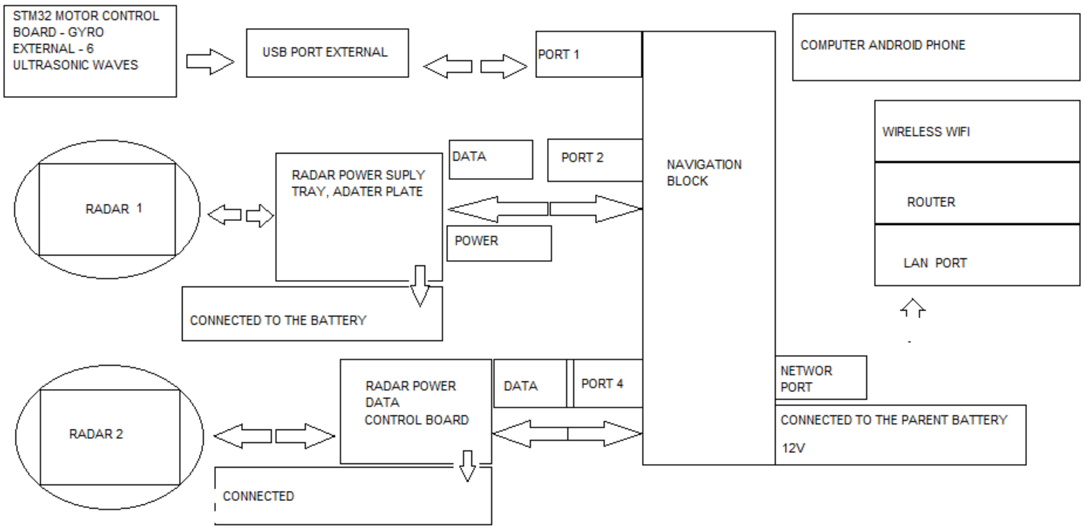
User Manual of Mobile System

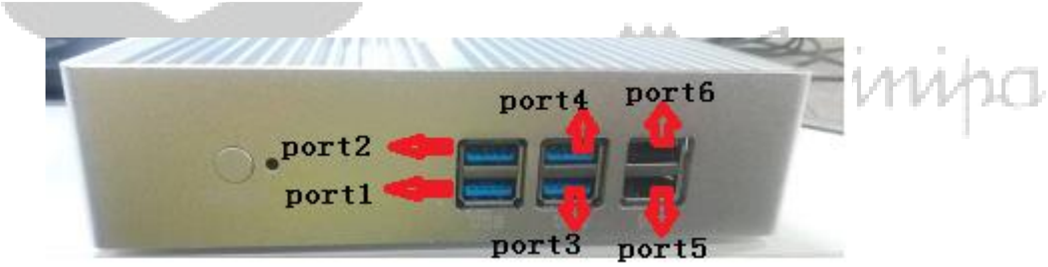
Table of Contents

User Manual of Mobile System

The hardware connection of mobile system

Normally, all hardware will be connected when leaving the factory. If any hardware has been plugged or removed, please reconnect it according to the following figure:





The above figure shows the ports of the navigation module:

Port 1 is connected to the chassis.

Port 2 is connected to the upper Lidar and used for mapping, navigation and obstacle avoidance.

Port 3 is reserved for a camera.

Port 4 is connected to the lower Lidar and only used for navigation and obstacle avoidance.

Port 5 and Port 6 are reserved for use.

Controlling Mobile System to Map and Navigate Using a Computer

1.1.1 Environmental preparation-the installation of ROS system on a computer

ROS system must run in an Ubuntu Linux system, so an Ubuntu system must be installed on the computer first, followed by a ROS system. If an Ubuntu 16.04 system has been installed on the computer, a corresponding ROS kinetic system should be installed (if an Ubuntu 14.04 system has been installed, an Indigo version of ROS should be installed).

For the installation of Ubuntu system, please refer to:

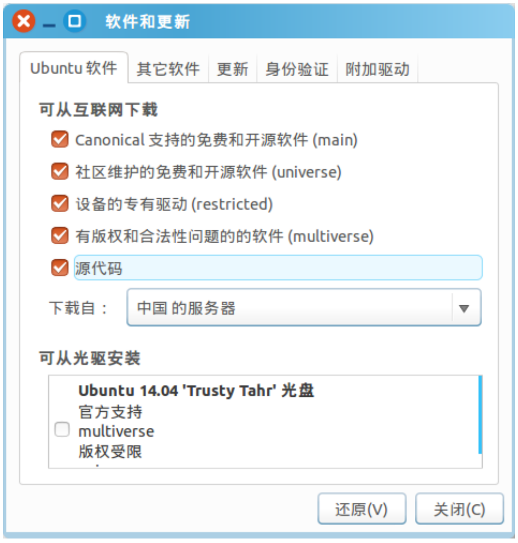
How to install a dual Ubuntu system on Windows 7:
http://blog.csdn.net/eaibot/article/details/53640828

How to directly install an Ubuntu system:
https://jingyan.baidu.com/article/3c48dd348bc005e10be358eb.html

Configuring Ubuntu software repositories

Configure your Ubuntu software repositories to enable three installation modes, i.e., restricted, universe and multiverse. The server must be domestic.

System Settings> Software and Update> Ubuntu Software, modify the settings as follows:



可从互联网下载

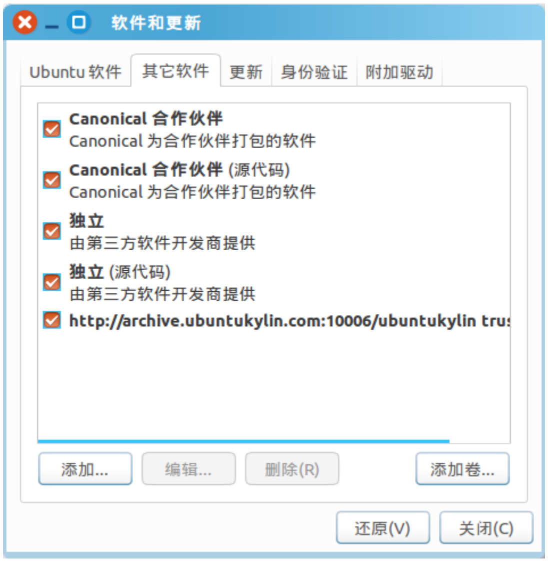
☑ Canonical 支持的免费和开源软件 (main)
☑ 社区维护的免费和开源软件 (universe)
☑ 设备的专有驱动 (restricted)
☑ 有版权和合法性问题的的软件 (multiverse)
☑ 源代码

下载自： 中国 的服务器

可从光驱安装

□ Ubuntu 14.04 'Trusty Tahr' 光盘
官方支持
multiverse
版权受限

System Settings> Software and Update> Other Software, modify the settings as follows:



☑ Canonical 合作伙伴
Canonical 为合作伙伴打包的软件
☑ Canonical 合作伙伴 (源代码)
Canonical 为合作伙伴打包的软件
☑ 独立
由第三方软件开发商提供
☑ 独立 (源代码)
由第三方软件开发商提供
☑ http://archive.ubuntukylin.com:10006/ubuntukylin tru

Click “Close” (C) button and wait for the cache to be updated.

