YUHESEN

NV-magic LiDAR SLAM Navigation for Industrial Applications

User manual V1.0.6

Shenzhen Yuhesen Technology Co., Ltd.

CONTENTS

1. Introduction to Navigation Module

1.1. Navigation Components

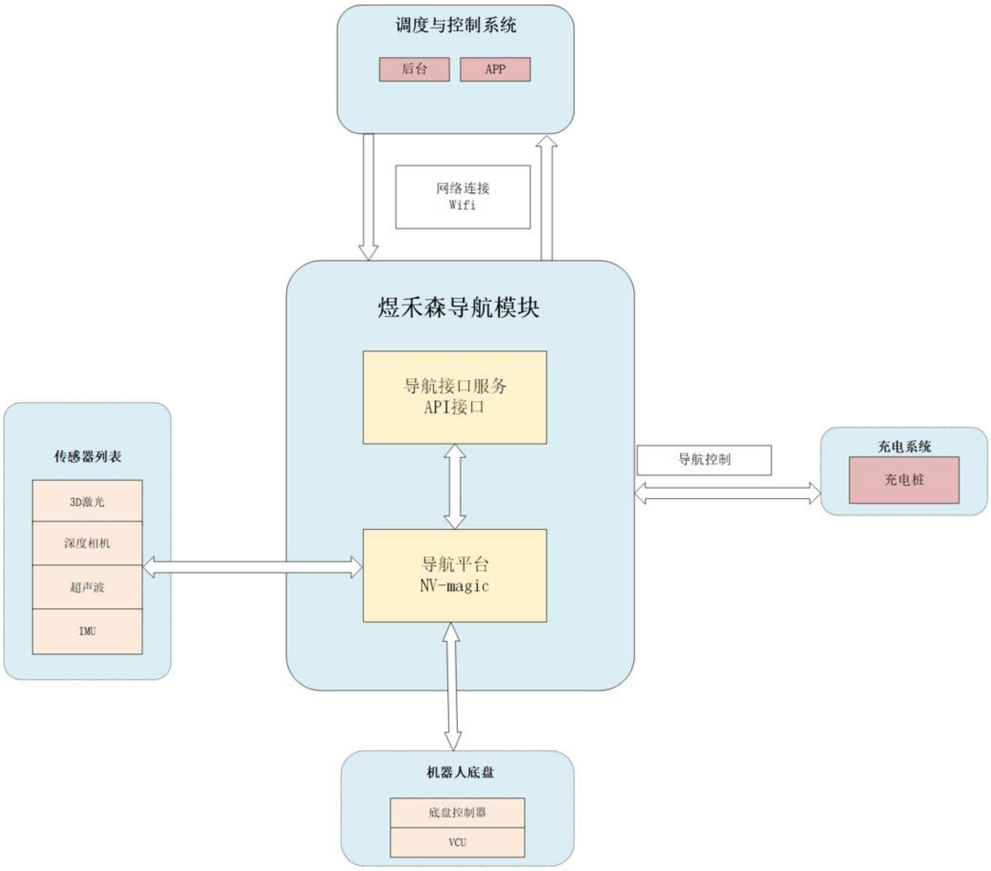


Figure 1.1.1

1.2. Navigation Topology

1.2.1. Electrical Overview

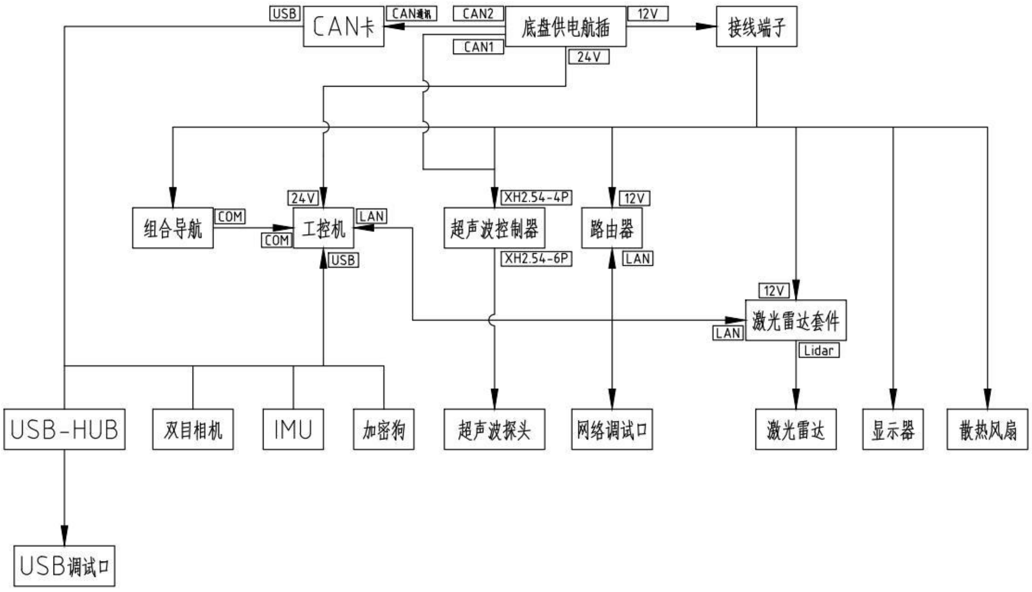


Figure 1.2.1

1.2.2. Network Topology

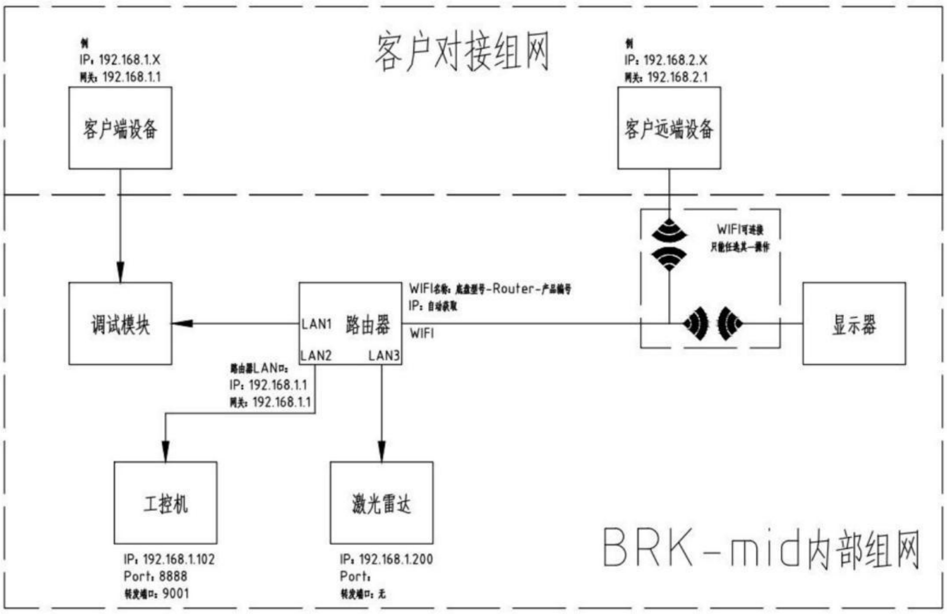


Figure 1.2.2

1.2.3. Communication Topology

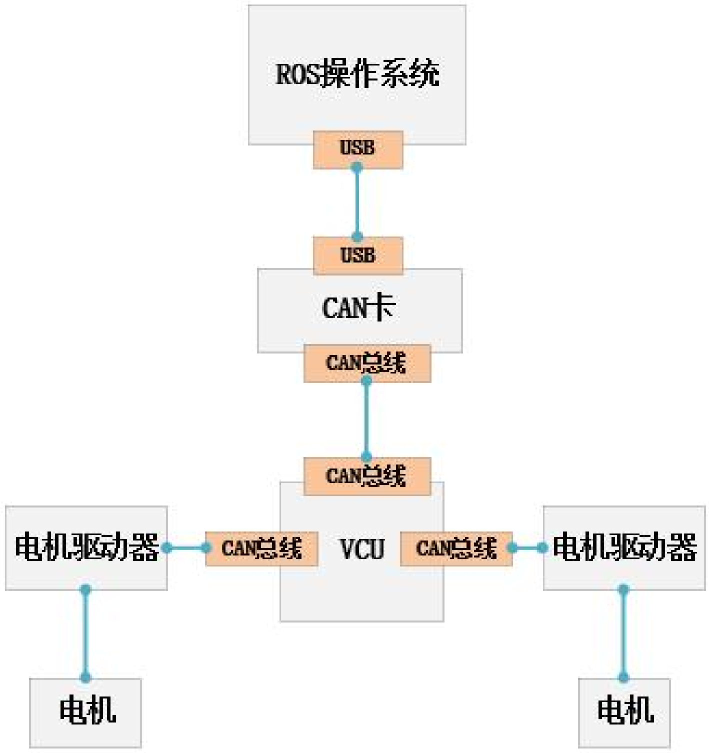


Figure 1.2.3

2. Hardware Configuration Information

2.1. Navigation Hardware Configuration Information1

Computing Unit	APQ	Specification	CPU:	Intel Core i5-8500
			Memory:	16G
			Storage:	512G
		Interface	USB:	6*USB 3.0
3D LiDAR	RS-Helios-16P	Specification	Measurement Range:	0.2m~150m
			Ranging Accuracy:	±2cm

			FOV – Horizontal:	360°
			FOV – Vertical:	-15°~+15°
			Scanning Frequency:	10/20HZ
			Angular resolution (horizontal):	0.2°/0.4°
		Interface	Communication:	LAN port
Depth Camera	Hp60c	Specification	Detection Range:	0.2~4m
			Working Temperature:	-10~50°C
			FOV:	Depth FOV: 73.8°× 58.8° Color FOV: 80.9°× 51.7°
			Resolution@fps	1920×1080 640×480
		Interface	Communication:	USB2.0
IMU	CH110_USB	Specification	Measuring range:	Gyro: ±2000°/s Accelerometer: ±8G
			Resolution:	Gyro: 0.01°/s Accelerometer: 1uG
			Maximum zero offset:	Gyro: 8°/h Accelerometer: 60uG
		Interface	Communication:	USB to Serial port
Router	1900M Gigabit port	Specification	Network standard:	802.11ac
			Wireless network support frequency:	2.4G
CAN Adapter	PCAN-FD	Interface	Communication:	CAN to USB

3. Navigation Functions

3.1 Sensing

Sensors	Functions	Performance
LiDAR	Detecting dynamic and static obstacles.	Wide detecting range
Depth Camera	Detecting dynamic and static obstacles.	Detecting≥20cm*20cm*30cm object
Ultrasonic Radar	Detecting dynamic and static obstacles.	Detecting glass wall

3.2 Mapping

Scene	Functions	Performance
Indoor	Scanning to create a 2D grid map.	30,000 ㎡
Outdoor	Scanning to create a 2D grid map.	150,000 ㎡

3.3 Positioning

Scene	Functions	Performance
Indoor	Using grid-based LiDAR localization in a pre-existing 2D grid map.	Positioning accuracy ±5cm
Outdoor	Using grid-based LiDAR localization in a pre-existing 2D grid map.	Positioning accuracy ±10cm

3.4 Navigation

Scene	Functions	Performance
Indoor	Free navigation between path points.	Point-to-point repeated positioning accuracy of ±5cm, direction tolerance of ±5 degrees.
	Hand-drawn path following.	Point-to-point repeated positioning accuracy of ±5cm, direction tolerance of ±5 degrees.
	Recorded path following.	Point-to-point repeated positioning accuracy of ±5cm, direction tolerance of ±5 degrees.
Outdoor	Free navigation between path points.	Point-to-point repeated positioning accuracy of ±10cm, direction tolerance of ±10 degrees.
	Hand-drawn path following.	Point-to-point repeated positioning accuracy of ±10cm, direction tolerance of ±10 degrees.
	Recorded path following.	Point-to-point repeated positioning accuracy of ±10cm, direction tolerance of ±10 degrees.

3.5 Obstacle Avoidance

Scene	Functions	Performance
Indoor	Obstacle avoidance during free navigation.	Smooth obstacle avoidance without continuous stuttering.
	Hand-drawn path following with obstacle avoidance/stoppage.	Smooth obstacle avoidance without continuous stuttering.
	Recorded path following with obstacle avoidance/stoppage.	Smooth obstacle avoidance without continuous stuttering.
Outdoor	Obstacle avoidance during free navigation.	Smooth obstacle avoidance without continuous stuttering.
	Hand-drawn path following with obstacle avoidance/stoppage.	Smooth obstacle avoidance without continuous stuttering.
	Recorded path following with obstacle avoidance/stoppage.	Smooth obstacle avoidance without continuous stuttering.

3.6 APP

Scene	Functions
Indoor & Outdoor	Mapping operations
	Image editing operations
	Path point management editing operations
	Navigation task scheduling operations
	System setting operations
	Update operations

3.7 API

Scene	Functions
Indoor & Outdoor	Mapping operations
	Image editing operations
	Path point management editing operations
	Navigation task scheduling operations

4. Navigation Update Instructions

4.1. Offline Updates

4.1.1. Obtaining Update Packages

Providing by YUHESEN.

4.1.2. Update Process

- Connect to the robot's Wi-Fi and open the web tool 'yhs_net_updater_tool.html' using a browser.

Figure 4.1.1

- The default IP does not need to be changed, click on 'Login'. Note: Do not use a proxy when logging into the webpage.

Figure 4.1.2

- To view version information, click on “Version Information”.

Figure 4.1.3

- Starting update process.

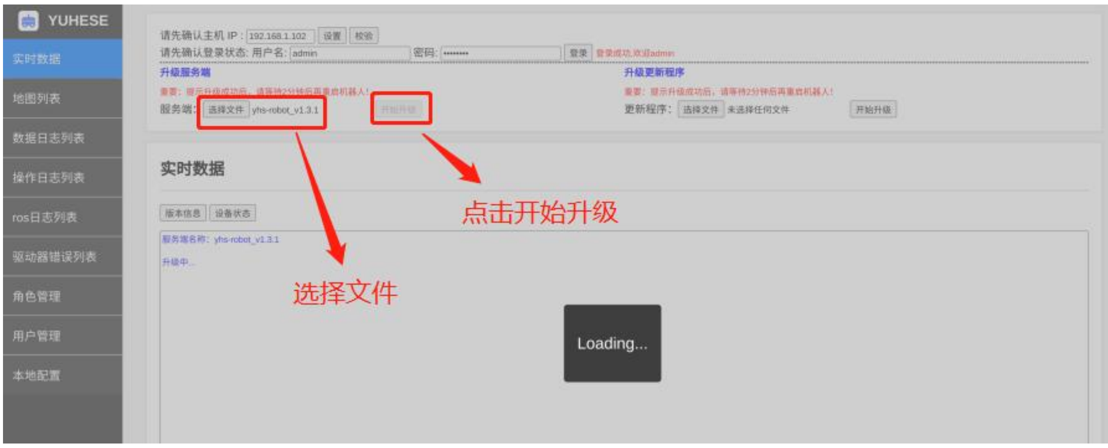


Figure 4.1.4

- Upgrade completed, wait for two minutes and then power off and restart.

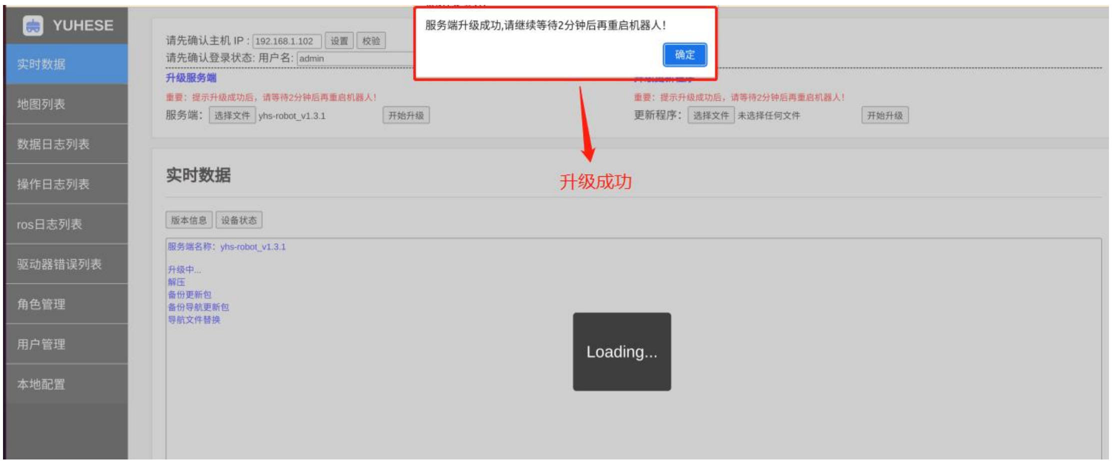


Figure 4.1.5

5. App Usage Instructions
5.1. App Introduction and Terminology

5.1.1. App Introduction
The Yuhesen Magic APP supports Android smartphones and tablets, enabling full control of the robot.

5.1.2. Terminology in the App
- Loop closure: In the mapping process, loop closure refers to identifying and correcting areas in the map that have already been explored to ensure the consistency and accuracy of the map. In essence, it means returning to the starting point of the mapping process after completing the map, which helps create a more accurate map.
- Reference layer: Used to assist in drawing routes, defining areas, or obtaining map dimension information.。
- Recorded Path: Control the robot remotely to move and record its real-time movement trajectory.
- Hand-drawn Path: Manually draw the robot's navigation path on the map.
- Initialization: When the actual position of the robot on the map differs from the position indicated in the app, an initialization of the position is required to ensure successful robot localization.
- Waypoints: Waypoints refer to various points, including starting points, charging points, and navigation points.
- Starting point: A point used for initializing the robot's position or for navigation purposes.
- Charging point: A point where the robot autonomously returns to recharge.
- Mark Current Point: Set the position and orientation of the waypoint to the current position and orientation of the robot. Ensure successful robot localization before setting.
- Navigation point: A point that the robot navigates to freely.
- Path combination: A combination of various types of tasks.
- Temporary point: A navigation point temporarily assigned.
- Free navigation: The robot autonomously moves on paths or trajectories that are not predefined, without relying on pre-drawn paths.
- Line-following navigation: The robot moves by following paths or trajectories drawn on the app.

5.2. App Function Topology

Figure 5.2.1

5.3. App Installation and Connection

5.3.1. App Installation

- Locate the app installation package, click on install as shown in Figure 5.3.1.
- Click on continue installation, as shown in Figure 5.3.2, then wait for the installation to complete.

Figure 5.3.1

Figure 5.3.2

Figure 5.3.3

5.3.2. App Connection

Figure 5.3.4

- Connect your mobile phone or tablet device to the robot's WiFi.
- WWiFi name: YHS-Magic-XX, WiFi password: 12345678
- Enter the default username "admin".
- Enter the default password "admin123".
- Click on login, which will take you to the function selection interface.
- If there are login issues, check if the username and password are entered correctly, or if the device's WiFi is properly connected to the robot.

5.4. App Interface Introduction

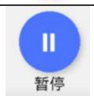
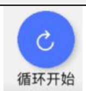
5.4.1. Main Interface

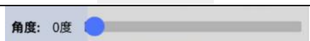
Figure 5.4.1

NV Magic includes several major modules such as map management, path and point management, task management, system settings, and robot status information. The schematic diagram of the app functions is shown in Figure 5.4.1.

5.4.2. Button Definitions

The buttons appearing on the app interface are as follows:

Name	Buttons	Functions
Start		Click to start task execution.
Save		Click to save map
Cancel		Click to cancel mapping
Pause		Click to pause the task in progress.
Resume		Click to resume the task in progress.
Stop		Click to stop the task in progress.
Start and loop		Click to loop and start task execution.
Task selection		Click to select a navigation point, hand-drawn path, recorded path, or combined path as the current task.
Temporary point		Navigation temporary point switch. When clicked to open, allows temporary setting of a navigation point on the map.
Configuration switch		Functional switch button, right for on, left for off.
Waypoint switch-previous one		In the path action editing, click to select the previous point.
Waypoint switch-next one		In the path action editing, click to select the previous or next point.
Save		In the path action editing, save the configured data.
Maps switch		Map selection switch in the top right corner.
Joystick control		Click to toggle the joystick control on or off.
Battery capacity		Display the real-time battery percentage of the robot.
Reference layer		Click to pop up the display settings.
Undo		Click to undo the added waypoints.

Complete		Click to finish adding waypoints.
Connect to point		Click to add a new point.
Save and exit		Click to save the edited map data.
Angle setting		Navigation point angle setting.
Curvature setting		Path curvature setting.
Robot position		Current location of the robot.

5.5. App Homepage Function Introduction

5.5.1. Map Management

Mapping

In the function selection interface, click on map management to enter the map management page, as shown in Figure 5.5.1.

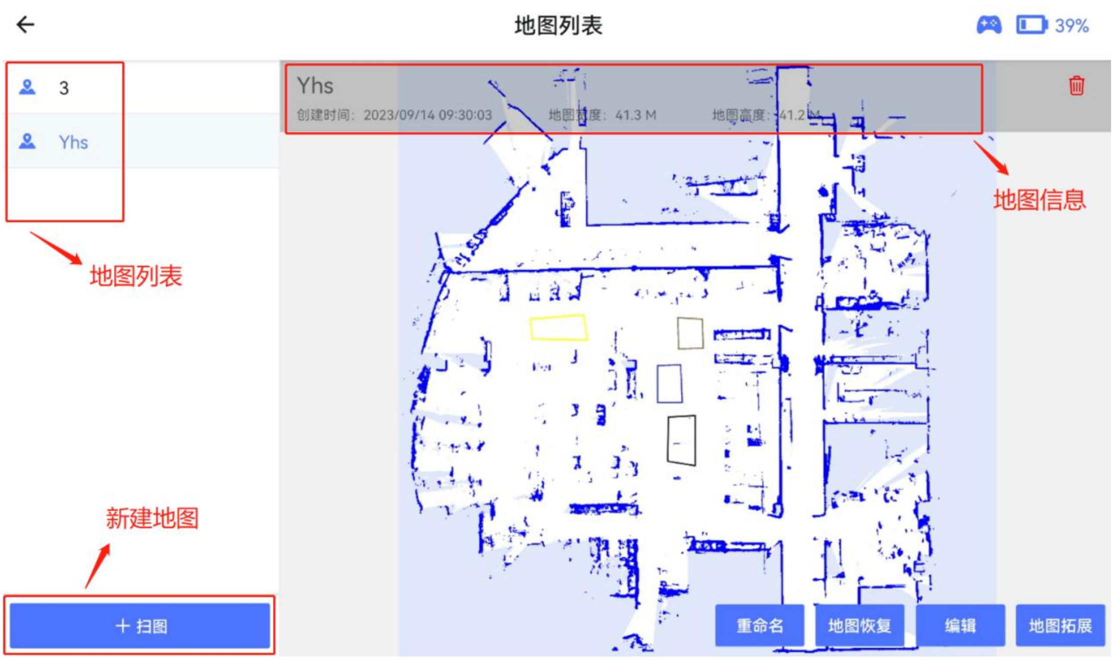


Figure 5.5.1

From the list on the left, you can see the names of the maps that have been created. Click on the corresponding map name to display a thumbnail and detailed information about the map.

- Click on "Scan Map". After entering the map name in the popup box, wait for the map building interface to appear as shown in Figure 5.5.2. You can use the remote control to navigate the robot for mapping. During the mapping process, the map is displayed in real-time on the app. After mapping is completed, remotely guide the robot back to the starting point and wait for the algorithm to optimize the loop closure. If the area is large, you can wait an additional 1-2 minutes for loop closure. Then, click on "Save Map". If there are cracks in the map during loop closure, guide the robot manually to those areas and wait for the map to be repaired.

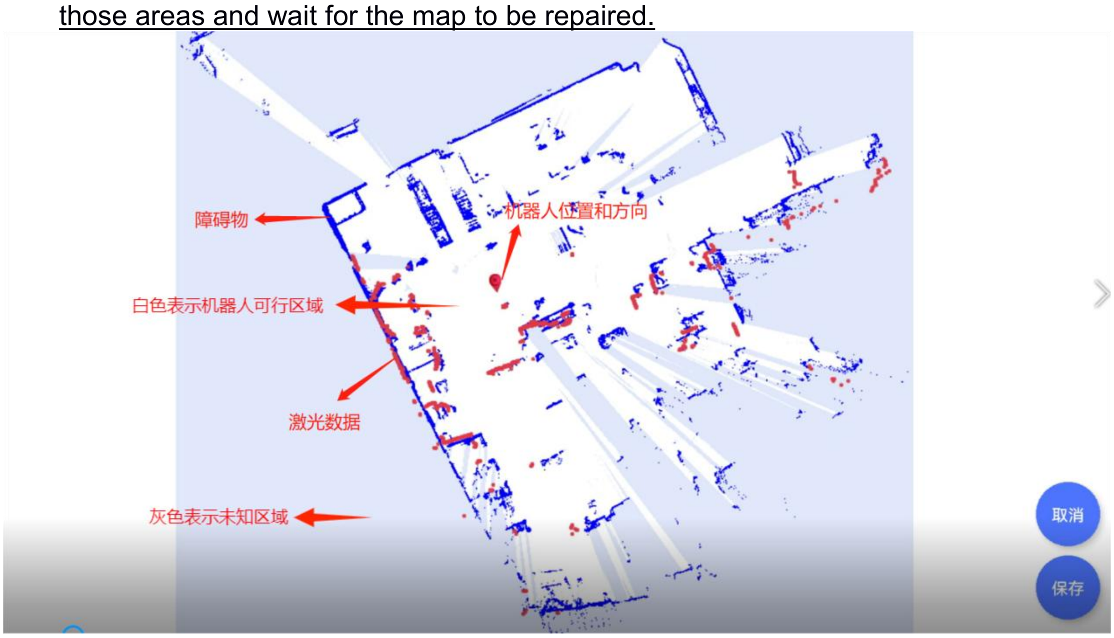


Figure 5.5.2

- Click on the delete button in the top right corner to delete the map. If the current map is in use, you may encounter a situation where it cannot be deleted, as shown in the image below in Figure 5.5.3.

Figure 5.5.3

You can switch to another map before attempting to delete. Refer to Figure 5.5.4 and Figure 5.5.5.

Figure 5.5.4

Figure 5.5.5

Map Editing

After selecting the map, click on the edit button to enter the map editing mode, as shown in Figure 5.5.6.

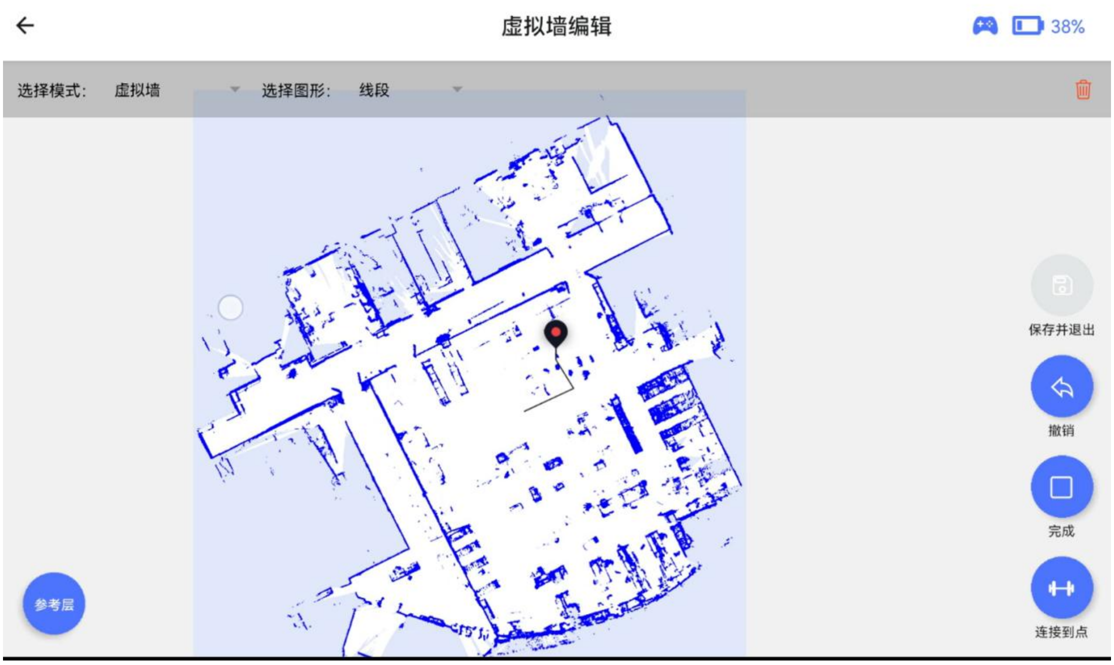


Figure 5.5.6

- **Virtual Wall Editing:** During the mapping process, some low-lying objects may not be detected by the laser radar, or there may be areas that are impassable. In such cases, virtual walls can be set on the map, as shown in Figure 5.5.7.

