5、Voice controlled multi-point navigation

This course needs to be combined with the hardware of the Rosmaster-X3 car, and only code analysis will be done here. Firstly, let's take a look at the built-in voice commands，

functional word	Speech recognition results	Voice broadcast content
Go to the point A	19	OK, I'm going to the point A.
Go to the point B	20	OK, I'm going to the point B.
Go to the point C	21	OK,I'm going to the point C.
Go to the point D	32	OK, I'm going to the point D.
Back to origianl point	32	OK, I'm return back.

1、Program startup

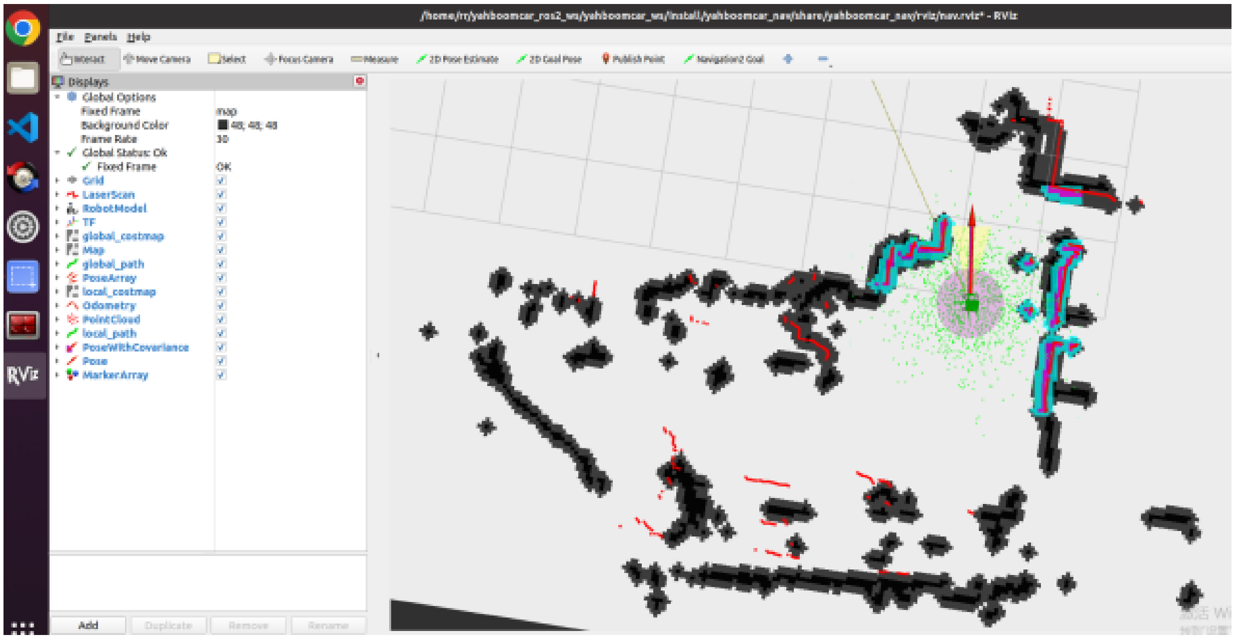
1.1、Calibration target point

Terminal input，

```
ros2 launch yahboomcar_nav laser_bringup_launch.py
ros2 launch yahboomcar_nav display_nav_launch.py
ros2 launch yahboomcar_nav navigation_teb_launch.py
```

Click on [2D Pose Estimate] in the rviz interface of the virtual machine, and then compare the pose of the car to mark the initial pose of the car on the map;；

The display after marking is as follows：


/home/rr/yahboomcar_ros2_ws/yahboomcar_ws/install/yahboomcar_nav/share/yahboomcar_nav/rviz/nav.rviz* - RViz

