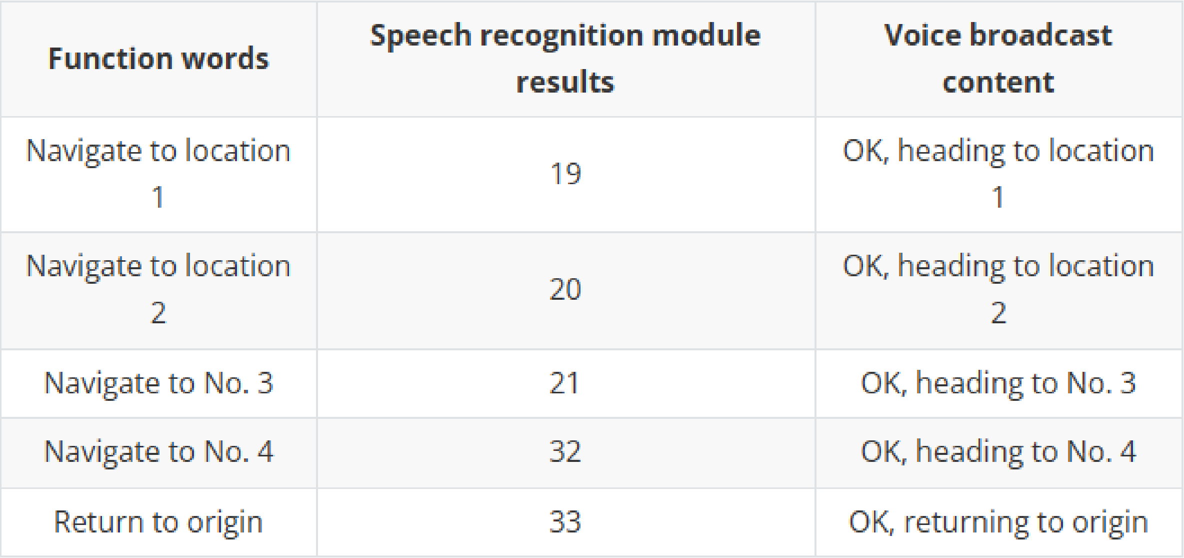
5. Voice-controlled multi-point navigation

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



Function words	Speech recognition module results	Voice broadcast content
Navigate to location 1	19	OK, heading to location 1
Navigate to location 2	20	OK, heading to location 2
Navigate to No. 3	21	OK, heading to No. 3
Navigate to No. 4	32	OK, heading to No. 4
Return to origin	33	OK, returning to origin

1. Program startup

1.1. Calibrate the target point

Terminal input,

```
roslaunch yahboomcar_nav laser_bringup.launch #laser + yahboomcar
roslaunch yahboomcar_nav yahboomcar_navigation.launch use_rviz:=false map:=house
#Open navigation, house is changed to the map name for mapping
```

[use_rviz] parameter: whether to open rviz.

[map] parameter: map name, map to load.

Open the visualization interface (virtual machine side)

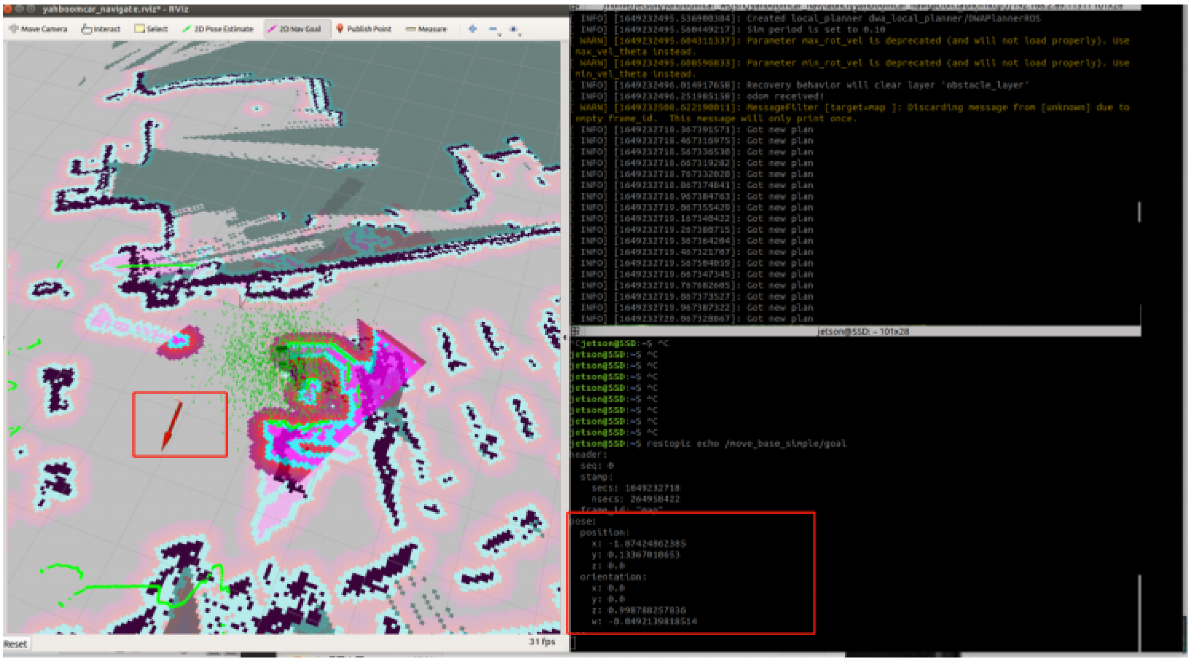
```
roslaunch yahboomcar_nav view_navigate.launch
```

