7. Voice control multi-point navigation

7.1. Function description

By interacting with the voice recognition module on ROSMASTER, it can realize the function of navigating to No. 1 / No. 2 / No. 3 position by voice on the established map, and the R2 key on the handle can cancel/enable this function at any time. .

7.2. start

7.2.1. function package path

```
~/yahboomcar/src/yahboomcar_voice_ctrl/
```

7.2.2. the target point to calibrate the voice navigation

robot side

```
roslaunch yahboomcar_nav laser_bringup.launch  #laser + yahboomcar
roslaunch yahboomcar_nav yahboomcar_navigation.launch use_rviz:=false map:=house
# start navigation, change house to the map name of the map
```

[use_rviz] parameter: whether to open rviz.

[map] Parameters: map name, the map to be loaded.

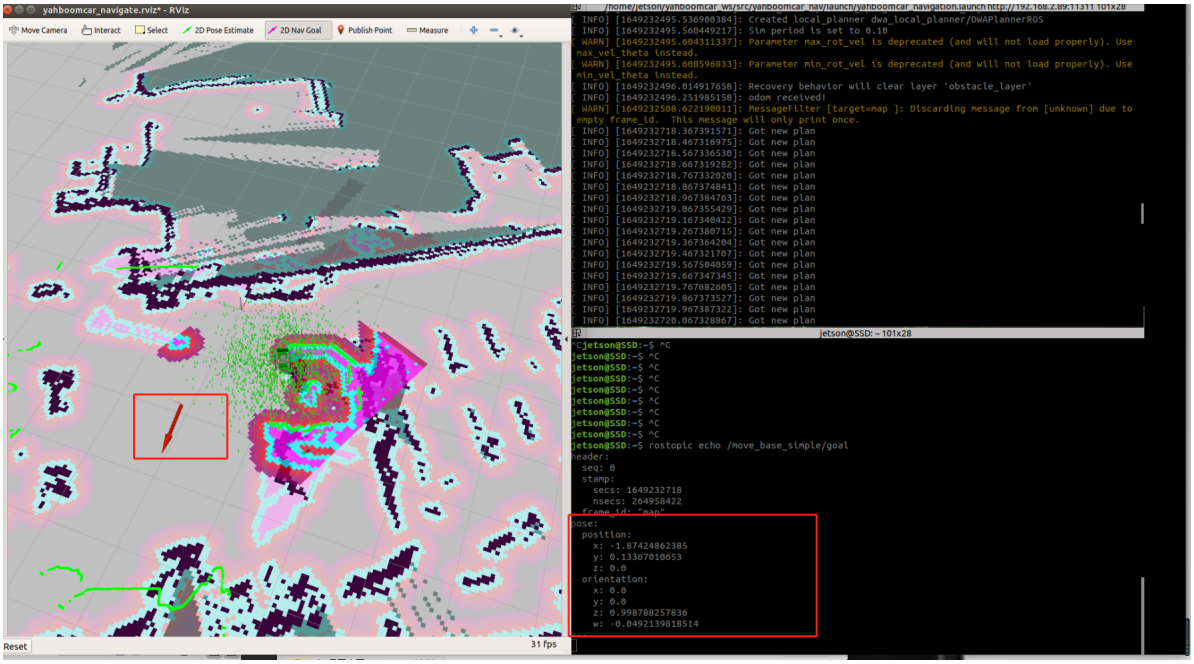
Open the visual interface (virtual machine side)

```
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. Using the 2D Nav Goal tool in rviz, a target point of the car is given in rviz, which is recorded as position 1. At this time, the coordinates of the target point will be printed out in the terminal window that just viewed the data of/move_base_simple/goal,As shown below,



It is enough to record the pose part of the data, and then you need to manually send this part of the content through the program.

4. open ~/yahboomcar/src/yahboomcar_voice_ctrl/scripts/voice_Ctrl_send_mark.py, and modify the pose data just recorded to the corresponding location,

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

For the other two points BC, the position is also calibrated first, and then the calibrated value is modified as an actual parameter into the function.

7.2.2. Voice Navigation