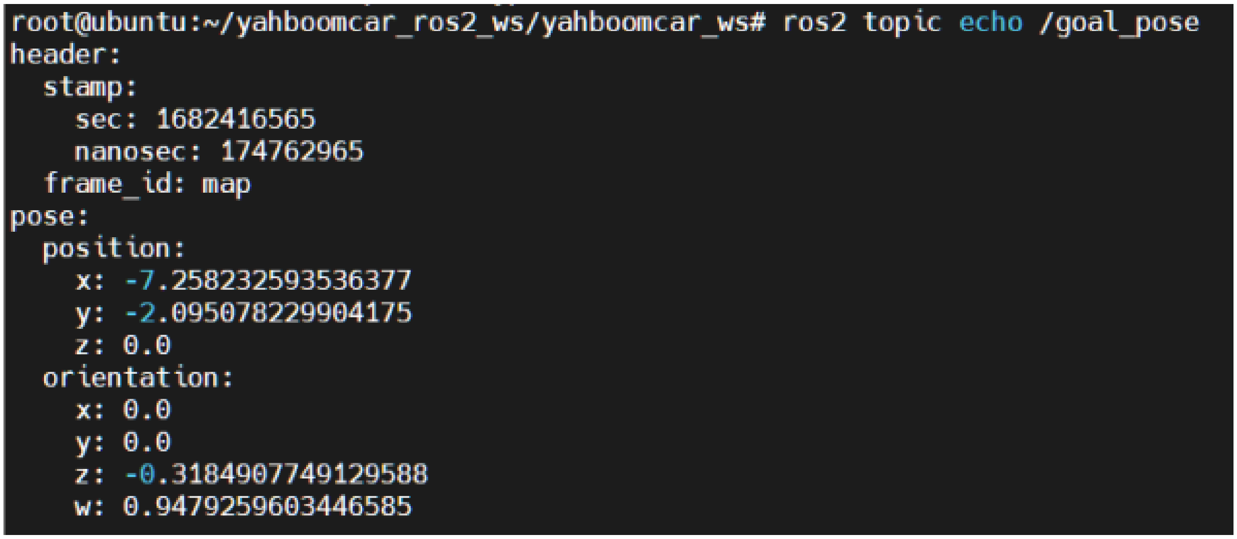
Global Options
Fixed Frame map
Background Color 48; 48; 48
Frame Rate 30
Global Status: Ok
Fixed Frame OK

By comparing the overlap of radar scanning points and obstacles, the initial pose of the car can be set multiple times until the radar scanning points roughly coincide with the obstacles；

Terminal input，

```
ros2 topic echo /goal_pose
```

Click on [2D Goal Pose] to set the first navigation target point. At this point, the car starts navigation and the terminal will print out topic data：



```
root@ubuntu:~/yahboomcar_ros2_ws/yahboomcar_ws# ros2 topic echo /goal_pose
header:
  stamp:
    sec: 1682416565
    nanosec: 174762965
  frame_id: map
pose:
  position:
    x: -7.258232593536377
    y: -2.095078229904175
    z: 0.0
  orientation:
    x: 0.0
    y: 0.0
    z: -0.3184907749129588
    w: 0.9479259603446585
```

1.2、Write the target point position to the program

Edit voice_ Ctrl_ Send_ Mark. py file, the file path is as follows：
~/driver_ws/src/yahboomcar_voice_ctrl/yahboomcar_voice_ctrl/voice_Ctrl_send_mark.py

Modify the pose of the first navigation point to the one printed in the terminal：

```
pose.pose.position.x = 2.15381097794
pose.pose.position.y = -5.02386903763
pose.pose.orientation.z = 0.726492681307
pose.pose.orientation.w = 0.687174202082
```

The marking results of several other points are also calibrated in the rviz according to the above steps, and the coordinates of the pose points are recorded. Then modify to the corresponding position。