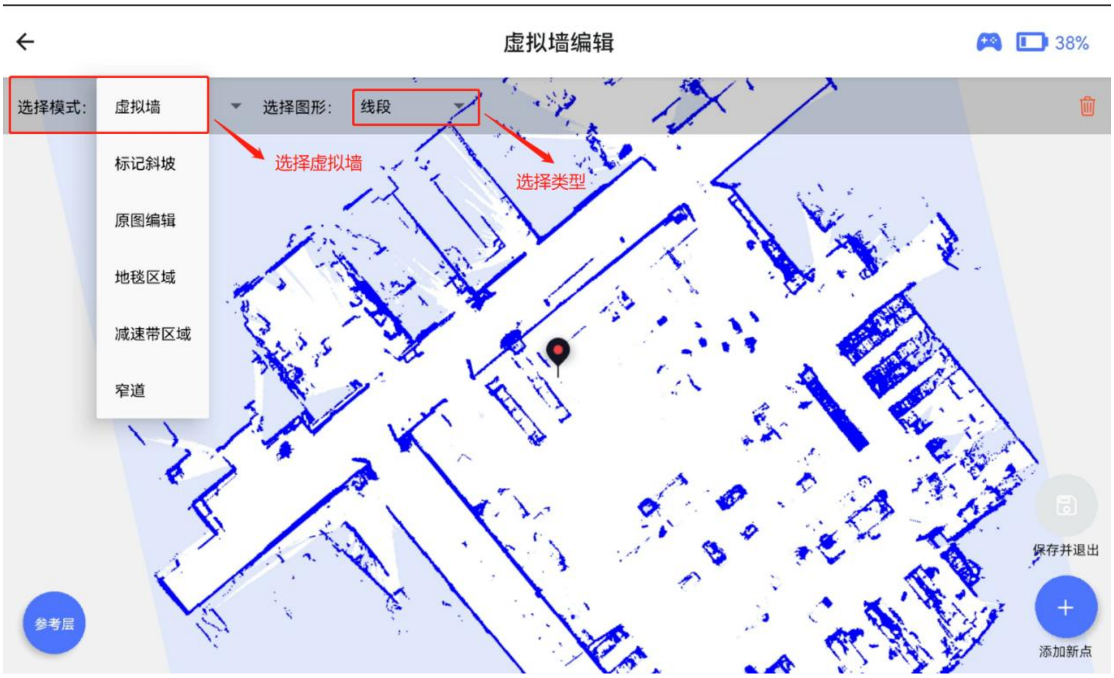


Figure 5.5.7

Select the mode for drawing virtual walls, such as the line segment mode. Click on "Add new point" to create the first point, as shown in Figure 5.5.8. Then, drag the map to change the length and position of the line segment, and click on "Connect to point", as illustrated in Figure 5.5.9, to connect the line segments together. If there is a mistake in this step, click on "Undo" to go back to the previous step. Once the drawing is complete, click on the "Finish" button to end the drawing, as shown in Figure 5.5.10. You can then choose to draw other areas or click on "Save and exit" to complete the map editing.

The drawn virtual walls will appear as shown in Figure 5.5.11.

The modes for drawing virtual walls include: line segment, polygon, circle. In polygon mode, you define an area by determining at least 3 points. For circles, you add new points to determine the center of the circle, and then drag the map to set the radius of the circle.

Figure 5.5.8

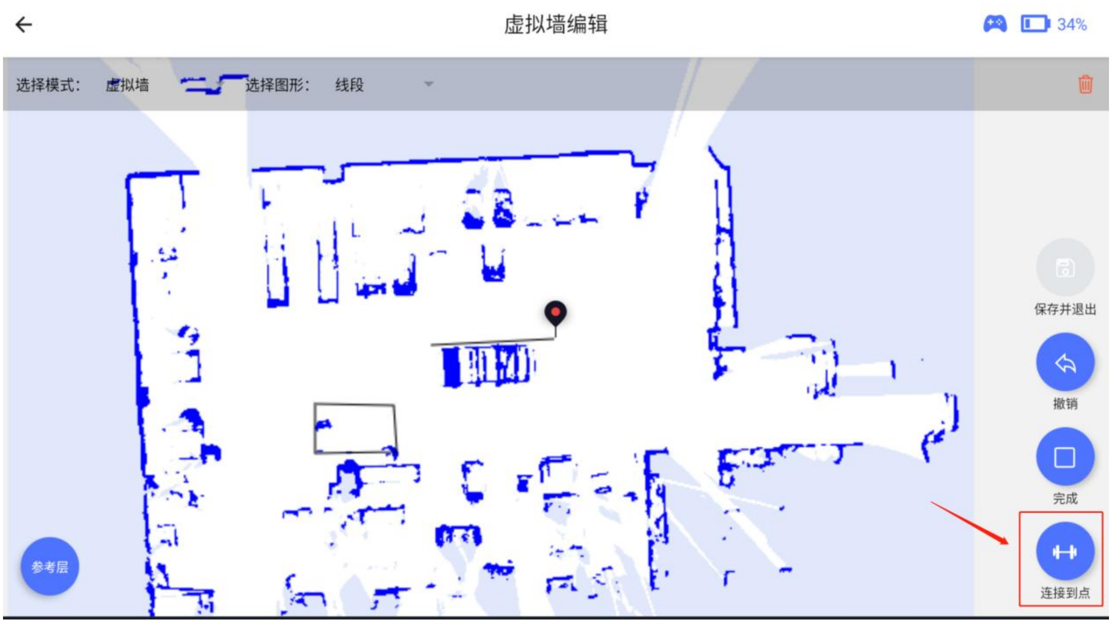


Figure 5.5.9

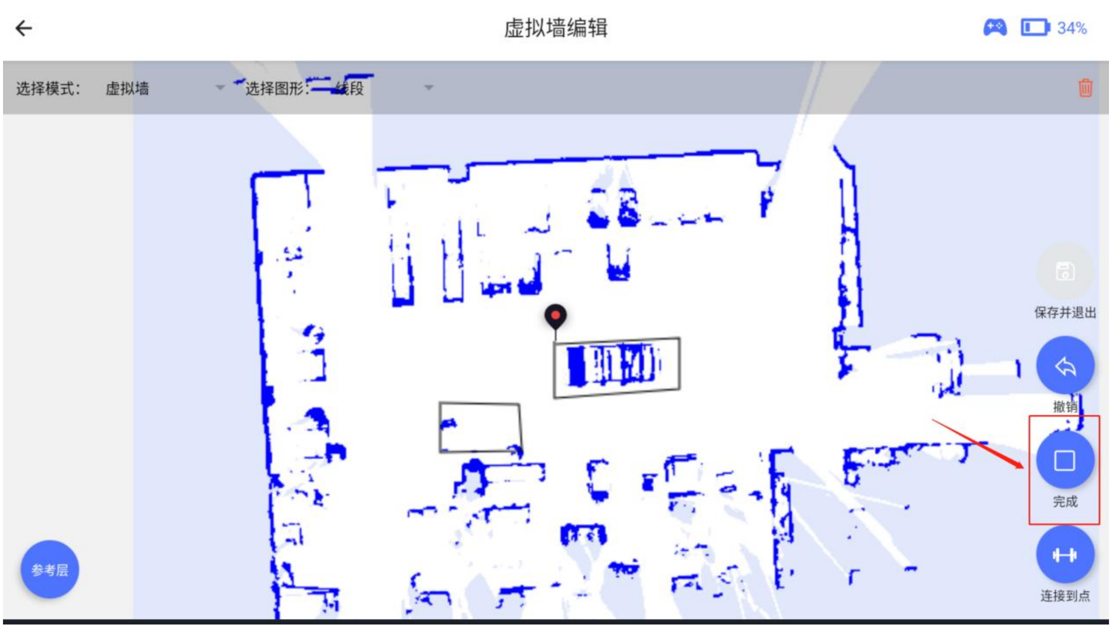


Figure 5.5.10

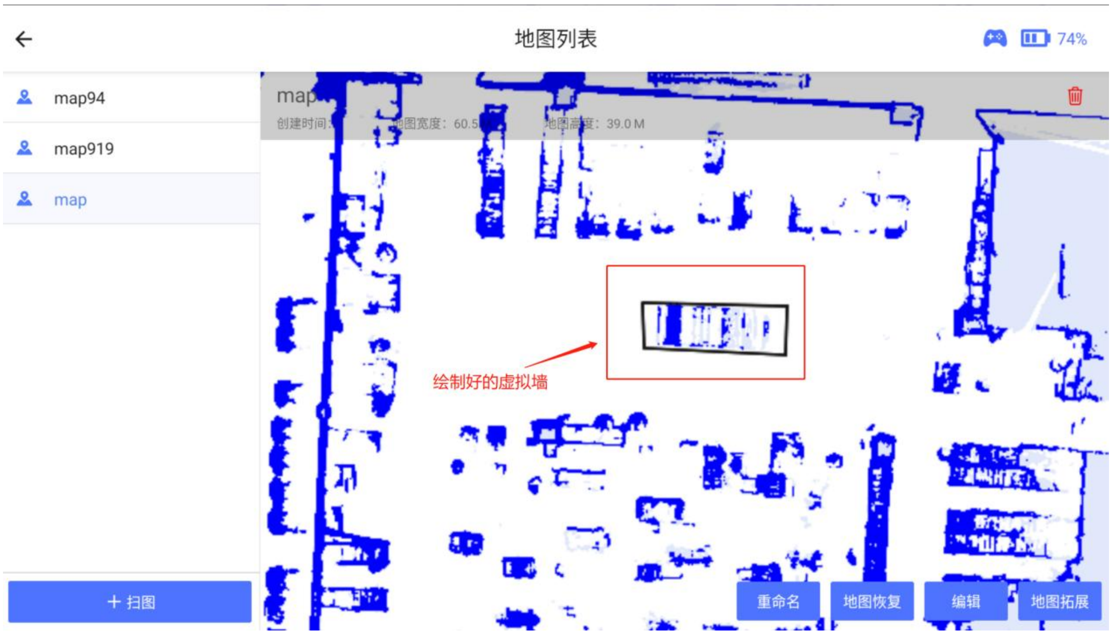


Figure 5.5.11

- **Original Map Editing:** In contrast to virtual wall editing, original map editing involves clearing or filling a specific area on the map. After an area is cleared, the robot can navigate through that space. As shown in Figure 5.5.12, select original map editing; then draw the area to be cleared or filled, as depicted in Figure 5.5.13.

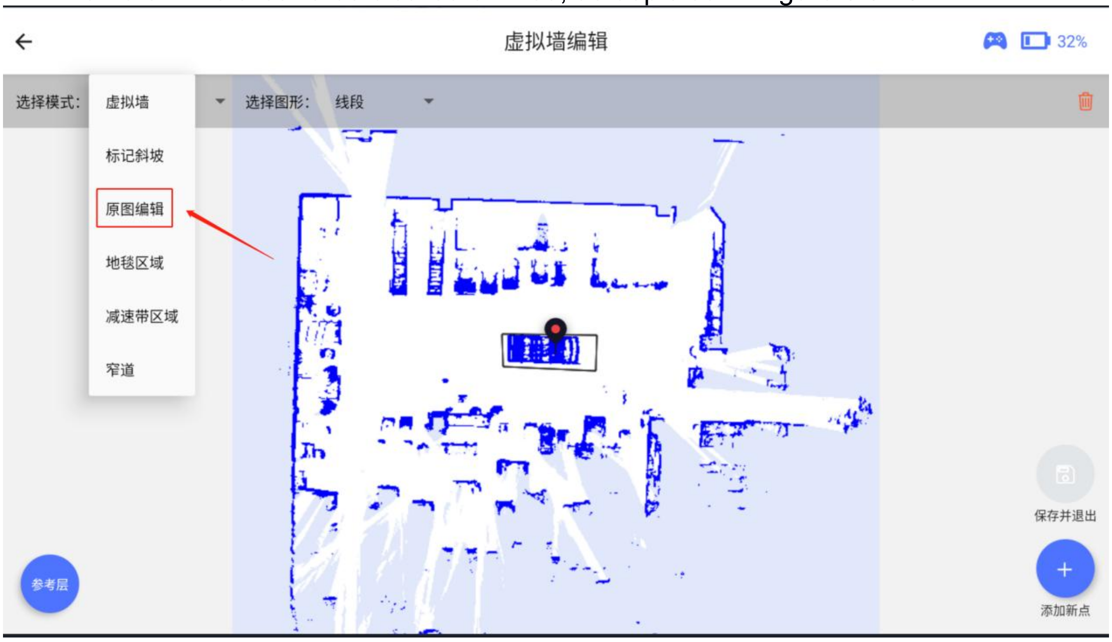


Figure 5.5.12

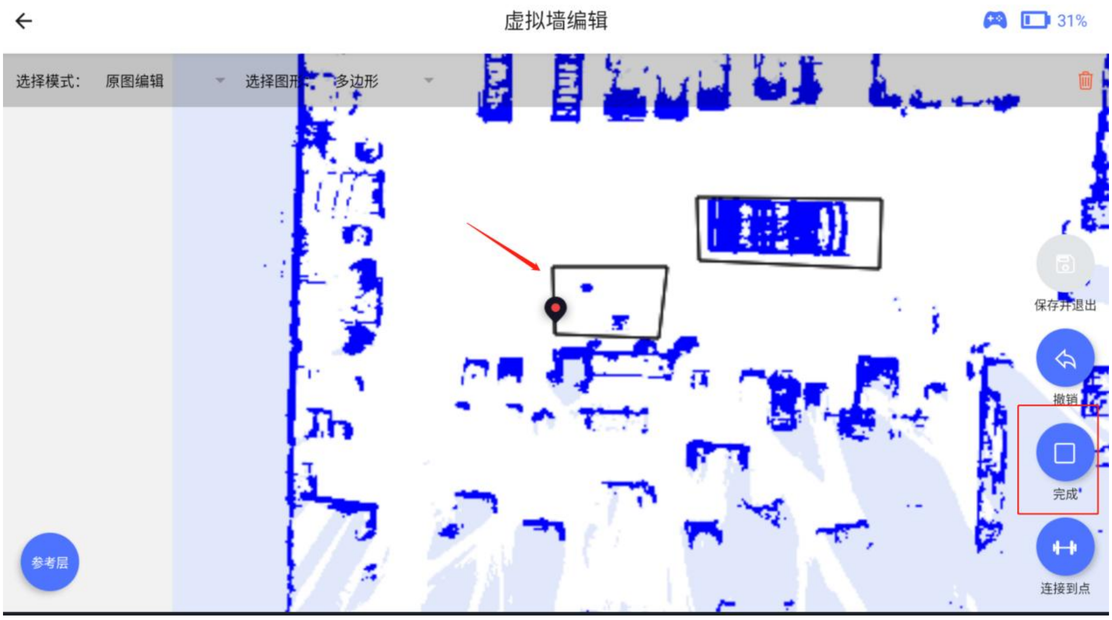


Figure 5.5.13

Then, click on "Finish" and choose to fill or clear the area, as shown in Figure 5.5.14. When clearing an area, all obstacles within that area will be removed, as illustrated in Figure 5.5.15. When filling an area, the entire region will be filled with obstacles, as shown in Figure 5.5.16.

Figure 5.5.14

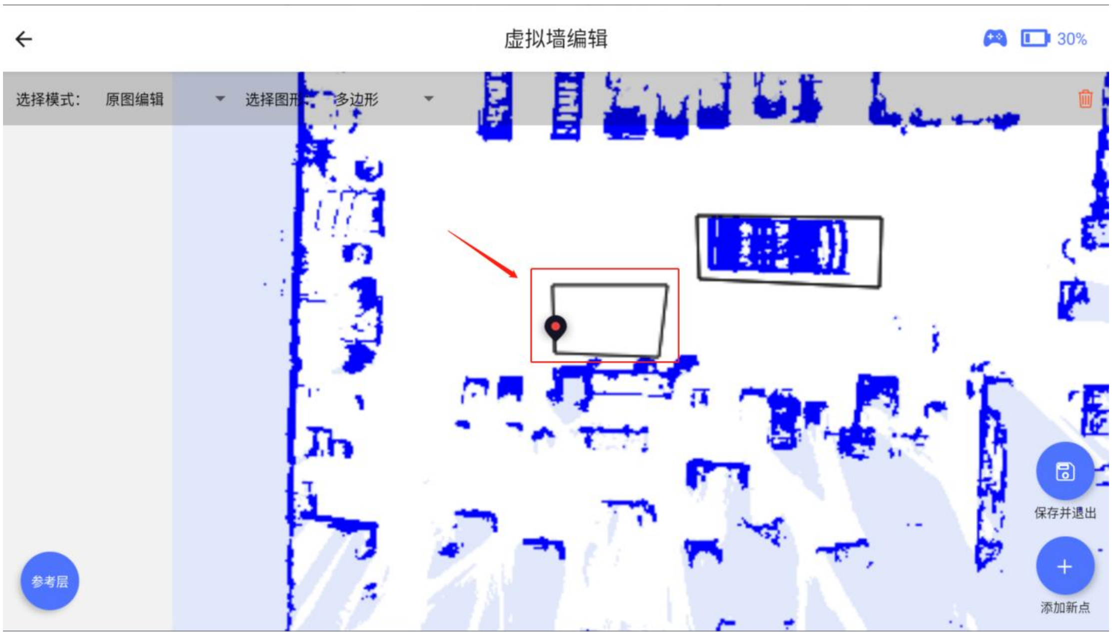


Figure 5.5.15

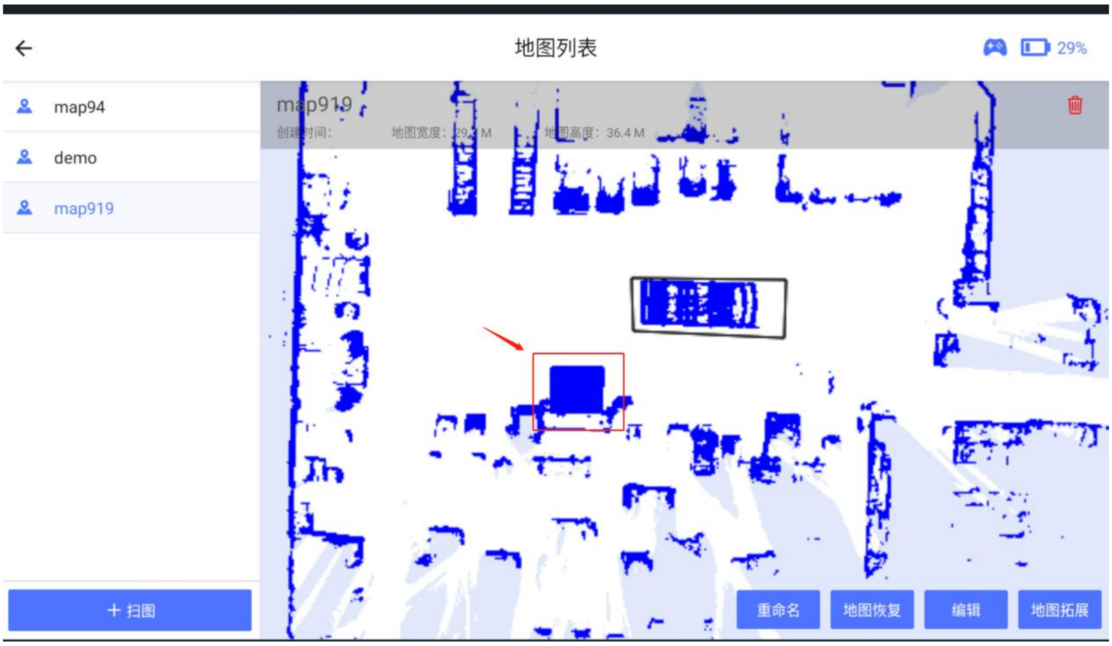


Figure 5.5.16

- **Marking Slopes:** Generally, robots navigate on ground with slight slopes without considering it as an obstacle. However, when navigating in scenes with slopes exceeding a certain degree, it is necessary to mark these areas as slope regions. Otherwise, the laser radar may identify the sloped surface as an obstacle, preventing the robot from passing through the slope, as shown in Figure 5.5.17.

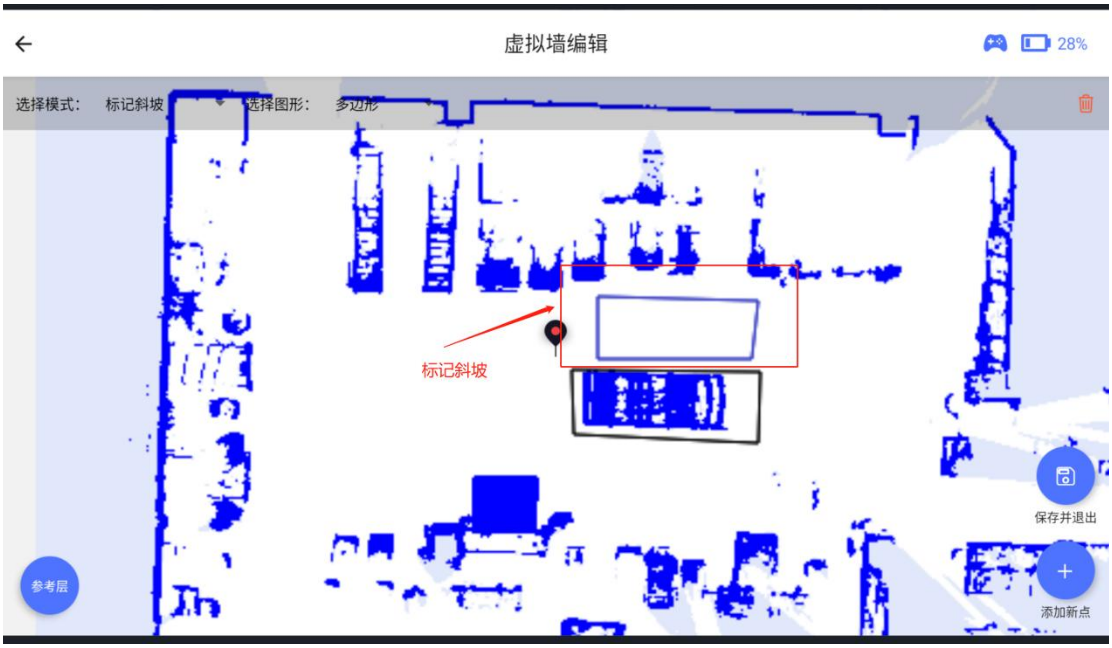


Figure 5.5.17

- **Deceleration Zone:** When encountering speed bumps or areas where the robot needs to slow down, the robot should decelerate accordingly. In such cases, a deceleration zone can be set up, as shown in Figure 5.5.18. Once the robot enters this area, it will travel at a reduced speed.

Figure 5.5.18

- **Narrow Passage:** When the robot needs to navigate through narrow spaces such as doors or other tight areas, it typically avoids them. However, by setting up a narrow passage zone, the robot can pass through such areas. The actual width of the narrow passage should be slightly wider than the robot's actual width, with approximately 20 centimeters of clearance on each side. Additionally, a hand-drawn path should be sketched in the middle of the narrow passage on the map. The robot will follow this hand-drawn path during navigation. The narrow passage setup is illustrated in Figure 5.5.19.

Figure 5.5.19

- If you want to delete these areas, click on "Edit", select the area, and then click on "Delete" in the top right corner to remove the region. Refer to Figure 5.5.20 for guidance.

Figure 5.5.20

Map Renaming

Click on "Rename", enter the new name for the map, and click "OK" to confirm. Please note that Chinese characters are not supported in the map name.

5.5.2. Path and Point Management

Click on "Path and Point Management" in the function options interface to access the Path and Point Management page, as shown in Figure 5.5.21.

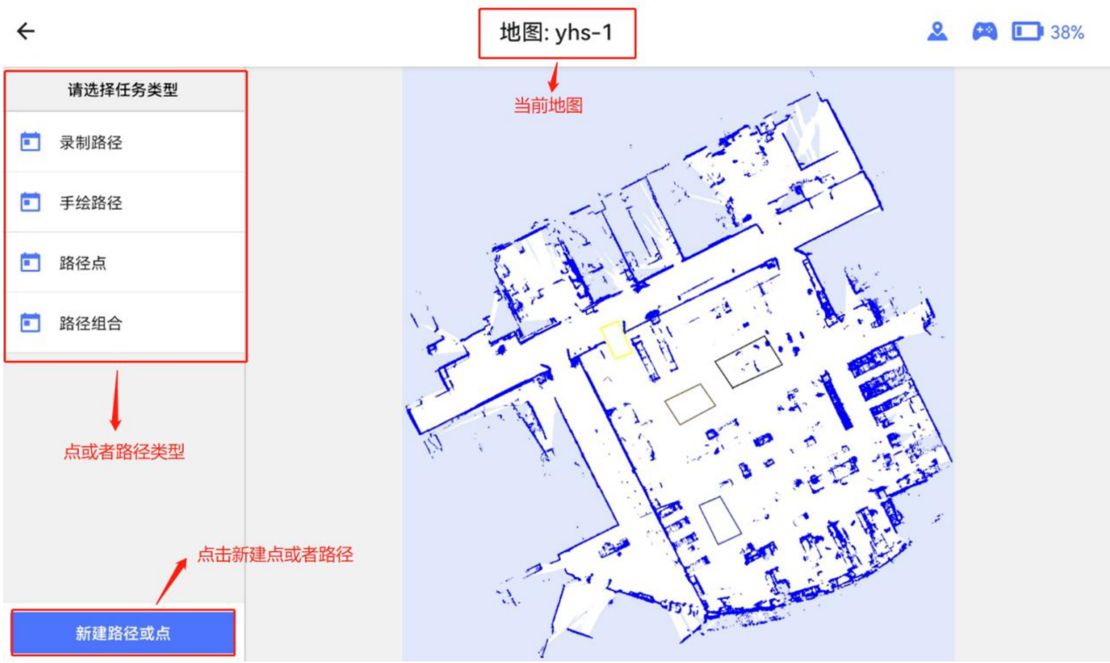


Figure 5.5.21

Hand-drawn Path

- To create a new hand-drawn path, click on "New Path or Point", then select "Draw New Path", as shown in Figure 5.5.22.

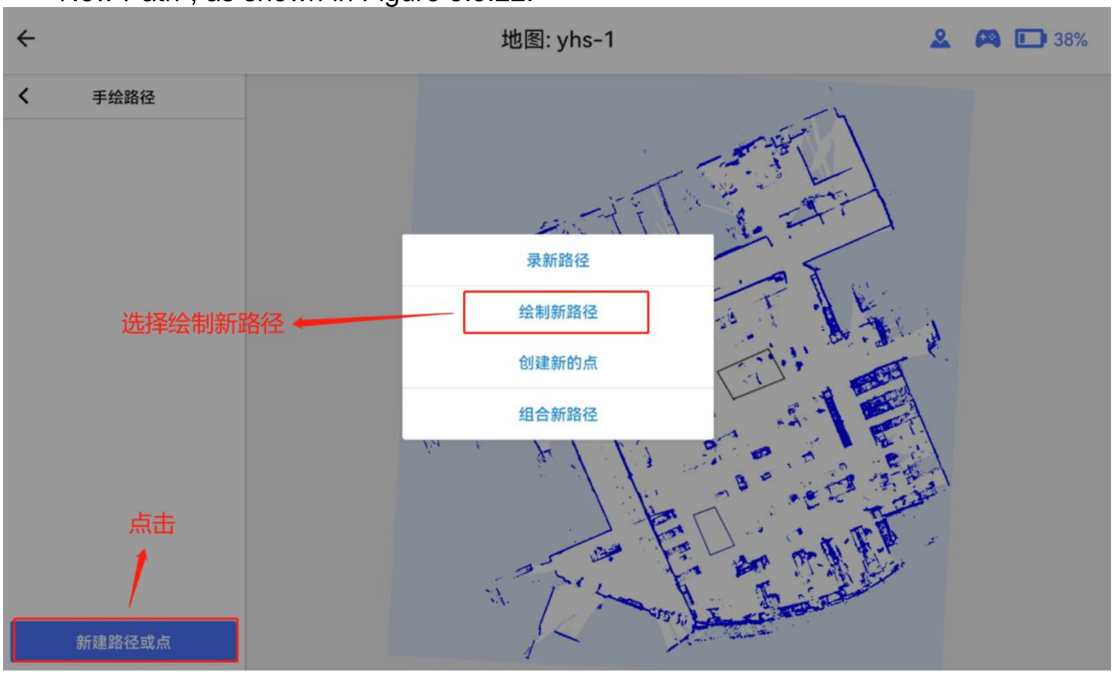


Figure 5.5.22

To connect points and form line segments, drag the map to position new points, as shown in Figure 5.5.23. The operation is similar to drawing areas.

Adjust the curvature of path segments based on the actual scenario, as depicted in Figure 5.5.24.

