

# Rosmaster\_X3 Navigation Using APP

## Quick to use tutorial

### 1、Rosmaster\_X3 power on

Turn on the power of X3, as shown in the following figure, and turn it to the ON direction.

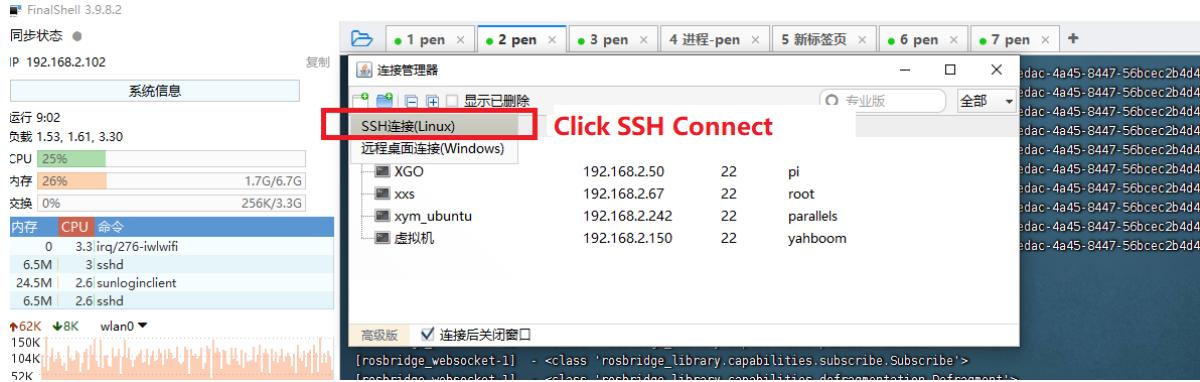


Connecting to the network can be visualized through the X3's built-in touch screen display, connecting to WiFi within the local area network

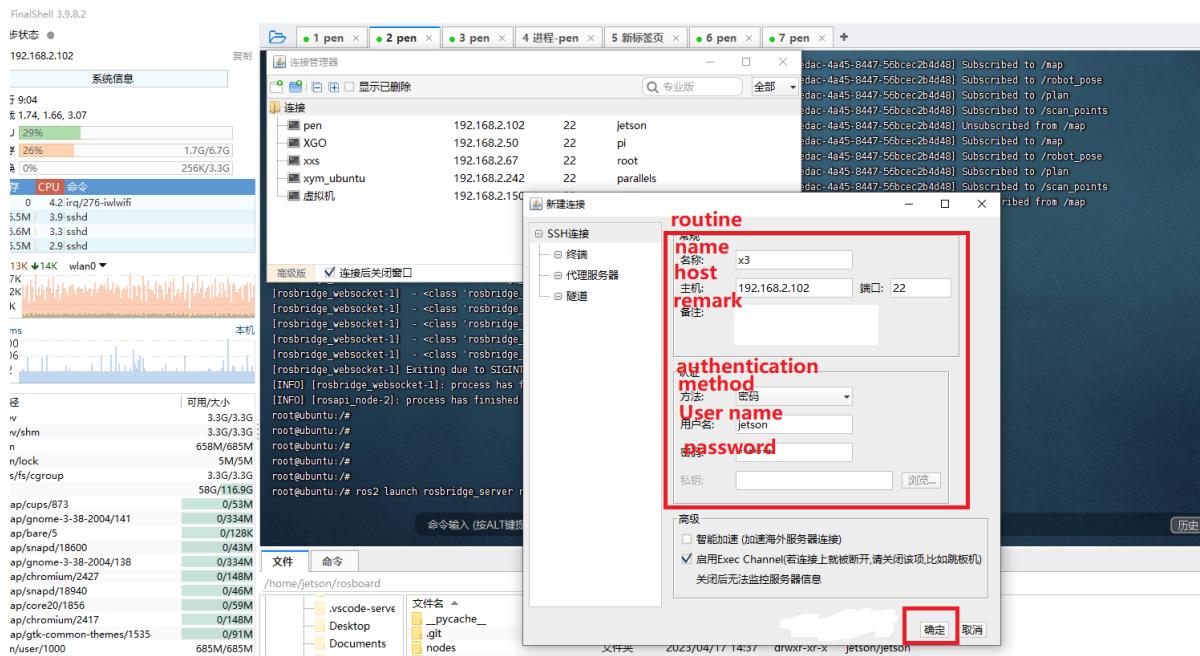
## 2. Open the shell to connect to Rosmaster\_X3

