robot to manual mode (Control mode)
3. Then select a map (in the case of multiple maps)

Figure 47: Relocation illustration.

Select the manual relocate button to judge the general orientation of the robot. It is suggested that the robot be moved to the position with obvious reference, click and drag the direction of the robot, so that the virtual robot on the map is roughly the same as the actual position direction.

1. Manual repositioning: manual remotely control robot, in situ rotating robot, green line will slowly overlap with black line. until the green line and the black contour line on the map exactly match. The green line represents the contour information perceived by the lidar, and the black contour line represents the stationary obstacle in the scene. At this point, the robot has been repositioned successfully.

2. Automatic repositioning: first manually reposition the robot icon on the map so that it is close to the real robot within 2 meters of the location on the map. Click on automatic repositioning, the robot can calculate its accurate position on the map.

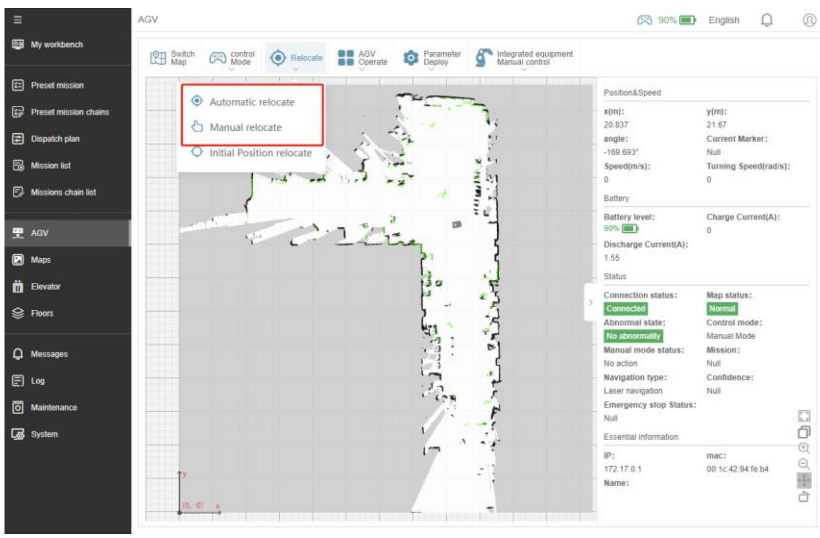


Figure 58: Automatic relocation. For better performance try both relocation methods several times.

Recording Points

Introduction

The recording point can record the navigation point, charging point, standby point and so on while moving the robot, and can quickly deploy the robot.

Add points to AGV

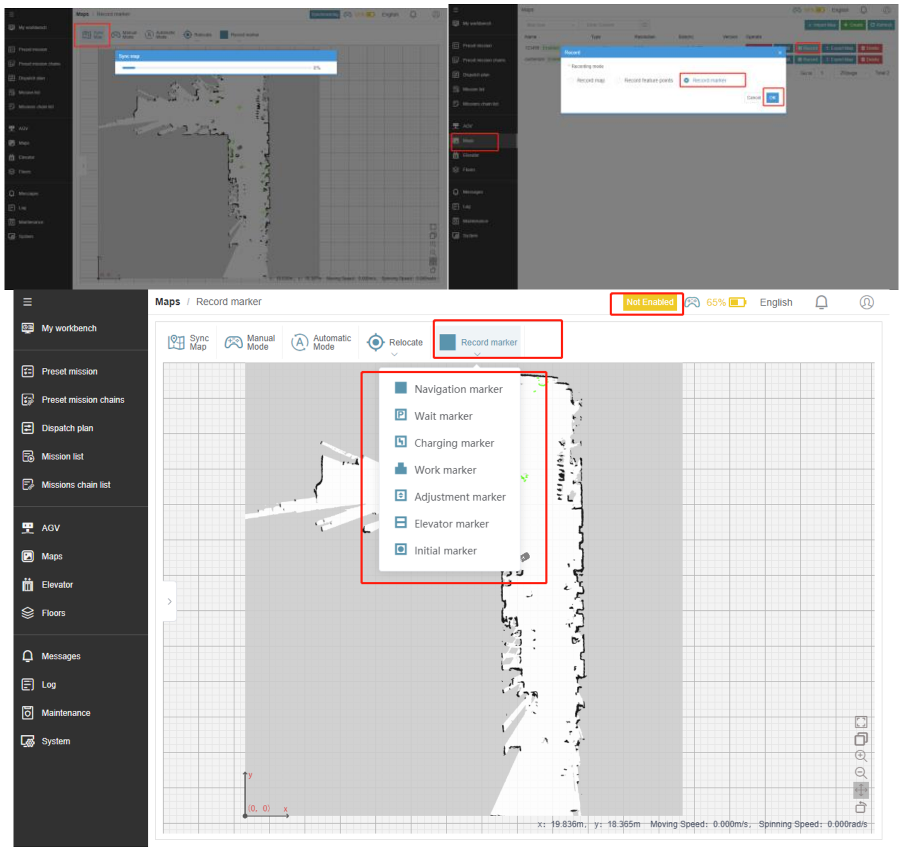


Figure 29: Adding points to a map.

icon	Note
Synchronous maps	Synchronize the current map to the robot.
Manual mode	The working mode of the switching robot is manual mode. In manual mode, the user can manually remote control the robot to move.
Automatic Mode	The working mode of switching robot is automatic mode. In automatic mode, the robot can execute the task independently. When switching automatic mode, such as path navigation tasks must move the robot to the site or path.
Automatic Repositioning	The position of the robot on the map may deviate from the actual position, and the robot calculates the exact position according to the surrounding environment
Manual repositioning	After the first synchronous map is given to the robot, the position of the robot on the map may deviate from the

	actual position. At this time, the robot needs to be given a general position on the map according to the actual position of the robot. Reset the positioning state of the robot.
Initial point repositioning	The robot body can be moved to the initial point of site calibration. Click on the initial point reposition to reset the position and direction of the direct virtual robot to the initial point.
Navigation points	Add a navigation point to the current position of the robot, which is the position used by the robot navigation.
Work points	Add a working point to the current position of the robot, which is the position of the robot working site.
Standby point	Robot waiting for the command to send the position, intelligent standby.
Charge point	Charging pile in front of 50 cm position, intelligent charging.
Initial point	Fixed calibration position of robot repositioning, initial point repositioning.
Delete	Click the delete button and then click on the map point to delete it, this operation is irreversible, please be careful.

Synchronizing maps while disabling maps
Manually manipulate the robot by adding points to the robot
Note: multiple clicks on the point icon leave multiple points in the same location and need to be deleted.

Add points directly to the map

Editing map

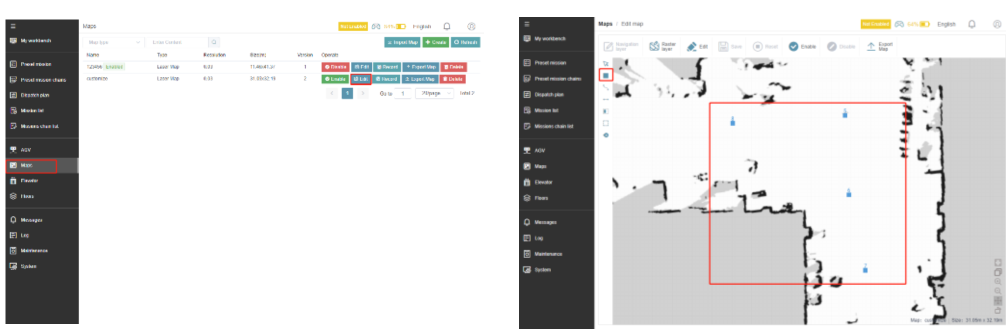


Figure30: Click directly on the map blank space to add points.

Add path

Draw a straight line

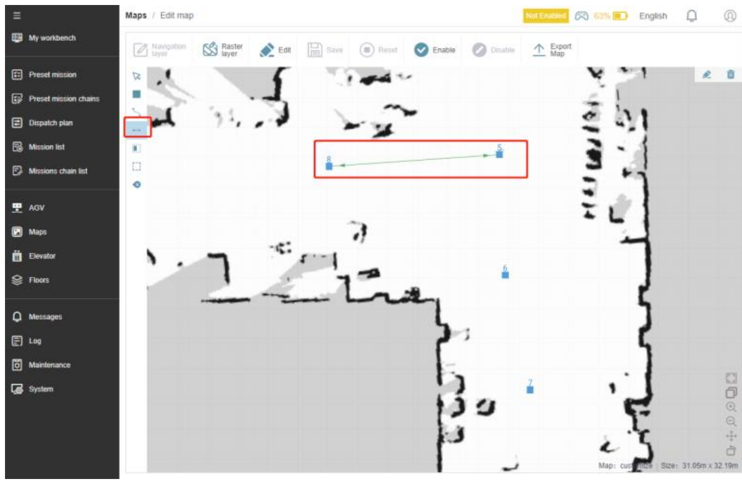


Figure 31: Straight line adding.

Drawing curves

First select the curve path tool, by clicking to connect two navigation points, you can draw a curve path.

Figure 32: Curve drawing.

Click on the selected tool and click on the line segment to appear two anchors. Drag the anchor to set the Bessel curve.

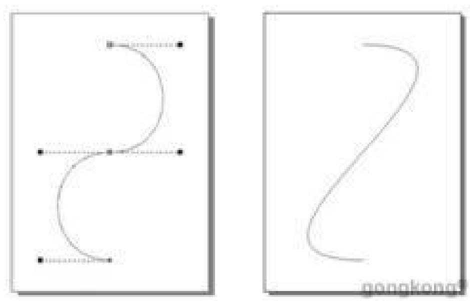


Figure 33: Bessel curve.

Bessel curve, also called Bez curve or Bezier curve, is a mathematical curve applied to two-dimensional graphics applications. It consists of line segments and nodes, which are draggable fulcrum, line segments like stretchable leather tendons, by controlling the four points on the curve (starting point, termination point and two separated middle points) to create and edit graphics. Using the Bessel tool, whether it is a straight line or a curve, is very simple and readily available.

Edit path parameters

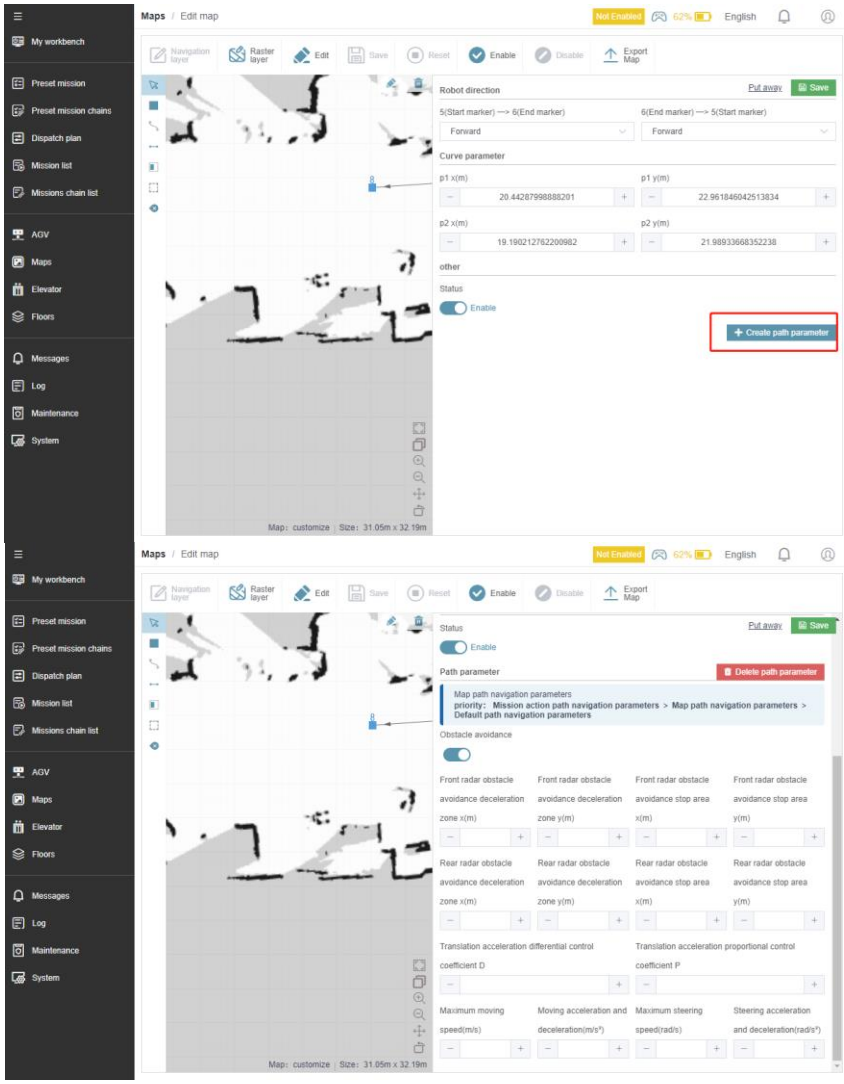
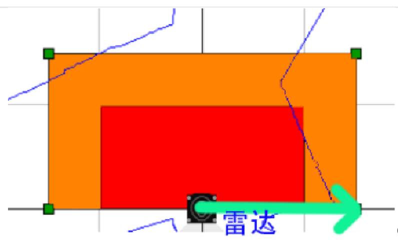
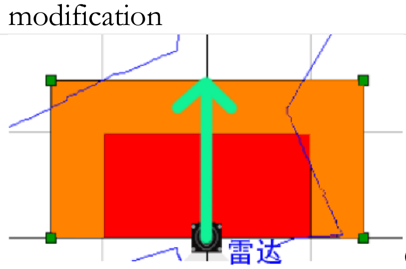
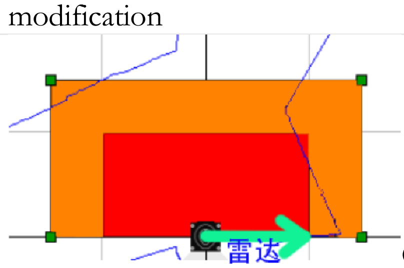
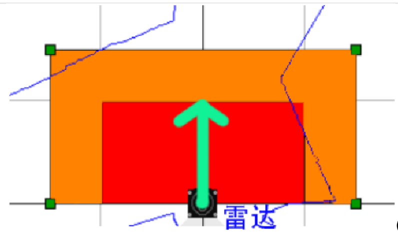