Before and after generating a segment between two points, the system automatically checks if the path is safe. If it's deemed unsafe, the segment cannot be created. <u>For instance, when drawing a hand-drawn path, if there is an obstacle in the path, the laser radar will detect it, resulting in the path not being generated.</u>

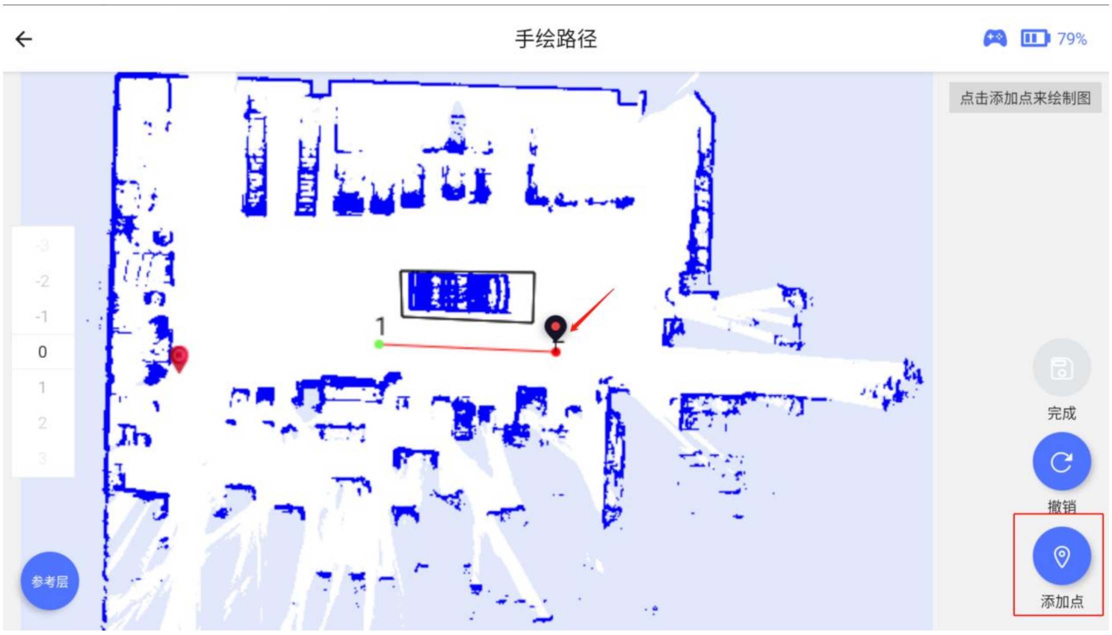


Figure 5.5.23

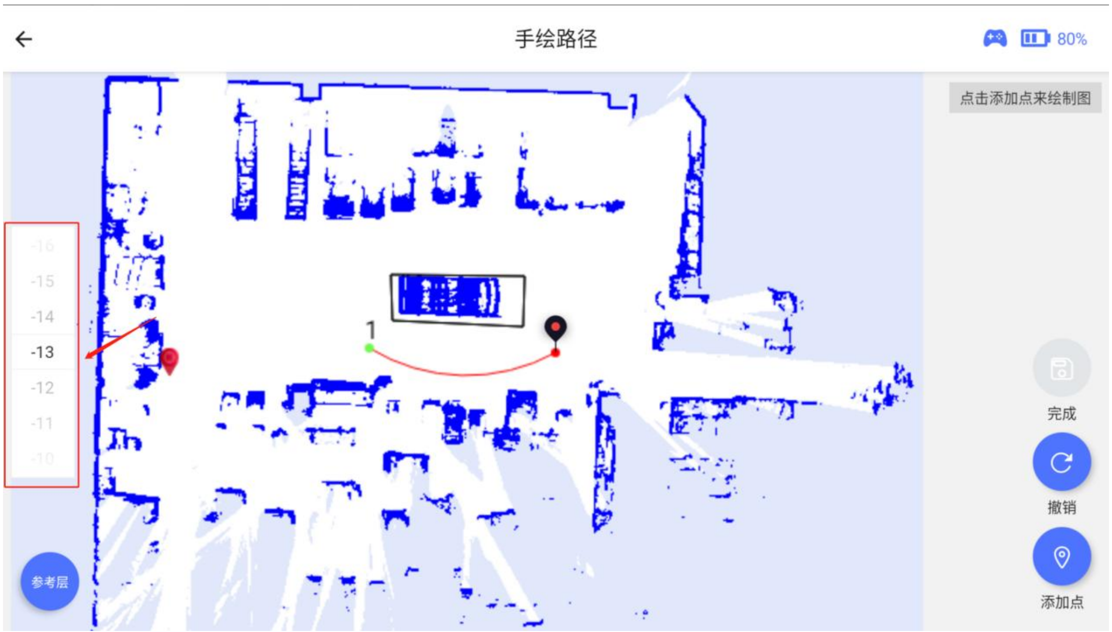


Figure 5.5.24

If you want to draw a closed-loop hand-drawn path, you can move the last point to overlap with another point when confirming the final point, as shown in Figure 5.5.25. Then, click on "Add Point", select "Closed Loop", and click "OK" to complete the process, as illustrated in Figure 5.5.26.

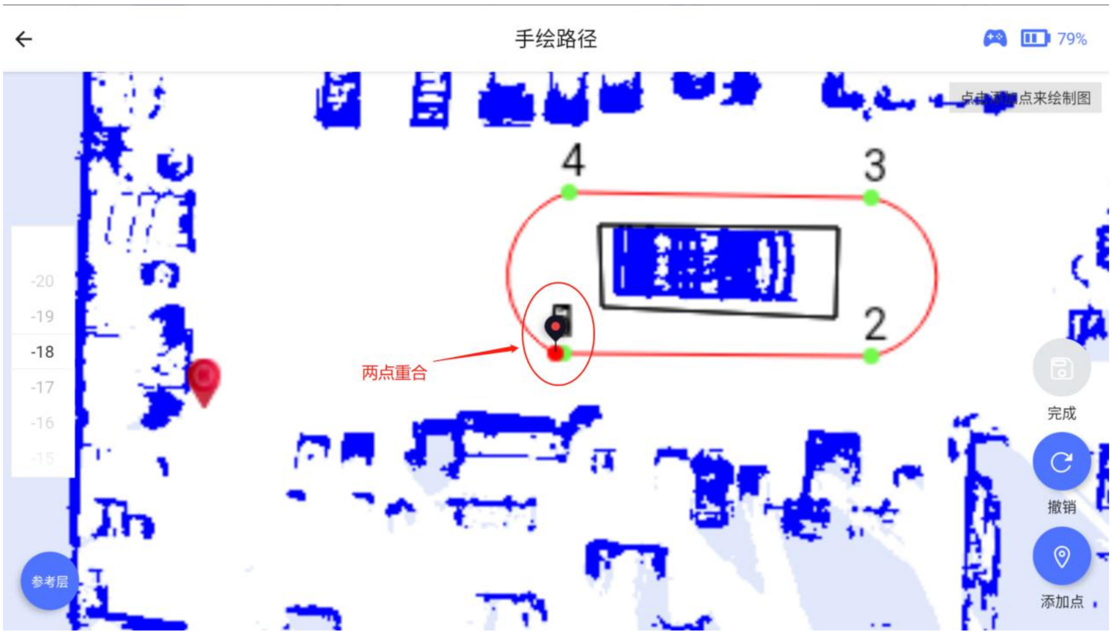


Figure 5.5.25

Figure 5.5.26

The generated path is depicted in Figure 5.5.27.

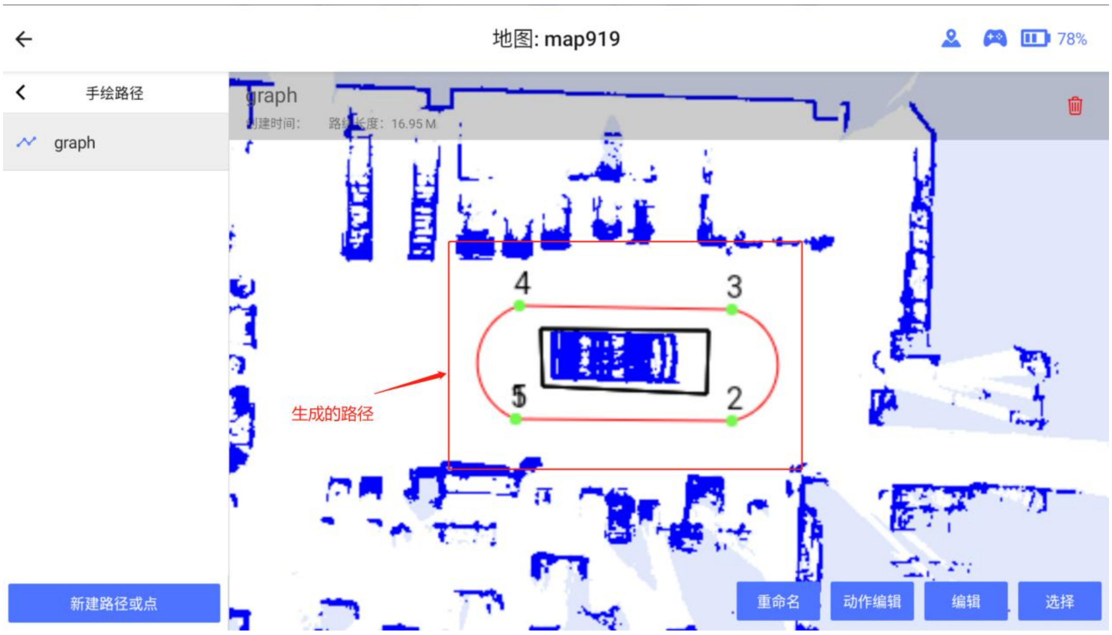


Figure 5.5.27

- **Hand-drawn Path Action Editing:** In practical applications, the robot may need to pause or turn to a certain angle at a specific point along the hand-drawn path during navigation. This can be achieved by utilizing action editing. To enter the action editing mode, refer to Figure 5.5.28.

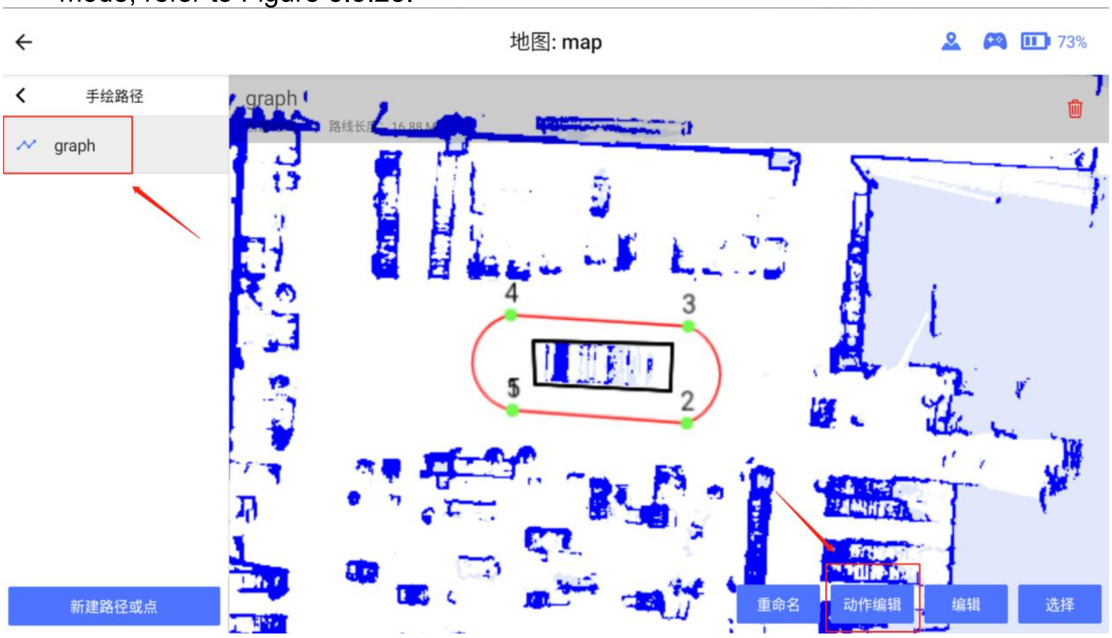


Figure 5.5.28

By long-pressing on the path, an action point will appear, as shown in Figure 5.5.29.

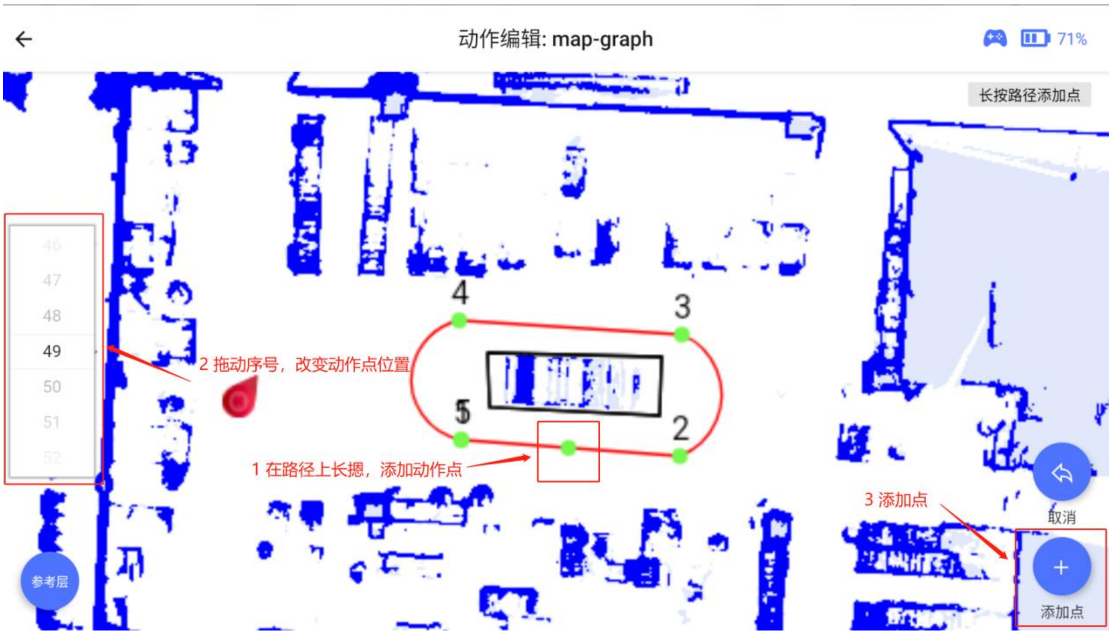


Figure 5.5.29

Enter the name of the action point and click "OK."

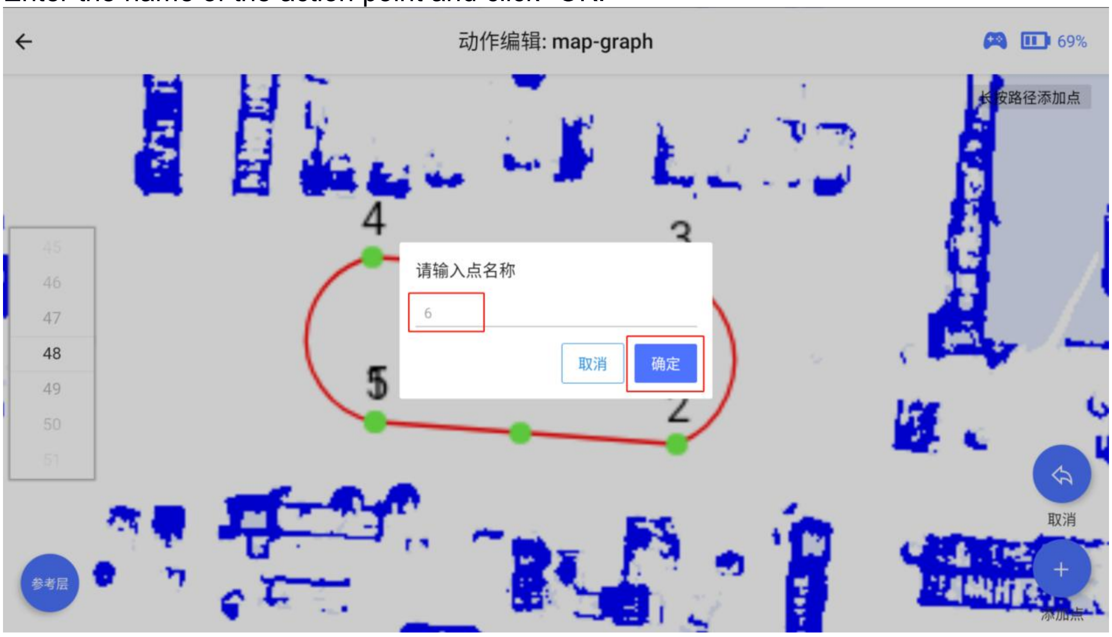


Figure 5.5.30

Once the action point is created, you can configure the pause duration and turning angle for the action point, as shown in Figure 5.5.31. It's important to note that after the robot reaches the action point, it will first turn and then pause. The angle within the action point can be confirmed using the robot's angle information on the main interface of the app, as depicted in Figure 5.5.32.

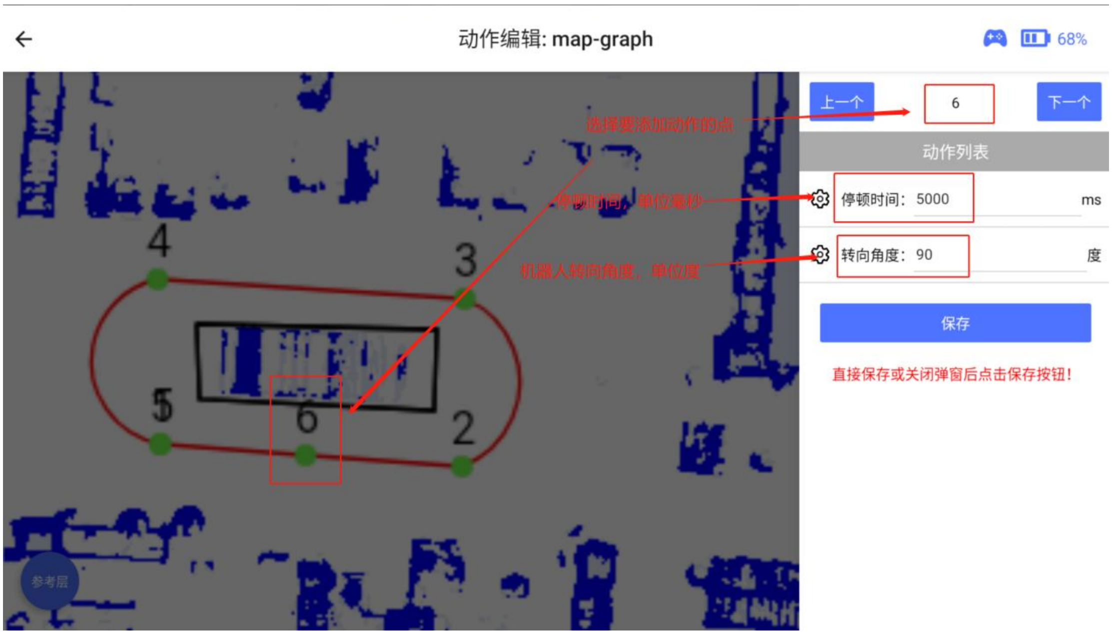


Figure 5.5.31

Figure 5.5.32

- **Hand-drawn Path Renaming:** Click on "Rename", enter a new path name, and click "OK", as shown in Figure 5.5.33. Please note that the name does not support Chinese characters.

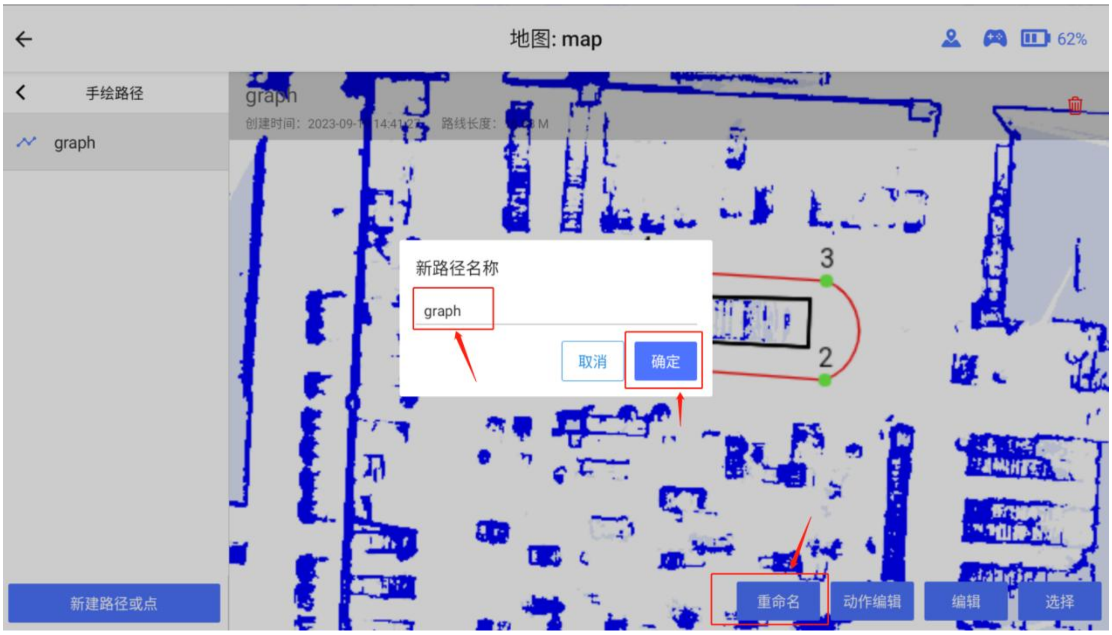


Figure 5.5.33

Hand-drawn Path Editing: Select the hand-drawn path, click on "Edit", then click on a point or line on the hand-drawn path, and choose an object, as shown in Figure 5.5.34.

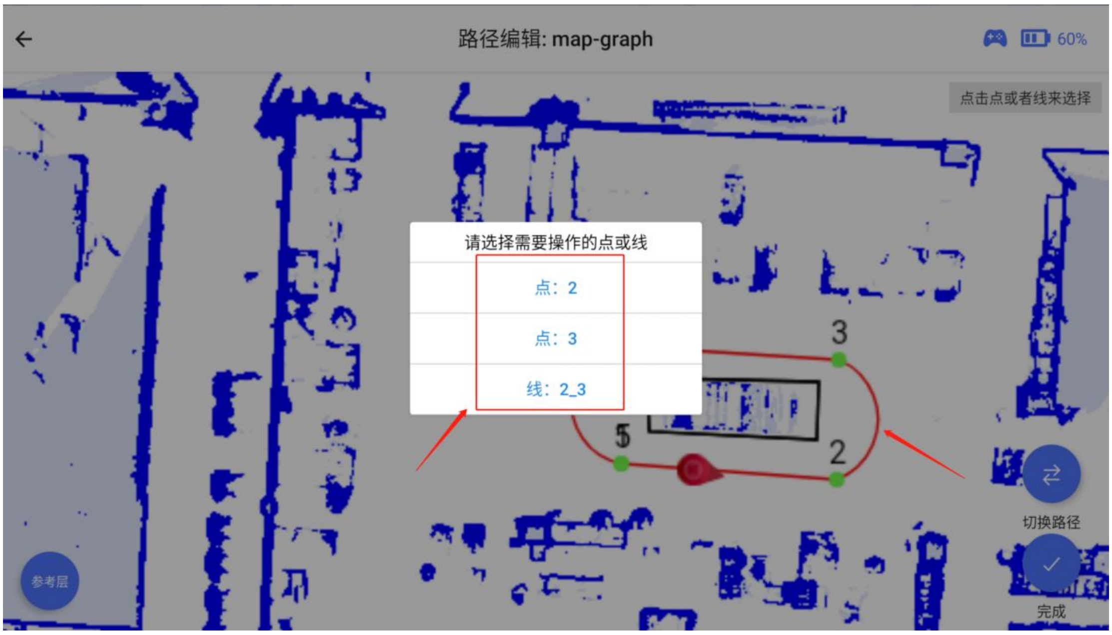


Figure 5.5.34

If you are editing a point on the hand-drawn path, as shown in Figure 5.5.35, you can delete the point or rename it. After deleting point "5", the outcome will be as illustrated in Figure 5.5.36.

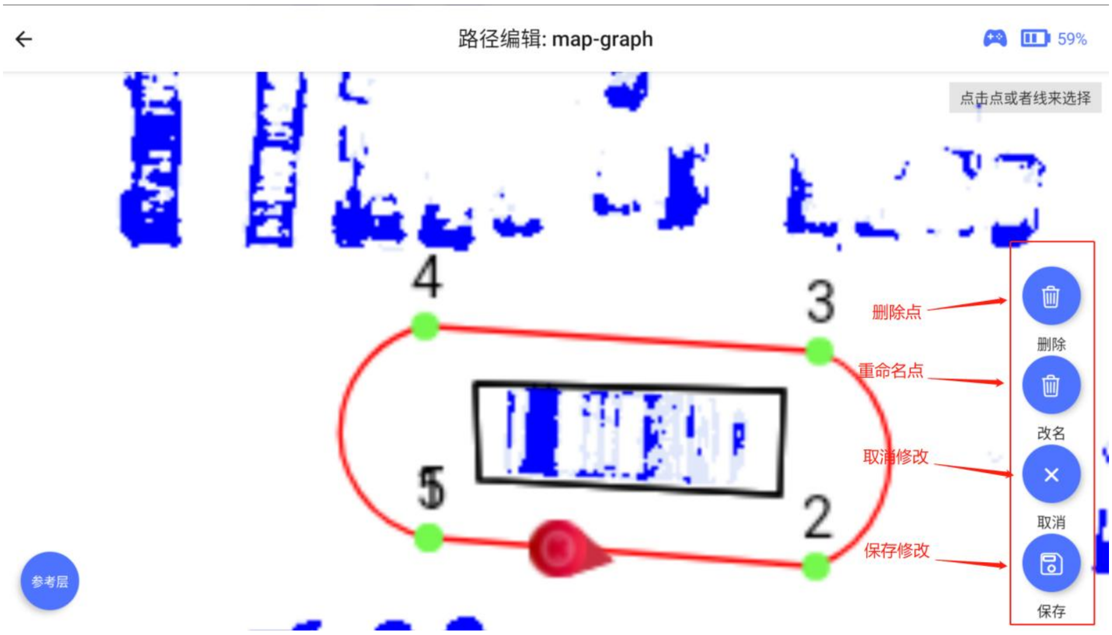


Figure 5.5.35

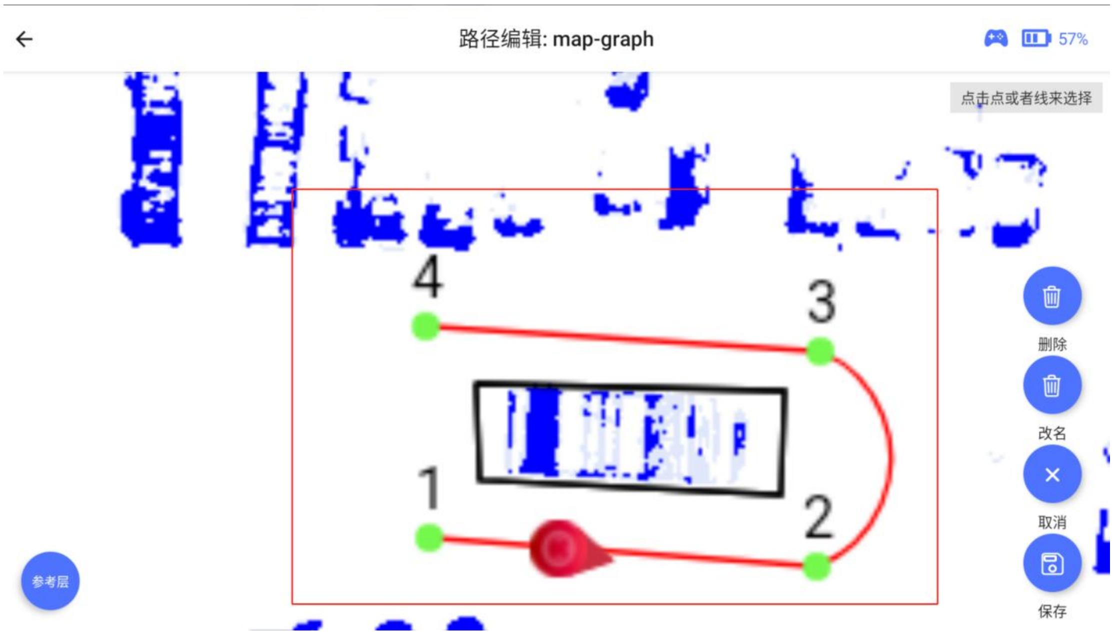


Figure 5.5.36

If you are editing a line segment on the hand-drawn path, as depicted in Figure 5.5.37, you can adjust the curvature of the line segment. After changing the curvature of segment "3_4," the result will appear as shown in Figure 5.5.38.

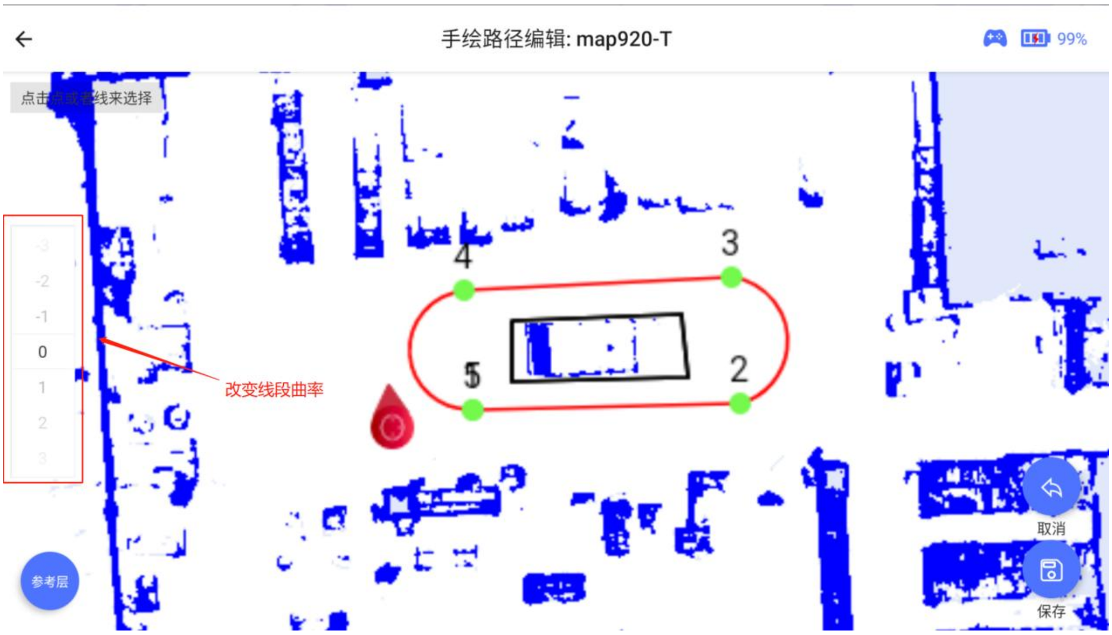


Figure 5.5.37

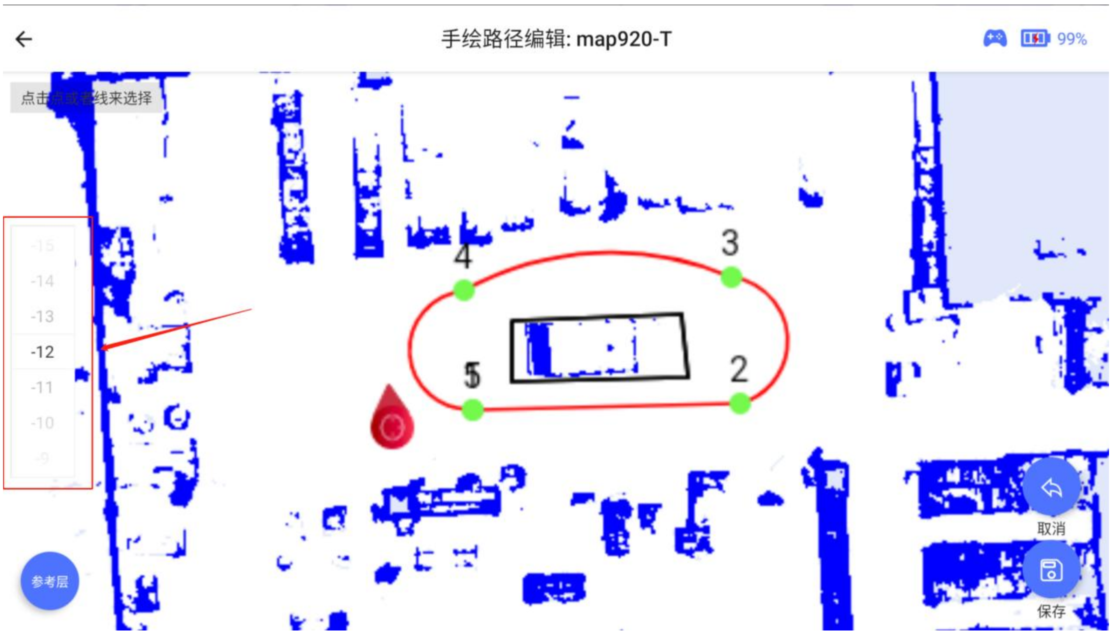


Figure 5.5.38

Deleting a Hand-drawn Path: Select a hand-drawn path, click on the delete button at the top right corner, to remove the hand-drawn path.