Configuring the apt source of ROS

For the apt source of ROS, official source, domestic USTC source or Singapore source are optional. Just choose one from them. It is recommended to use domestic USTC source or Singapore source. The installation speed will be much faster. (During installation, it is recommended to use wired network, so that it is not easy to make mistakes.)

◆ Mode 1: Official source

```
$ sudo sh -c ‘echo “deb http://packages.ros.org/ros/ubuntu $(lsb_release -sc) main” > \
/etc/apt/sources.list.d/ros-latest.list’
$ sudo apt-key adv --keyserver hkp://ha.pool.sks-keyservers.net:80 --recv-key \
421C365BD9FF1F717815A3895523BAEEB01FA116
$ sudo apt-get update
```

Mode 2: Domestic USTC source

URL: http://mirrors.ustc.edu.cn/ros/

```
$ sudo sh -c ‘. /etc/lsb-release && echo “deb http://mirrors.ustc.edu.cn/ros/ubuntu/ \
$DISTRIB_CODENAME main” > /etc/apt/sources.list.d/ros-latest.list’

$ sudo apt-key adv --keyserver hkp://ha.pool.sks-keyservers.net:80 --recv-key \
421C365BD9FF1F717815A3895523BAEEB01FA116
$ sudo apt-get update
```

- Mode 3: Singapore source

URL: http://mirror-ap.packages.ros.org/

```
$ sudo sh -c ‘. /etc/lsb-release && echo “deb http://mirror-ap.packages.ros.org/ros/ubuntu/ \
$DISTRIB_CODENAME main” > /etc/apt/sources.list.d/ros-latest.list’
$ sudo apt-key adv --keyserver hkp://ha.pool.sks-keyservers.net:80 --recv-key \
421C365BD9FF1F717815A3895523BAEEB01FA116
$ sudo apt-get update
```

sudo apt-get update may have an error when performing an update because of the network (ROS installation can continue as long as the ROS installation source is correct). You can re-perform the command to update.

Installing an ROS software package

```
$ sudo apt-get install ros-kinetic-desktop-full

$ sudo apt-get install python-rosinstall
```

71 software packages are updated, 799 new software packages are installed, 0 package is to be uninstalled and 314 packages are not updated.

It is necessary to download 390 MB software package.

When the package is decompressed, 1,620 MB of extra space will be consumed.

---

sudo apt-get install ros-kinetic-desktop-full If ROS Kinetic is not decompressed when fully downloaded during installation, there will be no ROS directory under /opt/. Probably, the update source is wrongly selected. As a result, the package cannot be fully downloaded and ROS cannot be decompressed and installed. It is necessary to change to a domestic source, and then run sudo apt-get update to reinstall.

Configuring the environment variable

```
$ sudo rosdep init

$ rosdep update


$ echo “source /opt/ros/kinetic/setup.bash” >> ~/.bashrc

$ source ~/.bashrc
```

Testing whether ROS is successfully installed

Input roscore –h into the terminal. When the following output appears, the installation is successful.

```
$ roscore -h

Usage: roscore [options]
```

```
roscore will start up a ROS Master, a ROS Parameter Server and a rosout

logging node


Options:

-h, --help show this help message and exit

-p PORT, --port=PORT master port. Only valid if master is launched

-v verbose printing


See http://www.ros.org/wiki/roscore
```

Input roscore into the terminal. When the following output appears, the configuration of environment variable is successful and ROS runs normally.

```
eaibot@eaibot:~$ roscore

... logging to /home/eaibot/.ros/log/45d93ed8-a23a-11e6-99b1-4437e63de0fc/roslaunch-eaibot-3460.log

Checking log directory for disk usage. This may take awhile.

Press Ctrl-C to interrupt

Done checking log file disk usage. Usage is <1GB.


started roslaunch server http://eaibot:35377/

ros_comm version 1.11.20



SUMMARY

========


PARAMETERS

* /rosdistro: kinetic

* /rosversion: 1.11.20
```

```
NODES


auto-starting new master
process[master]: started with pid [3472]
ROS_MASTER_URI=http://eaibot:11311/


setting /run_id to 45d93ed8-a23a-11e6-99b1-4437e63de0fc
process[rosout-1]: started with pid [3485]
started core service [/rosout]
```

1.1.2 Building an environment

Setting the user's access to a serial port

```
$ sudo usermod -a -G dialout your_user_name
```

your_user_name is replaced with an actual user name.

Installing a dependent package

```
$ sudo apt-get install git python-serial ros-kinetic-serial g++ \
  ros-kinetic-turtlebot-rviz-launchers ros-kinetic-teleop-twist-keyboard \
ros-kinetic-navigation ros-kinetic-slam-gmapping ros-kinetic-teb-local-planner
```

Obtaining and compiling dashgo_ws engineering package

Please ascertain whether your environment is Ubuntu 14.04 +ROS Indigo or Ubuntu 16.04 +ROS Kinetic and select a proper dashgo_ws version from the repositories and then place the dashgo_ws folder in the home folder of active user (i.e., in ~/ directory).

```
eaibot@eaibot:~$ cd ~
eaibot@eaibot:~$ cd dashgo_ws
eaibot@eaibot:~$ sudo chmod 777 ./*   -R
```

```
eaibot@eaibot:~/dashgo_ws$ ls
build    devel    src
eaibot@eaibot:~/dashgo_ws$ rm -rf build/
eaibot@eaibot:~/dashgo_ws$ rm -rf devel/
eaibot@eaibot:~/dashgo_ws$ catkin_make
```

After dashgo_ws folder is copied, put it into the home folder of active user, switch it to dashgo_ws, delete build and devel folders using rm command and recompile using catkin_make.