Figure 34: Path parameters settings.

Using the selection tool, double-click the line segment to pop up the path editing menu, click create path parameters to set the Bessel curve properties, including motion direction, maximum speed, maximum acceleration, maximum deceleration, maximum rotation acceleration and so on. Users can accurately set the robot path navigation and motion parameters.

icon	Note
Robot running direction	Set whether the robot is walking or backward on the line segment
Avoidance of obstacles	On/off robot active obstacle avoidance switch

Average acceleration differential control coefficient	Exercise control PID parameter adjustment
Average acceleration proportional control factor	Exercise control PID parameter adjustment
Maximum translation velocity (m/s)	The maximum speed at which the robot moves along the curve
Average acceleration (m/s ²)	The average acceleration and subtraction of the robot moving along the curve
Maximum rotation speed (rad/s)	The maximum rotation speed of the robot moving along the curve
Acceleration and subtraction (rad/s ²)	The rotation and acceleration of the robot along the curve
Forward radar obstacle avoidance deceleration zone x(m)	 General no modification
Forward radar obstacle avoidance deceleration zone y(m)	 General no modification
Front radar obstacle avoidance stop area x(m)	 General no modification

Front radar obstacle avoidance stop area y (m)	 General no modification
Back radar obstacle avoidance deceleration zone x(m)	The schematic diagram is the same as above, and the rear radar is set according to the actual number of radars
Back radar obstacle avoidance deceleration zone y(m)	The schematic diagram is the same as above, and the rear radar is set according to the actual number of radars
Back radar obstacle avoidance stop area x (m)	The schematic diagram is the same as above, and the rear radar is set according to the actual number of radars
Back radar obstacle avoidance stopping area y(m)	The schematic diagram is the same as above, and the rear radar is set according to the actual number of radars

Configuration Tasks

Introduction

This part establishes a robot killing task through practical simulation, and demonstrates how to create the task, which is divided into fixed point killing and mobile killing.

Preconditions

Build a clear map of the kill area, record each point that needs to stop and plan the path.

New tasks

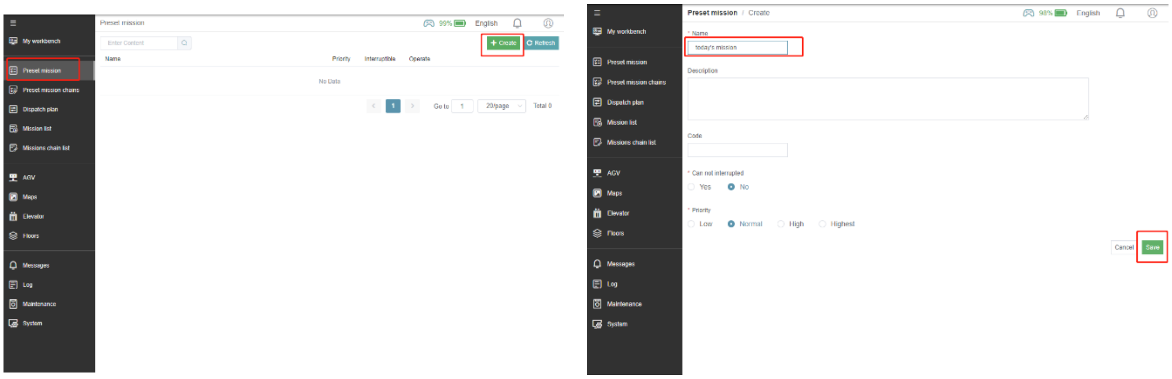


Figure 65: New tasks adding.

Enter the task name: my mission (customer custom), can interrupt the selection of no, map selection created map, others can use the default do not need to fill in, click OK to complete the creation task.

Choose the task just created called "my mission ", click on the action, and add the action the robot needs to complete in this task. A task can consist of many actions, and the robot executes all the actions under the task in sequence.

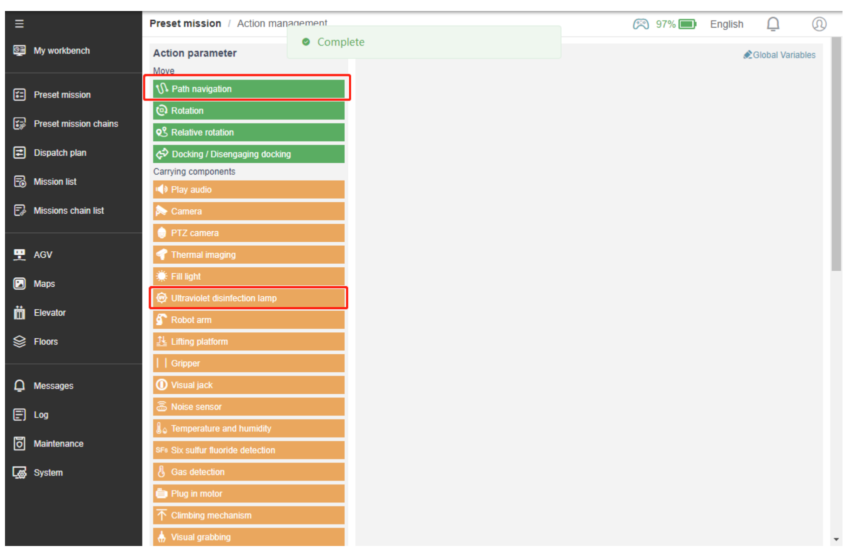


Figure 36: Actions list.

Set up disinfection task

Path navigation

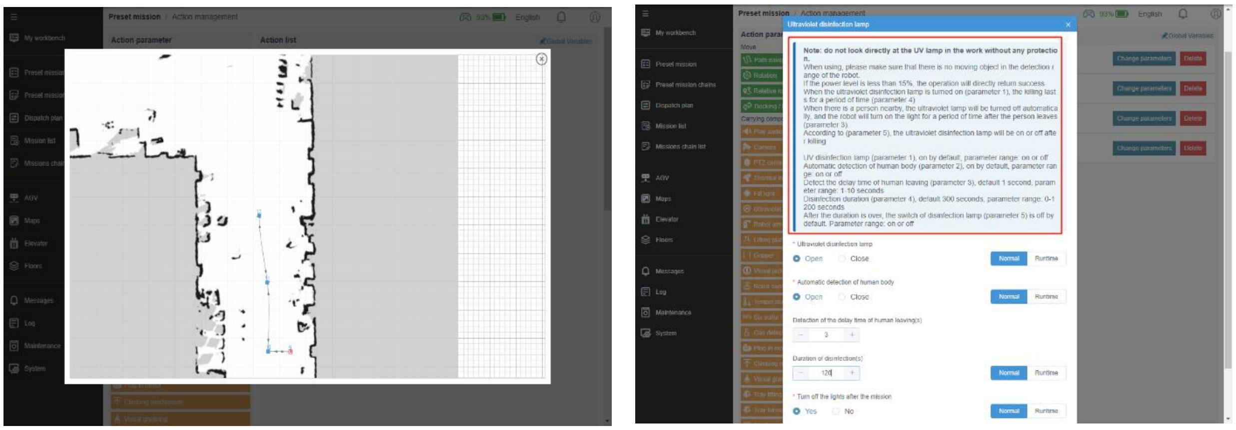


Figure 37: Selection of the map of new tasks according to your needs.

In the process of planning a task, only the position of the target point is input, and the robot can proceed from the current position (navigation point or path) to the target point. If you need to stop the movement process, you can set multiple intermediate point actions.

If it is found that the curve path only sets the target point task, the robot will stop the rotation angle at the navigation point of the midway path, please check whether the tangent direction of each path is less than 15°.

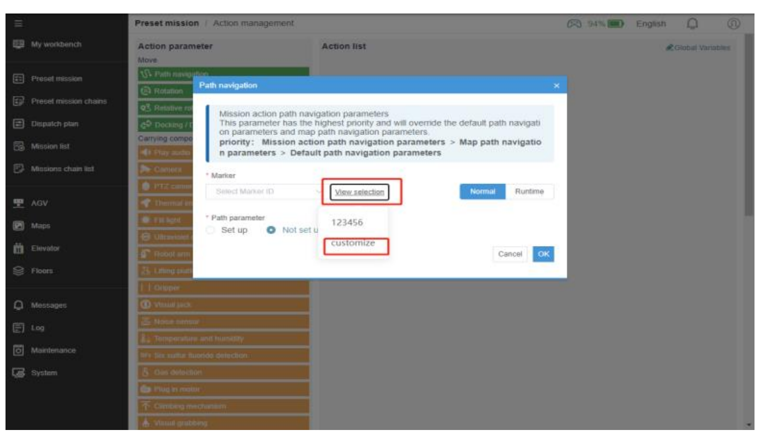


Figure 38: The following figure shows the action list of the task.

The UV lamp is equipped with a human sensor and has the function of extinguishing the lamp, that is, when the robot is disinfect at a fixed point, if someone moves, the UV lamp will automatically extinguish, so as not to cause harm to the human body. At the same time, you can set whether to turn off the lights after the end of the task, if you do not turn off the lights after the end of the task, the UV lights will always be on, so that the robot can be moved to the next navigation point or task point

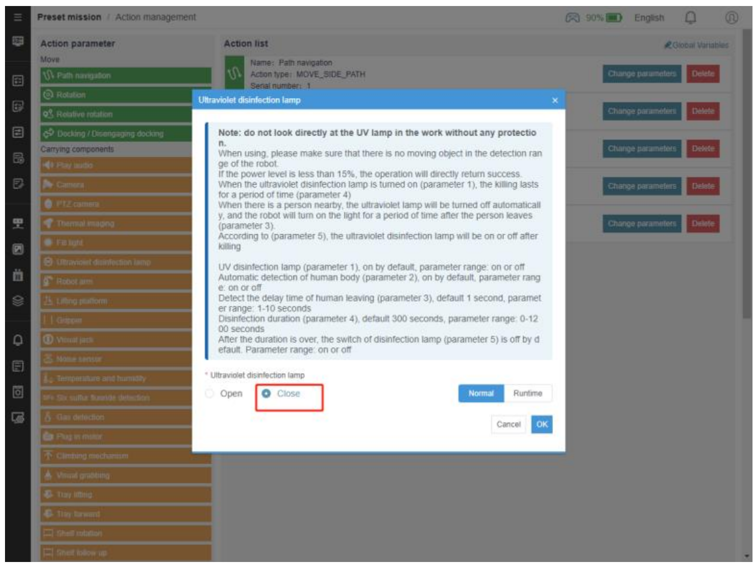


Figure 39: Adding actions according to needs.