Recording Path

- To create a new recording path, click on "New Path or Point", then select "Draw New Path", as shown in Figure 5.5.39.

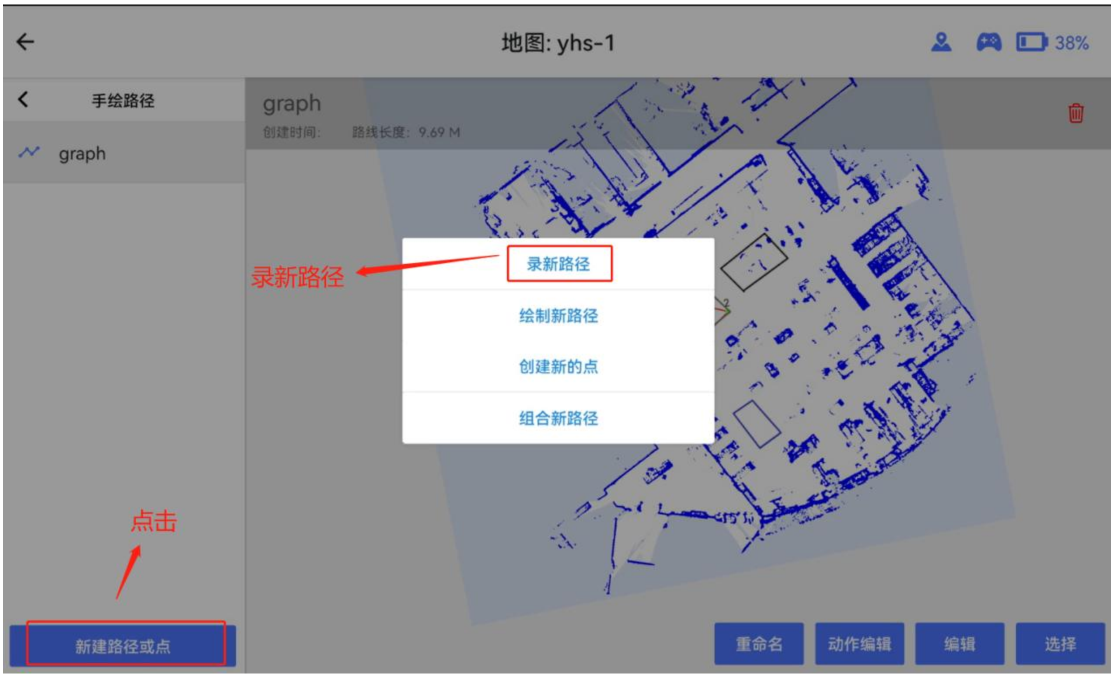


Figure 5.5.39

To remotely control the robot's movement, with the app displaying the trajectory in real-time, ensure that when recording a path, there are no intersections in the path. After completing the recording, you can click on "Save" to store the path or choose to cancel the recording process, as shown in Figure 5.5.40.

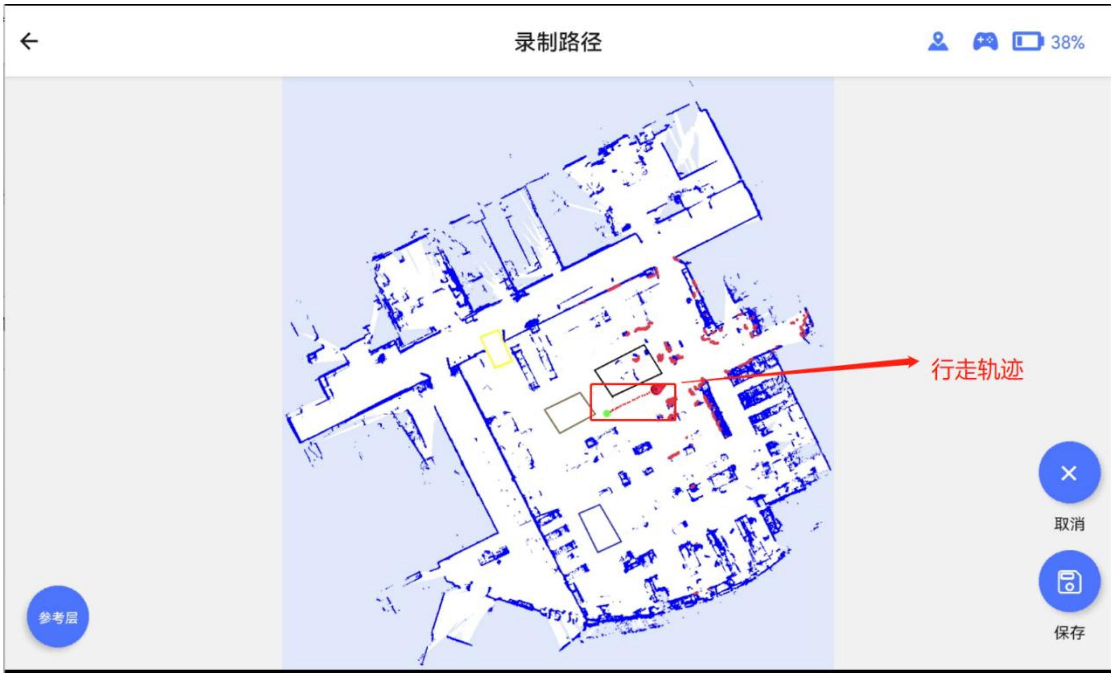


Figure 5.5.40

- **Recording Path Action Editing:** In practical applications, the robot may need to pause or turn to a certain angle at a specific point along the recorded path during navigation. This can be achieved by using action editing. Adding action points on a recorded path follows a similar process to adding them on a hand-drawn path. Select a recorded path, click on Action Editing, as shown in Figure 5.5.41.

Figure 5.5.41

Next, long-press on the recorded path to add an action point, then proceed to edit the actions associated with that point, as shown in Figure 5.5.42 and Figure 5.5.43.

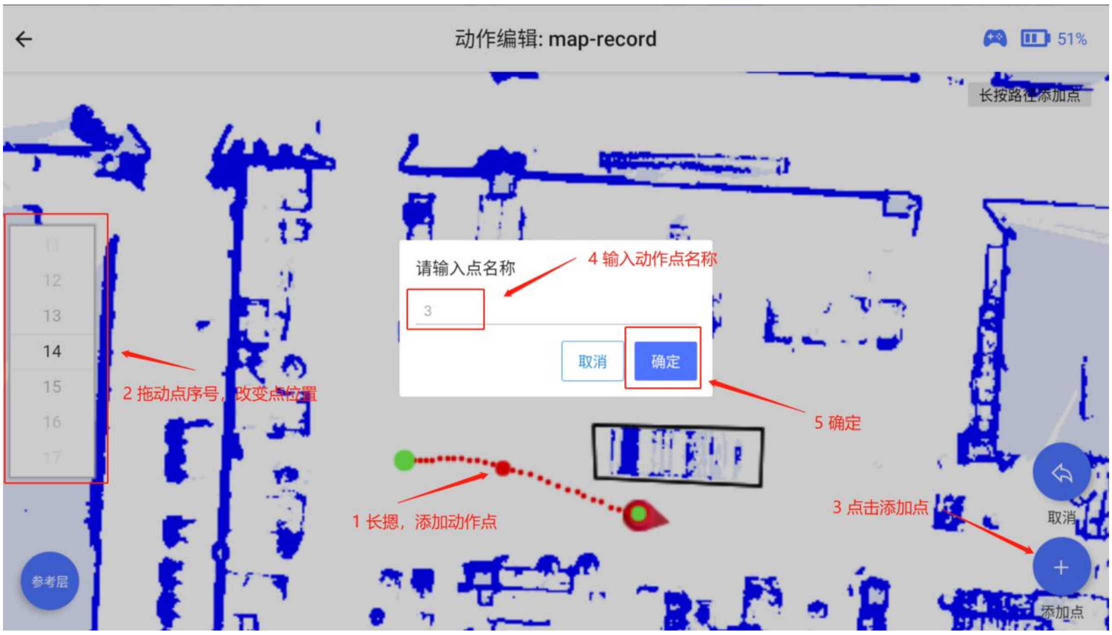


Figure 5.5.42

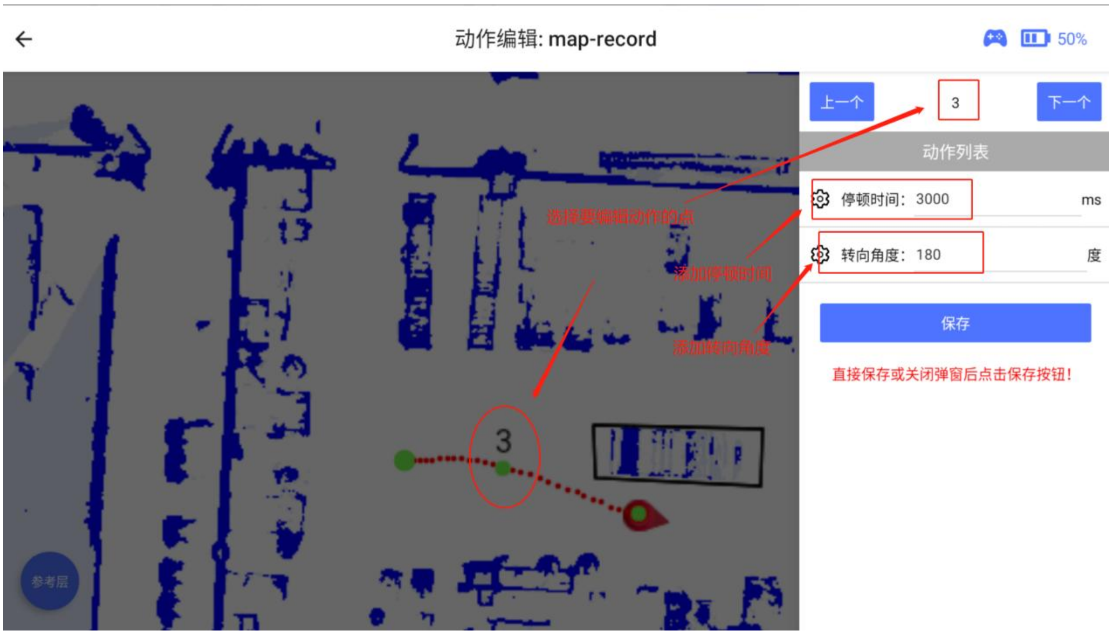


Figure 5.5.43

- **To rename a recorded path:** Select a recorded path, click on "Rename", enter a new name, and click "OK" to confirm. Please note that the name does not support Chinese characters.

- **Editing a recorded path:** In recorded path editing, you can modify existing action points by deleting them, renaming them, or changing their positions. Select a recorded path, click on "Edit", as shown in Figure 5.5.44.

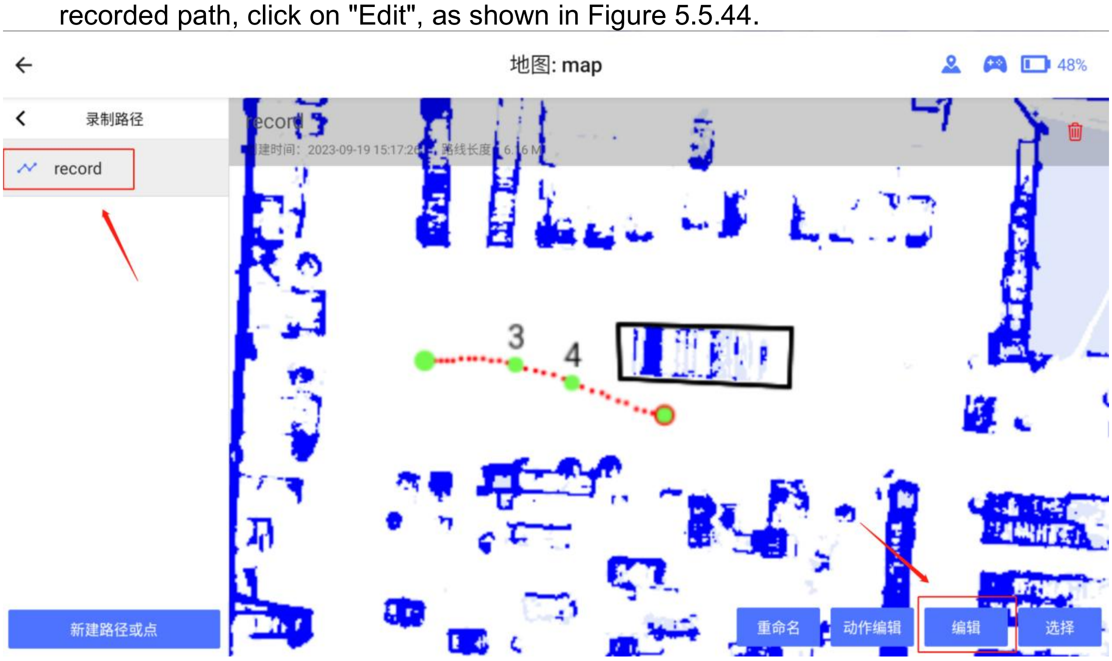


Figure 5.5.44

Upon entering the editing interface, click on "Select", as demonstrated in Figure 5.5.45. Then, choose the action point you want to edit, as depicted in Figure 5.5.46.

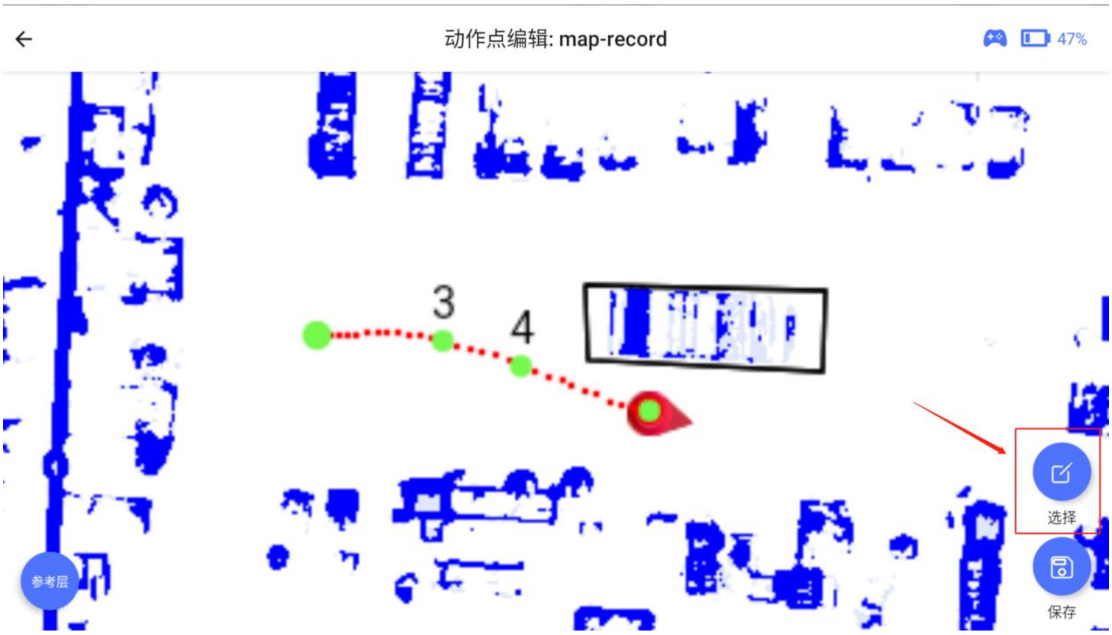


Figure 5.5.45

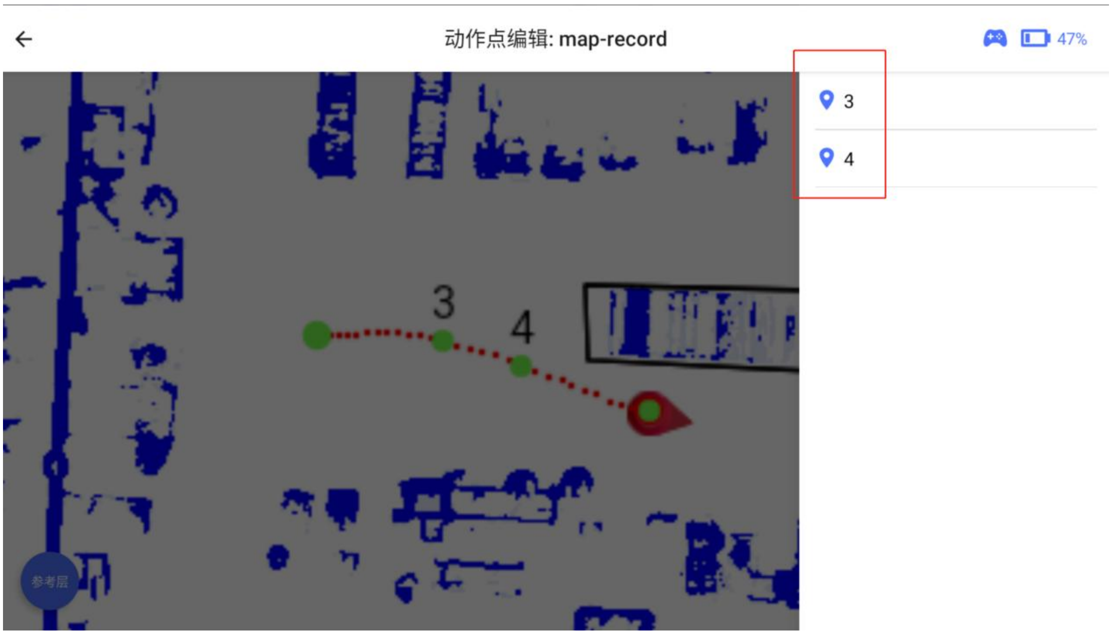


Figure 5.5.46

Once you have selected the action point for editing, you can rename it, delete it, or modify its position, as shown in Figure 5.5.47.

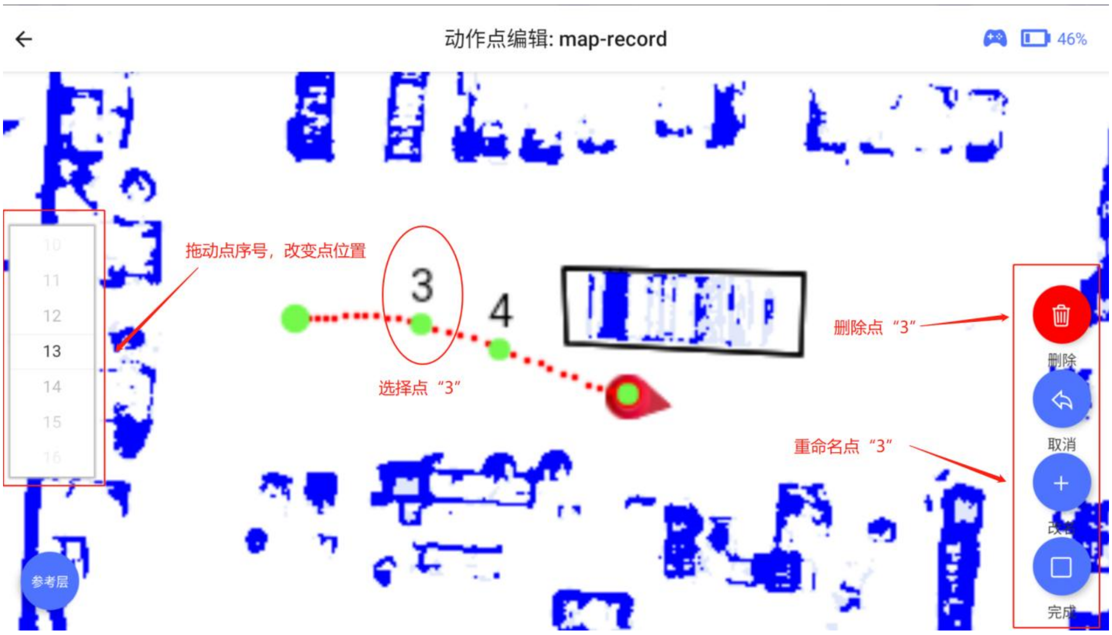


Figure 5.5.47

- **Deleting a recorded path:** Select a recorded path, then click on the delete button located in the top-right corner to remove the recorded path.

Path Points

- **Creating a new point:** Path points include: starting point, navigation point, and charging point. Click on "New Path or Point", then select "Create a New Point", as shown in Figure 5.5.48.

Figure 5.5.48

Creating a navigation point: Configure the settings as shown in Figure 5.5.49.

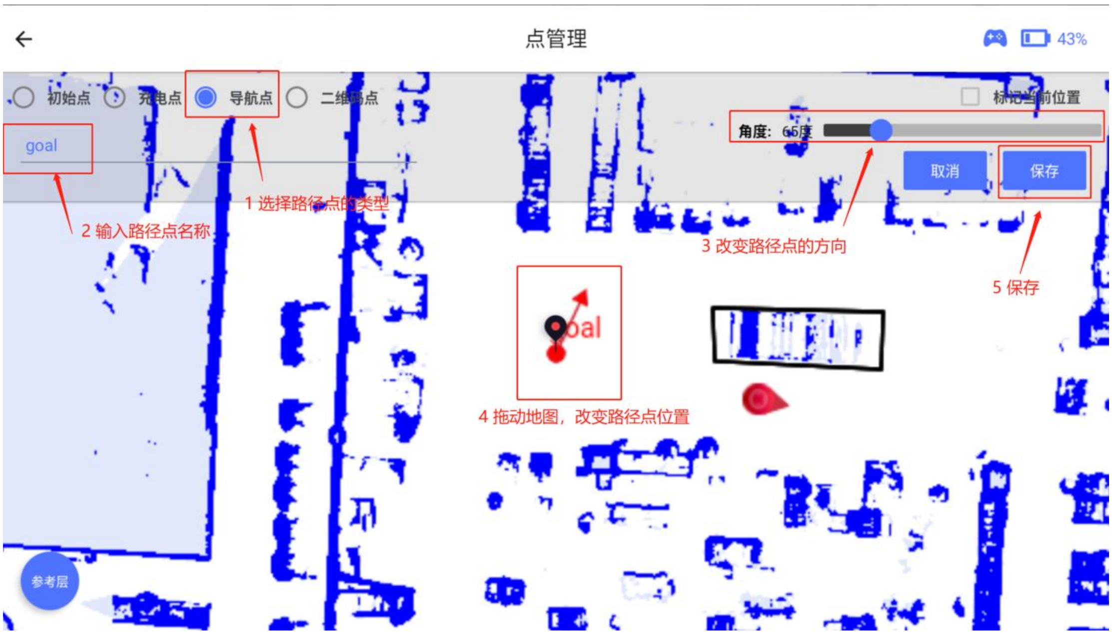


Figure 5.5.49

Creating a charging point: Before setting up the charging point, remotely control the robot to align and make contact between the charging end and the robot's power end, ensuring they are in a vertical position to each other. Then proceed with the settings as shown in Figure 5.5.50.

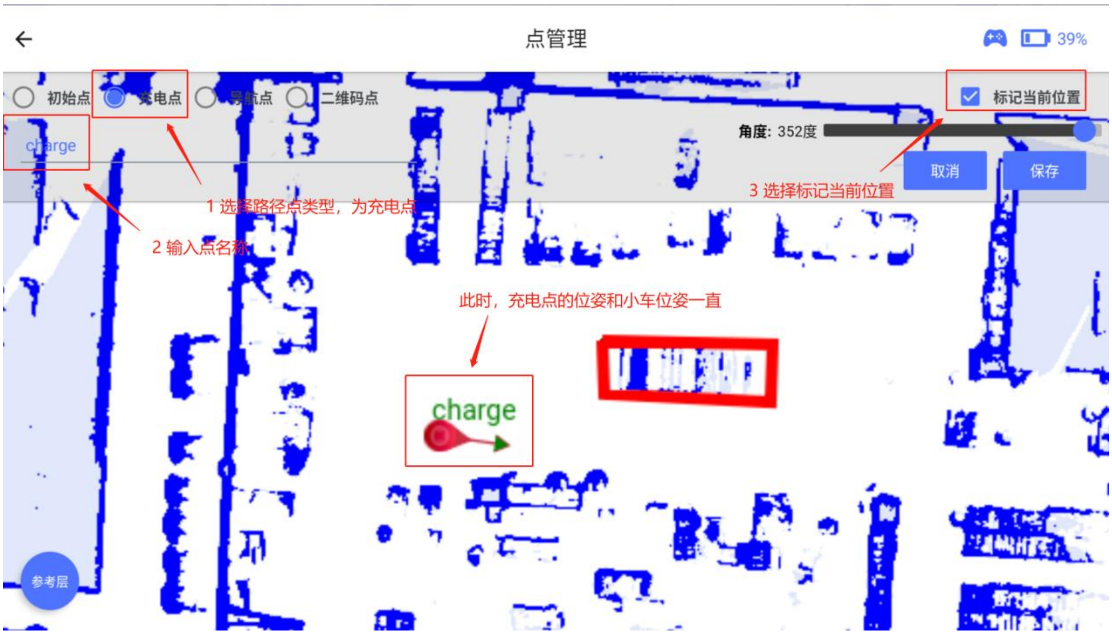


Figure 5.5.50

Creating a starting point: Creating a starting point is similar to creating a navigation point.

- **Editing Path Points:** You can modify the type, angle, name, and position of a path point by selecting a path point and clicking on "Edit", as shown in Figure 5.5.51.

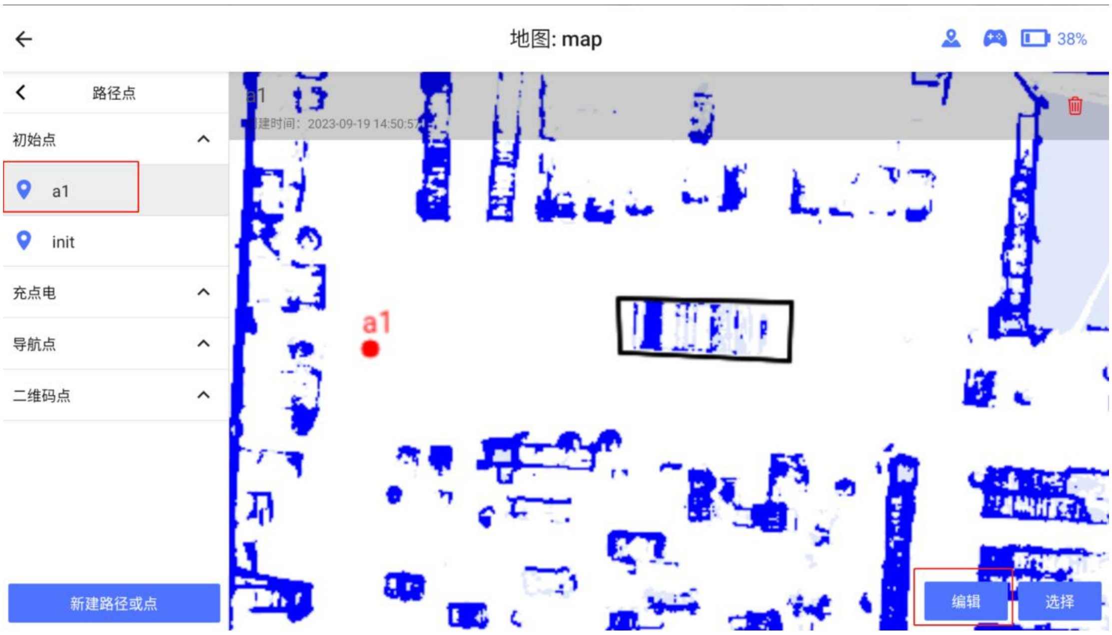


Figure 5.5.51

Upon entering the editing interface, click on "Move", as shown in Figure 5.5.52. Then, drag the map to change the position of the path point. Save the changes and exit the interface.

Figure 5.5.52

Upon entering the editing interface, click on "Edit" to modify the name, type, and angle of the path point, as depicted in Figure 5.5.53. After making the necessary changes, save and exit the interface.

Figure 5.5.53

- **Deleting a path point:** Select a path point, then click on the delete button located in the top-right corner to remove the path point.

Path Combination

- **Creating a path combination:** Click on "New Path or Point", then select "Draw New Path", as shown in Figure 5.5.54.

