Neor_mini Ackerman Mobile Base

Contents:

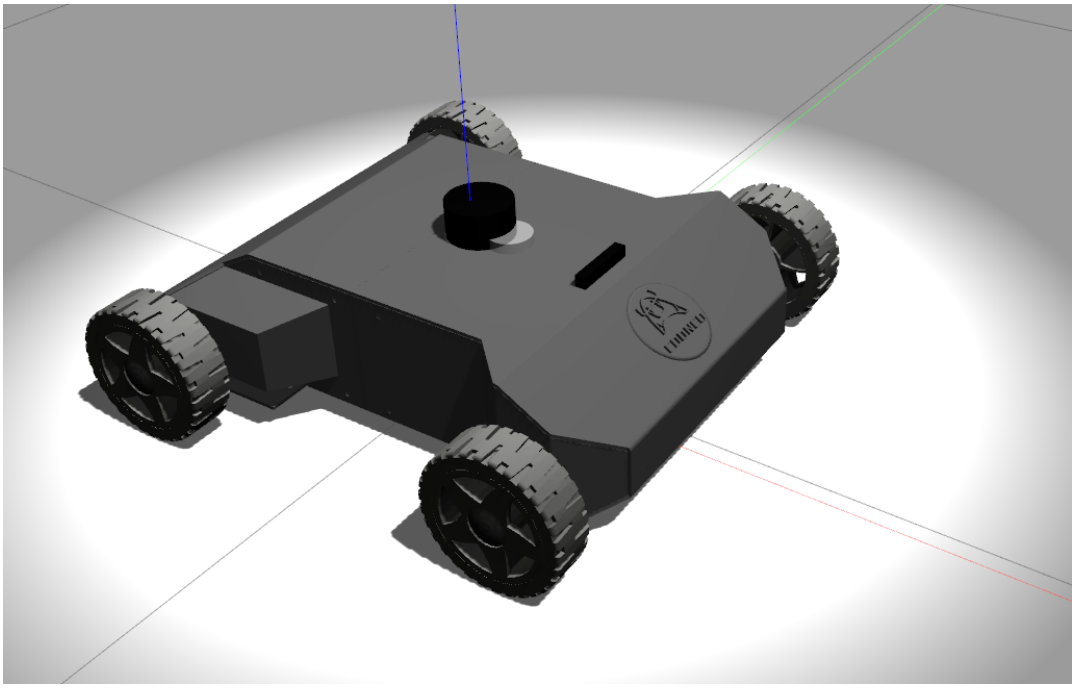
Chapter 1: Make a indoor autonomous mobile robot

Corresponding WeChat article

从零开始搭建一台ROS开源迷你无人车

原创 Lee COONEO 3月12日

收录于话题
#人工智能 5 #机器人 8



Developing Environments:

```
ubuntu 18.04 + ROS Melodic desktop full
```

Explaination:

```
mini_sim18_ws              #   this folder is ROS Workspace, you can run launchs and look at every demo.
original_neor_mini         #   this folder is an original neor_mini urdf file, you can construction by yourself
pictures                   #   the total process pictures
```

Neor mini Simulation in Gazebo with ROS, Follow below steps:

Step 1:

```
# open your Terminal
git clone https://github.com/COONEO/neor_mini.git
cd neor_mini/mini_sim18_ws
rosdep install --from-paths src --ignore-src -r -y    # you need wait a moment

# first see the below tips .
catkin_make                # if failed,please retry again and again
```