Detect the accuracy of the robot's repositioning position

Check that the robot's position on the map is accurate, and that the outline of the point cloud matches the outline of the map.

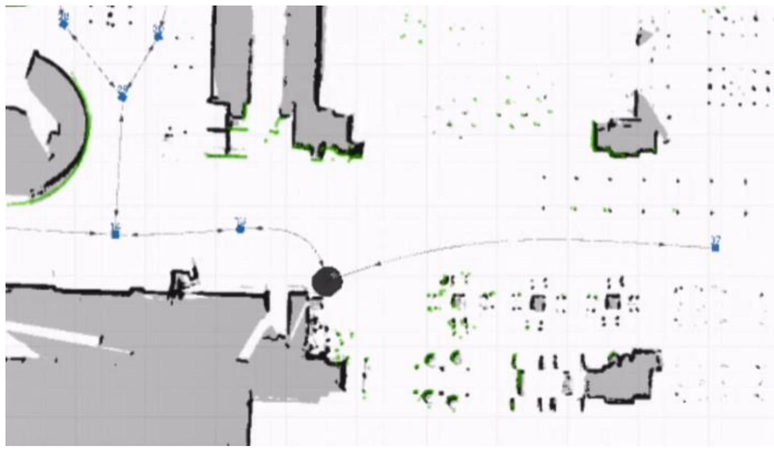


Figure 409: Check that the green line and the black line are completely overlapped

Stop the robot in orbit

After accurate positioning, the robot is moved to the navigation point or path. The task is to use the path navigation method. All robots must be on the path to start the task. The robot can be moved to the path or point by manual remote control. Note the need to park the robot in a position ±10 cm of orbit.

Enter Auto Mode

Click "automatic mode" AGV the operation interface, only in automatic mode can the task be executed, manual mode needs manual operation robot to move.

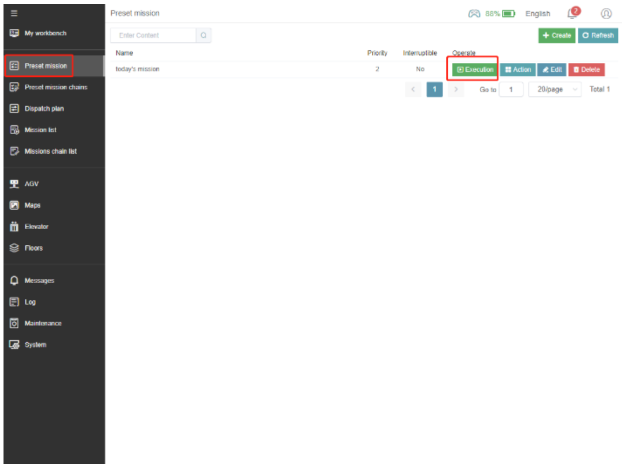


Figure 41: Manual execution of a task.

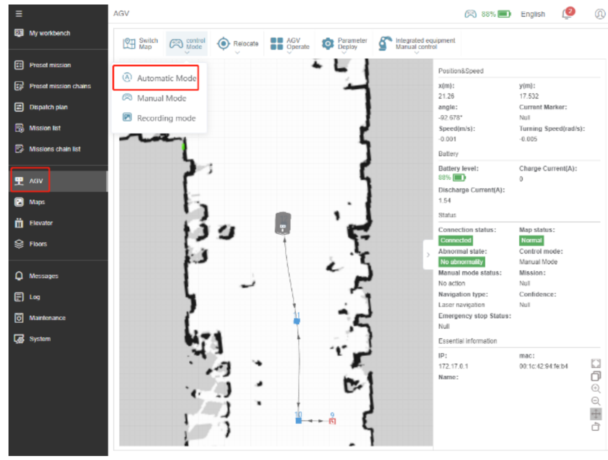


Figure 4210: AGV automatic mode.

Figure 43: Check the status of AGV.

Mission observations

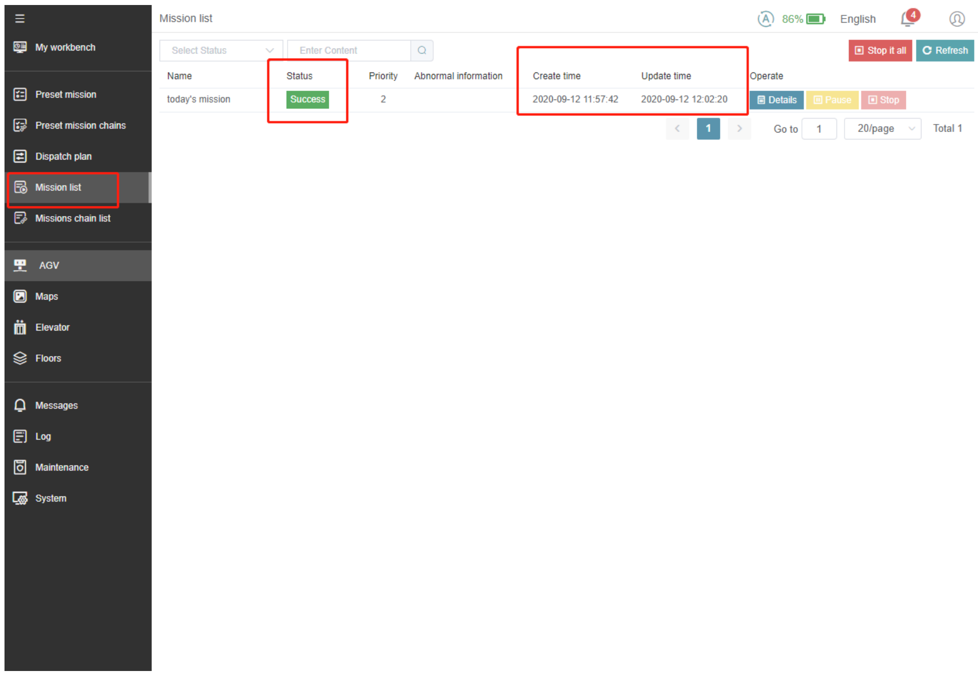


Figure 44: Missions completion and status.

Exception handling

1.Robot derailment

If the robot does not move, you can first view the execution status in "execution management ", showing that" create "is because the robot has not entered automatic mode, showing that" wait "is still in execution, or there is a task error. It is also possible that the map has not been updated to the latest version. Remember that map editing must be synchronized. If you trigger obstacle avoidance, please refer to the obstacle avoidance area regulation section.

2.Execution error

When there is a task error, you need to stop the error task, otherwise the subsequent task cannot run properly. Select the wrong task in the execution management and click "stop ".

3.Prompt derailment

1. Please check if you have synchronized the map, each time you modify the map need to synchronize the map.
2. Check if the robot stops on the path ±10 cm.

If the system prompt never reaches this point: check path connectivity.

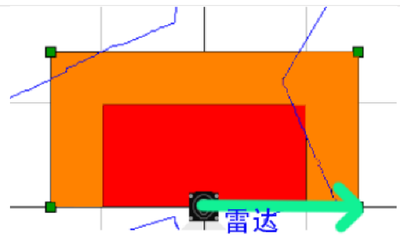
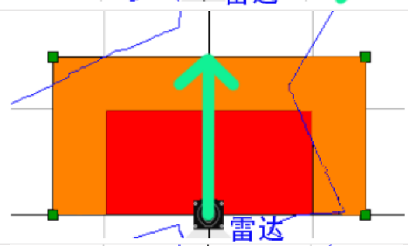
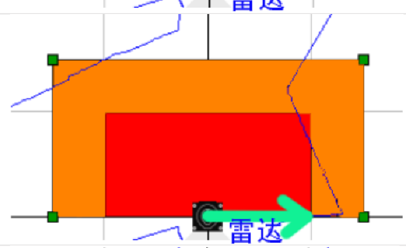
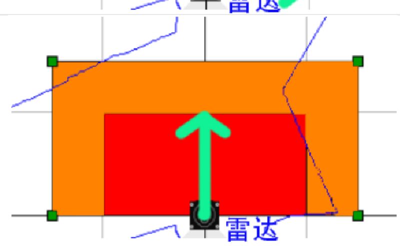
ADJUSTMENTS

Instruction

The obstacle avoidance area of the robot can be changed, and the modified area will also be displayed by the indicator lamp. Blue is the deceleration zone, red is the stop zone, and green indicates no trigger deceleration or stop zone.

Adjustment of obstacle avoidance areas

Map of the obstacle avoidance area

Forward radar obstacle avoidance deceleration zone x(m)	
Forward radar obstacle avoidance deceleration zone y(m)	
Front radar obstacle avoidance stop area x(m)	
Front radar obstacle avoidance stop area y (m)	

The minimum distance of obstacle avoidance area is set at 0 m. And the deceleration area needs to be larger than the stop area…35
Just set up the front radar
The X axis represents the front and rear direction of the robot, and the Y axis represents the left and right direction of the robot

Priority

Path task > path parameter >AGV configuration, parameter content can be default.
If path tasks and path parameters are not set, the path parameters A GV configured are used by default.

Path task settings

. 1. AGV modification of path parameters

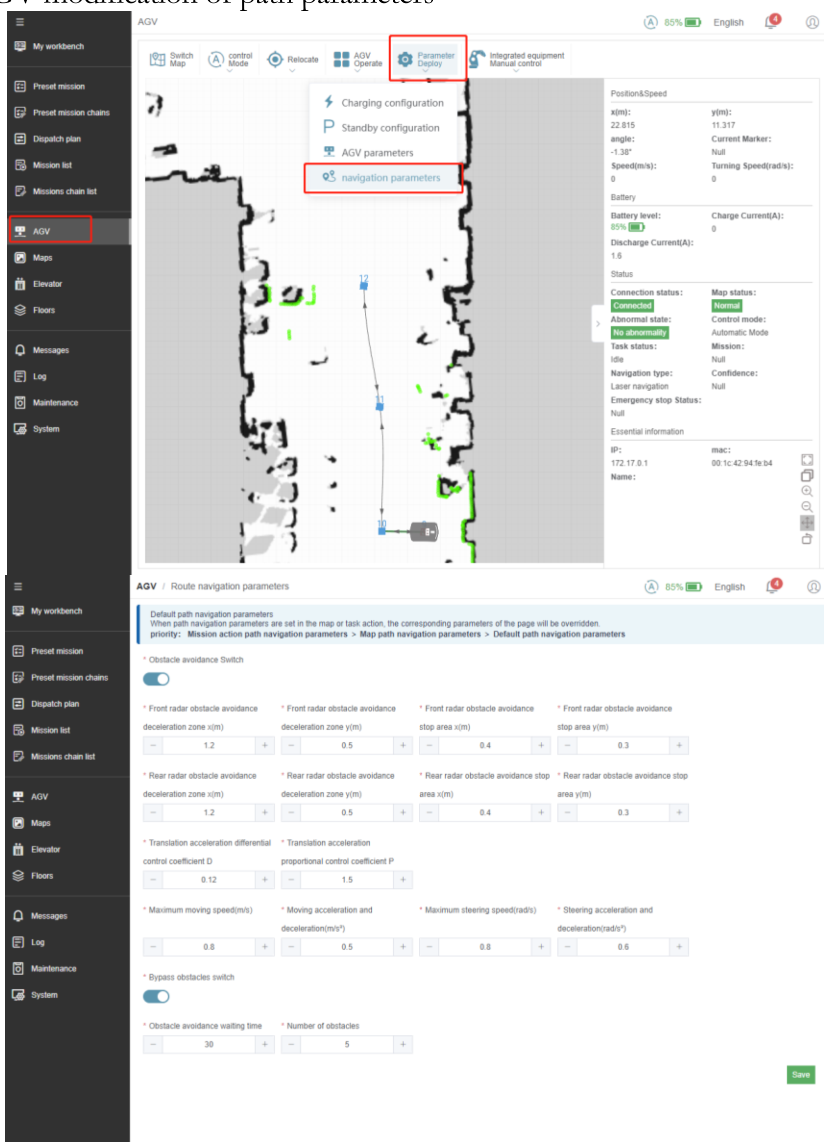


Figure 45: Path parameters modification.