After catkin_make is compiled, add it to ~/.bashrc file of Dashgo environment variable.

```
$ echo “source ~/dashgo_ws/devel/setup.bash” >> ~/.bashrc
$ source ~/.bashrc
```

source ~/.bashrc can enable the configuration of environment variable.

1.1.3 Launching and controlling B1 system to move and map using the computer

ROS system has been installed in the computer and EAI-Dashgo environment has been built. K4 hardware has been normally connected.

Configuring /etc/hosts files in the computer and the navigation module respectively

First of all, connect the computer to a chassis router. At the computer terminal,

```
feibot@feibot:~$ hostname       //view the host name of Ubuntu system of the computer.
Mine is feibot.
feibot@feibot:~$ ifconfig      //view the IP address of WLAN card of the computer. Mine is
192.168.31.143.
feibot@feibot:~$ sudo vim /etc/hosts    //open /etc/hosts in the computer
```

Add IP address and host name of the navigation module to the end of the opened /etc/hosts in the computer, which are invariably 192.168.31.200   PS3B-B1.

Add IP address and host name of the Ubuntu system previously viewed to the end of /etc/hosts in the navigation module.

```
feibot@feibot:~$ ssh  eaibot@192.168.31.200  //enter the navigation module remotely
eaibot@PS3B-K4:~$ sudo vim /etc/hosts  //open /etc/hosts in the navigation module
```

Add IP address and host name of the navigation module to the end of the opened /etc/hosts in the computer. What I previously viewed was 192.168.31.143　feibot.

Note: If /etc/hosts files in the computer and navigation module are not correctly modified, when mapping, rviz in the Ubuntu system will not be able to display the map. During navigation, there will be no response if you click rviz to set the starting point and goal.

Launching B1 mapping remotely and displaying a map using the computer

Enter the navigation module remotely in the computer and launch mapping in the navigation module.

```
feibot@feibot:~$ ssh  eaibot@192.168.31.200  //enter the navigation module remotely
eaibot@PS3B-K4:~$ roslaunch dashgo_nav gmapping_imu.launch  //launch mapping
```

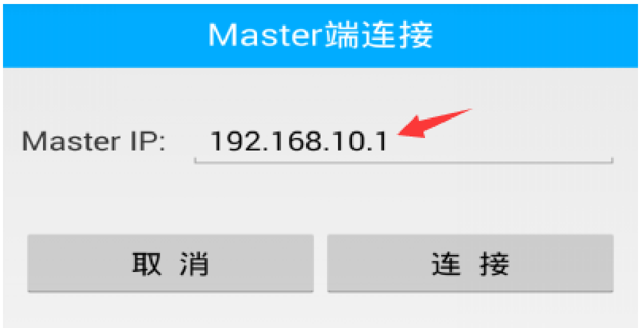
Open rviz directly on the computer and observe the map.

```
export ROS_MASTER_URI=http://192.168.31.200:11311
roslaunch dashgo_rviz view_navigation.launch
```

Constructing a map by moving the chassis and scanning with Lidar

(1) Mode I: Controlling the chassis to move using DashgoApp

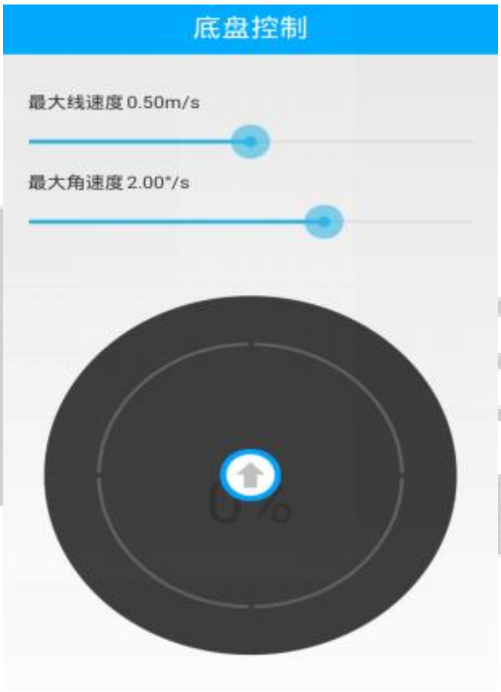
Connect the chassis router on the mobile phone, open DashgoAPP and select “WIFI” to enter WIFI connection interface, as shown in the following figure:



Master IP: 192.168.10.1

The Master IP to be input is the IP address of the navigation module, i.e., 192.168.31.200.

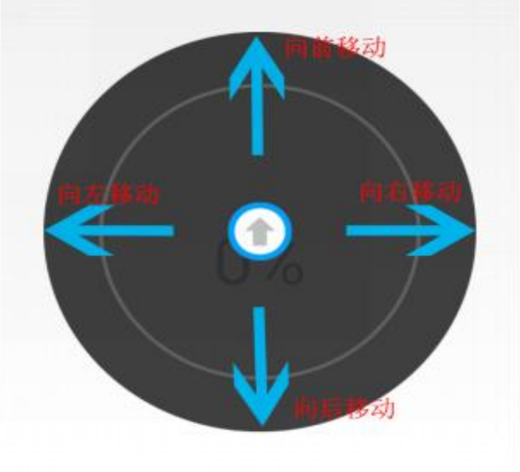
Click “Connect”. After WIFI is successfully connected, the interface is as follows:



最大线速度 0.50m/s

最大角速度 2.00°/s

The control of direction is shown below:



Note: DashgoApp cannot be connected and used unless the chassis is launched (e.g. gmapping_imu.launch). Also it can only be used to control the chassis to move.

(2) Mode 2: Controlling the chassis to move and map using the keyboard

Keep gmapping_imu.launch running normally, launch keyboard control at another terminal of the navigation module and move K4 to scan a map.

```
feibot@feibot:~$ ssh    eaibot@192.168.31.200    //enter the navigation module remotely
eaibot@PS3B-K4:~$ rosrun dashgo_tools    teleop_twist_keyboard.py    //launch keyboard control to move
```

After keyboard control is successfully launched, on the keyboard, “i” stands for forward, “,” for backward, “j” for left, “l” for right and “k” for stop.

Save the map

Keep gmapping_imu.launch running normally and scan a map. At this point, enter the map directory of the navigation module: dashgo_ws/src/dashgo/dashgo_nav/maps and save the new map in the directory.

```
feibot@feibot:~$ ssh    eaibot@192.168.31.200    // enter the navigation module remotely
eaibot@PS3B-K4:~$ roscd    dashgo_nav/maps     // enter the map directory of the navigation module
eaibot@PS3B-K4:~$ rosrun    map_server    map_save –f    eai_map_imu    // save the map
```

After the map is saved, press ctrl+c to stop mapping and keyboard control.