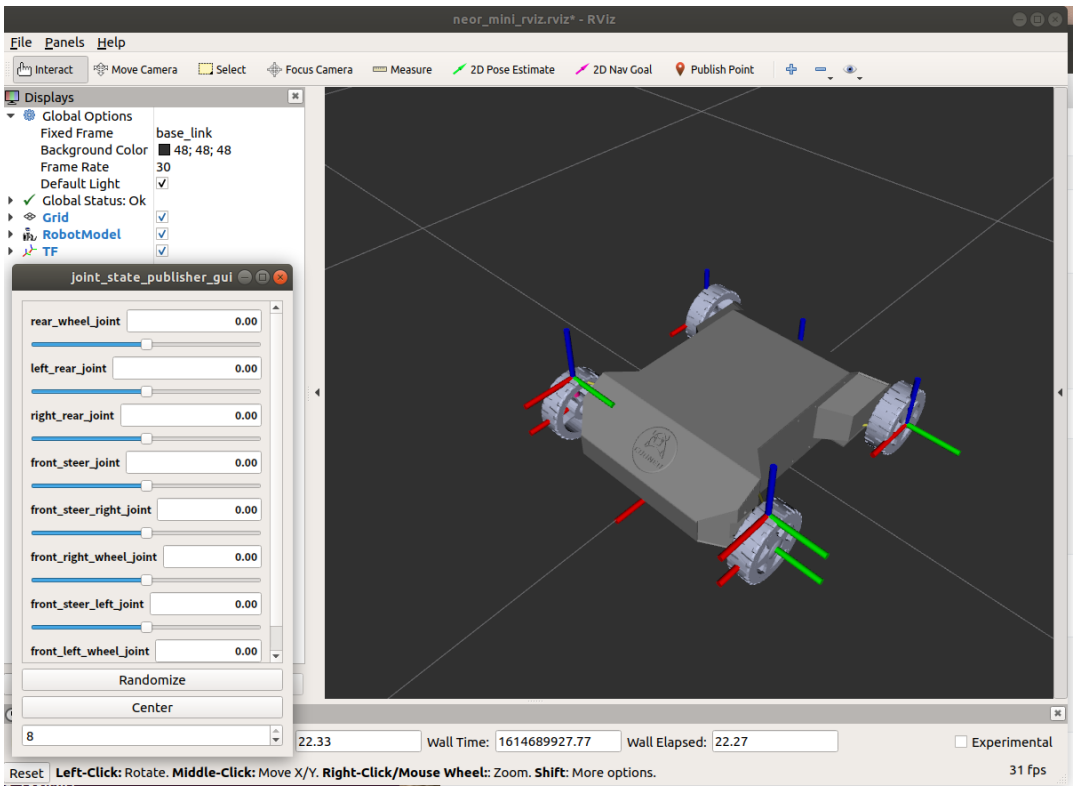
if "catkin_make " error,please install the dependencey of lego_loam(See chapter 3) and lio_sam(See chapter 8). or remove LeGO-LOAM && lio_sam folder.

You can see 5 ROS packages in mini_sim18_ws/src folder,lists:

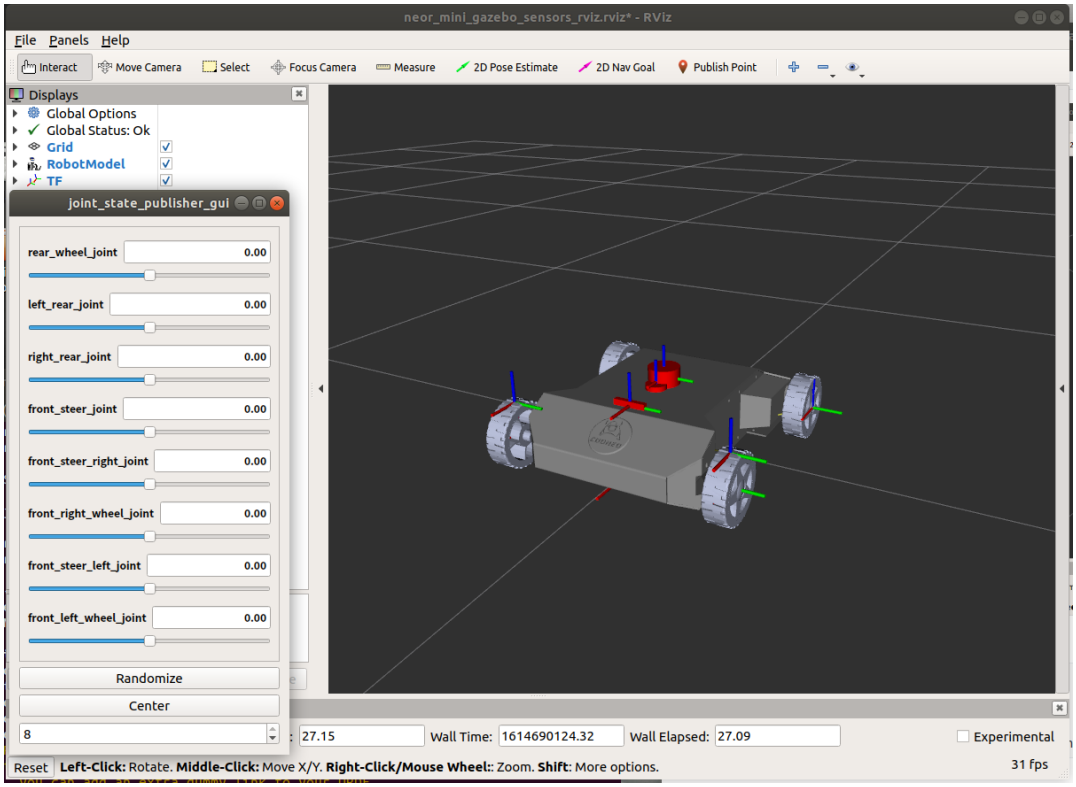
```
neor_mini              # Storing the description of neor mini's appearance with urdf file
steer_drive_ros        # Ackermann kinematics ROS plugins
steer_mini_gazebo      # Storing the launch files of neor mini model visual in Gazebo
mini_gmapping          # Storing the launch files and gmapping params files
mini_navigation        # Storing the launch file and navigation params files
```

Step 2: launch neor_mini's launch file, visualize the urdf in Rviz.

```
# show the neor_mini.urdf in Rviz
cd ~/neor_mini/mini_sim18_ws
source devel/setup.bash
roslaunch neor_mini display.launch
```

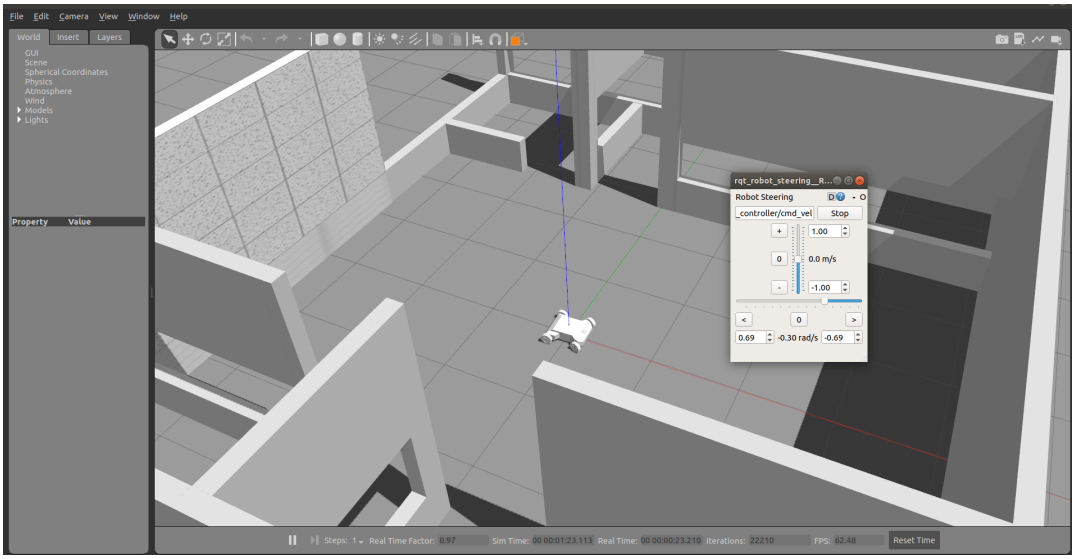



```
#show the neor_mini_gazebo_sensors.urdf in Rviz
cd ~/neor_mini/mini_sim18_ws
source devel/setup.bash
roslaunch neor_mini display_gazebo_sensors.launch
```