2. Modify the path parameter in the action list

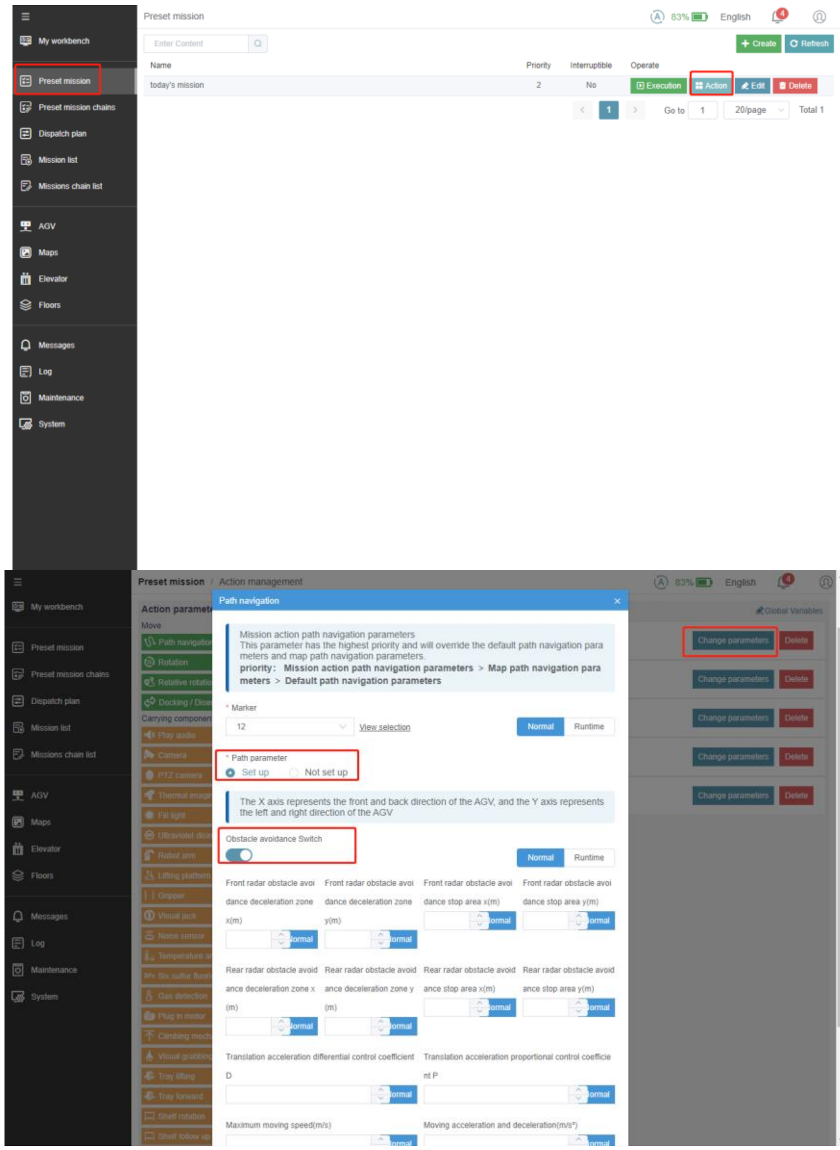


Figure 4611: Path parameter modification from action list.

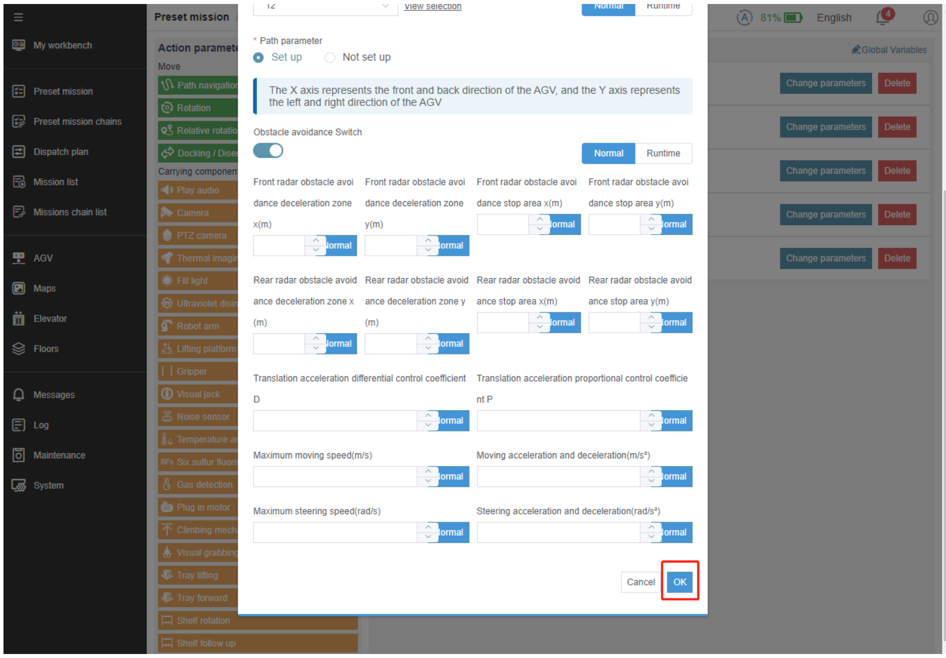


Figure 4712: Path parameter modification step

A GV default settings

1. Robot Configuration

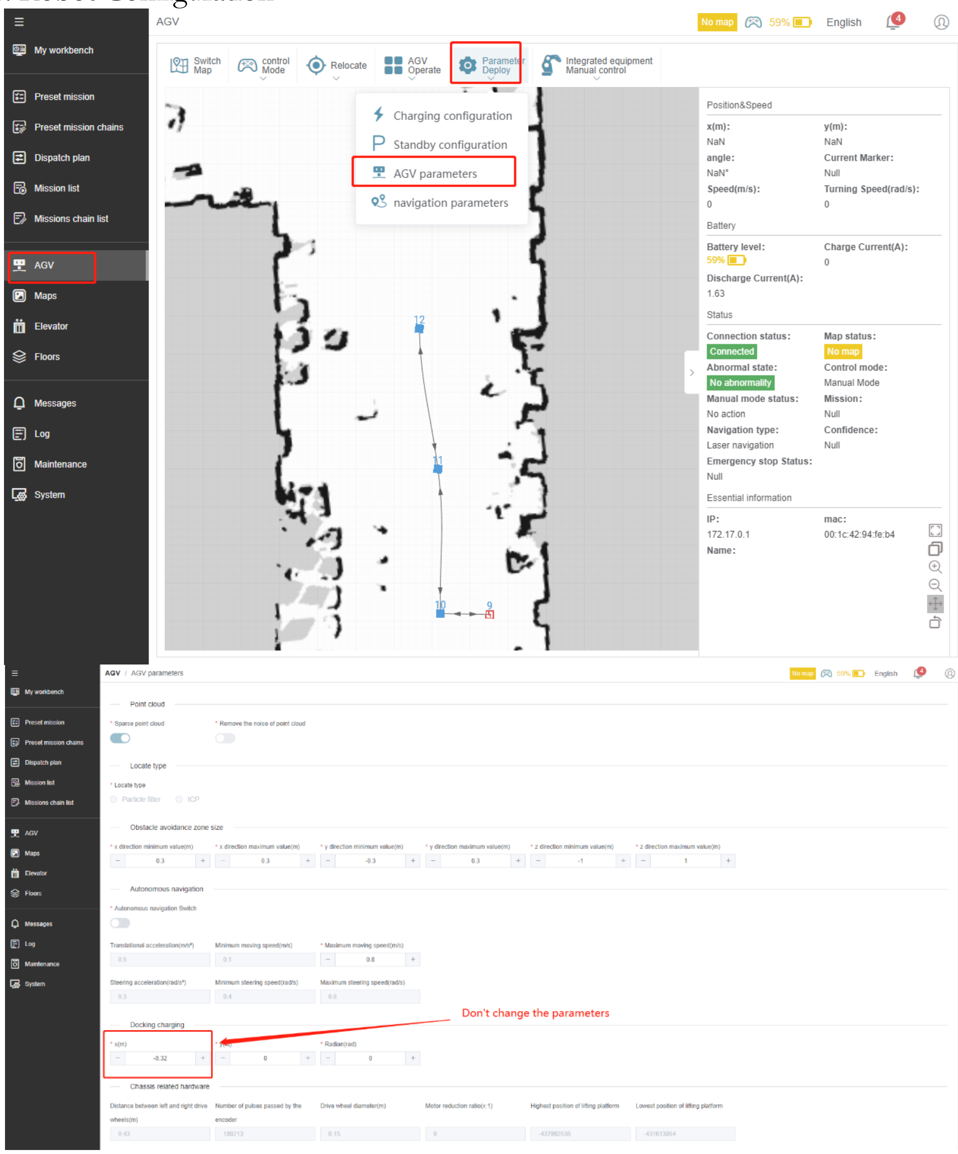


Figure 48: Robot configuration settings.

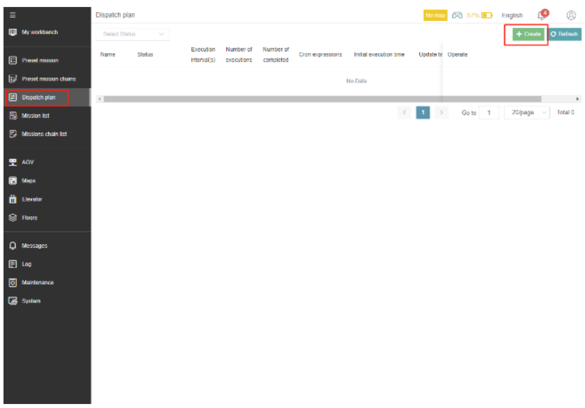
2. Click Save to apply changes made.

Setting Periodic/Timing tasks

Introduction

Create a scheduling plan and specify a task, and the scheduling system automatically executes the task according to the plan set by the user.

Create daily kill implementation plan



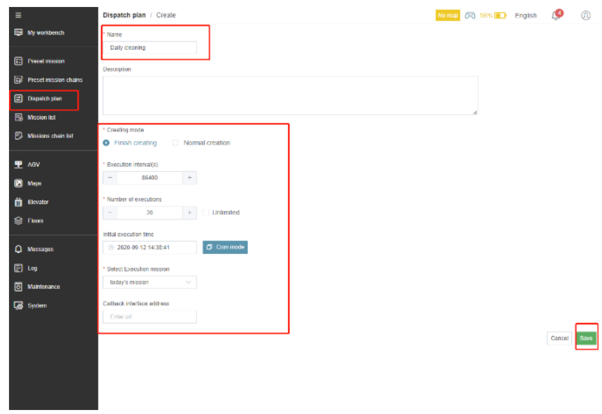


Figure 49: Daily plan dispatch plan using cron mode.

Note: do not select normal creation and execution intervals are too low, resulting in short-term task creation.