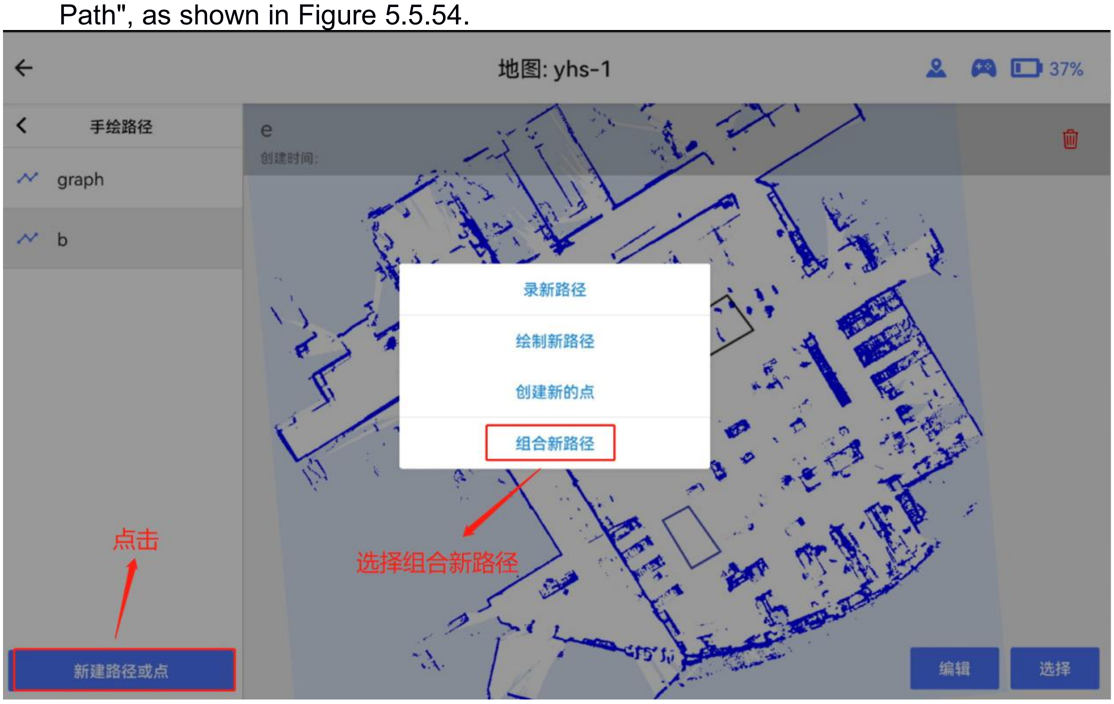


Figure 5.5.54

After entering the task planning interface, click on "Add Task", as shown in Figure 5.5.55. When executing a path combination, the order of tasks is based on the sequence in which they were added.

Figure 5.5.55

Select the paths and navigation points you want to add, keeping in mind that charging points cannot be included in the combination, as illustrated in Figure 5.5.56.

Figure 5.5.56

After adding tasks, click on the right side "Added Tasks" to view the current tasks added and their execution order. You can drag tasks to change their order, click on "Delete" to remove a task, and click on the plus sign to continue adding tasks, as shown in Figure 5.5.57.

Figure 5.5.57

After completing the additions, click on "Save", as shown in Figure 5.5.58. The execution sequence for this task combination is: init -> a1 -> hand-drawn path.

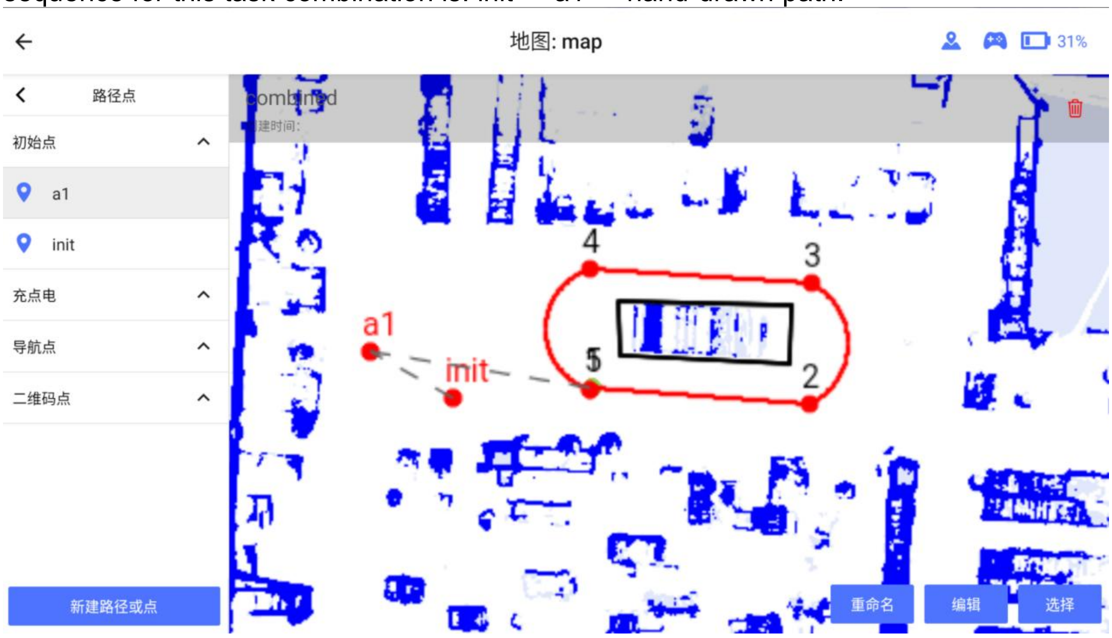


Figure 5.5.58

- **To rename a path combination:** Select a path combination, click on "Rename", enter a new name (without Chinese characters), and click "Save" to confirm.

- **Editing a path combination:** You can add or remove tasks and change the order of task execution as needed, as illustrated in Figure 5.5.59.

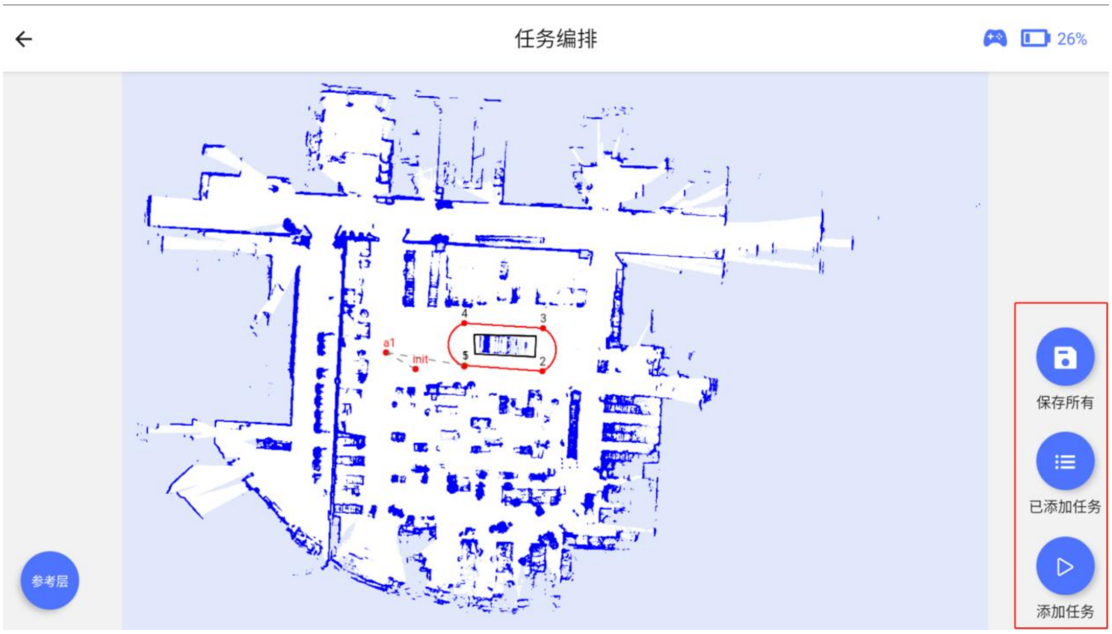


Figure 5.5.59

- **Deleting a path combination:** Select a path combination, then click on the delete button located in the top-right corner to remove the path combination.

5.5.3. Task Management

In the function options interface, click to enter the task management interface, as shown in Figure 5.5.60.

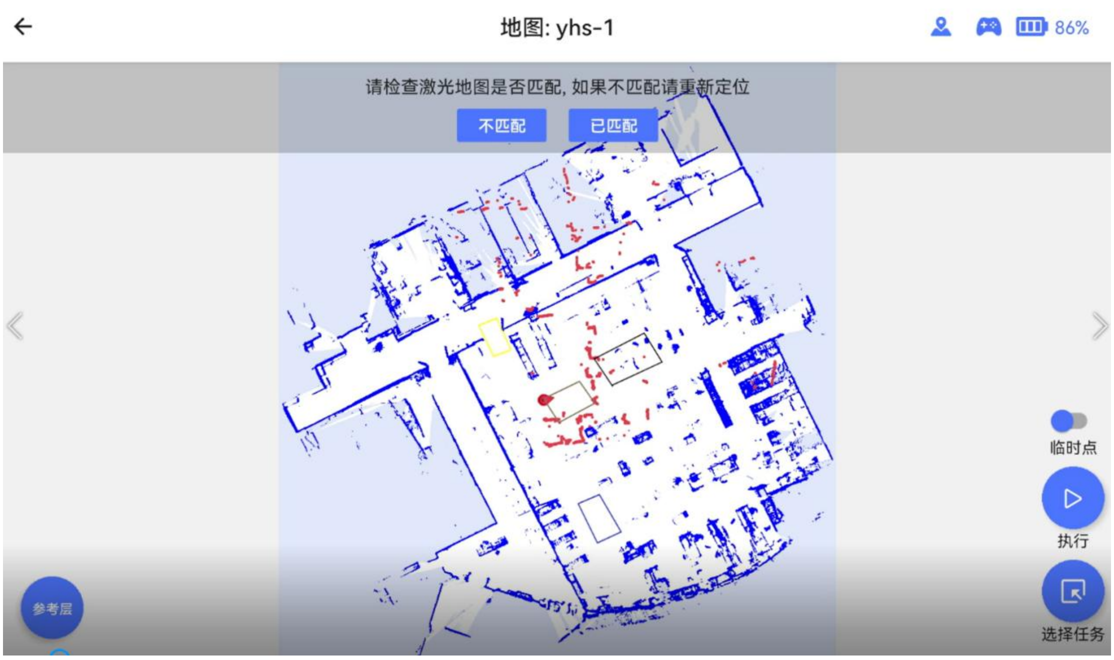


Figure 5.5.60

Initialization

If the message "Is it matching?" is displayed at the top of the interface, it indicates that the robot is in a state of localization loss and requires initialization, as shown in Figure 5.6.61.

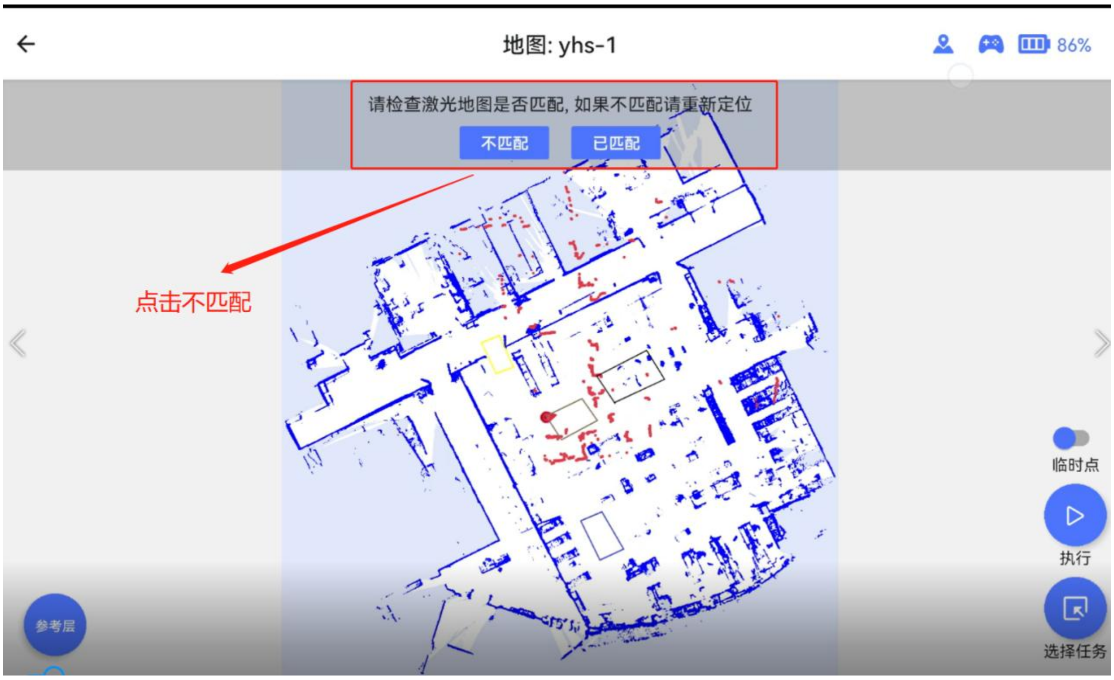


Figure 5.5.61

- **Dynamic Initialization:** Manually rotate the screen to align the red laser with obstacles on the map as closely as possible, as shown in Figure 5.5.62. Select dynamic pose, click on custom initialization, and switch the remote control to command control mode.

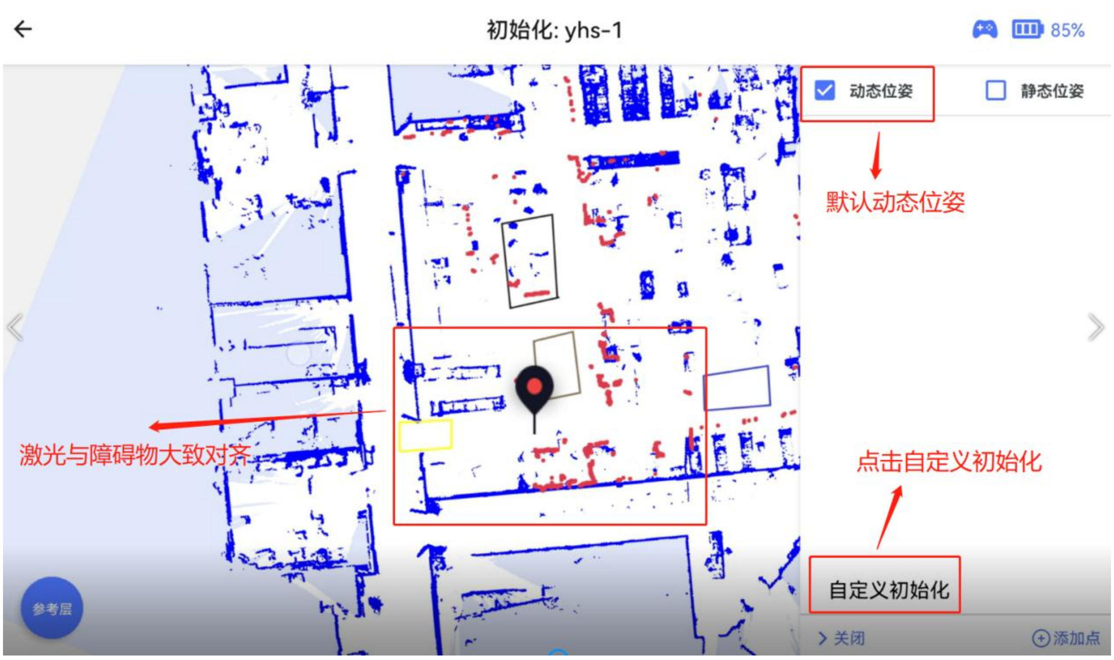


Figure 5.5.62

At this point, the robot will rotate in place(FW series). It will stop after successful localization or if localization fails due to a timeout. If localization is unsuccessful, continue adjusting the position of the laser to align it more closely with the obstacles. Retry the initialization until successful.

- **Static Initialization:** Manually rotate the screen to align the red laser with obstacles on the map as closely as possible. Then select static initialization, click on custom initialization, and wait for successful initialization, as shown in Figure 5.5.63.

Figure 5.5.63

<u>Note: If the red laser point on the map is close to the obstacles, you can manually control the robot to move around freely. The LiDAR point cloud will gradually match the surroundings. If initialization fails, you can try multiple times or move the robot to a smaller area for initialization.</u>

Task Execution

- Select the task as shown in Figure 5.5.64.

Figure 5.5.64

Select a task that has already been created, as shown in Figure 5.5.65.

Figure 5.5.65

Next, choose "Start" or "Start Loop" for the task. When you select "Start," the task will execute only once. If you click on "Start Loop," you can input the number of times to repeat the task, enabling multiple executions, as depicted in Figure 5.5.66.

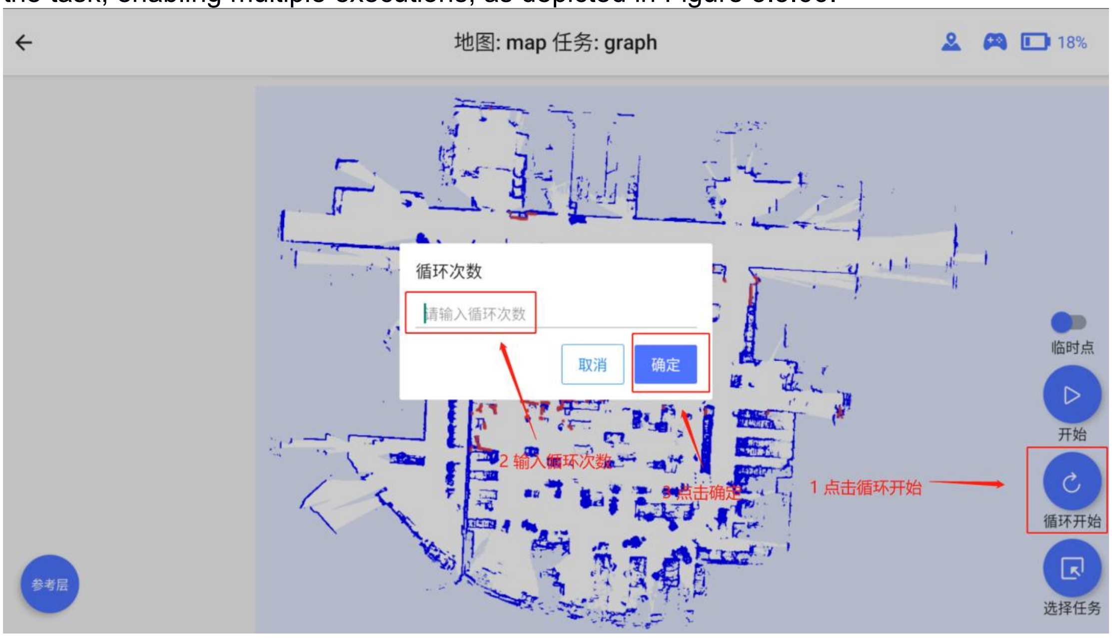


Figure 5.5.66

After clicking on "Start" or "Start Loop", switch the remote controller to navigation mode, and the robot will begin executing the task.

- **Temporary Point:** When sending a temporary navigation point, you can utilize a temporary point, as shown in Figure 5.5.67.

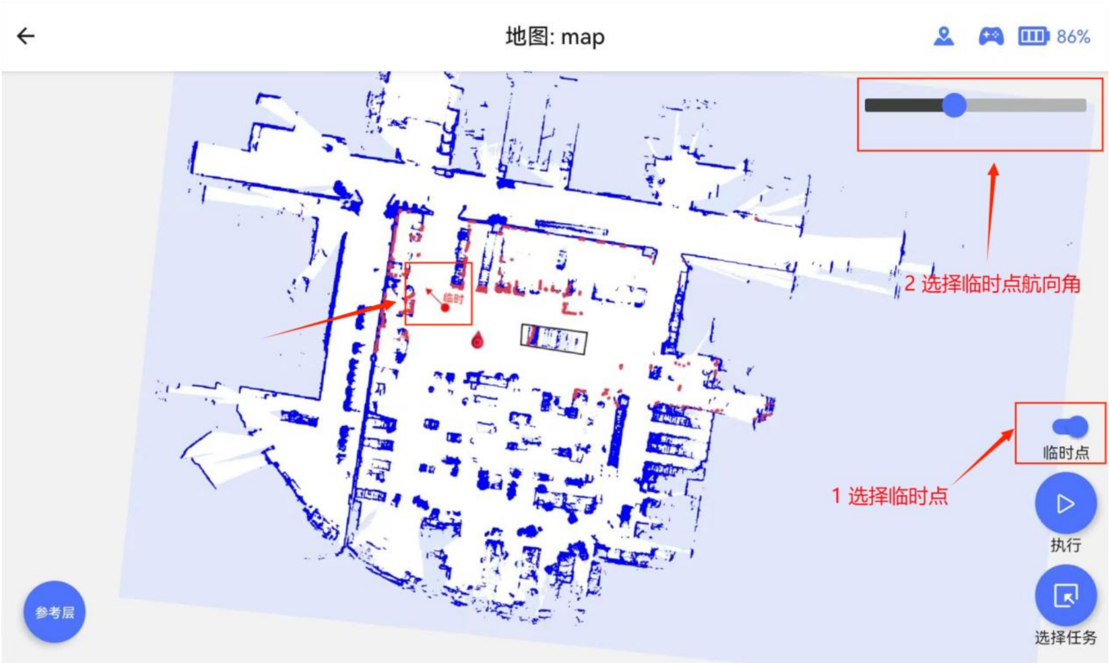


Figure 5.5.67

5.5.4. System Settings

In real-world applications, where specific requirements exist for robot speed or obstacle avoidance and stopping, these can be achieved through system settings. Click on "System Settings" to enter the system settings interface, as shown in Figure 5.5.68.

Figure 5.5.68

Speed Setting

Click on "Robot Configuration", as shown in Figure 5.5.69.

Figure 5.5.69

Click on "max_speed", as shown in Figure 5.5.70.

Figure 5.5.70

The maximum speed setting is displayed as shown in Figure 5.5.71.

Figure 5.5.71

Obstacle Avoidance and Stopping Settings

Click on "System Settings" to access the system settings interface. Then, click on "avoid_obstacle" as shown in Figure 5.5.72.

Figure 5.5.72

According to your requirements, you can enable or disable obstacle avoidance mode for hand-drawn paths or recorded paths. Refer to Figure 5.5.73. Please note that in free navigation mode, obstacle avoidance is enabled by default.

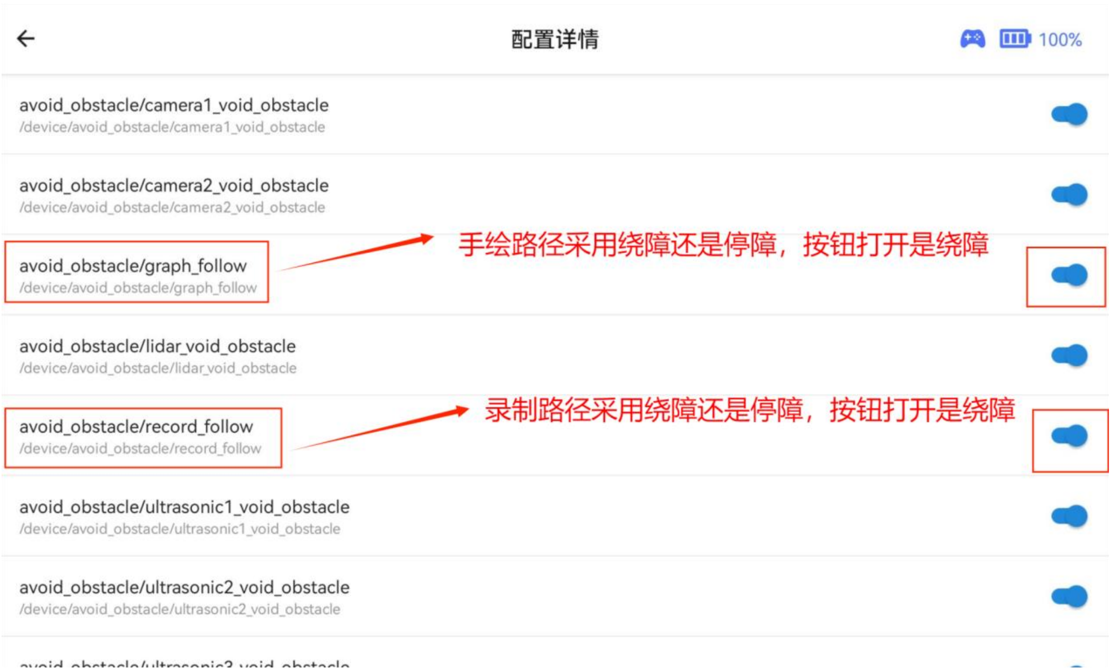


Figure 5.5.73

Please note that after modifying these parameters, you will need to restart the navigation system or reboot the robot for the changes to take effect.

6. API Usage Instructions

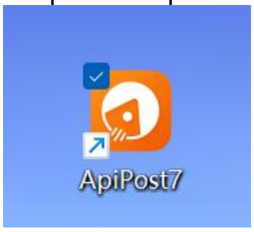
6.1. Importing ApiPost Projects

The NV magic Navigation API interface is based on the HTTP protocol and can be debugged using software such as ApiPost. Visit the ApiPost official website to download the latest software: https://www.apipost.cn/.

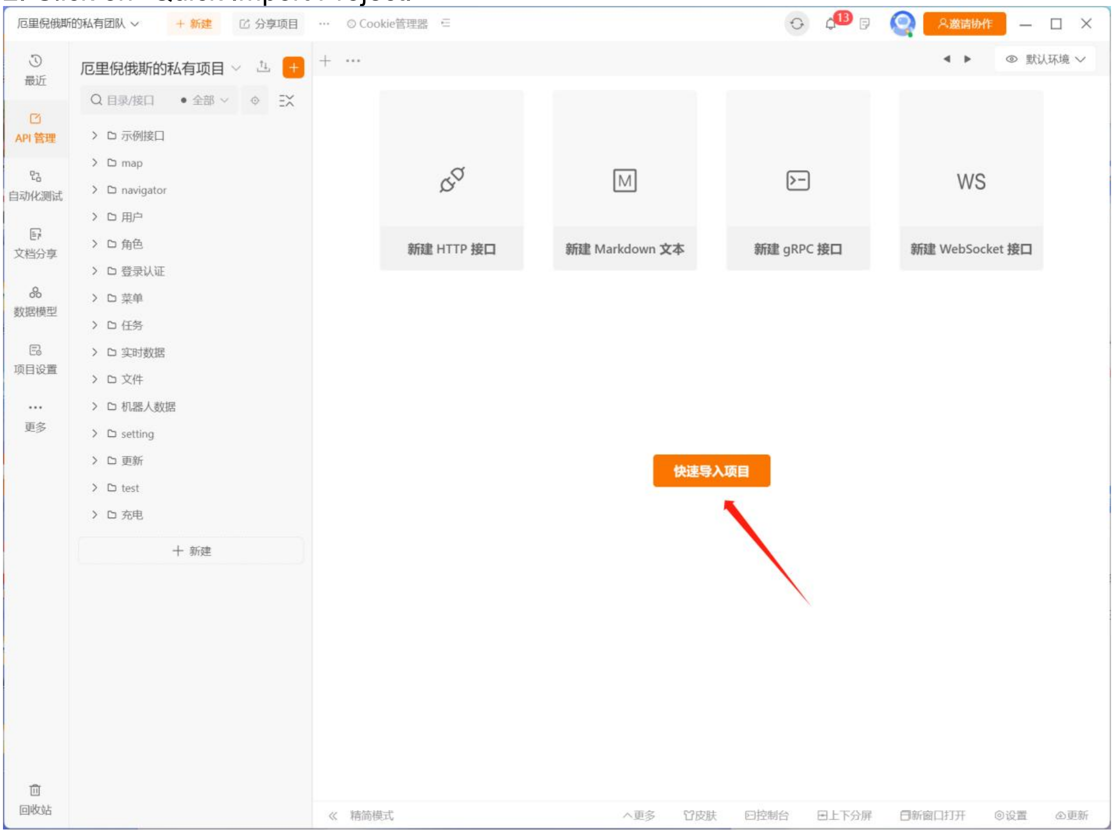
For the NV magic Navigation interface documentation, please refer to: https://doc.apipost.net/docs/detail/37e3705b580b000?target_id=53f169d.

Copy the link and open it in a web browser to view the details.