```
roslaunch yahboomcar_nav laser_bringup.launch    #laser + yahboomcar
roslaunch yahboomcar_nav yahboomcar_navigation.launch use_rviz:=false map:=house
 # start navigation, change house to the map name of the map
python ~/yahboomcar_ws/src/yahboomcar_voice_ctrl/scripts/voice_Ctrl_send_mark.py
```

After calibrating the initial pose in rviz, say "Hi Yahboom" to ROSMASTER to wake up the voice module, and after hearing the feedback broadcast of the voice module "Hi , I'm here.", say to ROSMASTER "Go to the point A"; the voice module It will report back "OK, I'm going to the point A.". The same is true for other positions, as long as the coordinates of the destinations of other points are written in the program.

7.3. core code analysis

7.3.1. For the principle of navigation, please refer to the tutorial "11. Lidar/12. Navigation and Obstacle Avoidance" , here is mainly to judge the results of speech recognition and package and send

target point data,

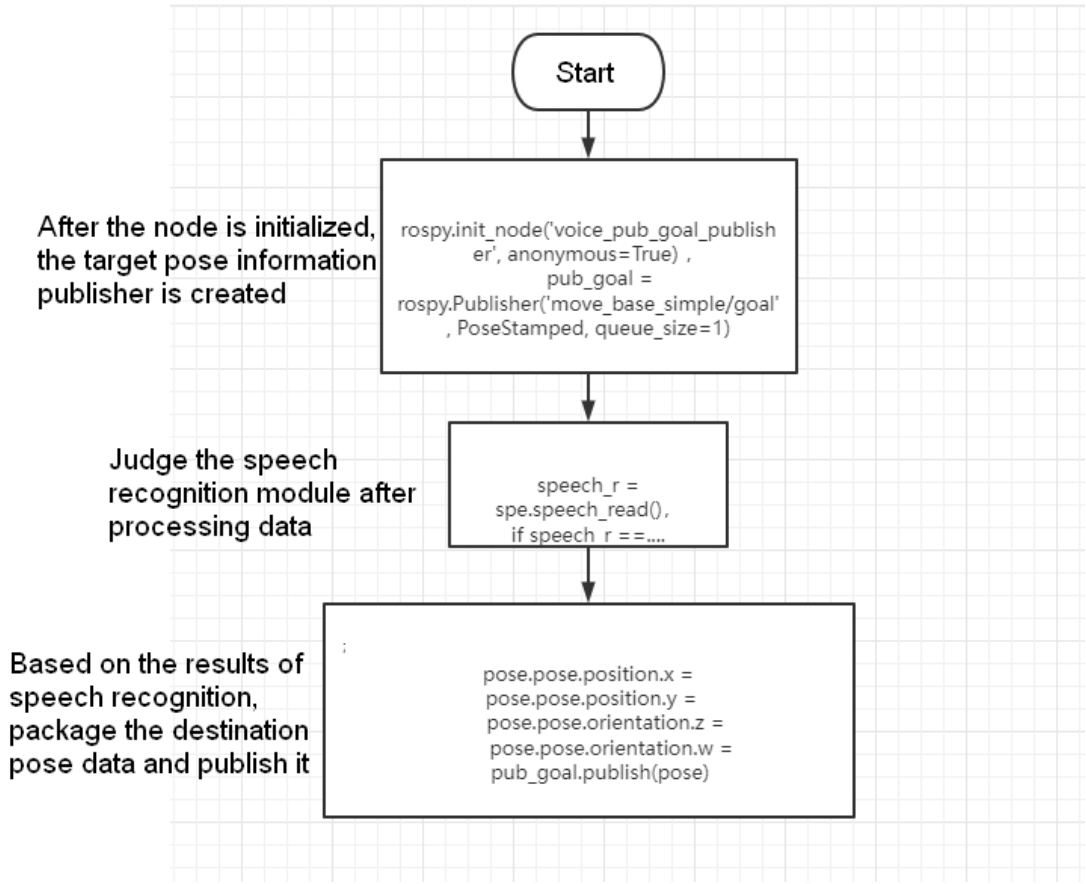
```
        speech_r = spe.speech_read()
        if speech_r == 19 :
            print("goal to one")
            spe.void_write(speech_r)
            pose.pose.position.x = 2.15381097794
            pose.pose.position.y =  -5.02386903763
            pose.pose.orientation.z = 0.726492681307
            pose.pose.orientation.w = 0.687174202082
            pub_goal.publish(pose)
        elif speech_r == 20 :
            print("goal to tow")
            spe.void_write(speech_r)
            pose.pose.position.x = 1.57744419575
            pose.pose.position.y = 4.8174996376
            pose.pose.orientation.z = -0.683335654604
            pose.pose.orientation.w = 0.730104364558
            pub_goal.publish(pose)

        elif speech_r == 21 :
            print("goal to three")
            spe.void_write(speech_r)
            pose.pose.position.x =  -1.08106160164
            pose.pose.position.y = 1.30198049545
            pose.pose.orientation.z =  -0.0132771070267
            pose.pose.orientation.w =  0.99991185533
            pub_goal.publish(pose)

        elif speech_r == 32 :
            print("goal to four")
            spe.void_write(speech_r)
            pose.pose.position.x =  -1.08106160164
            pose.pose.position.y = 1.30198049545
            pose.pose.orientation.z =  -0.0132771070267
            pose.pose.orientation.w =  0.99991185533
            pub_goal.publish(pose)

        elif speech_r == 33 :
            print("goal to Origin")
            spe.void_write(speech_r)
            pose.pose.position.x =  -1.08106160164
            pose.pose.position.y = 1.30198049545
            pose.pose.orientation.z =  -0.0132771070267
            pose.pose.orientation.w =  0.99991185533
            pub_goal.publish(pose)
        elif speech_r == 0 :
            pub_cmdVel.publish(Twist())S
```

7.3.2. Program flow chart



The complete code can refer to:

```
~/yahboomcar/src/yahboomcar_voice_ctrl/scripts/voice_Ctrl_send_mark.py
```

7.3.3. Function module communication table

function word	Speech Recognition Module 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.
Return to the original place	33	OK, I'm return back.