Scheduling plan attribute description

Title	Note
Name of name	User custom name
Description	User Custom Description
How tasks are created	To create is to create the next task at a time after completing a task ." Normal creation "means to create a new task directly after the interval is reached, regardless of whether the previous task is completed.
Callback URL address	The user can specify the URL address of a callback that the scheduling plan will call back during execution of the URL. And feedback the status of execution.
Number of	The total number of tasks performed.-1 means infinite
Implementation interval	How many seconds to set the interval to perform the task
Start time	Start time to implement the plan
CRON mode	Create tasks using CRON expressions

Tasks selected for execution	Selection of tasks performed by scheduling

Automatic charging of robots

Add charging point

After recording the map, add the charge point to the process of adding the point through the mobile robot

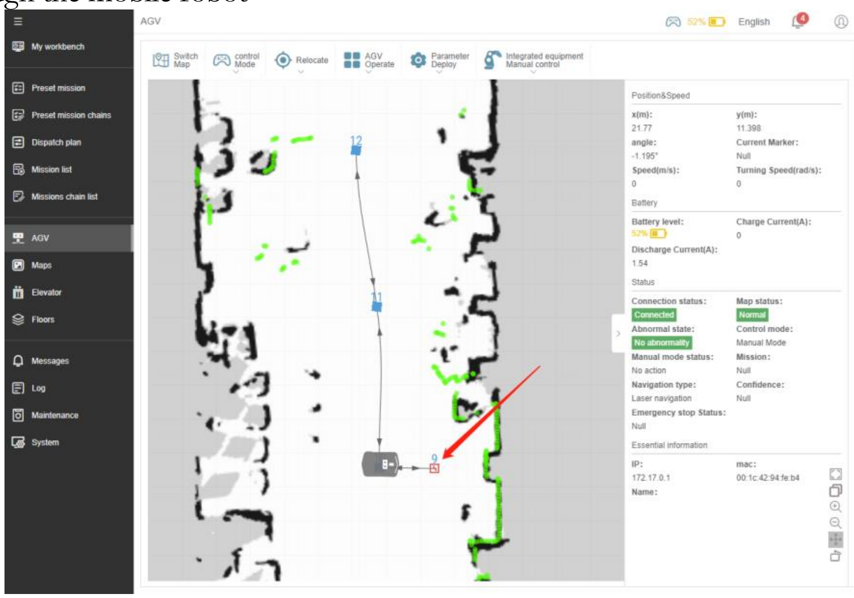
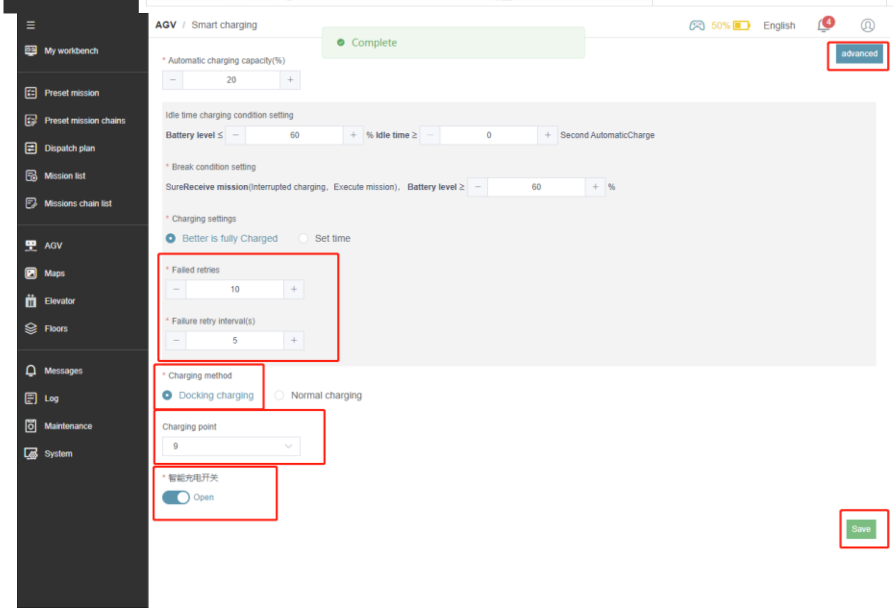


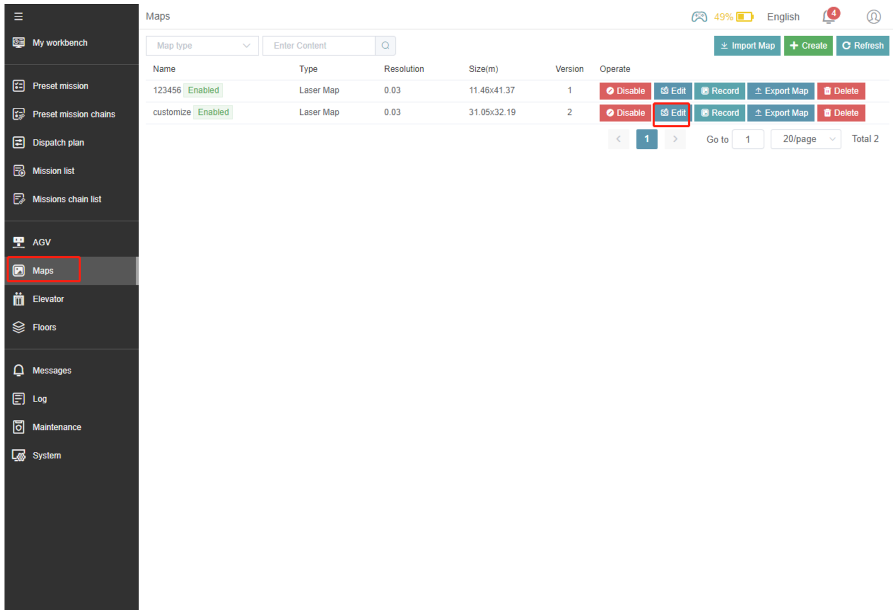
Figure 50: Adding charging point to the map.

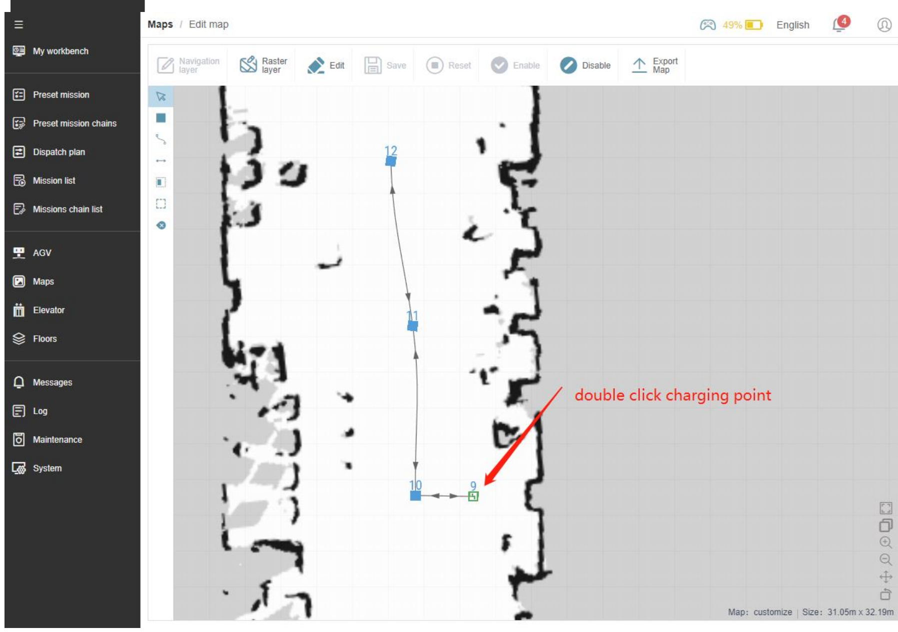
Advice: in the process of recording the point, record a charging point at the position of 50-70 cm in front of the charging pile so that the robot can charge automatically after performing the task.

My workbench
Preset mission
Preset mission chains
Dispatch plan
Mission list
Missions chain list
AGV
Maps
Elevator
Floors
Messages
Log
Maintenance
System
AGV
51%
English
Switch Map
control Mode
Relocate
AGV Operate
Parameter Deploy
Integrated equipment Manual control
Charging configuration
Standby configuration
AGV parameters
navigation parameters
Position&Speed
x(m):
21.77
y(m):
11.398
angle:
-1.195°
Current Marker:
Null
Speed(m/s):
0
Turning Speed(rad/s):
0
Battery
Battery level:
51%
Charge Current(A):
0
Discharge Current(A):
1.57
Status
Connection status:
Connected
Map status:
Normal
Abnormal state:
No abnormality
Control mode:
Manual Mode
Manual mode status:
No action
Mission:
Null
Navigation type:
Laser navigation
Confidence:
Null
Emergency stop Status:
Null
Essential information
IP:
172.17.0.1
mac:
00:1c:42:94:fe:b4
Name:



My workbench
Preset mission
Preset mission chains
Dispatch plan
Mission list
Missions chain list
AGV
Maps
Elevator
Floors
Messages
Log
Maintenance
System
AGV / Smart charging
Complete
50%
English
advanced
* Automatic charging capacity(%)
20
Idle time charging condition setting
Battery level ≤ 60 % Idle time ≥ 0 Second AutomaticCharge
* Break condition setting
SureReceive mission(Interrupted charging、Execute mission)、Battery level ≥ 60 %
* Charging settings
Better is fully Charged
Set time
* Failed retries
10
* Failure retry interval(s)
5
* Charging method
Docking charging
Normal charging
Charging point
9
* 智能充电开关
Open
Save


Maps
English
My workbench
Preset mission
Preset mission chains
Dispatch plan
Mission list
Missions chain list
AGV
Maps
Elevator
Floors
Messages
Log
Maintenance
System
Import Map
Create
Refresh
Name
Type
Resolution
Size(m)
Version
Operate
123456 Enabled
Laser Map
0.03
11.46x41.37
1
customize Enabled
Laser Map
0.03
31.05x32.19
2
Disable
Edit
Record
Export Map
Delete
Go to
20/page
Total 2


Maps / Edit map
English
My workbench
Preset mission
Preset mission chains
Dispatch plan
Mission list
Missions chain list
AGV
Maps
Elevator
Floors
Messages
Log
Maintenance
System
Navigation layer
Raster layer
Edit
Save
Reset
Enable
Disable
Export Map
12
11
10
9
double click charging point
Map: customize
Size: 31.05m x 32.19m

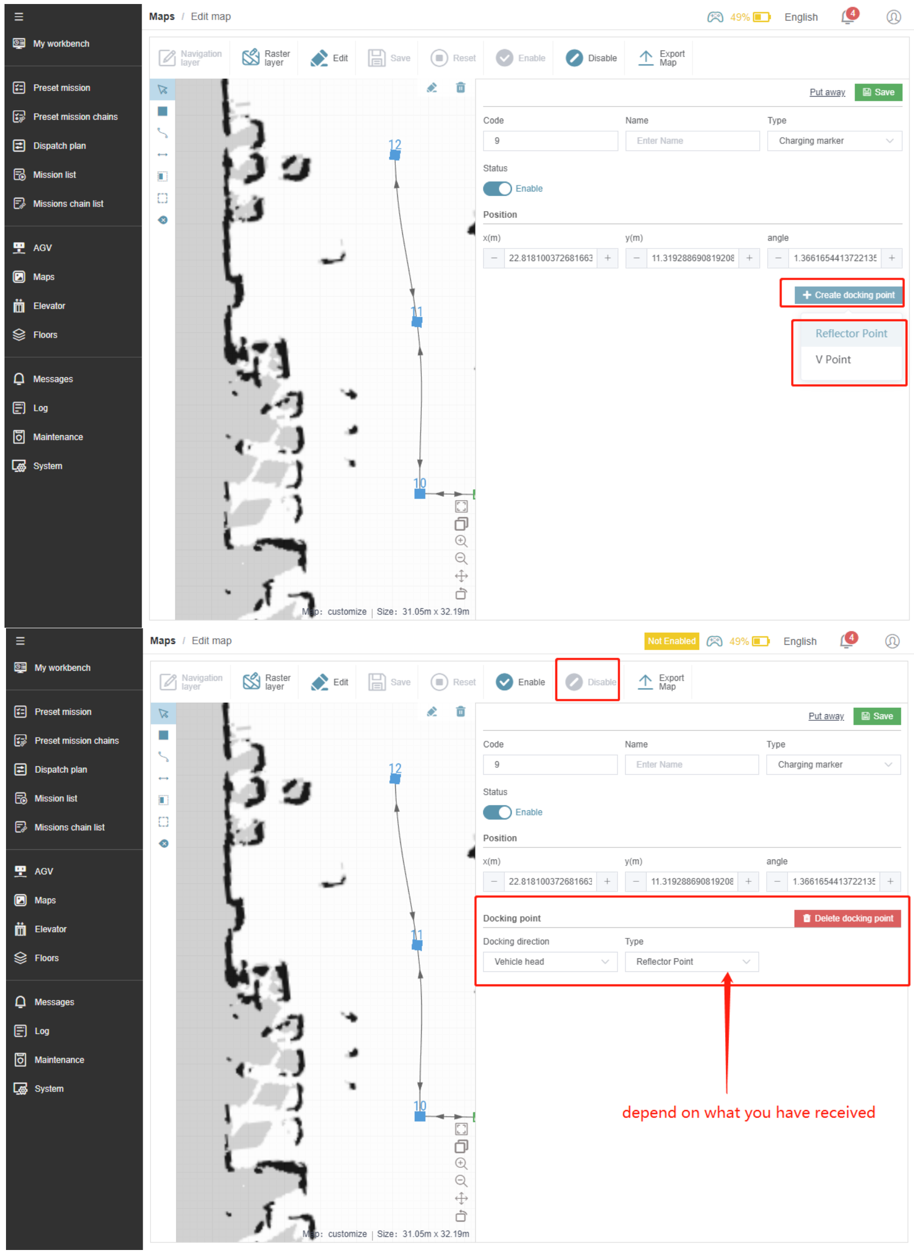


Figure 51: Automatic charging setup illustration.