Launching B1 navigation remotely using the computer

Make sure that the mapping program has already stopped and that the terminal is in the navigation module (if not, you need to log in the navigation module remotely, ssh eaibot@192.168.31.200).

```
$ roslaunch dashgo_nav navigation_imu.launch
```

In the computer, launch rviz and observe the map.

```
export ROS_MASTER_URI=http://192.168.31.200:11311
roslaunch dashgo_rviz view_navigation.launch
```

After rviz is opened, the default location of E6 is the central point of the grid, which is not necessarily the actual location of E6. Therefore, every time rviz is opened, it is necessary to check and set the starting point.

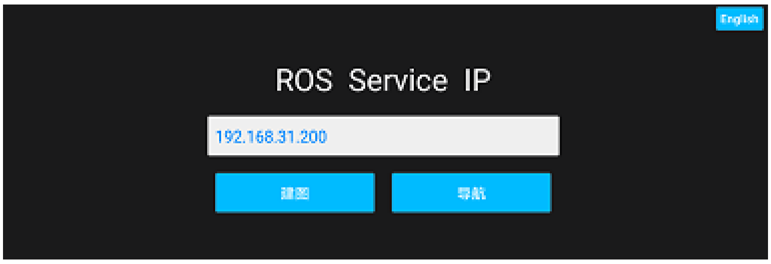
Set the starting point of the robot: click 2D Pose Estimate on rviz, click on a corresponding position on the map, set the right direction and set the starting point of the robot.

Set the goal of the robot: click 2D Nav Goal on rviz and click on the position of goal on the map. At this point, normally, the robot will plan the route leading to the goal and move to it.

Controlling B1 Mobile System to Map and Navigate Using Android APP

1.1.7 Mobile APP- the mapping and navigation of EAIGO

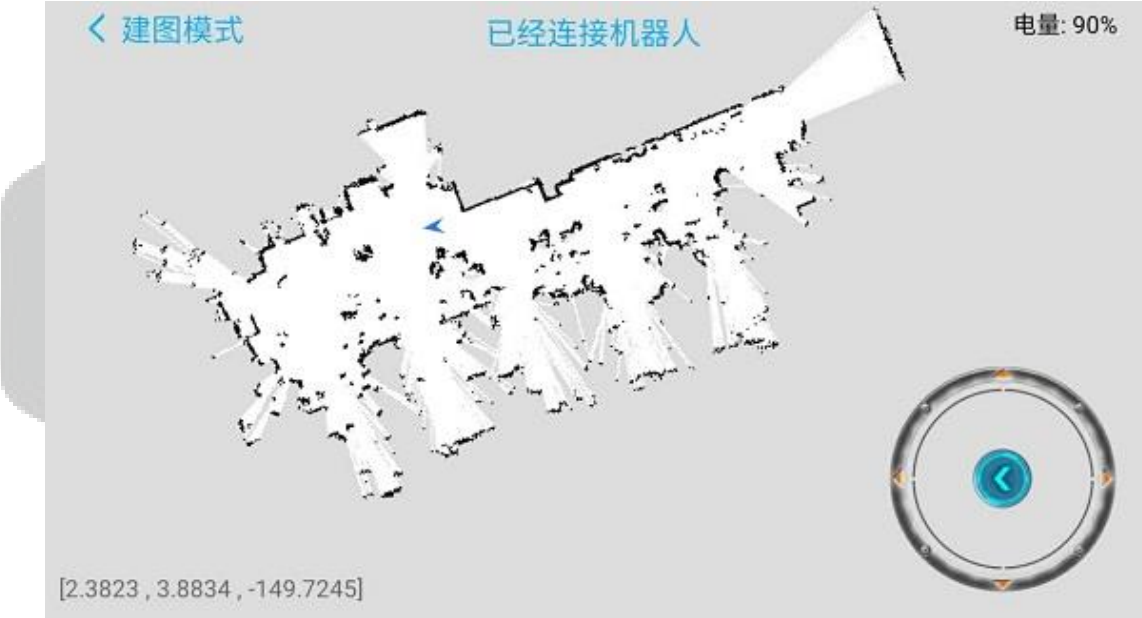
Step 1: Install EAIGO5X.apk on the mobile phone, connect the chassis router through WIFI, open EAIGO, enter the IP address of the navigation module, 192.168.31.200 and click mapping button to start mapping.



ROS Service IP

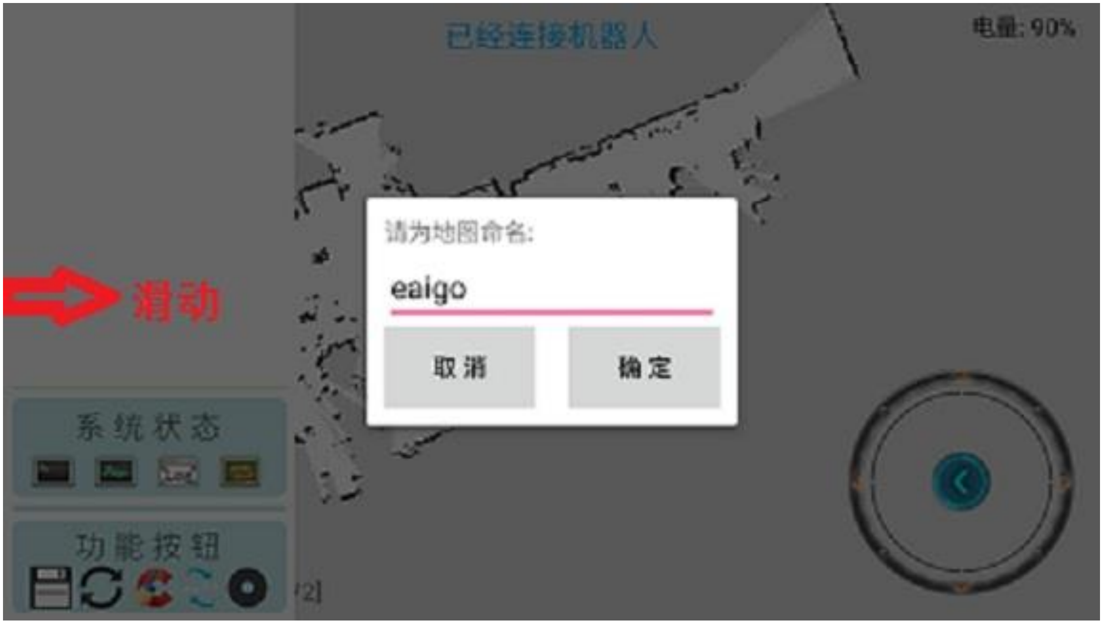
192.168.31.200

Step 2: Wait for about 10s until a map is shown on the mobile phone. At this point, control the cart to move using a steering wheel in the lower right corner and scan a map. The results are shown as follows:



[2.3823 , 3.8834 , -149.7245]

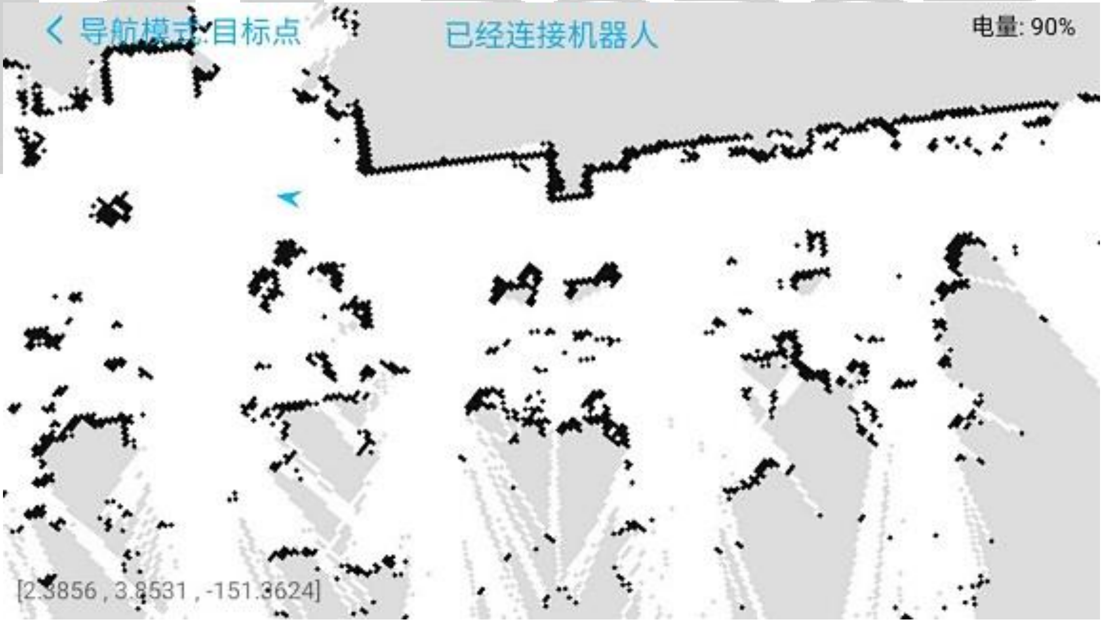
Step 3: After moving and scanning a map, slide out the selection bars on the left, click the first button of the function bar 💾, enter the map name you want to save and press OK.



请为地图命名:

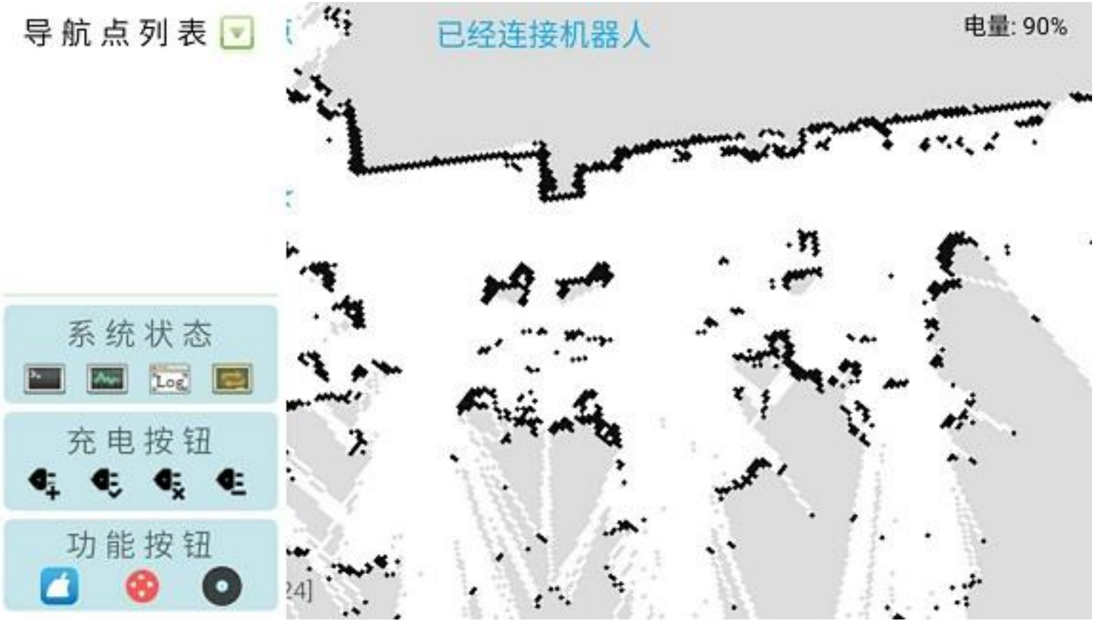
eaigo

Step 4: After the map is saved, click the fourth button of the system state bar, a dialog box will pop up. Choose "Yes" and switch from mapping mode to navigation mode. In the upper left corner, it shows that currently the platform is in navigation mode and a goal-setting state.