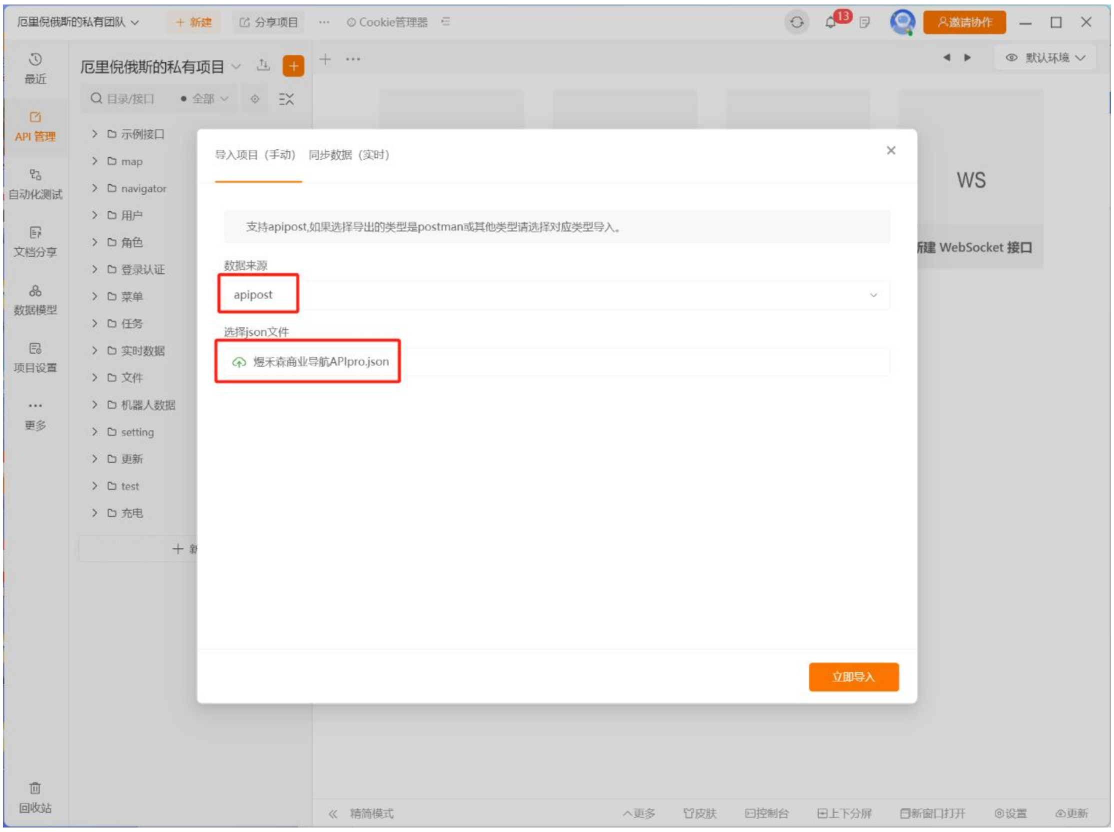
1. open the ApiPost software.



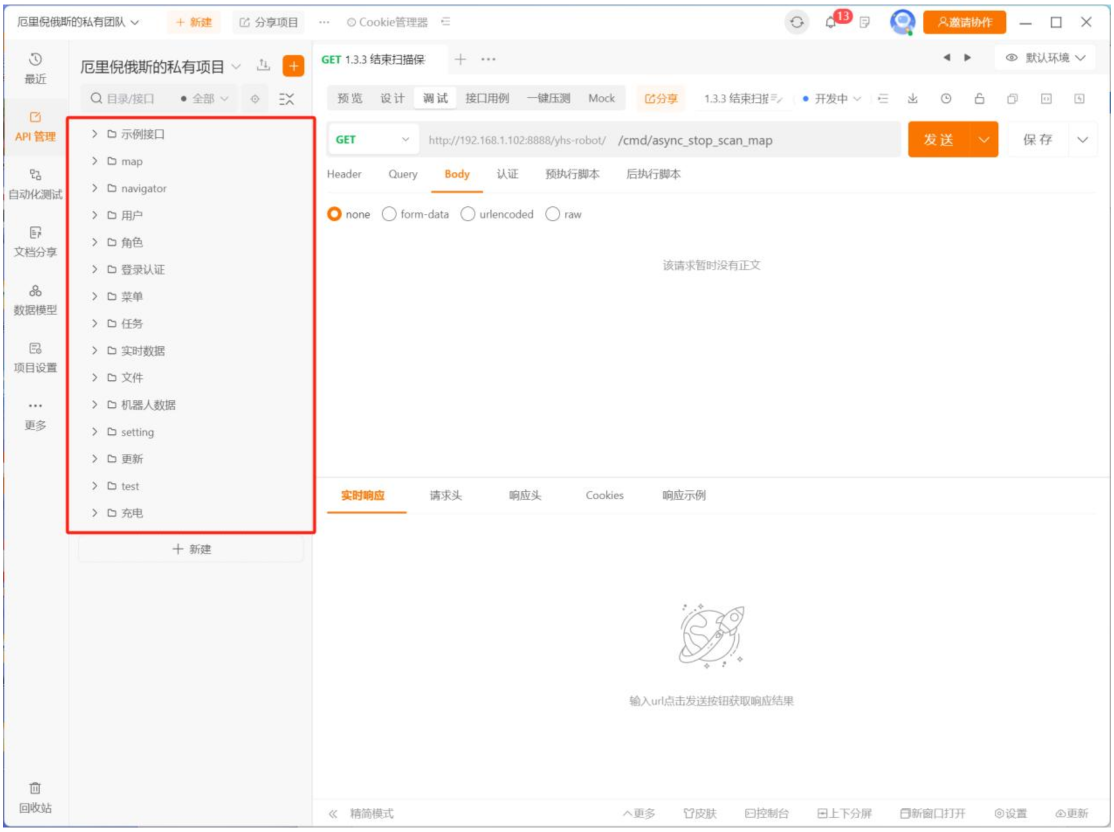
2. Click on "Quick Import Project.



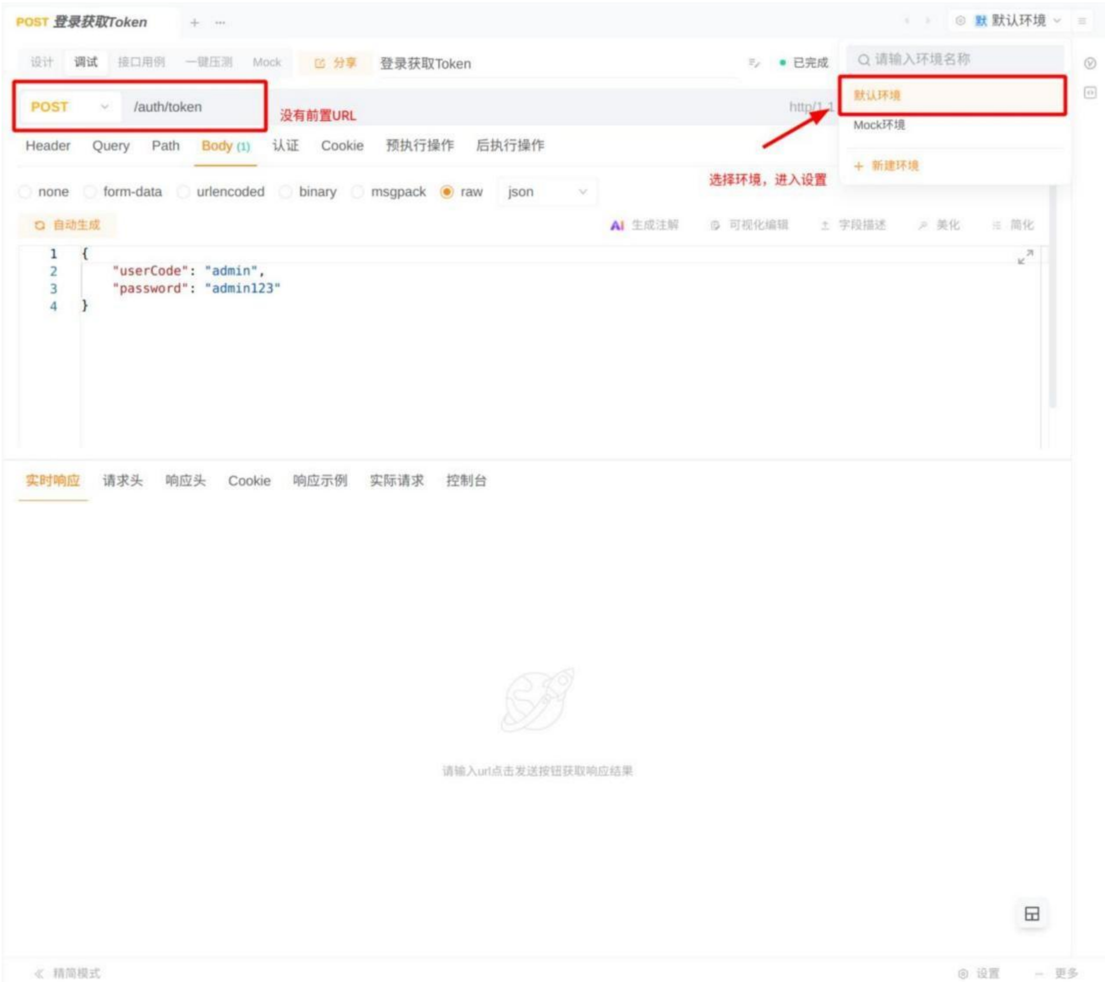
3. Select "Data Source" as "ApiPost", choose the JSON file "煜禾森商业导航 APIpro.json" (select the actual JSON file as needed), and then click on "Import Immediately".



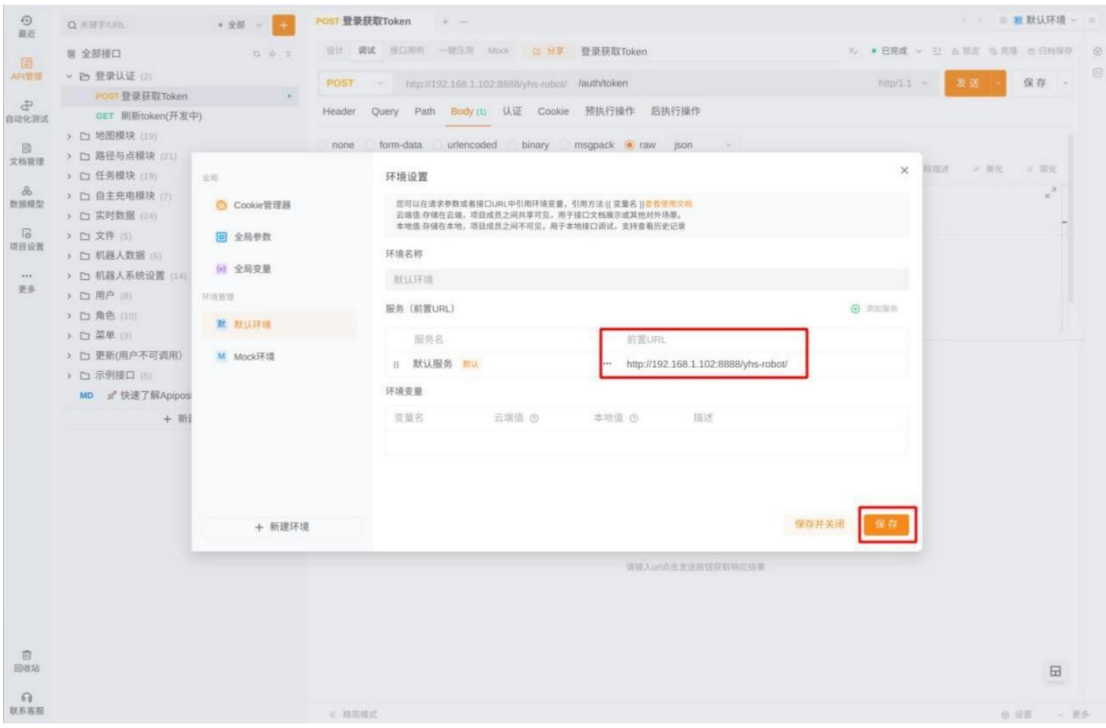
4. The project directory will display the following folders. At this point, the import process is successful.



5.If the interface URL does not have a prefix URL, you can add it manually.

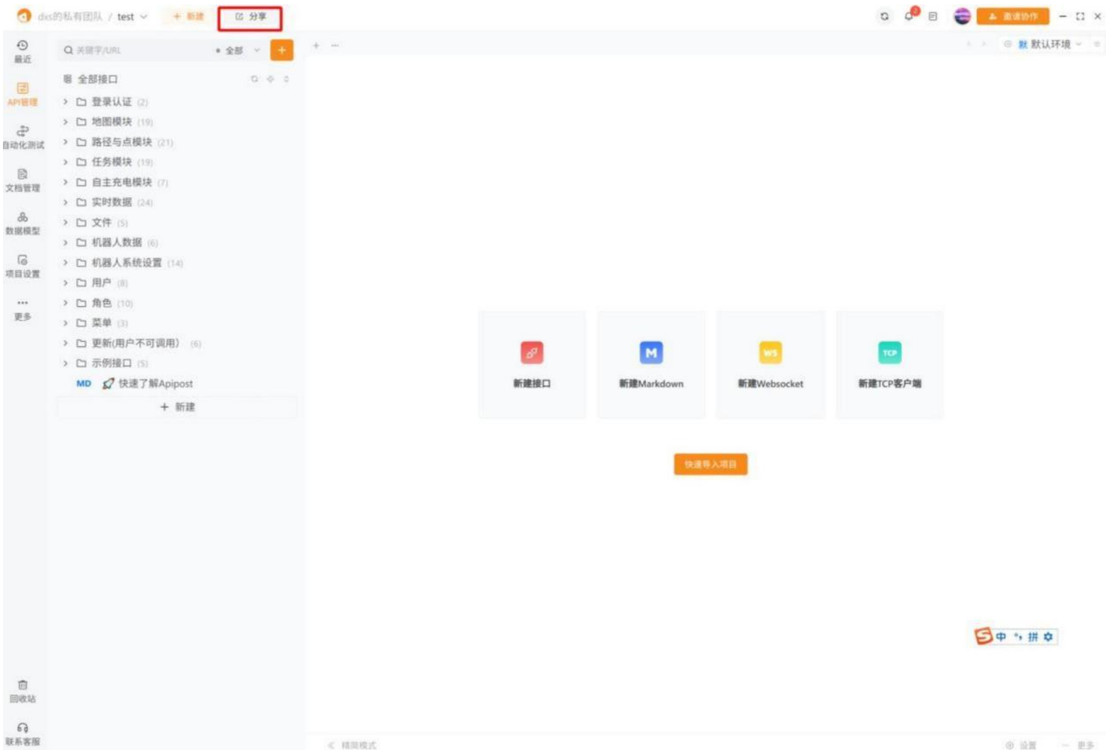

POST 登录获取Token
默认环境
设计
调试
接口用例
一键压测
Mock
分享
登录获取Token
已完成
请输入环境名称
POST
/auth/token
没有前置URL
默认环境
Mock环境
新建环境
Header
Query
Path
Body (1)
认证
Cookie
预执行操作
后执行操作
选择环境，进入设置
none
form-data
urlencoded
binary
msgpack
raw
json
自动生成
生成注解
可视化编辑
字段描述
美化
简化
{
"userCode": "admin",
"password": "admin123"
}
实时响应
请求头
响应头
Cookie
响应示例
实际请求
控制台
请输入url点击发送按钮获取响应结果
精简模式
设置
更多

Fill in the prefix URL: http://192.168.1.102:8888/yhs-robot/ and save it.

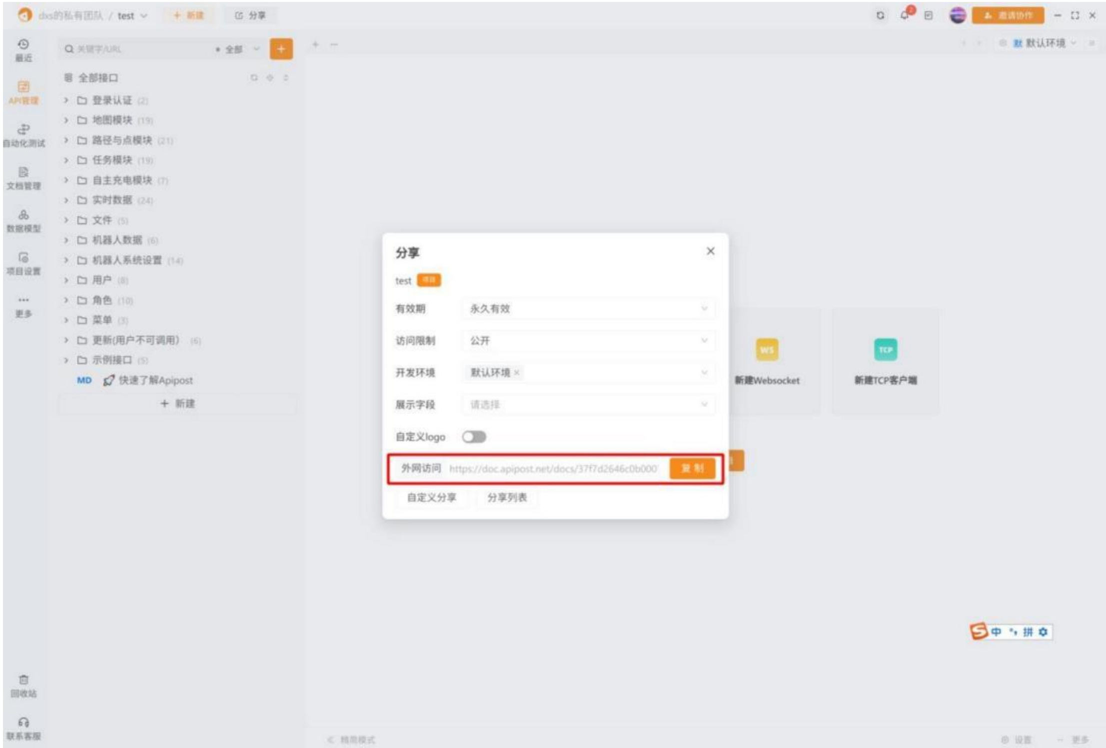


6. Generate Documentation

Open the ApiPost software, import the project, and click "Share."



Click the external access link to copy it.



Then, open the link in your browser to view the documentation.

6.2. Data Description

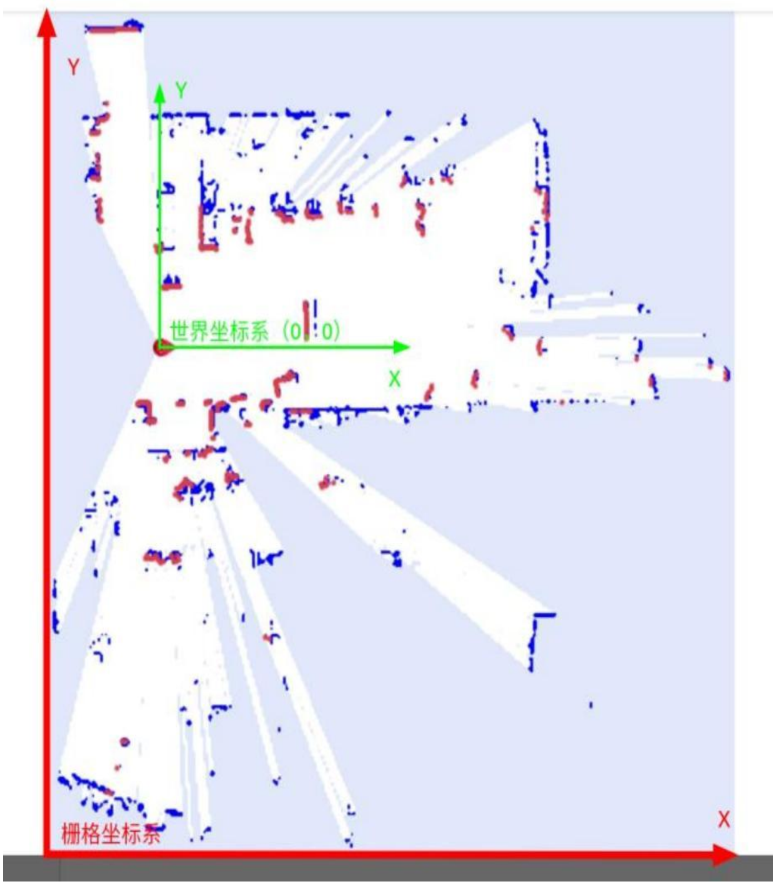
1. Coordinate System

1.1 Coordinate Origin

Grid Coordinate System (grid):
The origin is located at the bottom-left corner of the PNG image.
The positive direction of the x-axis is to the right, following the right-hand coordinate system.

World Coordinate System (world):
The origin is the starting position of the cleaning map.
The robot's orientation at the start of mapping defines the positive direction of the x-axis, also following the right-hand coordinate system.

Note:
The two coordinate systems generally do not overlap.
The grid coordinate system uses only natural numbers, while the world coordinate system can include negative numbers and decimals.



1.2 Coordinate Conversion

The world coordinate system uses meters as units, while the grid coordinate system uses grid units.

The conversion between the two systems is based on a scale: **1 grid = 0.05 meters**, meaning **1 meter equals 20 grids.**

Conversion Formulas:

world_x = grid_x * resolution + originX

world_y = grid_y * resolution + originY

Map Information:

The current map's information can be obtained through the "Get Current Default Map" interface.

From the map information:

The origin coordinates of the map are **(-5.45, -18.51)**, representing the world coordinates of the bottom-left corner of the map.

The map resolution is **0.05**.

Map dimensions:

Width: **744 * 0.05 = 37.2 meters**

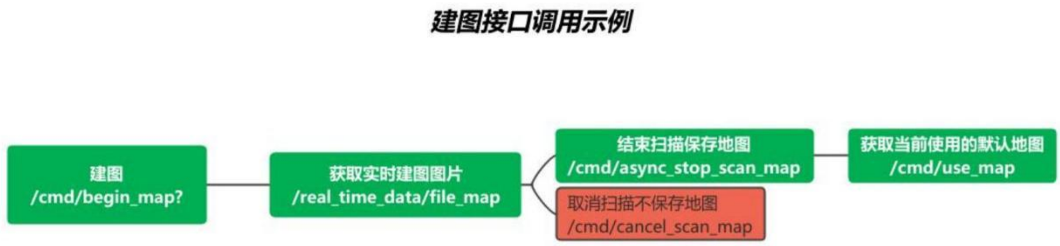
Height: **561 * 0.05 = 28.05 meters**


```json
{
    "data": {
        "createdAt": "2024/11/22 14:15:50",
        "creator": "董小菘",
        "default": true,
        "id": "3739c532-a899-11ef-b5dc-c4837210cf9c",
        "mapInfo": {
            "createdAt": "2024/11/22 14:15:50",
            "gridHeight": 744,
            "gridWidth": 561,
            "originX": -5.45,
            "originY": -18.512559509277345,
            "resolution": 0.05000000074505806
        },
        "name": "map"
    },
    "errCode": 0,
    "msg": "",
    "successed": true
}
```

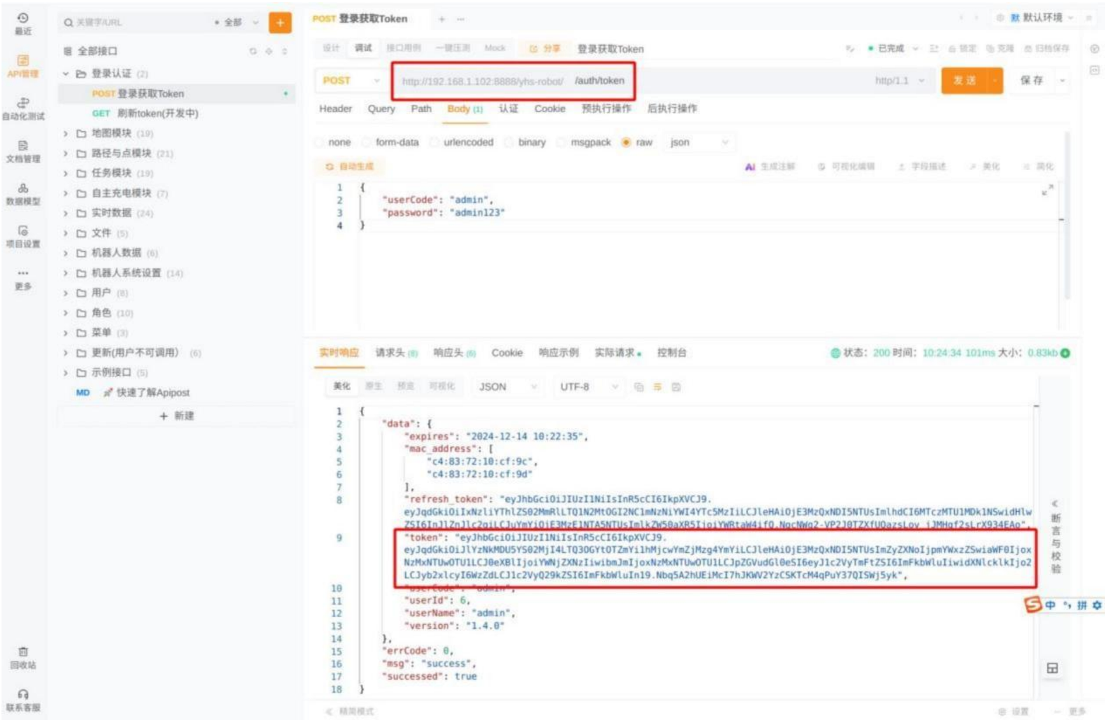
6.3. API Examples

Import the "煜禾森商业导航接口.json" file in the ApiPost software, then connect the computer to the robot's WIFI.

1.Mapping Interface Call Example

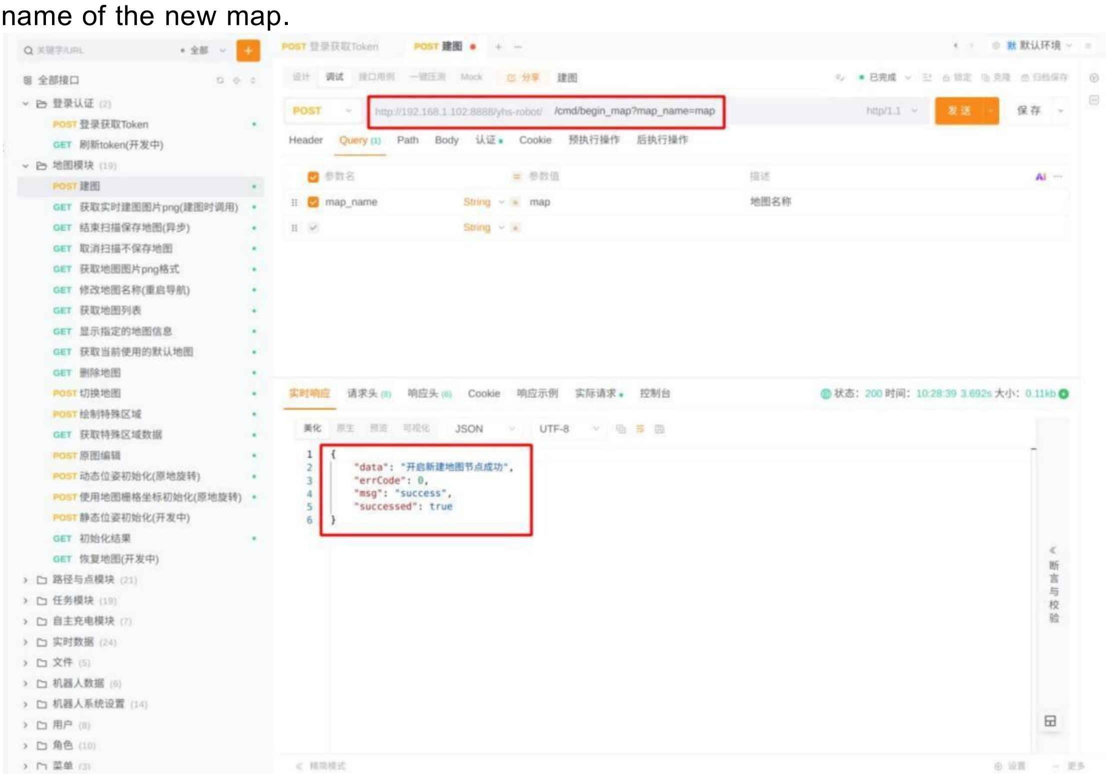


① Call /auth/token to get the token



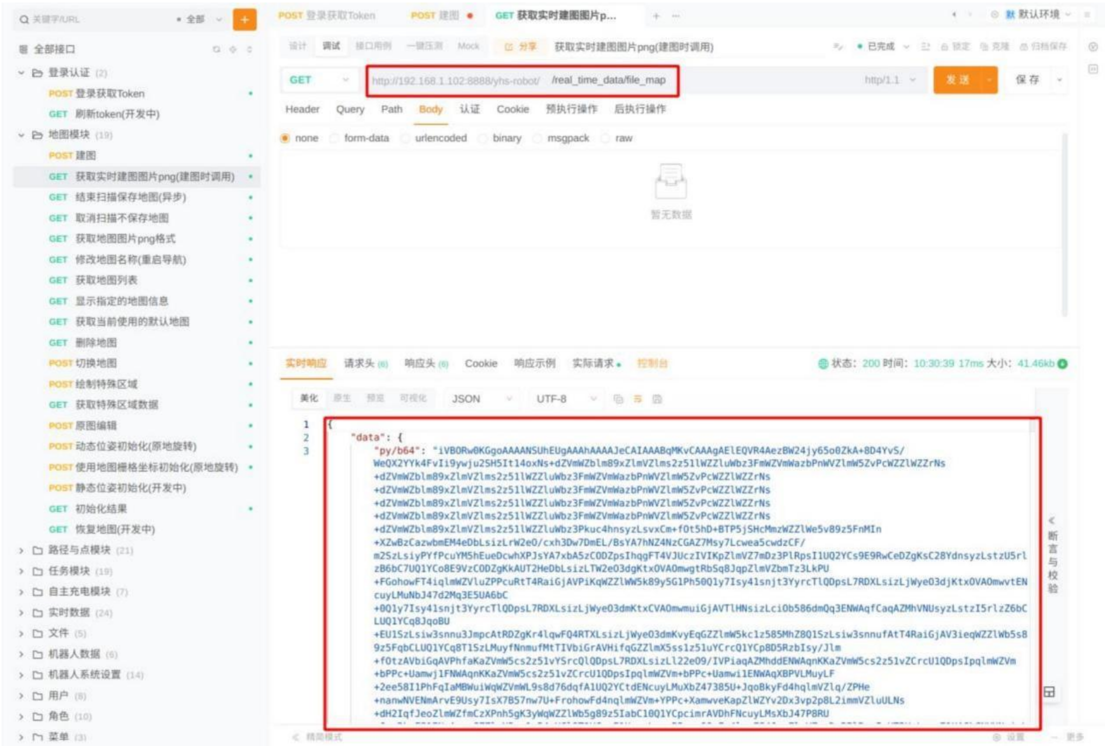
Some interfaces require Bearer Token authentication to be filled in, otherwise, when called, it will return msg: "Unauthenticated". Note that the value of the token is the content within double quotation marks. If copied incorrectly, it will return msg: "Invalid token".