Attention to automatic charging of robots:

1. Docking distance can't be too far or too close
2. Must set up docking charging at the A GV configuration interface and select charging point
3. Then select the docking type in the map editing interface, whether the V plate or the reflective paste depends on the situation
4. Charging pile in automatic mode.

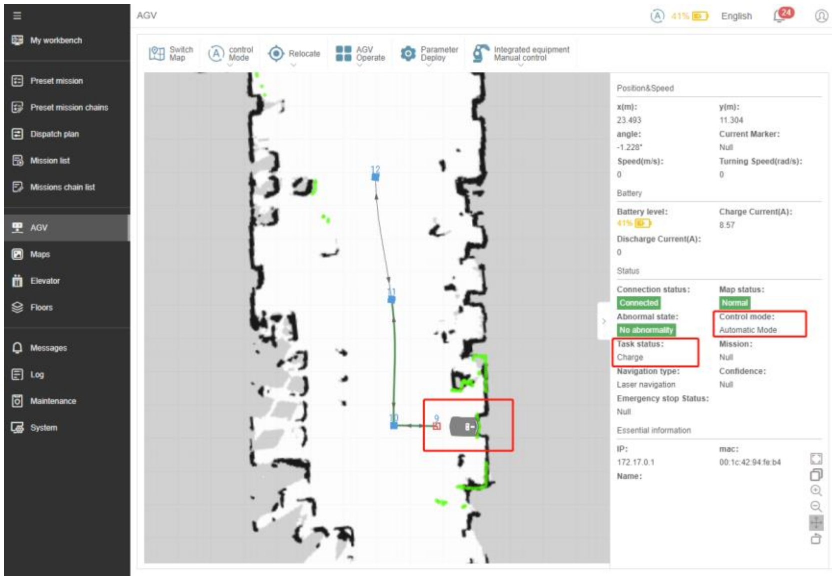


Figure 52: The robot is performing automatic charging.

Console → navigation bar function

The top navigation bar includes the modules of temperature measurement, disinfectioning, robot, disinfectioning report and entering the background.
Click on different navigation bar labels to enter different feature pages. Access to the background requires administrator or super administrator authorization.
1. click [temperature measurement], you can enter the temperature measurement function page, operate the robot to perform the temperature measurement task; when the robot is performing the temperature measurement task, click [disinfection] or [robot], prompt "is performing the temperature measurement task, please stop the temperature measurement and re-".
2. click [disinfection], you can enter the disinfection function page, operate the robot to perform the disinfection task; when the robot is performing the temperature measurement task, click [temperature measurement] or [robot], prompt "is performing the disinfection task, please stop disinfect and re-operate ".
3. click [robot], you can enter the robot control page, manually operate the robot to perform map change, re-positioning and other functions; when the robot is in "manual mode ", click on other navigation labels, then prompt" please switch to automatic mode before operating other functions ".

4. click [report], you can go to the report list page, click to view the report details; when the robot is performing disinfection or temperature measurement task status, you can click to view disinfection report task.
5. Click [enter background] to enter management background, edit area information, task information, user information and configuration parameters. The robot can support users to enter the background in the task of disinfecting or measuring temperature.

Robot status

You can see the current state of the robot in the upper right corner
Status includes standby, idle, working, charging, obstacle avoidance, emergency stop, abnormal
[on standby]: The robot is on line and is waiting at the standby point. It is shown as standby
[Idle]: Idle when the robot has finished the task and is not in standby;
[At Work]: When the robot performs the disinfection or temperature measurement task, it is displayed as work;
[In charge]: When the robot is in charge state, it is shown as in charge;
[Obstacle avoidance]: when the robot encounters obstacles, it is displayed as obstacle avoidance during obstacle avoidance;
[Emergency stop]: when the robot encounters a sudden stop, it is displayed as a state of emergency stop, the emergency stop is lifted, and the state before the emergency stop is restored;
[Abnormal]: to robot anomalies, such as derailment, broken network, etc. are shown as abnormal

Battery status

The upper right corner can view the battery surplus of the robot, according to the percentage, when the power is lower than the set charging threshold, it is shown as red low power state, when charging, the battery shows charging.

Console → disinfection page

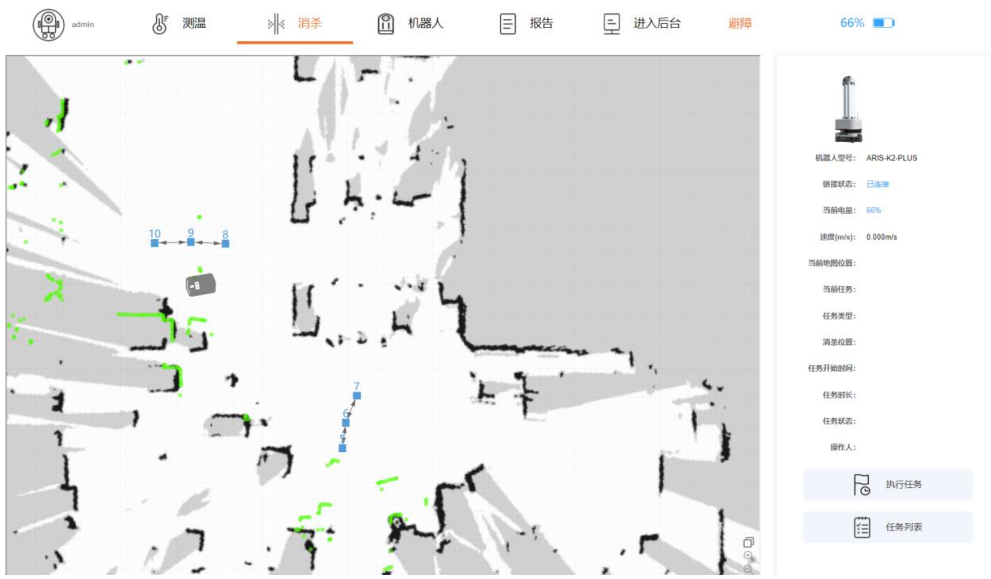


Figure 53: Desinfection page.

Status introduction

On the disinfection page, the left map shows the current location of the robot; the right side of the page shows the basic information and task status of the robot, including:

[Robot Model]: Display according to the robot model configured in the background

[Connection Status]: Real-time display of robot off-line status

[Current Power]: Real-time display of the current power of the robot

[Velocity (m/s)]: Real-time display of the robot's current speed

[Current Map Location]: Real-time display of the current map of the robot

[Current task]: Real-time display of the robot's current task, if there is a task, the task name is displayed, and if there is no task, the content is not displayed

[Task type]: Displays the type of task, whether it is a temporary manual task or a timed automatic task, and if there is no task, does not display the content

[Disinfection location]: real-time display of disinfection task location name, if temporary manual task, direct display of sub-area name, if timing automatic task, direct display of disinfection area name, details can be viewed by clicking on the robot logo; if no task, no content is displayed

[Task Start Time]: Displays the task start time and, if no task, does not display content

[Task duration]: Displays the time the current task has been executed, and the task status can start in time;

[Task Status]: Displays status information for the task, including not started, in progress, completed, stopped, and abnormal;

1. temporary manual task: the robot has not started before the disinfection, when the robot enters the disinfection area to complete the list confirmation and start disinfection, the state becomes in execution, when the execution reaches the exit point, the task state becomes stopped, and the task is abnormal.
2. timing automatic task: when the robot reaches the entry point is not started, when the entry point and automatically start execution, the task state is displayed as execution, when the completion of the disinfection reaches the exit position, the task state becomes stopped when the task is forced to stop.

[Operator]: Displays the name of the person who created the disinfection task

When the robot is idle, click [execute the task] to expand the list of temporary manual tasks, select the disinfection task execution location;
When the robot is performing the disinfection task, click [execute the task] and prompt "disinfection the task, please wait until the current task is completed.

When clicking [execute task], pop up disinfection location list, can choose disinfection location (note: only display the location information belonging to the current map, other areas of disinfection location information need to operate robot switch map before display, When the robot page operation switch map, disinfection page and temperature page map synchronous switch), click pop-up execution confirmation, the task starts to execute, as shown below, the robot status in the upper right corner becomes "working ".

Task List Page Overview: When the user clicks the task list under the page of the disinfection module, the task list can show the content including the task being executed and the task to be executed. If there is a task in the task list, the task information can be seen, and if there is no task at present.

Background

This document is to help users understand the interface functions and docking methods of the YOUIFleet operating system, and help users directly complete the scheduling of robots in Youifleet.

This document only includes the descriptions of mission-related calling interface . If you have other interface connection requirements, you can contact our consultants or technical support personnel.

The design style of the API complies with the HTTP Restful specification. Developers who using this interface need to be familiar with the Restful development specification. The detail of Specific Restful Development Dpecifications please refer to: http://www.restfulapi.nl/

YOUIFleet (Robot scheduling system, hereinafter referred to as YOUIFleet) is a software system independently developed by YOUIBOT for robot Mission work scheduling.

Users can complete map creation and editing, robot management, mission and mission work setting and other management work through YOUIFleet.

API above the 2.0.0 version of YOUIFleet is required.

The development process

1. Deploy the service package in the development environment according to the YOUIFleet deployment manual and the YOUIFleet manual. And configure maps, robots, missions.
2. Call the Create Mission Work API in the system.
3. Call the query Mission Work status API in the system.
4. Call the query Mission Work execution result API in the system.

The description of mission interface

Mission objects

Mission refers to the execution process of the robot configured by users. Users can configure the mission of the robot in YOUIFleet and then call and assign them to the robot for execution through the interface. Please refer to YOUIFleet user manual for specific mission configuration methods.

Field name	Field type	Field description
id	String	The unique ID that is automatically generated when the mission work is created
agvMapId	String	Map ID
missionGroupId	String	Mission work group ID

name	String	Name of Mission work
description	String	Description
agvId	String	Assign the AGVID to perform the mission work
agvGroupId	String	Assign the AGVID to perform the mission work
sequence	int	Priority.: 1: low, 2: ordinary, 3: high, 4: highest; The default is 2.
interrupt	boolean	Interruptible or not.If it's true, mission can be interrupted during execution by other higher priority missions.The default is false.
createTime	date	Mission create time
updateTime	date	Mission update time

Mission work project

Mission work refers to the robot execution process. The user creates a mission through the mission work and assigns this mission work to the robot for execution.

Field name	Field type	Field description
id	String	The unique ID that is automatically generated when the mission work is created
agvId	String	ID of the robot performing the mission
missionId	String	Mission ID
agvMapId	String	Map ID
schedulePlanId	String	If it was created through a dispatch plan, it refer to the dispatch plan ID.
callbackUrl	String	Callback URL，
name	String	Name of Mission work
sequence	int	Priority.: 1: low, 2: ordinary, 3: high, 4: highest; The default is 2.
description	String	Name of Mission work
status	String	Status, please check the status description of mission work below
message	String	Abnormal signal
interrupt	boolean	Interruptible or not.If it's true, mission can be interrupted during execution by other higher priority missions.The default is false.
runtimeParam	String	Run-time parameter
startTime	date	Mission work start time
endTime	date	Mission work end time
createTime	date	Mission work create time
updateTime	date	Mission work update time
currentActionSequence	Integer	Sequence number of currently executed action

Working state

When the robot is performing a mission, the working state will change accordingly. The docking system needs to adjust the operation logic according to the working state.