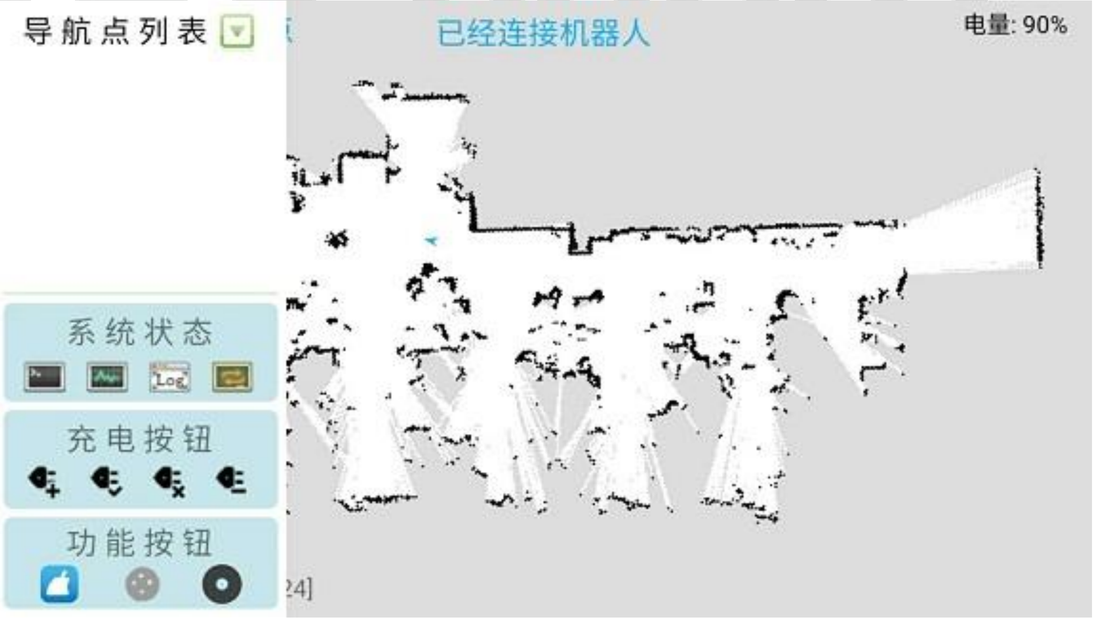


[2.3856 , 3.8531 , -151.3624]

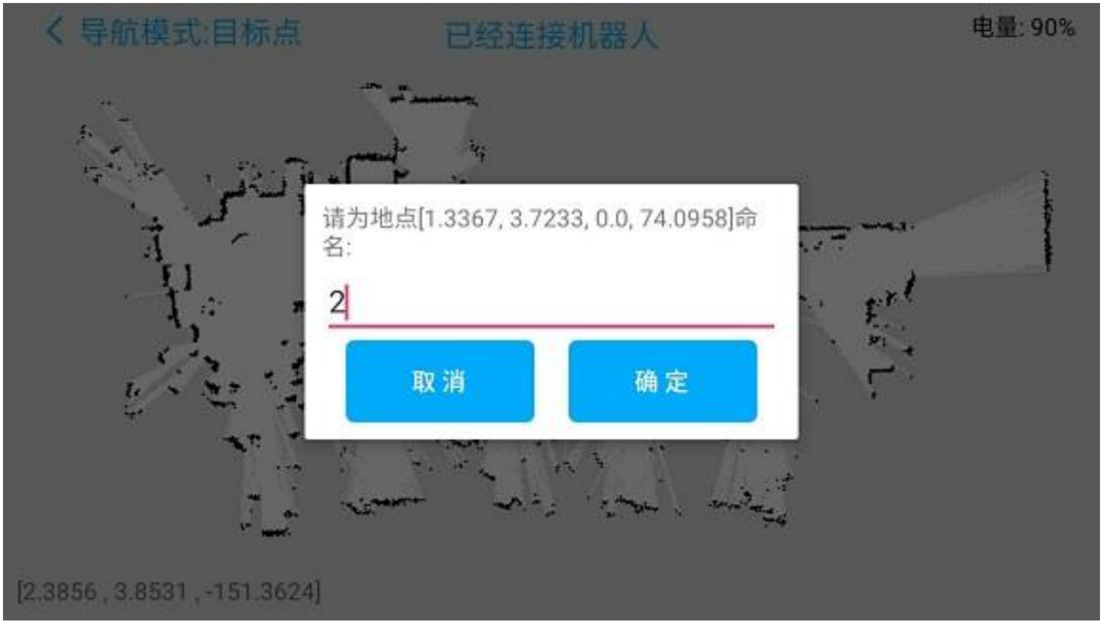
Step 5: (If the size of the displayed map is proper, this step can be omitted) If the size of the map is improper, it can be zoomed in or out. Slide out the selection bars on the left of APP and click ● to switch to the zoom state ● (red means the zoom state and gray means the goal-setting state. The default is gray).



Step 6: (If the size of the displayed map is proper, this step can be omitted). After the map is zoomed, it is necessary to reclick ⊕ to switch back to ⊕, i.e., the goal-setting state. Otherwise, a goal cannot be set by clicking on the map.



Step 7: After switching back to the goal-setting state, click on the map, set a goal and set the name of goal.

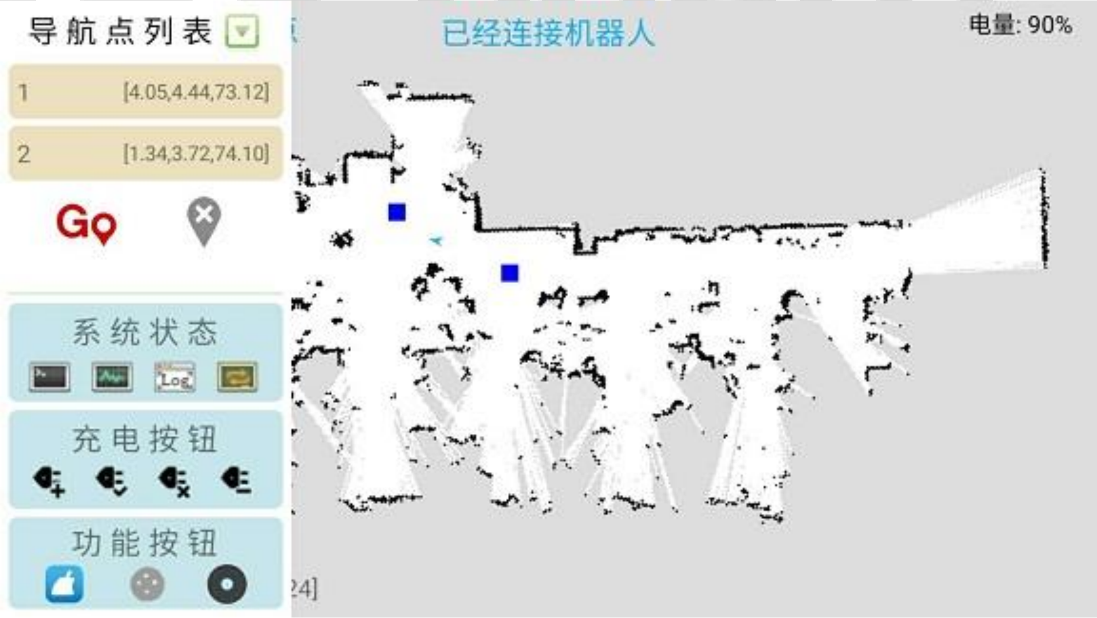


请为地点[1.3367, 3.7233, 0.0, 74.0958]命名:

2

[2.3856 , 3.8531 , -151.3624]

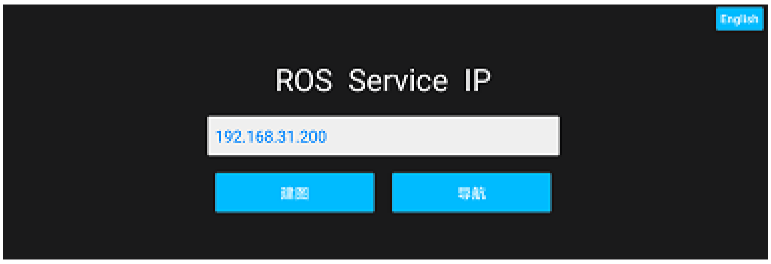
Step 8: After setting a goal, click Go in the function bar on the left to start navigation. Click ✖ to cancel navigation. Normally, if you set multiple goals, the system will navigate among multiple goals cyclically. If only a goal is set, it will navigate only once.



1	[4.05,4.44,73.12]
2	[1.34,3.72,74.10]

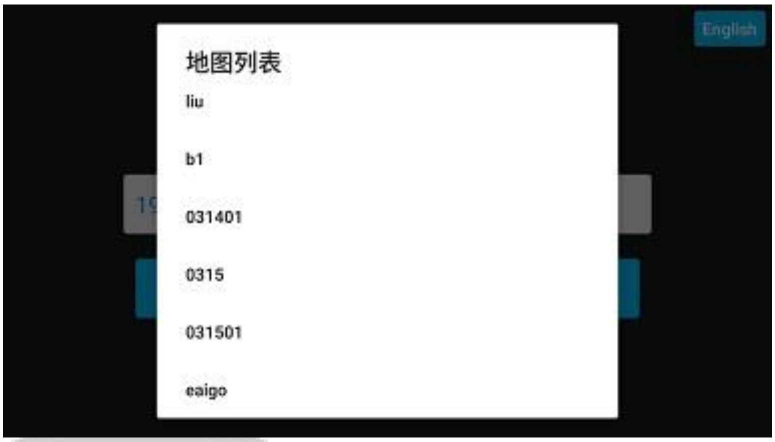
1.1.4 Directly selecting an existing map on APP to start navigation

Before opening EAIGO interface, input the IP address of the navigation module, 192.168.31.200 and click the navigation button directly. At this point, a selection box for existing maps will pop up. Just click and select a proper map to enter the navigation mode directly.



ROS Service IP

192.168.31.200



地图列表

liu

b1

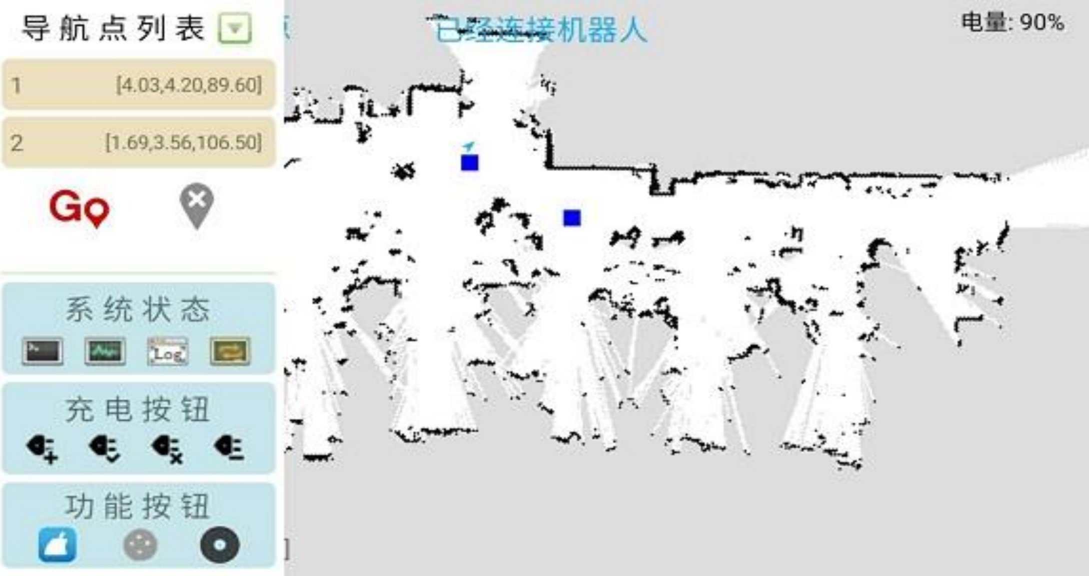
031401

0315

031501

eaigo

After entering the navigation mode, the operation is the same as the previous step, 5, 6, 7 and 8. First of all, click to begin to zoom the map. Then click to switch back to the goal-setting mode. Click on the map to set a goal. Click Go on the function bar on the left to start navigation.



导航点列表

1	[4.03,4.20,89.60]
2	[1.69,3.56,106.50]