Note: The IP address used during the writing of this tutorial is 192.168.2.102. The username is: Jetson, and the password is: yahboom. The actual IP address used in this tutorial shall prevail.

Open the shell tool, and the shell tool I am using here is FinalShell. Enter information such as username, password, port, and connection name.

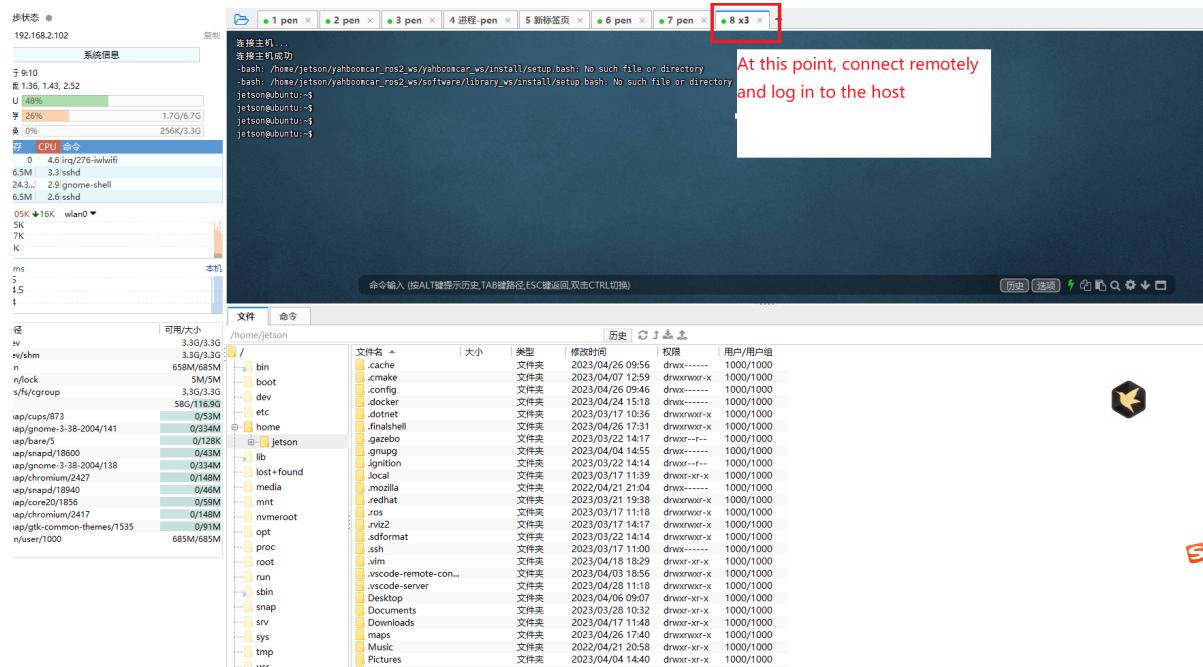


After clicking on SSH connection, the information input box shown in the following figure will appear:



Find your own newly created connection configuration names:





### 3. Start ROSBridge and related node services

**For Raspberry Pi P15 master, you need to enter the docker container first. For Orin master, you do not need to enter docker and do not need to do these four steps. You can just run the command directly.**