Name of state	Description
WAIT	Waiting for execution
START	Begin execution
RUNNING	Executing
SUCCESS	execution succeed
FAULT	Execution failed
PAUSE	Suspend execution
SHUTDOWN	Cancel execution
WAITINPUT	Waiting for input. During the mission execution, if there are parameters that need to be input, the status will change to WAITINPUT

Create Mission Work

Calling the interface to create the mission work, YOUIFleet will automatically assign mission to robots and execute them automatically. For more details, please refer to [YOUIFleet API Manual 2.13.1 Create Work According to mission work]

<table>
<tr><th>Interface content</th><th>Interface Description</th></tr>
<tr><td>Request URL</td><td>http://host:port/api/v2/missionWroks</td></tr>
<tr><td>Request method</td><td>POST</td></tr>
<tr><td>Request Parameter</td><td><table>
<tr><td>Parameter name</td><td>Parameter Type</td><td>Parameter Description</td></tr>
<tr><td>agvId</td><td>String</td><td>Assign agvId to the mission</td></tr>
<tr><td>callbackUrl</td><td>String</td><td>Call back URL</td></tr>
<tr><td>missionId</td><td>String</td><td>Mission ID</td></tr>
<tr><td>runtimeParam</td><td>String</td><td>Real-time parameter (json format) such as: {"marker1":"1001"}</td></tr>
</table></td></tr>
<tr><td>Response body</td><td>MissionWork, Please refer to the mission object</td></tr>
<tr><td>Response code</td><td>201 please refer to Response code explanation.</td></tr>
</table>

Sample code JAVA：Query mission work

```
String fleetHost = "127.0.0.1";
Integer fleetPort = 8080;
String missionWorkId = "2035ee2f-c965-11e9-b01f-b0416f03e8c4";
HttpRequest request = new HttpRequest();
request.method(HttpMethod.GET).host(fleetHost).port(fleetPort).path("/api/v2/missionWorks/" + missionWorkId).contentType("application/json");
HttpResponse response = request.send();
int status = response.statusCode();
if (status == HttpStatus.SC_OK) {
   System.out.println("Mission work query succeeded");
   System.out.println(JSON.parseObject(response.body(), MissionWork.class));
} else {
   System.out.println("Mission work query failed");
   System.out.println(response.body());
}
```

Sample code JAVA：Create mission work

```
String fleetHost = "127.0.0.1";
Integer fleetPort = 8080;
String missionId = "2035ee2f-c965-11e9-b01f-b0416f03e8c3";
MissionWorkPost missionWorkPost = new MissionWorkDemo.MissionWorkPost(missionId, null);
HttpRequest request = new HttpRequest();
request.method(HttpMethod.POST).host(fleetHost).port(fleetPort).path("/api/v2/missionWorks/").data(JSON.toJSONString(missionWorkPost)).contentType("application/json");
HttpResponse response = request.send();
int status = response.statusCode();
if (status == HttpStatus.SC_CREATED) {
   System.out.println("Mission work created successfully");
   System.out.println(JSON.parseObject(response.body(), MissionWork.class));
} else {
   System.out.println("Mission work creation failed");
   System.out.println(response.body());
}
```

Query Mission Work

The user can query the execution status and execution results of the mission work through the access interface.

Interface content	Interface Description
Request URL	http://host:port/api/v2/missionWroks/{id}
Request method	GET
Request Parameter	Parameter name / Parameter Type / Parameter Description: id / String / Mission ID
Response body	MissionWork Please refer to the mission work object
Response code	200 please refer to Response code explanation.

Pause mission work

This interface can be used to suspend the mission being executed. At this time, the robot that is performing the mission will temporarily stop working, Until the mission is resumed or canceled.

Interface content	Interface Description
Request URL	http://host:port/api/v2/missionWroks/{id}/conntrols/pause
Request method	POST
Request Parameter	Parameter name / Parameter Type / Parameter Description: id / String / Mission ID
Response body	MissionWork Please refer to the mission work object

Sample code JAVA：Pause Mission work

```
String fleetHost = "127.0.0.1";
Integer fleetPort = 8080;
String missionWorkId = "2035ee2f-c965-11e9-b01f-b0416f03e8c4";
HttpRequest request = new HttpRequest();
request.method(HttpMethod.POST).host(fleetHost).port(fleetPort).path("/api/v2/missionWorks/" + mission-WorkId + "/controls/pause").contentType("application/json");
HttpResponse response = request.send();
int status = response.statusCode();
if (status == HttpStatus.SC_OK) {
   System.out.println("Mission work resumed successfully");
   System.out.println(JSON.parseObject(response.body(), MissionWork.class));
} else {
   System.out.println("Mission work resumed unsuccessfully");
   System.out.println(response.body());
}
```

<table>
<tr><td>Request URL</td><td>http://host:port/api/v2/missionWroks/{id}/conntrols/resume</td></tr>
<tr><td>Request method</td><td>POST</td></tr>
<tr><td>Request Parameter</td><td><table><tr><td>Parameter name</td><td>Parameter Type</td><td>Parameter Description</td></tr><tr><td>id</td><td>String</td><td>Mission ID</td></tr></table></td></tr>
<tr><td>Response body</td><td>MissionWork Please refer to the mission work object</td></tr>
<tr><td>Response code</td><td>200 please refer to Response code explanation.</td></tr>
<tr><td></td><td></td></tr>
</table>

Sample code JAVA：Pause Mission Work

```
String fleetHost = "127.0.0.1";
Integer fleetPort = 8080;
String missionWorkId = "2035ee2f-c965-11e9-b01f-b0416f03e8c4";
HttpRequest request = new HttpRequest();
request.method(HttpMethod.POST).host(fleetHost).port(fleetPort).path("/api/v2/missionWorks/" + mission-
WorkId + "/controls/resume").contentType("application/json");
HttpResponse response = request.send();
int status = response.statusCode();
if (status == HttpStatus.SC_OK) {
    System.out.println("Mission work resumed successfully");
    System.out.println(JSON.parseObject(response.body(), MissionWork.class));
} else {
    System.out.println("Mission work resumed unsuccessfully");
    System.out.println(response.body());
}
```

Cancel Mission work

We can cancel the mission that the robot is performing. After canceling the mission, the robot will be assigned other mission.

Sample code JAVA：Cancel Mission Work

```
String fleetHost = "127.0.0.1";
Integer fleetPort = 8080;
String missionWorkId = "2035ee2f-c965-11e9-b01f-b0416f03e8c4";
HttpRequest request = new HttpRequest();
request.method(HttpMethod.POST).host(fleetHost).port(fleetPort).path("/api/v2/missionWorks/" + mission-
WorkId + "/controls/stop").contentType("application/json");
HttpResponse response = request.send();
int status = response.statusCode();
if (status == HttpStatus.SC_OK) {
    System.out.println("Mission work cancelled successfully");
    System.out.println(JSON.parseObject(response.body(), MissionWork.class));
} else {
    System.out.println("Mission work cancellation failed");
    System.out.println(response.body());
}
```

HTTP status code

status code	Description
200	Execution succeed
201	Created successfully
204	Deleted successfully or does not exist
401	Authentication failed
404	Interface does not exist
423	Certificate authentication failed
500	Internal server exception

Run-time Parameter interface

Run-time Parameter is a very important function, which allow us to pass parameters to the work during execution. Mission work can be performed according to parameters.

Development Process

1. Create mission to set dynamic parameters.
2. Create a mission.
3. The user needs to input parameters during the execution of the work, and the status becomes “WAITINPUT”.
4. The user enters parameters and the mission continues to execute.

Run-time parameter object

Run-time Parameter

Field name	Field type	Field description
id	String	Parameter ID
missionWorkId	String	Mission ID
parameterKey	String	Parameter Name
parameterType	String	Parameter Type
parameterValue	String	Parameter Value
type	String	Data Type
createTime	date	Create Time
updateTime	date	Update Time

Query parameter list

<table>
<tr><th>Interface Content</th><th>Interface Description</th></tr>
<tr><td>Request URL</td><td>http://host:port/api/v2/missionWorks/{id}/emptyRuntimeParameters</td></tr>
<tr><td>Request method</td><td>GET</td></tr>
<tr><td>Request Parameter</td><td><table><tr><td>Parameter name</td><td>Parameter Type</td><td>Parameter Description</td></tr><tr><td>id</td><td>String</td><td>Mission ID</td></tr></table></td></tr>
<tr><td>Response body</td><td>List<RuntimeParameter> Please refer to the mission work object</td></tr>
<tr><td>Response code</td><td>200 please refer to Response code explanation.</td></tr>
</table>

Sample code JAVA: Query dynamic parameter list

```
String fleetHost = "127.0.0.1";
Integer fleetPort = 8080;
String missionWorkId = "2035ee2f-c965-11e9-b01f-b0416f03e8c4";
HttpRequest request = new HttpRequest();
request.method(HttpMethod.POST).host(fleetHost).port(fleetPort).path("/api/v2/missionWorks/" + mission-WorkId + "/emptyRuntimeParameters").contentType("application/json");
HttpResponse response = request.send();
if (response.statusCode() == 200) {
    System.out.println("Run-time parameter list query succeeded");
    System.out.println(JSON.parseObject(response.body(), RuntimeParameter.class));}
```

Update parameter list

After obtaining the parameter list, the docking system needs to input parameters through the interface.

Interface Content	Interface Description
Request URL	http://host:port/api/v2/runtimeParameters/{id}
Request method	PUT
Request Parameter	Run-time Parameter object
Response body	Run-time Parameter object
Response code	200 please refer to Response code explanation.

Sample code JAVA: Update dynamic parameter list

```
String fleetHost = "127.0.0.1";
Integer fleetPort = 8080;
RuntimeParameter runtimeParameter = new RuntimeParameter();
runtimeParameter.setId("2035ee2f-c965-11e9-b01f-b0416f03e8c4");
runtimeParameter.setParameterValue("test");
HttpRequest request = new HttpRequest();
request.method(HttpMethod.PUT).host(fleetHost).port(fleetPort).path("/api/v2/runtimeParameters/" +
runtimeParameter.getId()).data(JSONObject.toJSONString(runtimeParameter)).contentType("applica-
tion/json");
HttpResponse response = request.send();
if (response.statusCode() == 200) {
   System.out.println("Run-time parameter successfully");
   System.out.println(JSON.parseObject(response.body(), RuntimeParameter.class));
} else {
   System.out.println("Run-time parameter update failed");
   System.out.println(response.body());
}
```

Robot status, human detection, and map as image extraction code samples.

First of all, all messages shared are in JSON format and here is the code that I implemented to obtain map in json format and calculate frequency.

```
import requests
import time

ts = time.time()
prev = 0

while True:
    response = re-
quests.get("http://192.168.1.52:8080/api/v3/M
aps")
    crt = time.time() - ts
    freq = (crt - prev)
    prev = crt
    print(1/freq)
    print(response.json())
```

The following code was used to separate map info from points and its information and then decoded back to image file.

```
import json
import base64
import cv2
import numpy as np

data = {}
object = {'zafar'}
solo_map = open("Ispic Demo Center.json","r")
json_map = json.load(solo_map)
#print(json_map)
with open('dest_file.json', 'w') as
dest_file:
    with open("Ispic Demo Center.json", 'r')
as source_file:
```

```
        for line in source_file:
            element =
json.loads(line.strip())
            if 'Marker' in element:
                point_file = element['Mark-
er']
                # for i in range (len(ele-
ment['Marker'])):
                #     print(element['Mark-
er'][0]['x'])
            if 'AGVMap' in element:
                string = ele-
ment['AGVMap'][0]['mapData']
                k = 0
                new_string = ""
                for n in string:
                    k += 1
                    if k>= 22:
                        new_string += n

json.dump(new_string,dest_file)

with open('dest_file.json','r') as output:
    for element in output:
        element = json.loads(element.strip())


with open("imageToSave.png", "wb") as fh:
    fh.write(base64.decode-
bytes(new_string.encode()))

im= cv2.imread('imageToSave.png',20)
red = [0,0,255]
for i in range(len(point_file)):
    im[round(point_file[i]['x']),
round(point_file[i]['y'])]=red
cv2.imwrite("result.png",im)
```

```
cv2.imshow('1',im)
cv2.waitKey()
```

To control robot from python environment following code should be used.

```
async def move_robot():

    async with websockets.connect(uri) as
websocket:
    async for greeting in websocket:

        greeting = await websocket.recv()
        data = json.loads(greeting)
        mod_d = data("count",10)
        print(message)
        with open('data3.json', 'w') as out-
file:
            json.dump(data, outfile)
        manual = "SWITCH_MANUAL_MODE"
        move = {
            "code": "MOVE",
            "data": {
                "vx": 0,
                "vy":0,
                "vtheta":-0.2
            }
        }
        await websocket.send(json.dumps(man-
ual))
        await web-
socket.send(json.dumps(move))
start_server = websockets.serve(counter,
"ws://192.168.1.52:8080/agv/status", 6789)
```

where any values for Vx, Vy, and Vtheta should be inserted to move for desired point in manual mode.