② Call /cmd/begin_map? to start mapping, fill in the parameter map_name with the name of the new map.

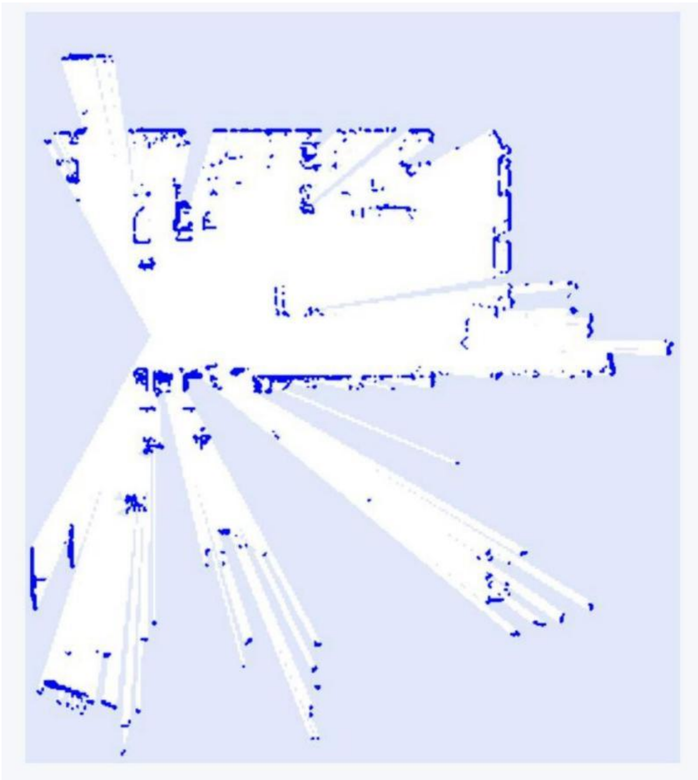


After the successful activation of the mapping node in the response, you can remotely control the robot to move and scan the site to create a map. During the mapping process, the speed should not be too fast.

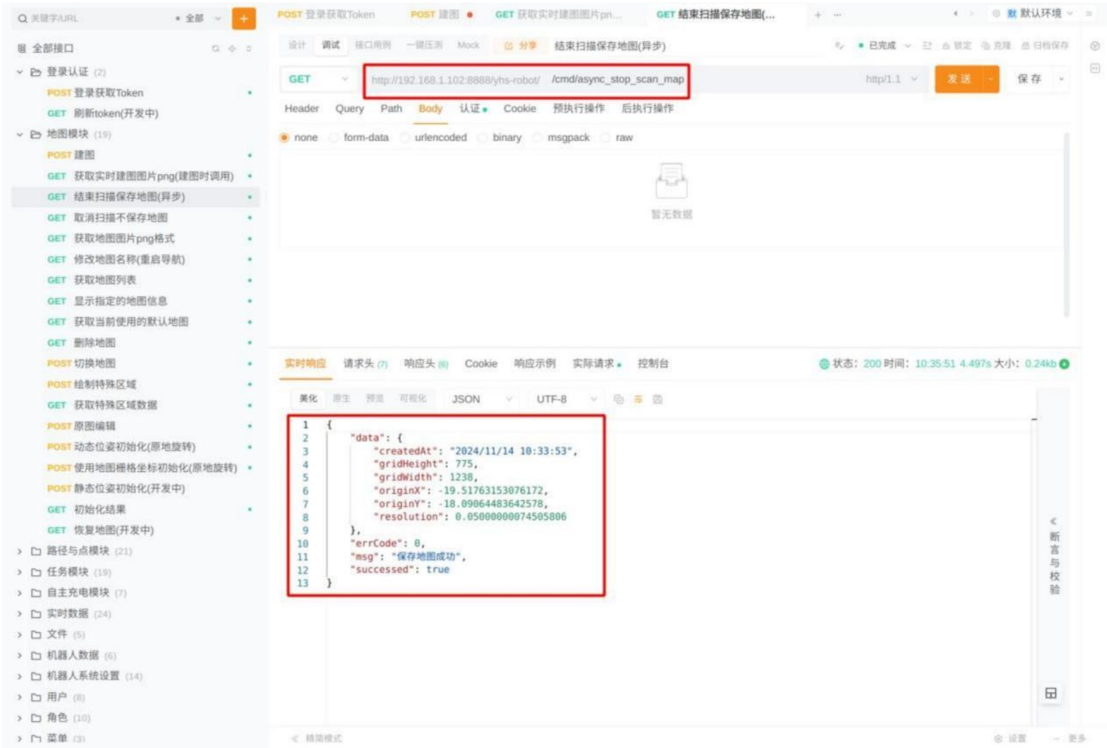
③ You can call the /real_time_data/file_map interface during the mapping process to obtain real-time images of the mapping.



The returned data is in base64 format and needs to be converted by the user into an image format.

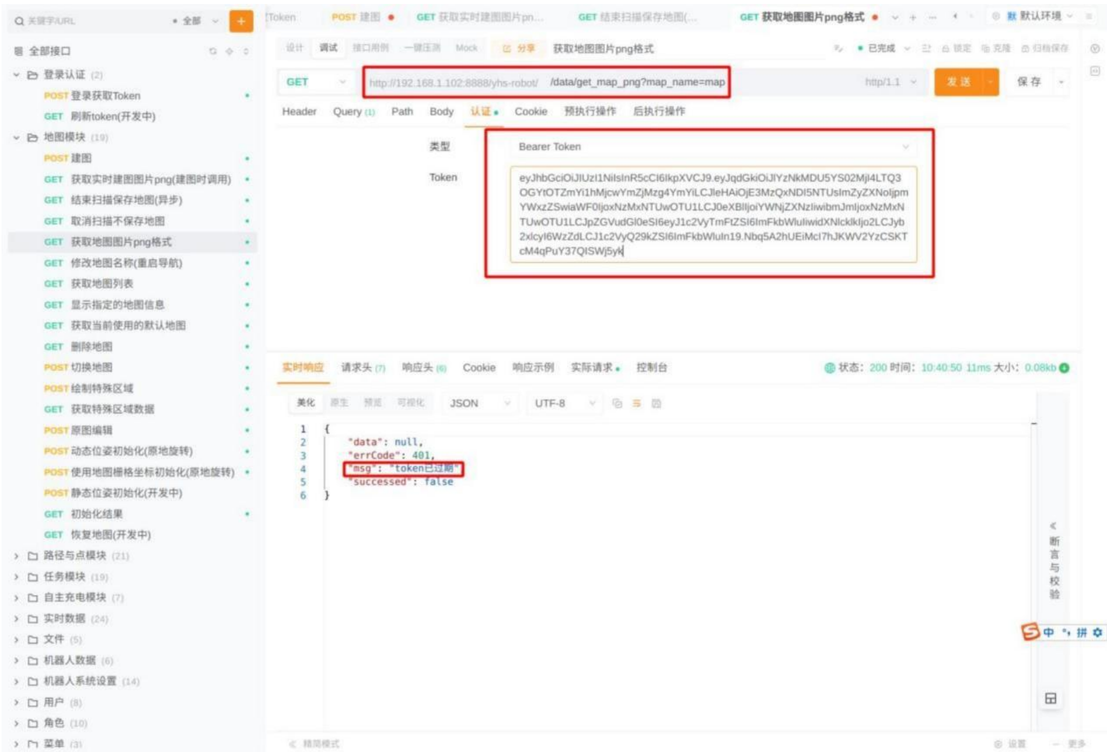


④After the robot has finished scanning the site, call the /cmd/async_stop_scan_map interface to save the map.



The duration of saving the map depends on the size of the map. After successful saving, the information of the map will be returned.

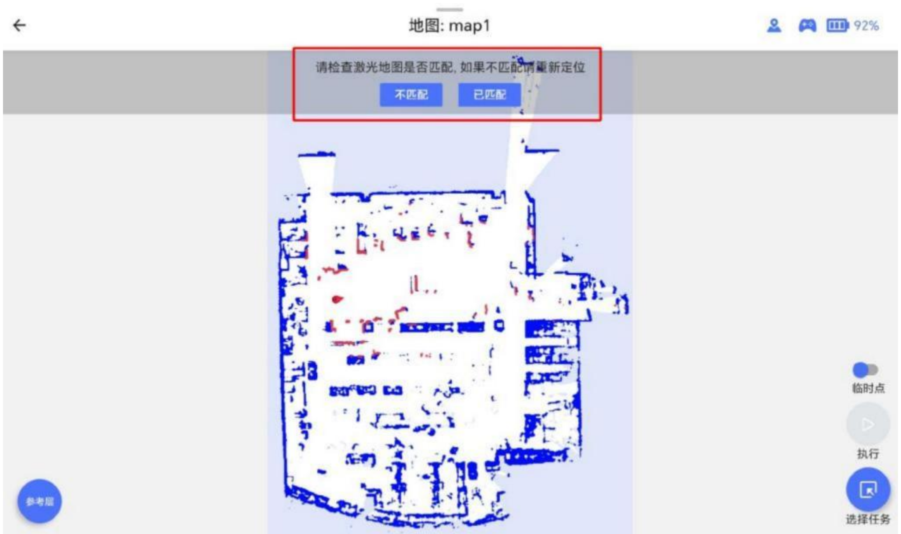
⑤ Call the /data/get_map_png? interface to get the image of the map by filling in the map name as a parameter. This interface requires token authentication; otherwise, the response will output "Token expired". As shown in the image below.



After the request, convert the received base64 format data into an image format to obtain a complete map image display.

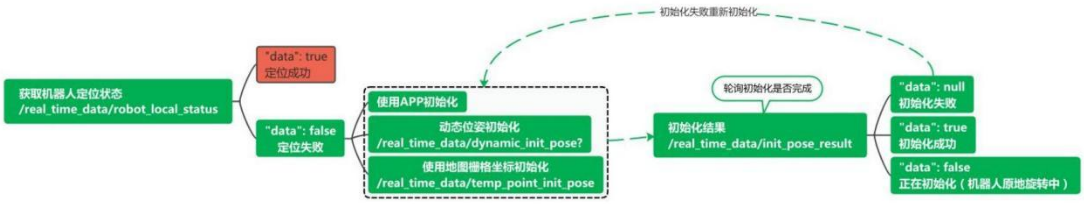
2.Initialization Interface Call Example

The purpose of initialization is to confirm the robot's position on the map and the direction it is facing. Only when the positioning is accurate can the robot accurately perform navigation work. If the positioning feedback is unsuccessful, initialization is required before navigation can proceed. When the APP displays a prompt like the one shown below, it indicates that the robot's positioning was unsuccessful and requires initialization:

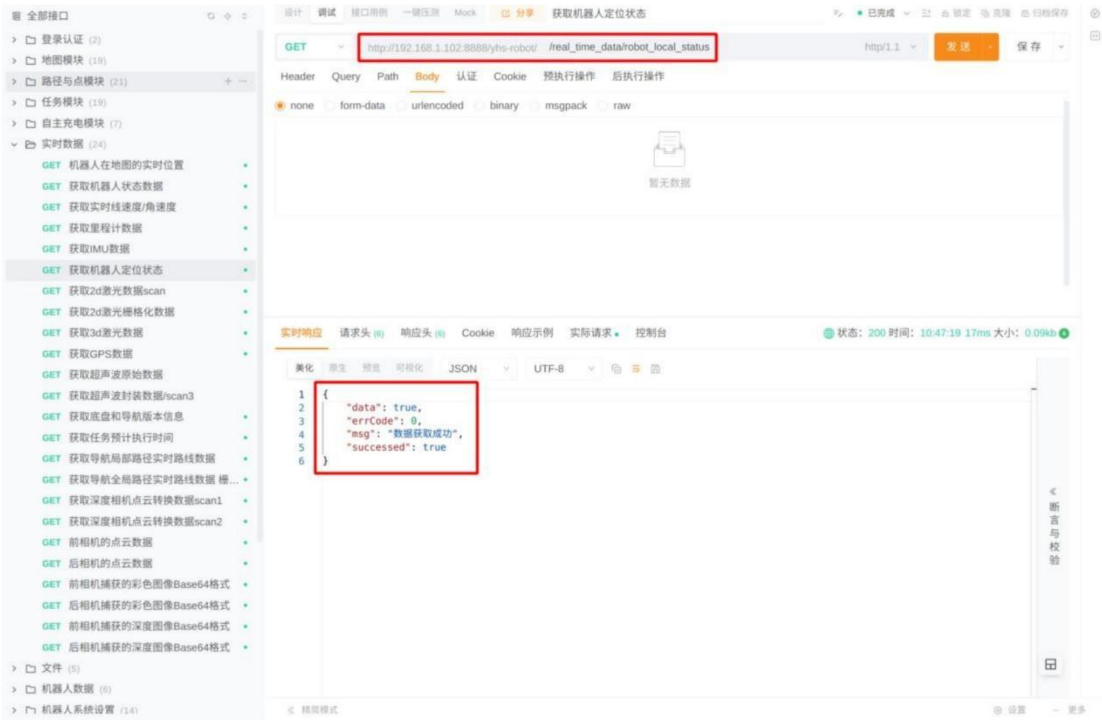


Alternatively, call the interface "Get Robot Positioning Status" to determine if the robot needs initialization. If the data returns false, it indicates that the positioning was unsuccessful.

初始化接口调用示例

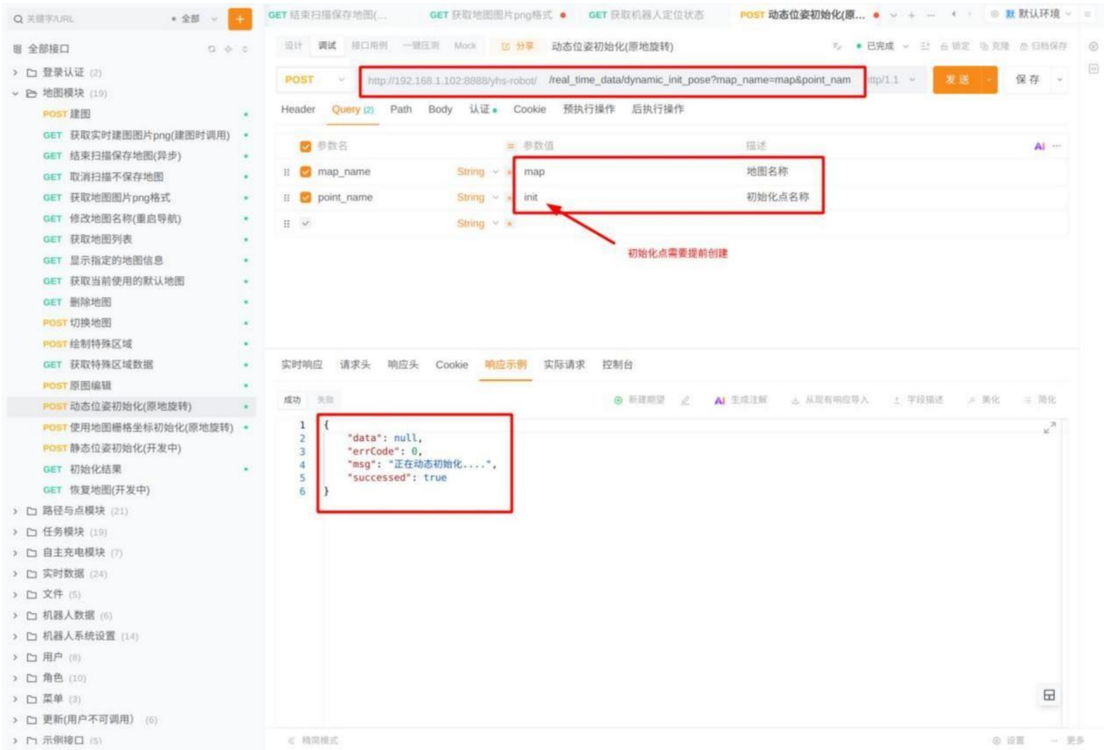


①Call /real_time_data/robot_local_status to get the robot's positioning status.



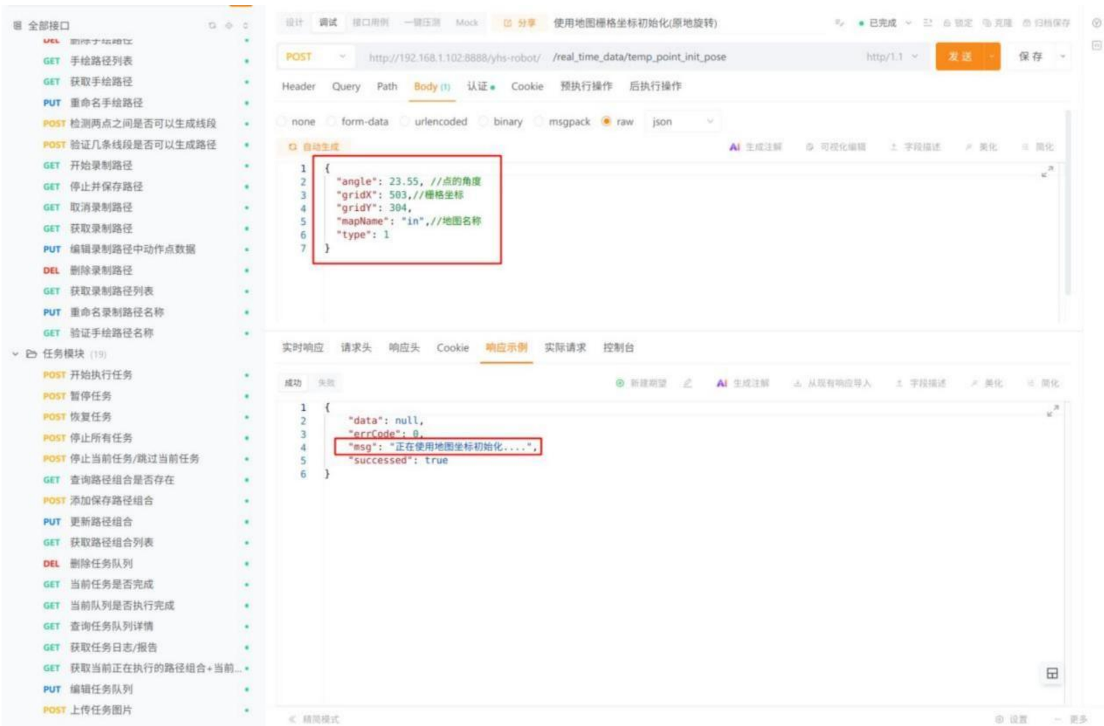
If the response data returns true, it means the robot's positioning was successful. If it returns false, it means the robot's positioning failed, and initialization is needed. After the first mapping, it is recommended to initialize using the APP first, then set the initialization point. This way, when starting navigation subsequently, you can call the interface for initialization.

②First method for interface initialization: Call /real_time_data/dynamic_init_pose? for dynamic pose initialization (the initialization point needs to be set in advance).



You can create an initialization point using the /cmd/pos interface for dynamic initialization. **Note that when initializing using this interface, you need to maneuver the robot roughly to the location of the initialization point and ensure that the robot's orientation matches that of the initialization point.**

③ Second method for interface initialization: Call /real_time_data/temp_point_init_pose for dynamic pose initialization, where grid coordinate values need to be filled in.

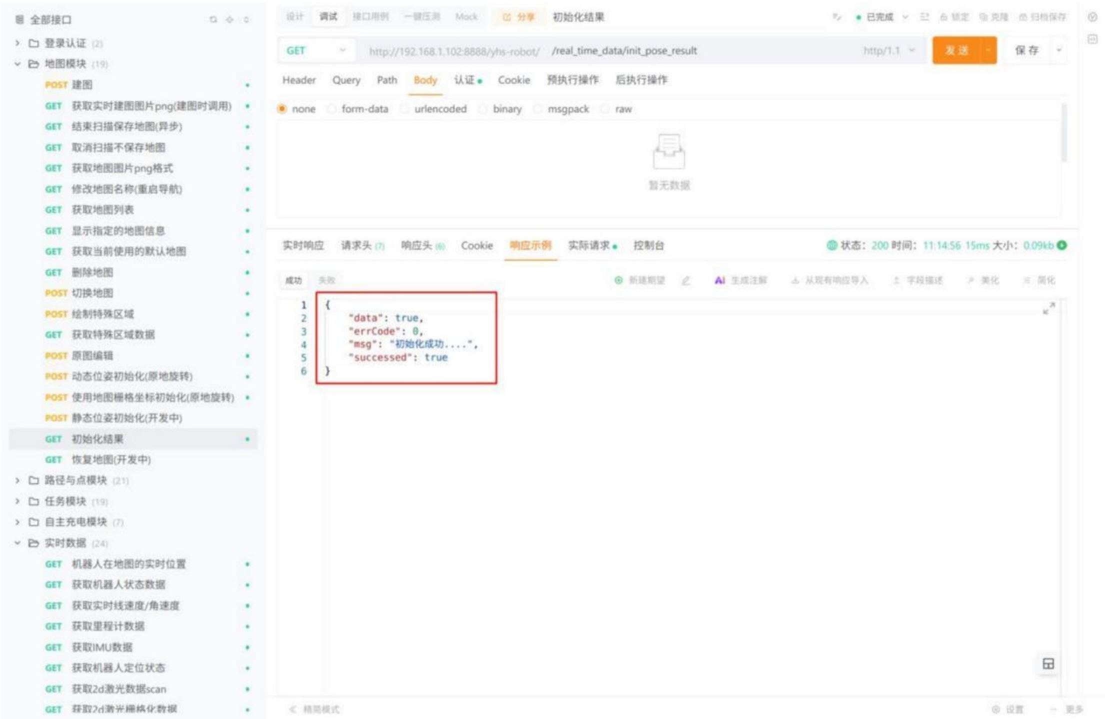


Similarly, **when calling this interface for initialization, the robot needs to be moved to the corresponding grid coordinate point with the robot's orientation aligned with the grid point angle.**

After calling the initialization interface, switch the remote control to the

command control mode, and the robot will rotate in place.

Call /real_time_data/init_pose_result to poll the initialization result.



{"data":false,"errCode":0,"msg":"Initializing...","successed":true}

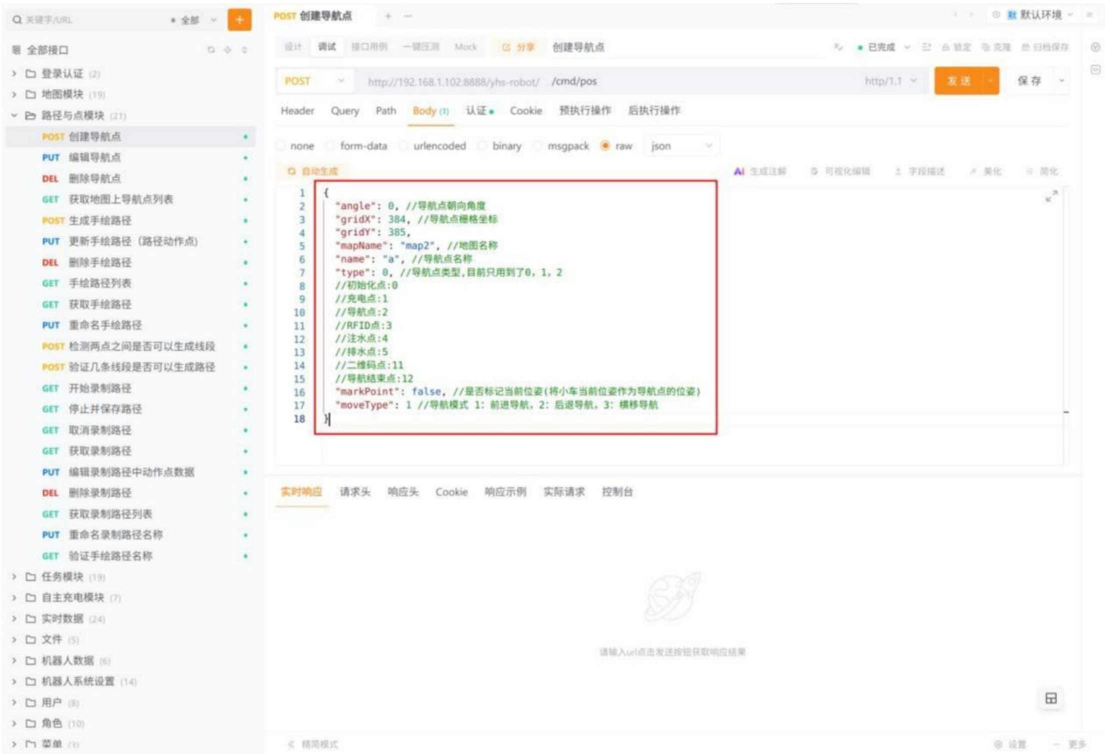
{"data":true,"errCode":0,"msg":"Initialization successful...","successed":true}

{"data":null,"errCode":-1,"msg":"Initialization failed...","successed":false}

Poll the initialization result, if initialization fails, repeat the above initialization operations.

3.Path and Point Creation Interface Call Example

①Create Navigation Points

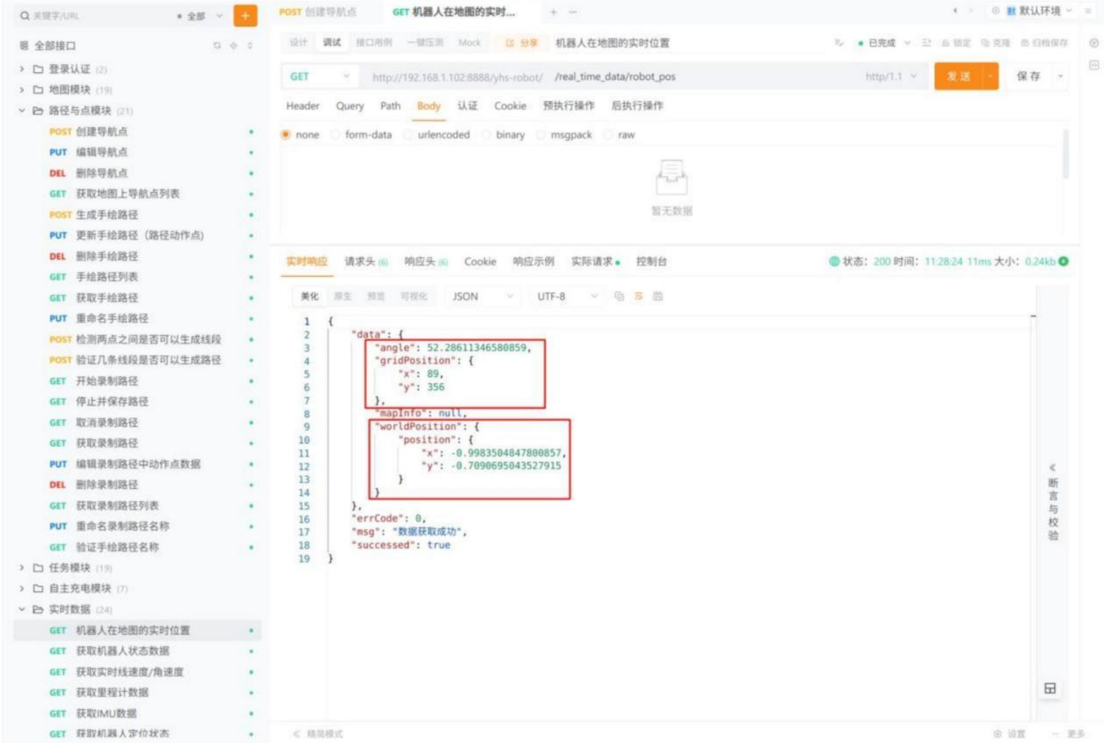


Initialization Point: Used for robot initialization or navigation.

Charging Point: Used for the robot's autonomous return to charging.

Navigation Point: Used as the target point for robot navigation

How to confirm the coordinates and angle of a point you want to navigate to? The simplest method is to move the robot to the desired navigation point, then obtain the robot's current pose through the APP main interface or interface, as shown in the image. Alternatively, if you know the world coordinate system, based on the components of the navigation point on the x-axis and y-axis of the world coordinate system, as well as the angle, you can determine the pose of that point.