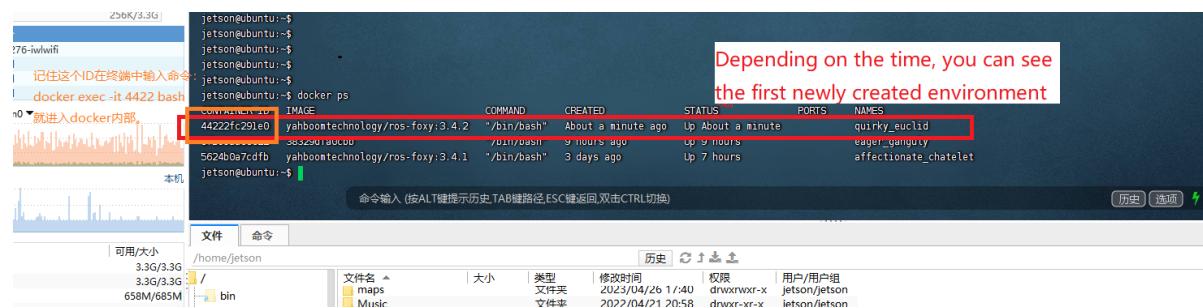
1. Enter the command in the terminal to create a docker environment.

```
~/run_docker.sh
```

2. Then enter the command:

```
docker ps
```

3. View the docker environment just created.



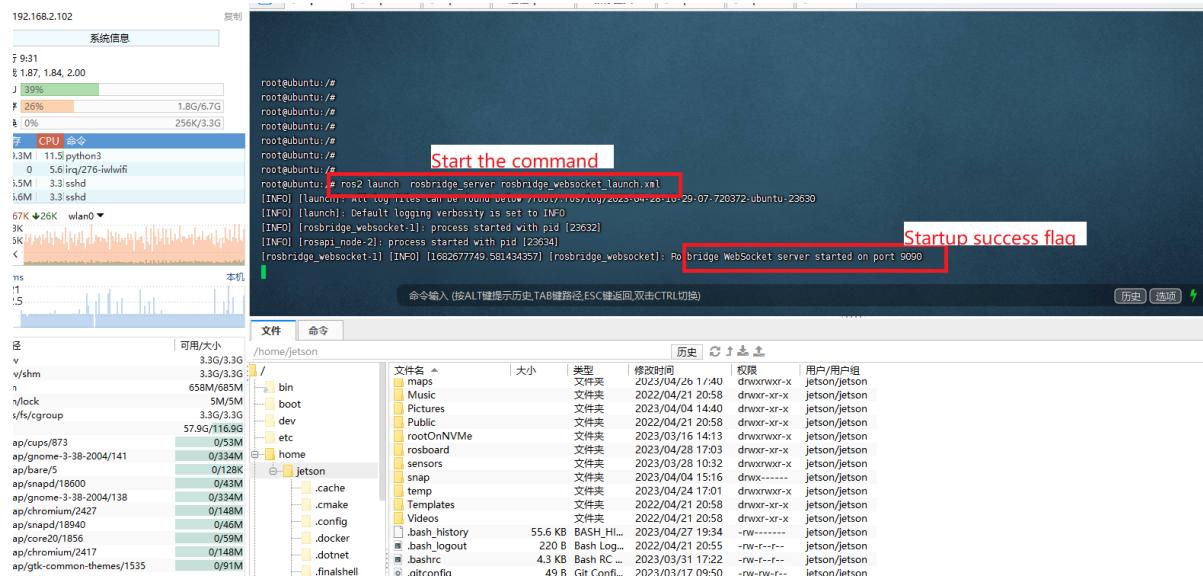
4. Find the docker environment just created and find the CONTAINER ID. Enter the command in the terminal:

```
docker exec -it CONTAINER_ID bash
```

Note: CONTAINER ID is the ID of the latest docker environment just found.

Enter the following command to start rosbridge

```
ros2 launch rosbridge_server rosbridge_websocket_launch.xml
```



**Raspberry Pi 5 master control,** Follow the above steps, reopen a terminal, and enter the docker environment. The docker you enter is newly created, so you don't need to execute the `~run_docker.sh` command. **Orin master control does not need to enter docker**

Enter command:

Launch for navigation

```
ros2 launch yahboomcar_nav navigation_dwa_launch.py
```

The map file directory for navigation startup is

`~/yahboomcar_ros2_ws/yahboomcar_ws/src/yahboomcar_nav/maps/yahboomcar.yaml`

Reopen a terminal and enter Docker, enter the following command:

Start publishing robot position nodes:

```
ros2 launch robot_pose_publisher_ross2 robot_pose_publisher_launch.py
```

```
root@ubuntu:~/yahboomcar_ros2_ws/yahboomcar_ws#
root@ubuntu:~/yahboomcar_ros2_ws/yahboomcar_ws#
root@ubuntu:~/yahboomcar_ros2_ws/yahboomcar_ws#
root@ubuntu:~/yahboomcar_ros2_ws/yahboomcar_ws#
root@ubuntu:~/yahboomcar_ros2_ws/yahboomcar_ws# ros2 launch robot_pose_publisher_ross2 robot_pose_publisher_launch.py
[INFO] [launch]: All log files can be found below /root/.ros/log/2023-06-05-09-56-03-330596-ubuntu-1943
[INFO] [launch]: Default logging verbosity is set to INFO
/root/yahboomcar_ros2_ws/yahboomcar_ws/install/robot_pose_publisher_ross2/share/robot_pose_publisher_ross2/launch/robot_pose_publisher_
' instead
  Node(
  /root/yahboomcar_ros2_ws/yahboomcar_ws/install/robot_pose_publisher_ross2/share/robot_pose_publisher_ross2/launch/robot_pose_publisher_
    Node(
```

**Note: If the point cloud information is not displayed after full startup, wait for the mapping module to start before restarting the robot position node**

Reopen a terminal and enter Docker, enter the following command:

Start the node for publishing laser data transfer points.

```
ros2 run laserscan_to_point_publisher laserscan_to_point_publisher
```

```
root@ubuntu:~/yahboomcar_ros2_ws/yahboomcar_ws#
root@ubuntu:~/yahboomcar_ros2_ws/yahboomcar_ws#
root@ubuntu:~/yahboomcar_ros2_ws/yahboomcar_ws#
root@ubuntu:~/yahboomcar_ros2_ws/yahboomcar_ws# source install/setup.bash
root@ubuntu:~/yahboomcar_ros2_ws/yahboomcar_ws# ros2 run laserscan_to_point_publisher laserscan_to_point_publisher
^CTraceback (most recent call last):
```

命令输入 (按ALT键提示历史,TAB键路径,ESC键返回,双击CTRL切换)

Re-open a terminal and enter docker, enter the following command

Start chassis and radar nodes

```
ros2 launch yahboomcar_nav laser_bringup_launch.py
```

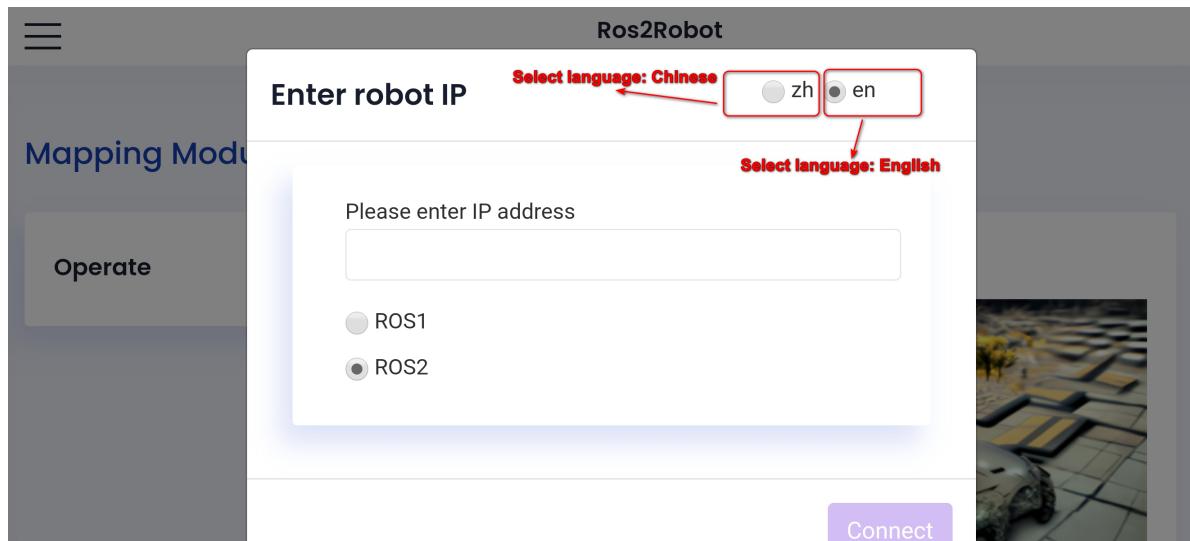
Open another terminal and start the camera

```
ros2 launch usb_cam demo.launch.py #pi5 motherboard
```