For camera streaming access following uri should be used:

- rtsp://admin:robot2020@192.168.1.97:554/h264/CH1/main/av_stream

- rtsp://admin:robot2020@192.168.1.97:554/h264/CH2/main/av_stream

The following code is used to access camera streaming and detect human face in rectangle, note that library for deep convolutional networks should be added before:

```
from mtcnn import MTCNN
import cv2

detector = MTCNN()
# Load a video, if we were using google colab
we would
# need to upload the video to Google Colab
cap = cv2.VideoCapture('rtsp://admin:ro-
bot2020@192.168.1.97:554/h264/CH1/main/av_str
eam')

while (True):
    ret, frame = cap.read()
    frame = cv2.resize(frame, (720, 540))
    boxes = detector.detect_faces(frame)
    if boxes:

        box = boxes[0]['box']
        conf = boxes[0]['confidence']
        x, y, w, h = box[0], box[1], box[2],
box[3]

        if conf > 0.5:
            cv2.rectangle(frame, (x, y), (x +
w, y + h), (255, 255, 255), 1)

    cv2.imshow("Frame", frame)
    if cv2.waitKey(25) & 0xFF == ord('q'):
        break

cap.release()
cv2.destroyAllWindows()
```

REFERENCES

1. World Health Organization. WHO Coronavirus Disease (COVID-19) Dashboard. (2020). https://covid19.who.int/.

2. G.-Z. Yang, B.J. Nelson, R.R. Murphy, H. Choset, H. Christensen, S.H. Collins, *et al.* **COVID-19- The role of robotics in managing public health and infectious diseases.** Sci Robot, 5 (2020), 10.1126/scirobotics.abb5589 eabb5589

3. Z.H. Khan, A. Siddique, C.W. Lee. **Robotics utilization for healthcare digitization in global COVID-19 management.** Int J Environ Res Public Health, 17 (2020), p. 3819, 10.3390/ijerph17113819

4. J. Kent. **Artificial intelligence could speed COVID-19, detection, treatment**(2020) https://healthanalytics.com/news/artificial-intelligence-could-speed-covid-19-detection-treatment

5. I.C. Yasa, H. Ceylan, U. Bozuyuk, A.-M. Wild, M. Sitti **Elucidating the interaction dynamics between microswimmer body and immune system for medical microrobots**

6. Sci Robot, 5 (2020), 10.1126/scirobotics.aaz2867eaaz3867 View PDF

7. Saltmarsh **Covid-19: disinfection robots are being deployed**(2020)

8. A. Chaturvedi. **Robots use UV rays to kill the virus that causes Covid-19: all you need to know**(2020) https://www.hindustantimes.com/world-news/robots-use -uv-rays-to-kill-virus-that-causes-covid-19-all-you-need-to-know/

9. A. Woodward, Gal.F S. **One graphic shows how long the coronavirus lives on surfaces like cardboard plastic and steel.** (2020) https://www.businessinsider.com/how-long-can-coronavirus-live-on-surfaces-how-to-disinfect-2020-3

10. L. Fornwalt, B. Riddell. **Implementation of innovative pulsed xenon ultraviolet (PX-UV) environmental cleaning in an acute care hospital.** Risk Manag Healthc Policy, 7 (2020), pp. 25-38, 10.2147/RMHP.S57082

11. U.S. SoftBank Robotics. **Deploy autonomous cleaning robots to fight COVID-19 in healthcare facilities**(2020). https://medium.com/@SoftBankRobotics/deploy-autonomous-cleaning-robots-to-fight-covid-19-in-healthcare-facilities-40da38bd51a6

12. M. Tavakoli, J. Carriere, T. Ali. **Robotics for COVID-19: how can robots help health care in the fight against coronavirus**(2020)

13. M. Zukowski, K. Matus, E. Pawluczuk, M. Kondratiuk, L. Ambroziak. **Patients' temperature measurement system for medical robotic assistant. Mechatronics systems and materials**

14. S. Sabanovic, W.-L. Chang, C.C. Bennett, J.A. Piatt, D. Hakken. **A robot of my own: participatory design of socially assistive robots for independently living older adults diagnosed with depression. ITAP 2015, Part 1, LNCS 9193** (2018), pp. 104-114, 10.1007/978-3-319-20892-3_11

15. B. Scassellati, M. Vazquez. **The potential of socially assistive robots during infectious disease outbreaks.** Sci Robot, 5 (2020), 10.1126/scirobotics.abc9014eabc9014

16. P. Vaishnavi, J. Agnishwar, K. Padmanathan, S. Umashankar, T. Preethika, S. Annapoorani, *et al.* **Artificial Intelligence and Drones to Combat COVID-19.** Preprints (2020), 10.20944/preprints202006.0027.v1

17. M. Sharma. **How drones are being used to combat COVID-19**(2020). https://www.geospatialworld.net/blogs/how-drones-are-being-used-to-combat-covid-19/

18. E. Rasmussen. **Drones against vector-borne diseases.** Sci Robot, 5 (2020), 10.1126/scirobotics.abc7642eabc7642 View PDF

19. M. Wojtynek, J.J. Steil, S. Wrede. **Plug, plan and produce as enabler for easy workcell setup and collabrative robot programming in smart factories.**

20. KI-Kunstliche Intelligenz, 33 (2019), pp. 151-161, 10.1007/s13218-019-00595-0 View PDF

21. L. Wang, S. Liu, H. Liu, X.V. Wang. **Overview of human-robot collaboration in manufacturing.** Proceedings of 5th international conference on the industry 4.0 model for advanced manufacturing (2020), pp. 15-58, 10.1007/978-3-030-4621-3_2

22. T.B. Sheridan. **Human–robot interaction: status and challenges.** Hum Factors (2016), 10.1177/0018720816644364 View PDF

23. A. Sheridan, E. Lagerstedt, J. Lindblom. **Foundation for classification of collaboration levels for human-robot cooperation in manufacturing.** Prod Manuf Res, 7 (2019), pp. 448-471, 10.1080/21693277.2019.1645628

24. J.E. Michaelis, A. Siebert-Evenstone, D.W. Shaffer, B. Mutlu. **Collaborative or simply uncaged? Understanding human–Robot interactions in automation.** Proceeding of the CHI conference on human factors in computing systems (2020), pp. 1-12, 10.1145/3313831.3376547

25. A.A. Malik, M.V. Andersen, A. Bilberg. **Advances in machine vision of flexible feeding of assembly parts.** Proc Manuf, 28 (2019), pp. 1228-1235, 10.1016/j.promfg.2020.01.214
ArticleDownload PDFView Record in ScopusGoogle Scholar

26. A.A. Malik, T. Masood, R. Kaousar. **Repurposing factories with robotics in the face of COVID-19.** Sci Robot, 5 (2020), 10.1126/scirobotics.abc2782eabc2782

27. N. Statt. **3D printers are on the front lines of the COVID-19 pandemic.** Verge (2020)

28. R. Tino, R. Moore, S. Antoline, P. Ravi, N. Wake, C.N. Jonita, *et al.* **COVID-19 and the role of 3D printing in medicine.** 3D Print Med (2020), 10.1186/s41205-020-00054-7

29. M. Salmi, J.S. Akmal, E. Pei, J. Wolf, A. Jaribion, S.H. Khajavi. **3D printing in COVID-19: productivity estimation of the most promising open source solutions in emergency situations.** Appl Sci (2020), 10.3390/app10114004

30. X. Mei, H. Lee, K. Diao, M. Huang, B. Lin, C. Liu, *et al.* **Artificial intelligence-enabled rapid diagnosis of patients with COVID-19** Nat Med (2020), 10.1038/s41591-020-0931-3

31. S. Yoo, H. Geng, T. Chiu, S. Yu, D. Cho, J. Heo, *et al.* **Deep learning-based decision-tree classifier for COVID-19 diagnosis from chest X-ray imaging**

32. L. Li, L. Qin, Z. Xu, Y. Yin, X. Wang, B. Kong, *et al.* **Artificial intelligence distinguishes COVID-19 from community acquired pneumonia on chest CT.** Radiology (2020), 10.1148/radiol.2020200905

33. R. Vaishya, M. Javaid, I.H. Khan, A. Haleem. **Artificial intelligence (AI) applications for COVID-19 pandemic.** Diab Metab Syndr, 14 (2020), pp. 337-339, 10.1016/j.dsx.2020.04.012

34. Frak Cimatu. **Baguio general hospital to use AI technology for coronavirus detection.** Rappler (2020)https://www.rappler.com/nation/baguio-general-hospital-artificial-intelligence-technology-coronavirus-detection

35. The Mount Sinai Hospital. Hospital is first in the US to use artificial intelligence to analyze COVID-19 patients. Medical Xpress. (2020). https://medicalxpress.com/news/2020-05-hospital-artificial-intelligence-covid-patients.html.

36. Brunel University. AI auto-scans lung X-rays for coronavirus. (2020). https://healthcare-in-europe.com/en/news/ai-auto-scans-lung-x-rays-for-coronavirus.html.

37. IANS. SMS Hospital brings in robots to serve COVID-19 patients. Economic Times. (2020). https://health.economictimes.indiatimes.com/news/industry/sms-hospital-brings-in-robots-to-serve-covid-19-patients/74827822.

38. Agencies. Covid-19: Robots to deliver food and medicines to patients at Chennai hospital. Economic Times. (2020). https://economictimes.indiatimes.com/news/politics-and-nation/covid-19-robots-to-deliver-food-and-medicines-to-patients-at-chennai-hospital/robot-zafi-in-chennai/slideshow/75026241.cms.

39. Catherine Clifford. COVID-19 pandemic process the need for 'social robots', 'robot avatars', and more, say experts. CNBC. (2020). https://www.cnbc.com/2020/04/03/covid-19-proves-the-need-for-social-robots-and-robot-avatars-experts.html.

40. PTI. Humanoid robots introduced in Chennai Airport. Economic Times. (2018). https://economictimes.indiatimes.com/news/science/humanoid-robots-introduced-at-chennaiairport/.

41. ANI. COVID-19: Mangalore Company develops UV light disinfection robot which sanitizes the entire room in 4 minutes. Economic Times. (2020). https://economictimes.indiatimes.com/news/politics-and-nation/covid-19-mangaluru-company-develops-uv-light-disinfection-robot-which-sanitises-entire-room-in-4-minutes/videoshow/75717325.cms.

42. We Robotics. How Delivery Drones are being used to tackle COVID-19 (Updated). (2020). https://blog.werobotics.org/2020/04/25/cargo-drones-covid-19/.

43. J. Vincent. **After the pandemic, doctors want their new robot helpers to stay.** Verge (2020). https://www.theverge.com/21317055/robot-coronavirus-hospital-pandemic-help-automation

44. Mark Jones. **Floor cleaning robots – could COVID-19 lead to more'public' automation.** Techwire Asia (2020). https://techwireasia.com/2020/05/floor-cleaning-robots-could-covid-19-lead-to-more-public-automation/

45. Chetan Kumar. **Covid-19 Bengaluru doc builds a UV robot to disable viruses,** Times of India (2020). https://timesofindia.indiatimes.com/india/covid-19-bengaluru-doc-builds-uv-robot-to-disable-virus/articleshow/75379109.cms

46. Keutal Sascha **disinfection robots fighting the coronavirus** Tectales (2020) https://tectales.com/bionics-robotics/9-disinfection-robots-fighting-the-coronavirus.html

47. EDx News Bureau. Cobots the new lab assistants. Express Healthcare. (2020). https://www.expresshealthcare.in/interviews/cobots-the-new-lab-assistants/414506/.

48. Mai Tao **Collaborative robots: a helping hand in healthcare** Robot Autom (2020) https://roboticsandautomationnews.com/2020/05/11/collaborative-robots-a-helping-hand-in-healthcare/32189/

49. University of Central Florida. AI can detect COVID-19 in the lungs like a virtual physician, new study shows. Science News. (2020). https://www.sciencedaily.com/releases/2020/09/200930144426.html.

50. UV Resources. UV-C lamps: playing it safe. (2020). http://www.uvresources.com/blog/uv-c-lampsstayingsafe/#:~:text=UV%2DC's/effects/on/the,known/carcinogen/for/human/skin.

51. Haas. **Haas joints project to 3D print ventilator parts and PPE masks**(2020) https://www.haas.co.uk/haas-3d-prints-ppe-masks-for-hospitals/

52. Kleinman Zoe **Coronavirus: can we 3D-print out way out of the PPE shortage** (2020) https://www.bbc.com/news/health-52201696