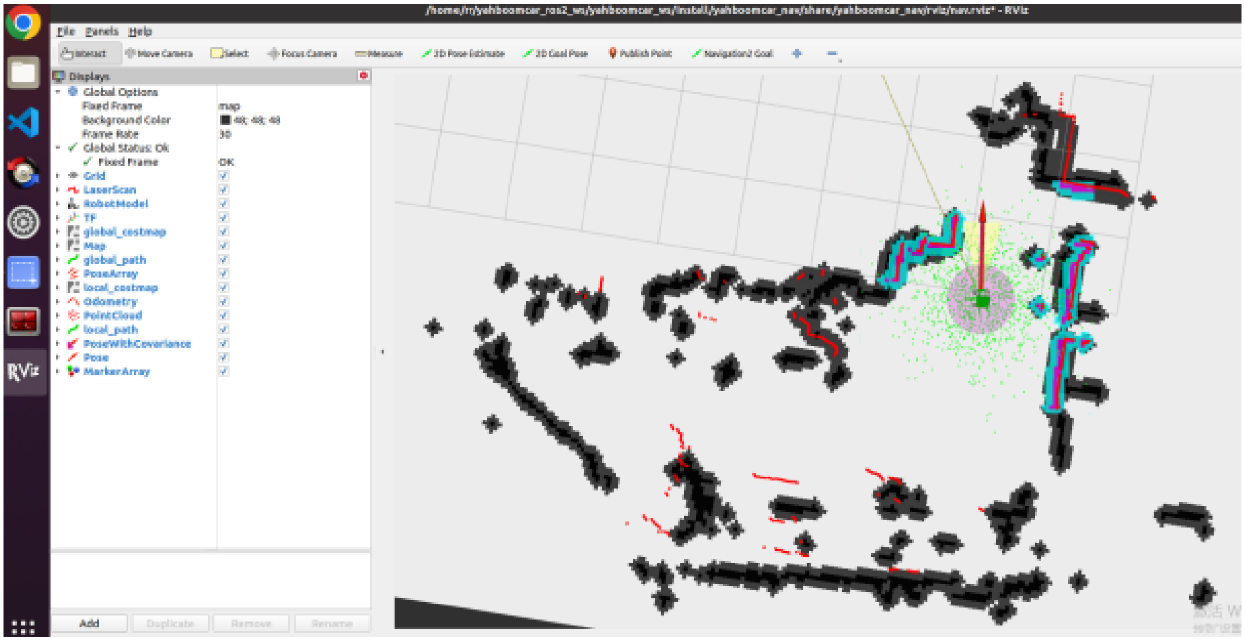
1.3、Voice navigation

Terminal input，

```
ros2 launch yahboomcar_nav laser_bringup_launch.py
ros2 launch yahboomcar_nav display_nav_launch.py
ros2 launch yahboomcar_nav navigation_teb_launch.py
```

At this point, click on [2D Pose Estimate] in the rviz interface of the virtual machine, and then compare the pose of the car to mark the initial pose of the car on the map；

The display after marking is as follows：



/home/m/yahboomcar_ros2_ws/yahboomcar_ws/install/yahboomcar_nav/share/yahboomcar_nav/rviz/nav.rviz* - RViz

Global Options
Fixed Frame map
Background Color 48; 48; 48
Frame Rate 30
Global Status: Ok
Fixed Frame OK

After marking, the display is as follows: Compare the overlap between the radar scanning point and the obstacle. You can set the initial pose of the car multiple times until the radar scanning point roughly coincides with the obstacle；

Terminal input，

```
ros2 run yahboomcar_voice_ctrl voice_Ctrl_send_mark
```

Say "Hi Yahboom" to the voice module on the car, wake up the voice module, hear the feedback from the voice module that "it is", then continue to say "Go to the point A"; the voice module will feedback that "OK, I'm going to the point A.", and the car will start navigating to position one at the same time. Use the same method to navigate to other positions.

2、Code parsing

```
#Import voice library
from Speech_Lib import Speech
#Create a target topic publisher
self.pub_goal = self.create_publisher(PoseStamped, "/goal_pose", 1)
def voice_pub_goal(self):
    self.pose.header.frame_id = 'map'
    #Obtain voice commands
    speech_r = self.spe.speech_read()
    # print("-------speech_r = ",speech_r)
    if speech_r == 19:
        print("goal to one")
        self.spe.void_write(speech_r)
        self.pose.header.stamp = Clock().now().to_msg()
        self.pose.pose.position.x = -7.1171722412109375
        self.pose.pose.position.y = -3.8613715171813965
        self.pose.pose.orientation.z = -0.6484729569092691
        self.pose.pose.orientation.w = 0.7612376922862854
        #Publish target point topic data
        self.pub_goal.publish(self.pose)
    elif speech_r == 20:
        print("goal to two")
        self.spe.void_write(speech_r)
        self.pose.header.stamp = Clock().now().to_msg()
        self.pose.pose.position.x = -5.434411525726318
        self.pose.pose.position.y = -3.575838088989258
        self.pose.pose.orientation.z = 0.041131907433507836
        self.pose.pose.orientation.w = 0.9991537250047569
        self.pub_goal.publish(self.pose)
```