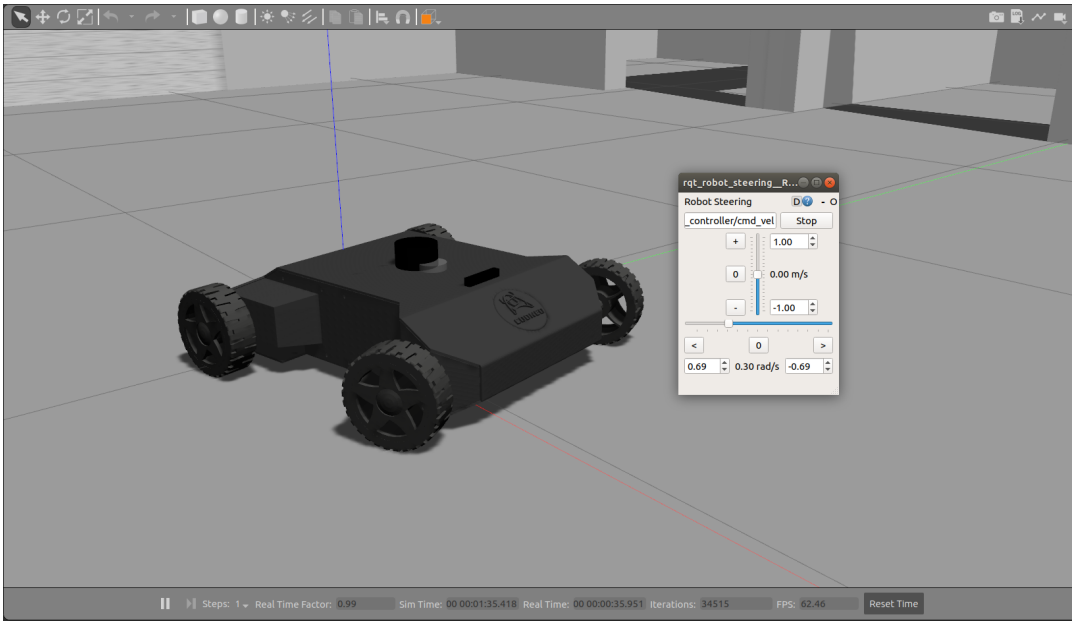


Step 3: launch steer_mini_gazebo's launch file. visualize the urdf in Gazebo and try to control neor_mini .

```
#show the neor_mini_gazebo.urdf in Gazebo
cd ~/neor_mini/mini_sim18_ws
source devel/setup.bash
roslaunch steer_mini_gazebo steer_mini_sim.launch
```