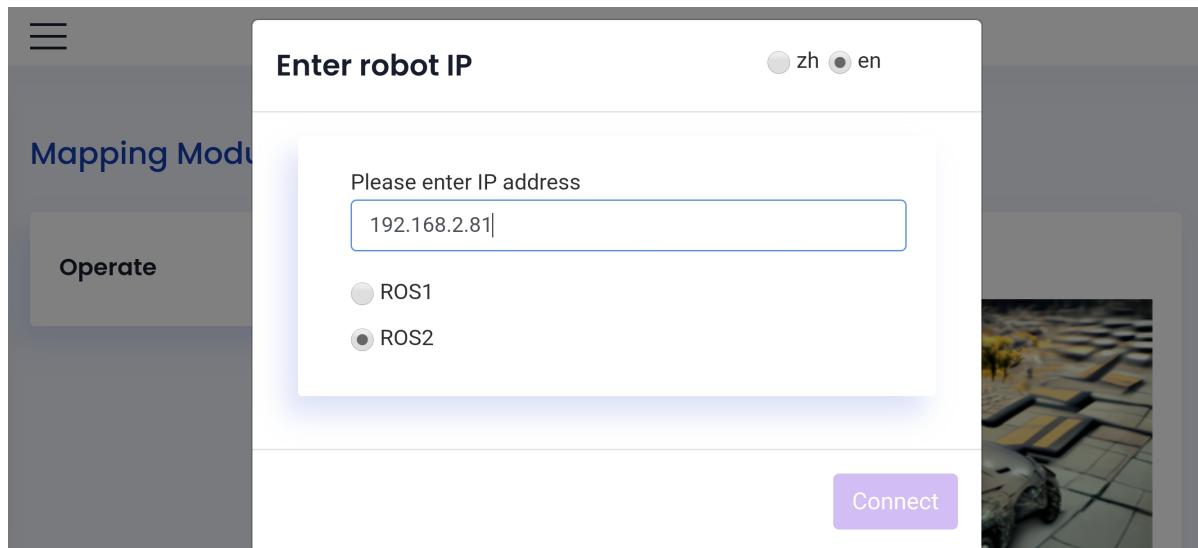
```
ros2 launch usb_cam camera.launch.py #orin motherboard
```