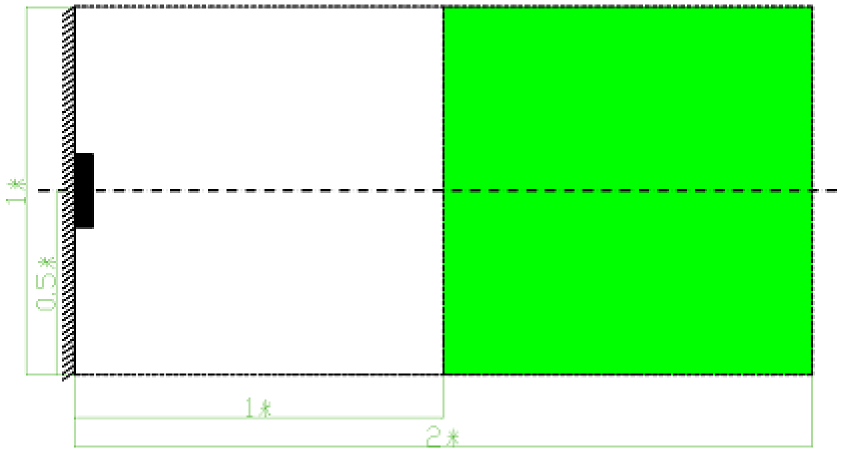
已经连接机器人

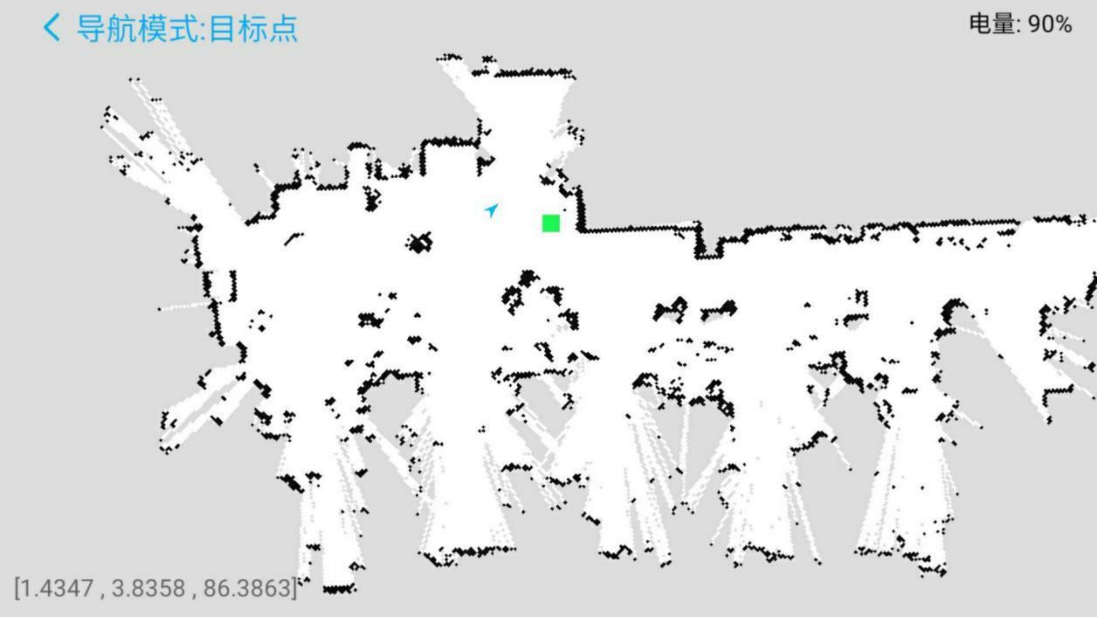
电量: 90%

1.1.5 The operations of other functions of APP

Automatic recharging.

Make sure that the APP has built a map and entered the navigation mode. Slide out the function bar on the left of APP. Click the first charge button and click on the map to set a proper recharging point. The recharging point is preferably about 1m right ahead of the charging pile, as shown in the following figure. The green area is the best recharging point.





[1.4347 , 3.8358 , 86.3863]

After the recharging point has been set,

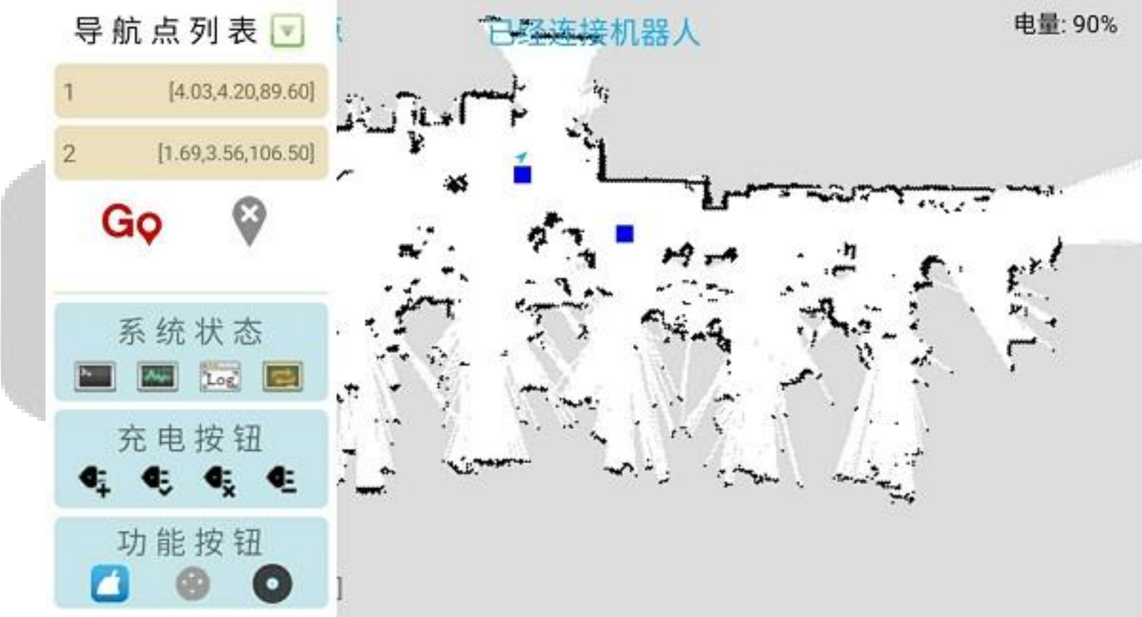
Click button and the chassis will first navigate to the recharging point, and then the infrared will look for the charging pile automatically. After the pile is found, recharging connection will begin.

Click button to cancel recharging and continue to navigate.

Click button to delete a recharging point that has been set.

Deleting a goal that has been set

To delete a single goal: in a list of navigation points, click Goal 1 and slide rightwards to delete it.



To delete all goals: click button to delete all goals that have been set.

The functions of other buttons

Under the mapping mode:

button is used to save the current map.

button is used to refresh the current map and request data.

button is used to clear the current map and rescan a map button is used to switch between manual mapping and auto mapping.

button is used to show/hide the virtual joystick.

Under the navigation mode, system state buttons:

button is used to show the launch of ROS services.

button is used to view basic information, such as the utilization rate of CPU of the navigation module and the occupancy of serial port.

Log button is used to view the error log of ROS.

button is used to switch between mapping mode and navigation mode.

Under the navigation mode, charge buttons:

button is used to set the position of recharging point.

button is used to make the chassis first navigate to the recharging point and then the infrared look for the charging pile automatically. After the pile is found, recharging connection will begin.

button is used to cancel recharging and continue to navigate.

button is used to delete a recharging point that has been set.

Under the navigation mode, function buttons:

button is used to delete all goals that have been set.

button is used to switch between zoom mode and goal-setting mode.

button is used to show the steering wheel button.

Revision History

Date	Content
March 19, 2018	First draft
March 19, 2018	An introduction to the framework and functions of BI software has been added.