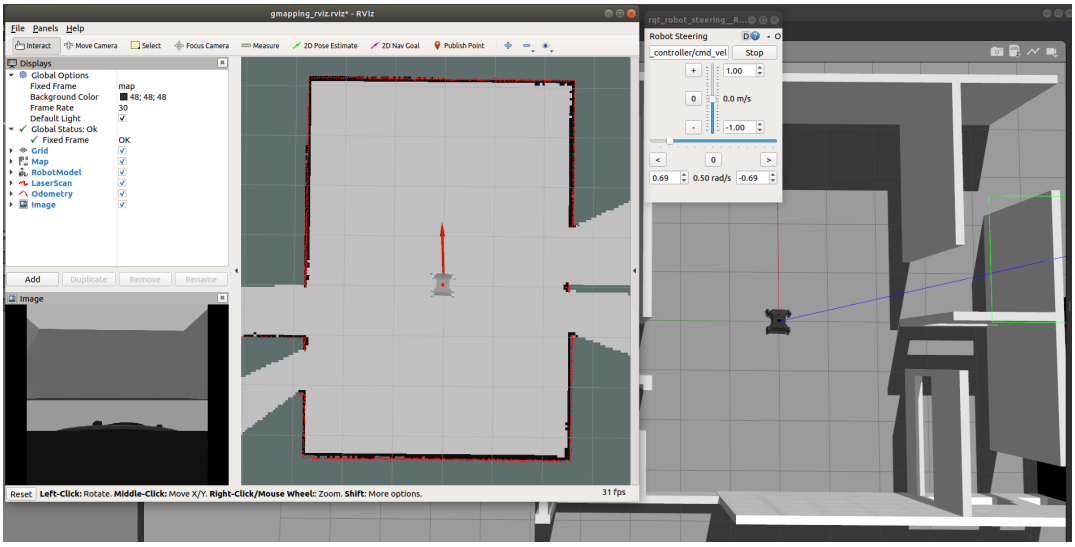



```
#show the neor_mini_gazebo_sensors.urdf in Gazebo
cd ~/neor_mini/mini_sim18_ws
source devel/setup.bash
roslaunch steer_mini_gazebo steer_mini_sim_sensors.launch
```



Step 4 : Gmapping with neor_mini urdf

```
# launch gmapping_steer_mini_sim.launch file and construction map
cd ~/neor_mini/mini_sim18_ws
source devel/setup.bash
roslaunch mini_gmapping gmapping_steer_mini_sensors.launch
```

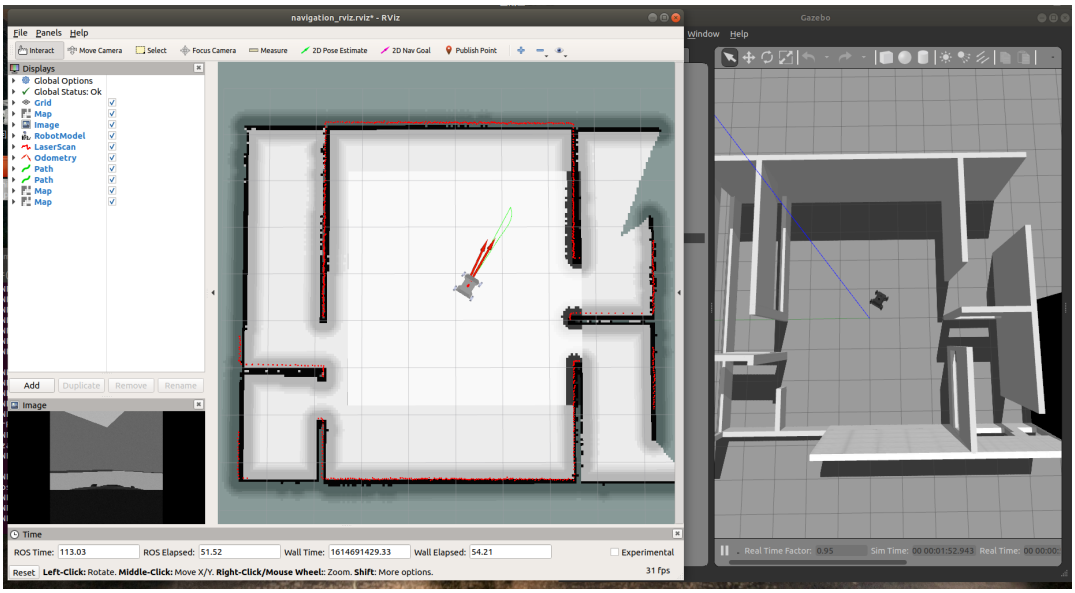


When you think the construction map is finished, Open a new terminal, you can run the below command to save the map.

```
cd ~/neor_mini/mini_sim18_ws/
source devel/setup.bash
cd src/mini_gmapping/map
rosrun map_server map_saver -f cooneo_office_map    # You can saved as another name
```