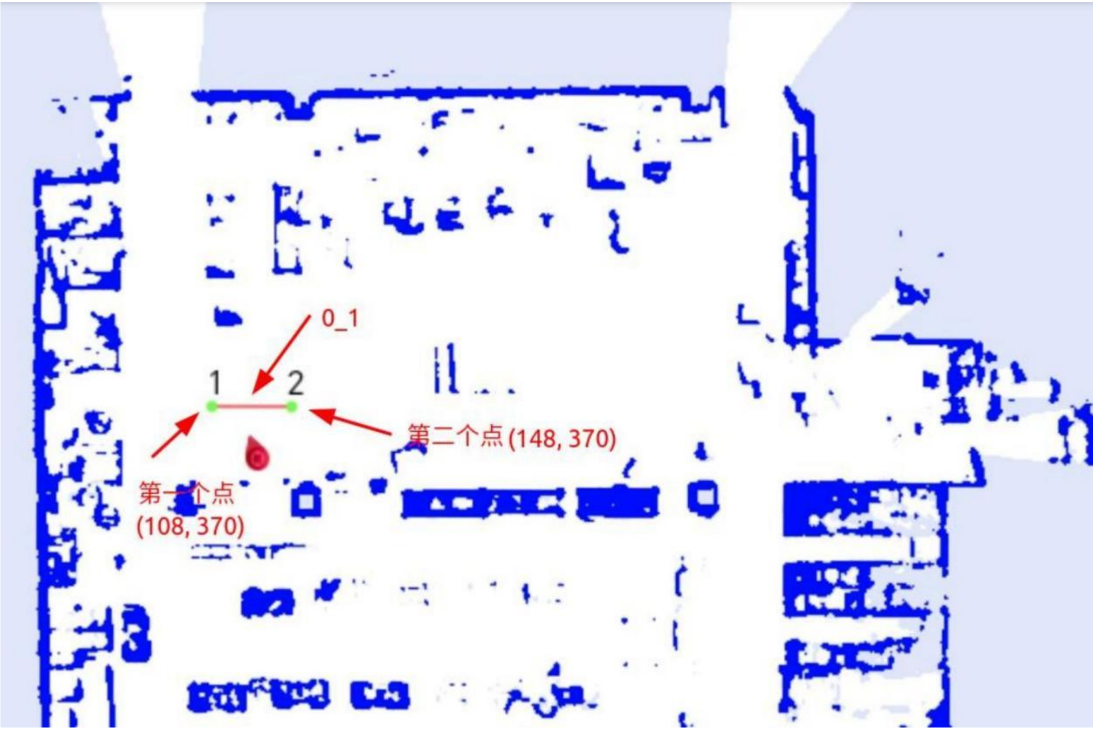
② Generate Hand-Drawn Paths, Example of the body parameter:

```
{
  "name": "graph_path", //手绘路径名称
  "mapId": "3739c532-a899-11ef-bd5dc-c4837210cf9c", //地图 ID
  "mapName": "map", //地图名称
  "points": [
    {
      "name": "1", //手绘路径第一个点的名称
      "actions": [], //动作
      "gridPosition": { //栅格坐标
        "x": 108,
        "y":
      370 },
      "angle": 0, //点的朝向 -180°-180°,这里没有用到，如果要使用角度，在 actions 里添加动作点
      "moveType": 0 //导航模式，0前进导航，1后退导航，2横移导航
    },
    {
      "name": "2", //手绘路径第二个点的名称
      "actions": [],
      "gridPosition":
        { "x": 148,
        "y":
      370 },
      "angle": 0,
```

```
    "moveType": 0
  }
],
"lines": [
  {
    "radius": 0, //曲率为0，则线段为直线，曲率为正，朝前进方向左边弯曲
    "begin": "1",
    "end": "2",
    "name": "1_2" //手绘路径中线段的名称
  }
],
"pathGroups": [],
"paths": [
  {
    "lines": [
      {
        "name": "1_2", //手绘路径线段的名称
        "reverse": false
      }
    ],
    "points": [
      {
        "name": "1",
        "actions": [],
        "gridPosition": {
          "x": 108,
          "y":
        370 },
        "angle": 0,
        "moveType": 0
      },
      {
        "name": "2",
        "actions": [],
        "gridPosition": {
          "x": 148,
          "y":
        370 },
        "angle": 0,
        "moveType": 0
      }
    ],
    "name": "path1_path0"
  }
```

```
  ],
  "type": 3,
  "close": false //是否为闭环路径(至少三个点才能形成闭环路径) }
```

Based on the above example, you can generate a straight hand-drawn path as shown in the following image.



If we want the robot to rotate to a direction of 90° at point "1" and stay for 5 seconds, you can fill in the body as follows:

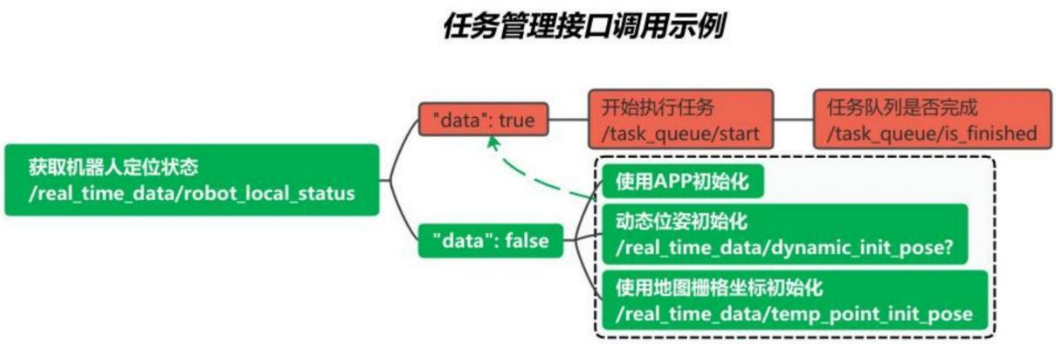
```
{
  "name": "graph_path2", //手绘路径名称
  "mapId": "3739c532-a899-11ef-b5dc-c4837210cf9c", //地图ID
  "mapName": "map", //地图名称
  "points": [
    {
      "name": "1", //手绘路径第一个点的名称
      "actions": [
        {
          "fields": [
            {
              "fields": [],
              "name": "millisecond",
              "type": "3",
              "value": "5000" //停留5000ms
            }
          ],
          "isChecked": false,
          "isDevice": false,
```

```
        "name": "Pause"
      },
      {
        "fields": [
          {
            "fields": [],
            "name": "rotation",
            "type": "double",
            "value": "90" //旋转到90° 范围:0°-360°
          }
        ],
        "isChecked": false,
        "isDevice": false,
        "name": "Rotate2d" //旋转
      }
    ], //动作
    "gridPosition": { //栅格坐标
      "x": 108,
      "y":
    370 },
    "angle": 0, //点的朝向 -180°-180°,这里没有用到，如果要使用角度，在actions里添加动作点
    "moveType": 0
  },
  {
    "name": "2", //手绘路径第二个点的名称
    "actions": [],
    "gridPosition":
      { "x": 148,
      "y":
    370 },
    "angle": 0,
    "moveType": 0
  }
],
"lines": [
  {
    "radius": 0, //曲率为正，朝前进方向左边弯曲
    "begin": "1",
    "end": "2",
    "name": "1_2" //手绘路径中线段的名称
  }
],
"pathGroups": [],
"paths": [
```

```json
{
  "lines": [
    {
      "name": "1_2", //手绘路径线段的名称
      "reverse": false
    }
  ],
  "points": [
    {
      "name": "1",
      "actions": [
        {
          "fields": [
            {
              "fields": [],
              "name": "millisecond",
              "type": "3",
              "value": "3000" //停留 3000ms
            }
          ],
          "isChecked": false,
          "isDevice": false,
          "name": "Pause"
        },
        {
          "fields": [
            {
              "fields": [],
              "name": "rotation",
              "type": "double",
              "value": "0" //旋转到45° 范围:0°-360°
            }
          ],
          "isChecked": false,
          "isDevice": false,
          "name": "Rotate2d" //旋转
        }
      ],
      "gridPosition": {
        "x": 108,
        "y":
      370 },
      "angle": 0,
      "moveType": 0
```

```
        },
        {
          "name": "2",
          "actions": [],
          "gridPosition": {
            "x": 148,
            "y":
          370 },
          "angle": 0,
          "moveType": 0
        }
      ],
      "name": "path1_path0"
    }
  ],
  "type": 3,
  "close": false //是否为闭环路径(至少三个点才能形成闭环路径) }
```

4.Task Interface Call Example



Before issuing a task, it is crucial to ensure that the robot's positioning was successful. If the positioning was unsuccessful, follow the initialization steps mentioned above for positioning.

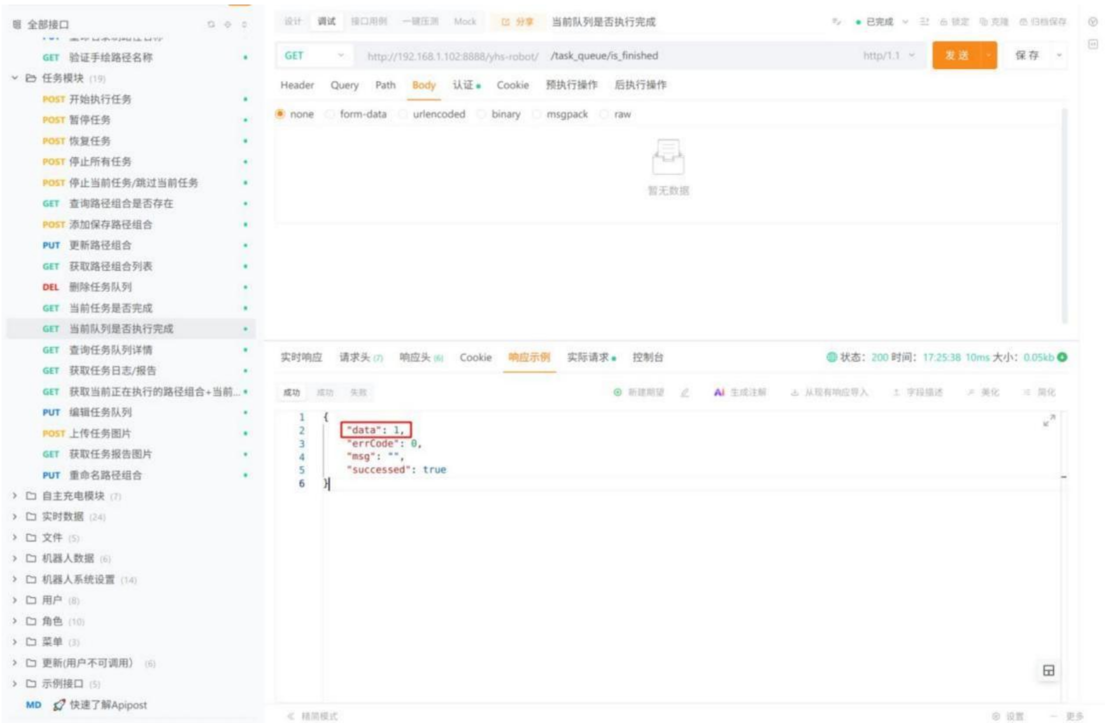
① Call the task execution interface /task_queue/start to begin executing the task.

```
{
    "loop": false, //是否循环执行
    "loop_count": 2, //循环次数，当 loop为false时，loop_count默认为1
    "map_name": "map", //地图名称
    "name": "", //路径组合名称，可选，需要执行该队列名称对应的任务时，传入该值。当对应的队列数据不存在时，则执行 tasks 中的数据。
    "tasks": [ //task中的任务,按照先后顺序进行执行
```

```
        {
            "name": "NavigationPointTask", // 导航点任务
            "start_param": {
                "map_name": "in", //地图名称
                "position_name": "a" // 导航点名
            称 }
        },
        {
            "name": "NavigationPositionTask", // 根据位姿下发导航点
            "start_param": {
                "destination": {
                    "angle": 0, //导航的角度
                    "gridPosition": { //导航点栅格坐标
                        "x": 449,
                        "y": 361
                    }
                }
            }
        },
        {
            "name": "PlayGraphPathTask", // 手绘路径任务
            "start_param": {
                "graph_name": "path2", //手绘路径名称
                "map_name": "" //地图名称
            }
        },
        {
            "name": "PlayPathTask", // 录制路径任务
            "start_param": {
                "map_name": "in1", //地图名称
                "path_name": "record" //录制路径名
            称 }
        }
    ]
}
```

If you want to execute a specific type of task, fill in the corresponding parameters in the task. There are four types of navigation tasks: navigation point task, navigation point task based on pose, hand-drawn path task, and recorded path task. Except for the navigation point task based on pose, the other three tasks require creating path points or routes in advance, as shown in the "Path and Point Creation Interface Call Example".

②Call the /task_queue/is_finished interface to check the current status of the task that was sent for execution.

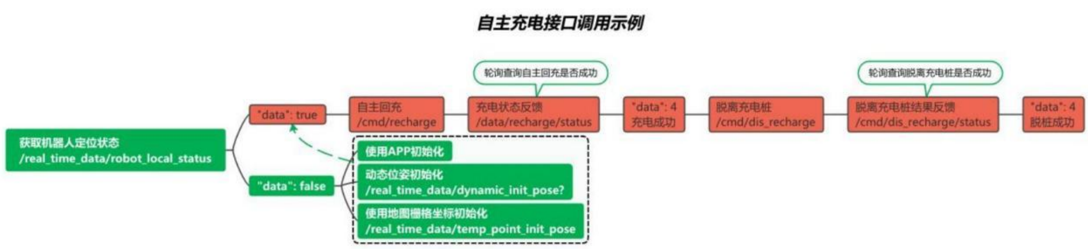


When calling the /task_queue/is_finished interface, the following responses indicate different statuses of the navigation task:

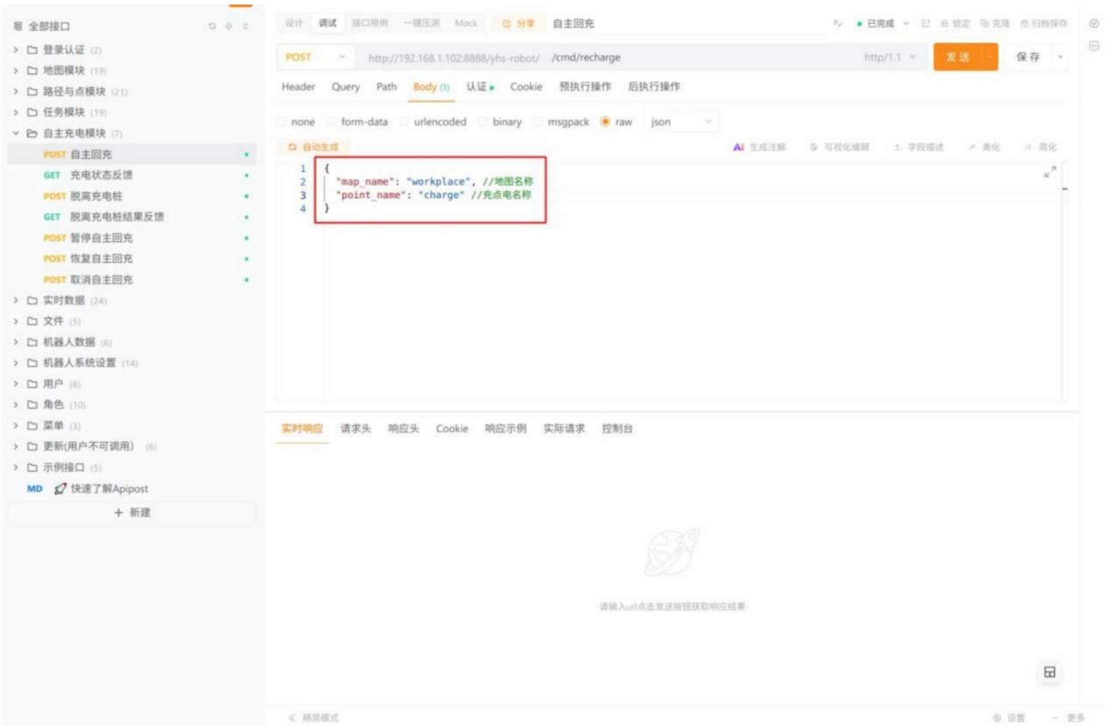
- {"data":1,"errCode":0,"msg":"","successed":true}: Indicates that the navigation task is currently in progress.
- {"data":2,"errCode":0,"msg":"","successed":true}: Indicates that the navigation task is paused.
- {"data":0,"errCode":0,"msg":"","successed":true}: Indicates that the navigation task has not started or has been completed.
- {"data":0,"errCode":0,"msg":"Cannot reach destination","successed":true}: Indicates that the navigation task cannot be completed.

It is important to note that task issuance cannot be preempted; you must wait for the previous task to finish before issuing a new task.

5. Autonomous Charging Interface Call Example

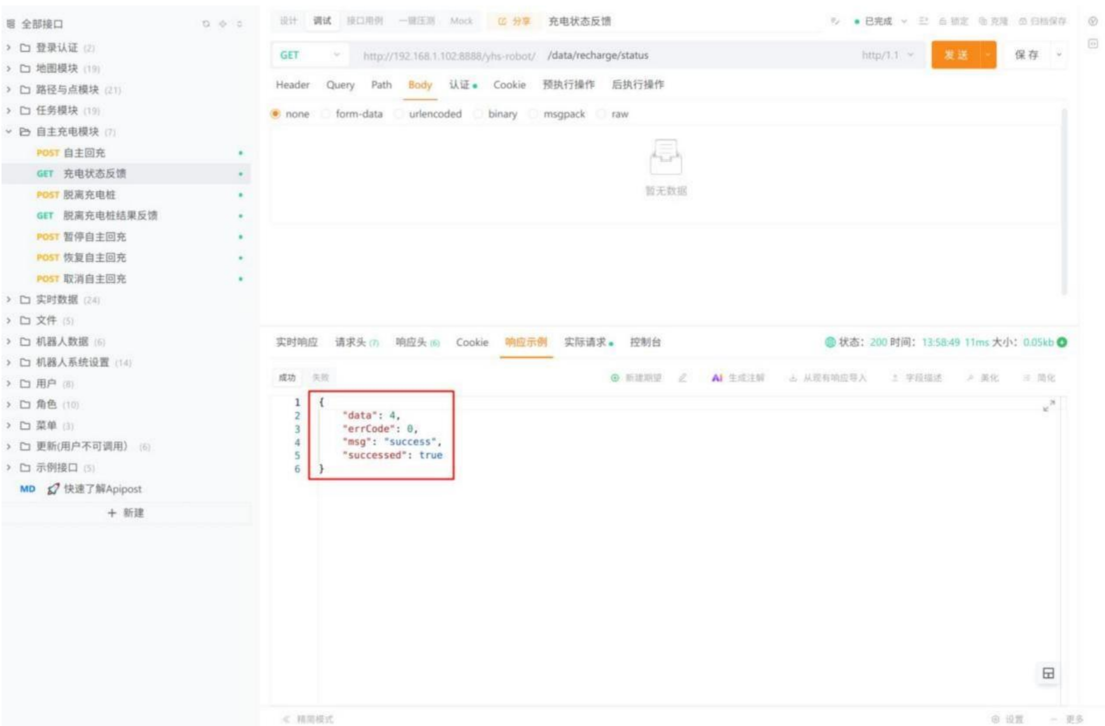


① Call the /cmd/recharge interface for autonomous return to charge.



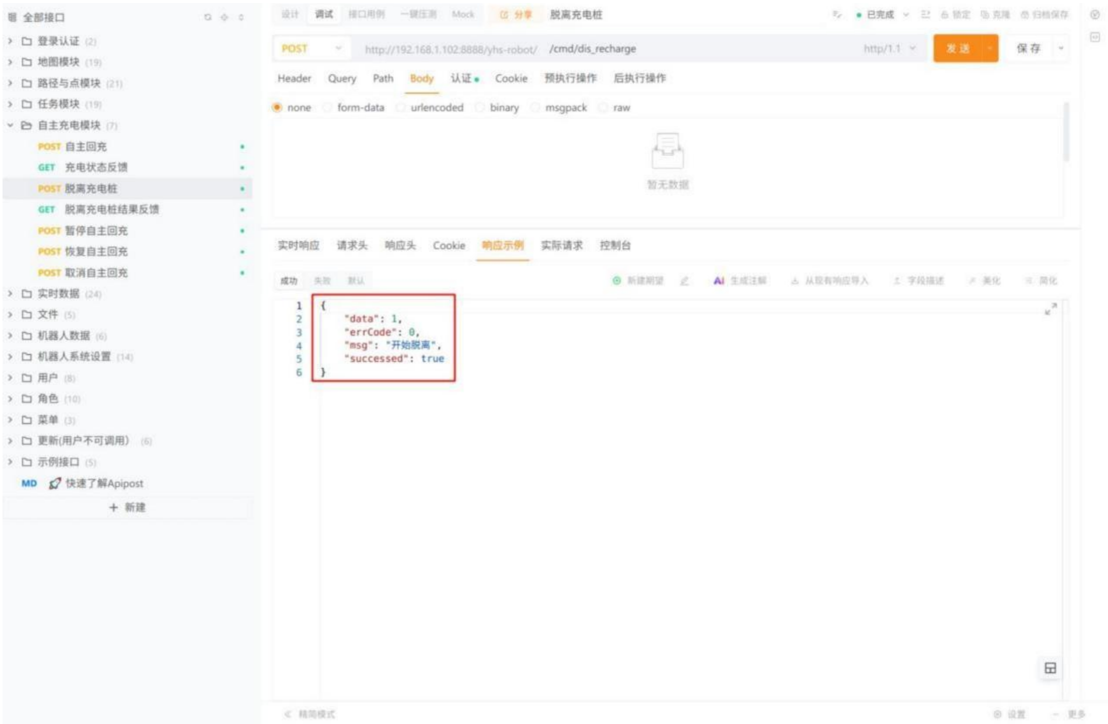
The charging point needs to be created in advance. After calling the interface, the robot will automatically navigate to a position approximately 1m in front of the charging station and then initiate the charging process. Ensure that there are no obstacles obstructing the path to the charging station.

②Poll for the charging status feedback during the charging process by calling the /data/recharge/status interface.



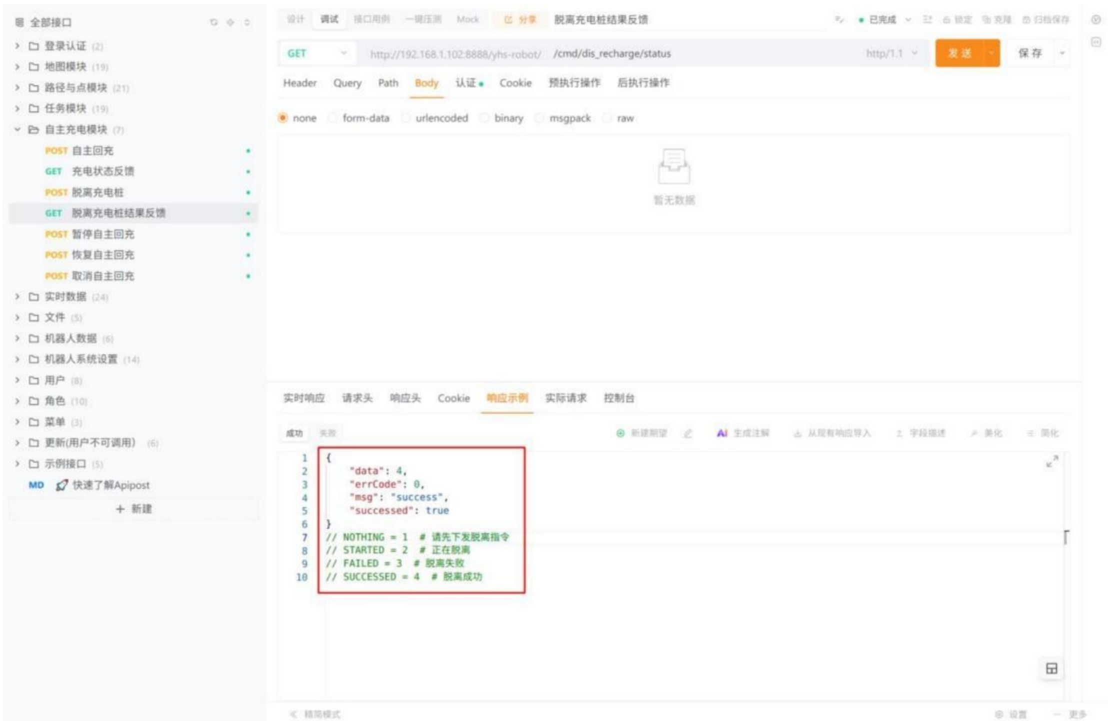
- "data": 1 # Please issue the charge command first.
- "data": 2 # En route to the charging station.
- "data": 3 # Failed.
- "data": 4 # Charging successful.

③Call the /cmd/dis_recharge interface to detach from the charging station.



After calling the interface, the robot will move forward a short distance to detach from the charging station. Ensure that there are no obstacles in front of the charging station.

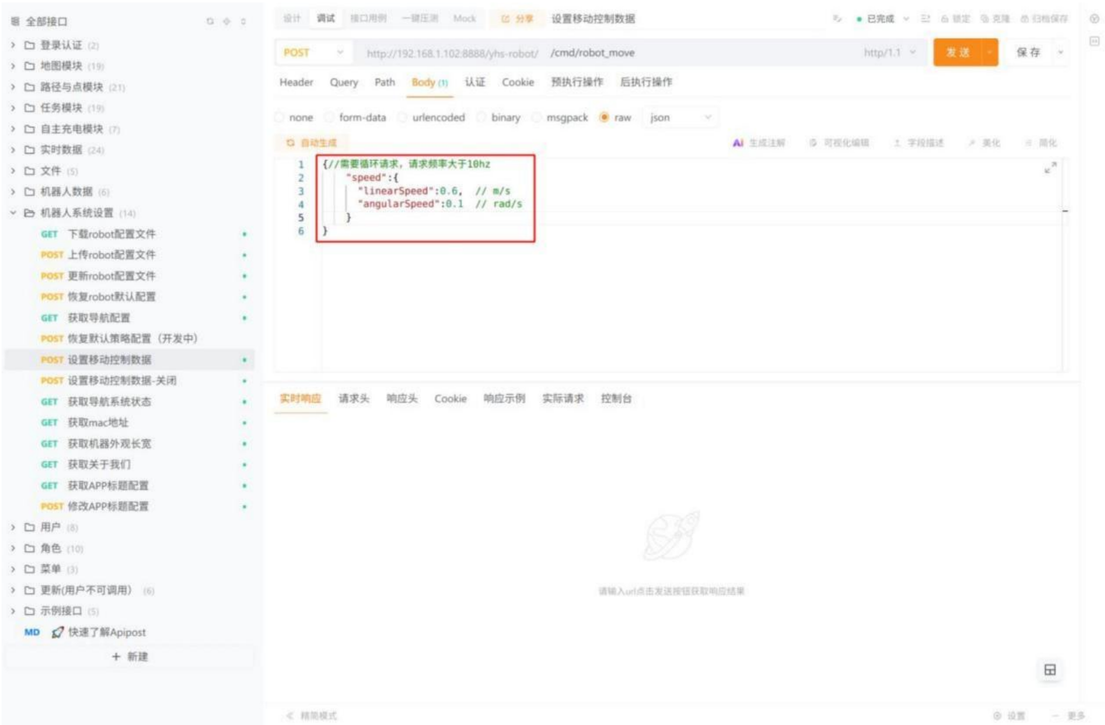
④During the detachment process, use the /cmd/dis_recharge/status interface to poll for the detachment result.



- "data": 1 # Please issue the detach command first.
- "data": 2 # Detaching in progress.
- "data": 3 # Detachment failed.
- "data": 4 # Detachment successful.

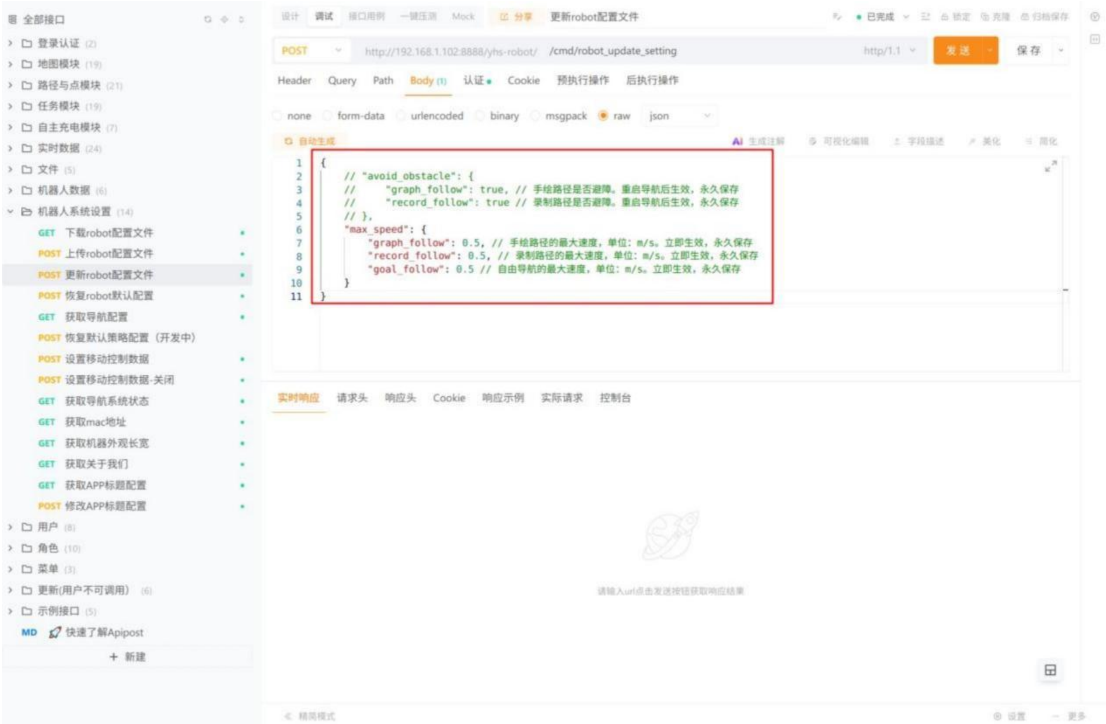
6. Other Interface Call Examples

①Call the /cmd/robot_move interface to directly control the movement of the base.



This interface requires polling calls.

②Call the /cmd/robot_update_setting interface to modify the navigation obstacle avoidance mode and the maximum navigation speed.



After modifying the obstacle avoidance mode, you need to restart the navigation for the changes to take effect. The changes to the maximum speed take effect immediately.

7. FAQs

1. Explanation of errCode in the interface:

- 0: Call successful
- -1: Call failed
- 401: Unauthorized or invalid token

2. **Does Magic Navigation have the functionality of extending maps?**
Currently, it does not have the capability of extending maps.

3. **What should be done if the log timestamps on the robot do not synchronize with the actual time?**
Access the robot's Ubuntu system, connect it to the network, and it should automatically synchronize the time.

4. **What is the IP address of the robot, and can it be modified?**
The robot's fixed IP address is 192.168.1.102, and it can be modified by contacting the relevant technical personnel.

5. How to connect the robot to the internet?

- Connect a mobile phone to the robot's USB port via a USB data cable and use the phone's USB tethering function for internet access.
- Connect a network cable to the Ethernet port of the robot's router.
- Use a router that supports inserting IoT cards for internet connectivity.