1) In the map rviz, adjust the initial posture of ROSMASTER;

2) Terminal input

```
rostopic echo /move_base_simple/goal
```

3) Use the 2D Nav Goal tool in rviz to give the car a target point in rviz, recorded as position 1. At this time, the coordinates of the target point will be printed in the terminal window where the data of /move_base_simple/goal was just viewed, as shown in the figure below,



```
[ INFO] [1649232495.536900384]: Created local_planner dwa_local_planner/DWAPlannerROS
[ INFO] [1649232495.560449217]: Sim period is set to 0.10
[ WARN] [1649232495.604311337]: Parameter max_rot_vel is deprecated (and will not load properly). Use
max_vel_theta instead.
[ WARN] [1649232495.608596833]: Parameter min_rot_vel is deprecated (and will not load properly). Use
min_vel_theta instead.
[ INFO] [1649232496.014917658]: Recovery behavior will clear layer 'obstacle_layer'
[ INFO] [1649232496.251985158]: odom received!
[ WARN] [1649232500.622190011]: MessageFilter [target=map ]: Discarding message from [unknown] due to
empty frame_id.  This message will only print once.
[ INFO] [1649232718.367391571]: Got new plan
[ INFO] [1649232718.467316975]: Got new plan
[ INFO] [1649232718.567336530]: Got new plan
[ INFO] [1649232718.667319282]: Got new plan
[ INFO] [1649232718.767332020]: Got new plan
[ INFO] [1649232718.867374841]: Got new plan
[ INFO] [1649232718.967384763]: Got new plan
[ INFO] [1649232719.067355429]: Got new plan
[ INFO] [1649232719.167348422]: Got new plan
[ INFO] [1649232719.267388715]: Got new plan
[ INFO] [1649232719.367364284]: Got new plan
[ INFO] [1649232719.467321787]: Got new plan
[ INFO] [1649232719.567504859]: Got new plan
[ INFO] [1649232719.667347345]: Got new plan
[ INFO] [1649232719.767682605]: Got new plan
[ INFO] [1649232719.867373527]: Got new plan
[ INFO] [1649232719.967387322]: Got new plan
[ INFO] [1649232720.067328867]: Got new plan
```

```
^Cjetson@SSD:~$ ^C
jetson@SSD:~$ ^C
jetson@SSD:~$ ^C
jetson@SSD:~$ ^C
jetson@SSD:~$ ^C
jetson@SSD:~$ ^C
jetson@SSD:~$ ^C
jetson@SSD:~$ ^C
jetson@SSD:~$ ^C
jetson@SSD:~$ rostopic echo /move_base_simple/goal
header:
  seq: 0
  stamp:
    secs: 1649232718
    nsecs: 264958422
  frame_id: "map"
pose:
  position:
    x: -1.87424862385
    y: 0.13367010653
    z: 0.0
  orientation:
    x: 0.0
    y: 0.0
    z: 0.998788257836
    w: -0.0492139818514
```

Just record the pose part of the data, and you need to send this part manually through the program later.

1.2. Write the target point position into the program

Open ~/yahboomcar/src/yahboomcar_voice_ctrl/scripts/voice_Ctrl_send_mark.py, and modify the pose data just recorded to the corresponding position.

```
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 the other points are also marked in rviz according to the above steps, and the coordinates of the pose points are recorded. Then modify them to the corresponding positions.

1.3, Voice Navigation

Terminal input,

```
roslaunch yahboomcar_nav laser_bringup.launch #laser + yahboomcar
roslaunch yahboomcar_nav yahboomcar_navigation.launch use_rviz:=false map:=house
#Open navigation, house is changed to the map name for mapping
python ~/yahboomcar_ws/src/yahboomcar_voice_ctrl/scripts/voice_Ctrl_send_mark.py
```

After calibrating the initial position in rviz, say "Hi Yahboom" to ROSMASTER to wake up the voice module. After hearing the voice module feedback "Here", say "Navigate to location 1" to ROSMASTER; the voice module will feedback "OK, going to location 1". The same is true for other locations, as long as the coordinate values of the destinations of other points are written in the program.

2. Code analysis

Code path: ~/driver_ws/src/yahboomcar_voice_ctrl/scripts/voice_Ctrl_send_mark.py

```
#创建目标点发布者发布目标位姿 Create a target point publisher to publish the target pose
pub_goal = rospy.Publisher('move_base_simple/goal', PoseStamped, queue_size=1)
#打包目标点数据 Packing target point data
speech_r = spe.speech_read()
if speech_r == 19 :
    print("goal to one")
    spe.void_write(speech_r)
    pose.pose.position.x = 2.15381097794
    pose.pose.position.y = -5.02386903763
    pose.pose.orientation.z = 0.726492681307
    pose.pose.orientation.w = 0.687174202082
    #发布坐标点位姿 Publish coordinate point pose
    pub_goal.publish(pose)
```