## 4、Open the app and activate navigation mode

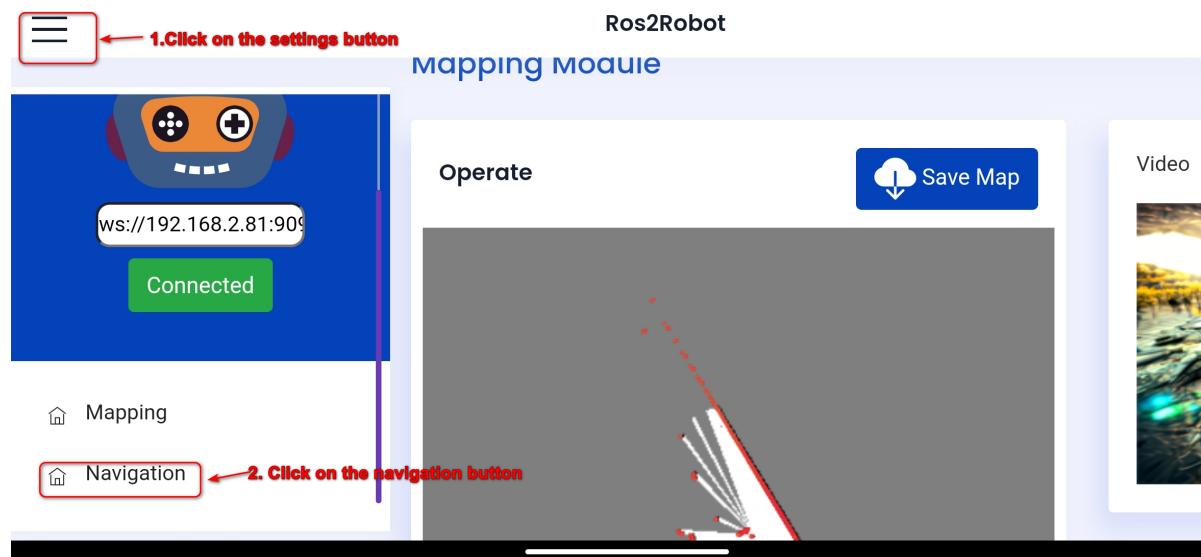
Install the app on your phone and open it. The following figure shows the APP opening interface:



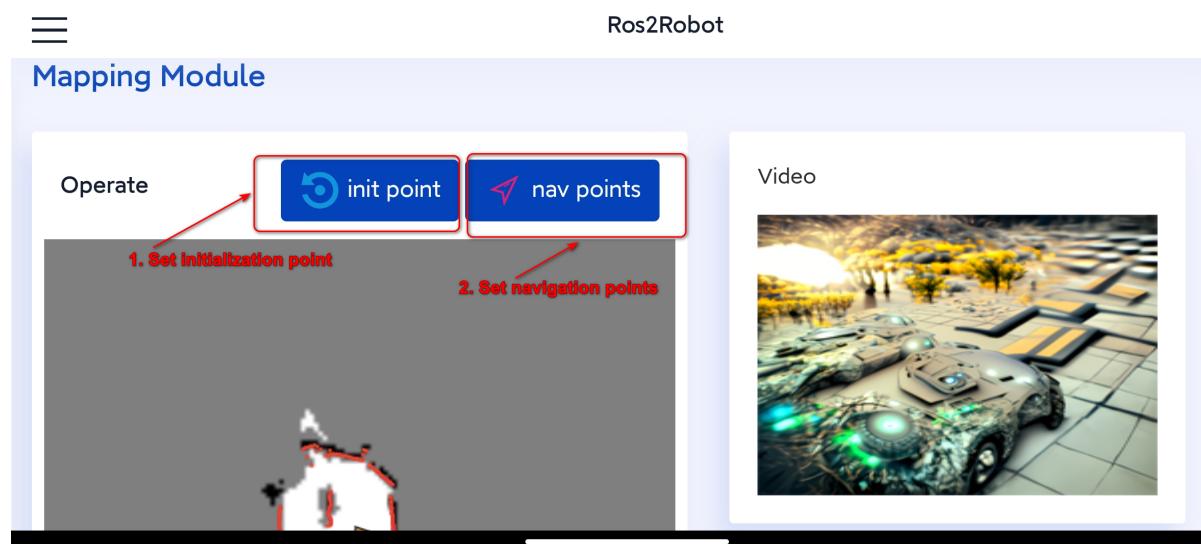
Enter the IP address in the input box. The IP address here is the IP address of the car, such as 192.168.2.81. This address is the actual IP address of the car. Please enter it according to the actual situation during operation. After completing the input, click the Next button.



Click on the settings button (three horizontal lines) in the upper left corner to pop up an option and select navigation mode.



After startup, you can see the current position of the X3 car and the laser point cloud. Set the initialization point before navigation. Click the Set Initialization Point button.



Then click on a point on the map and rotate it in the desired direction to set it as the initialization point.



After the initial point setting is completed, the robot position and point cloud will move to the position of the set point. Then click on the Set Navigation Point button. After waiting for the interface to load, set a target point on the map interface, wait for the car to automatically plan the route, and run to the target point.