Step 5 : Using cooneo_office_map to make a navigation demo.

```
cd ~/neor_mini/mini_sim18_ws/
source devel/setup.bash
roslaunch mini_navigation navigation_steer_mini_sensors.launch                # start a navigation demo
```



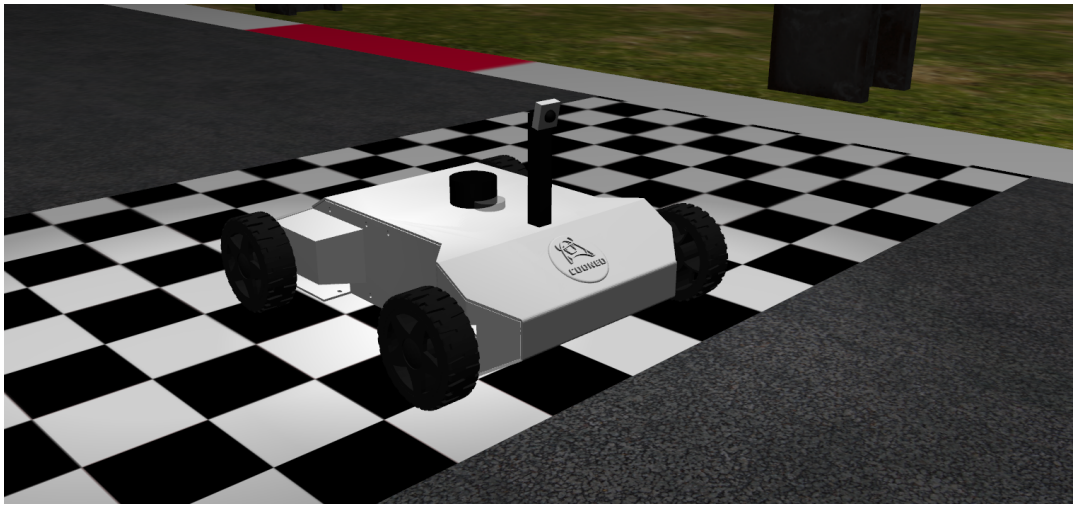
When you run up the list launch file, your monitor will show two windows, one is Rviz, another is Gazebo. looking at the rviz window up toolbar, you need to click "2D Nav Goal", and select a navigation goal on the map, soon the neor_mini model car will plan a route, and arrived.

Chapter 2: Make a outdoor line-tracking mobile robot

Corresponding WeChat article

NEOR mini在赛道上独自狂奔

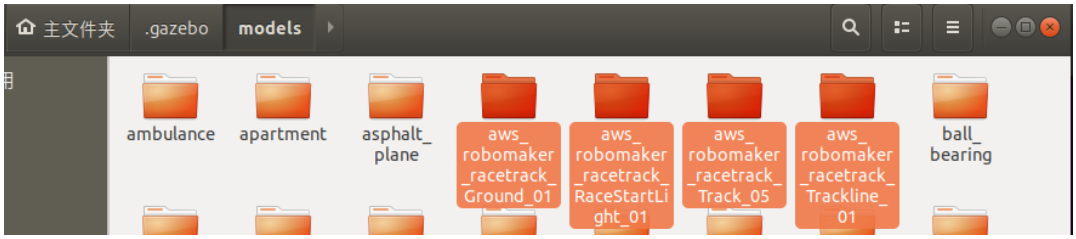
原创 Lee COONEO 4月30日



step 1 : Copy Gazebo-world from neor_mini_linetrack/models folder

```
# open a Terminal
cp -r ~/neor_mini/mini_sim18_ws/src/neor_mini_linetrack/models/* ~/.gazebo/models/
```

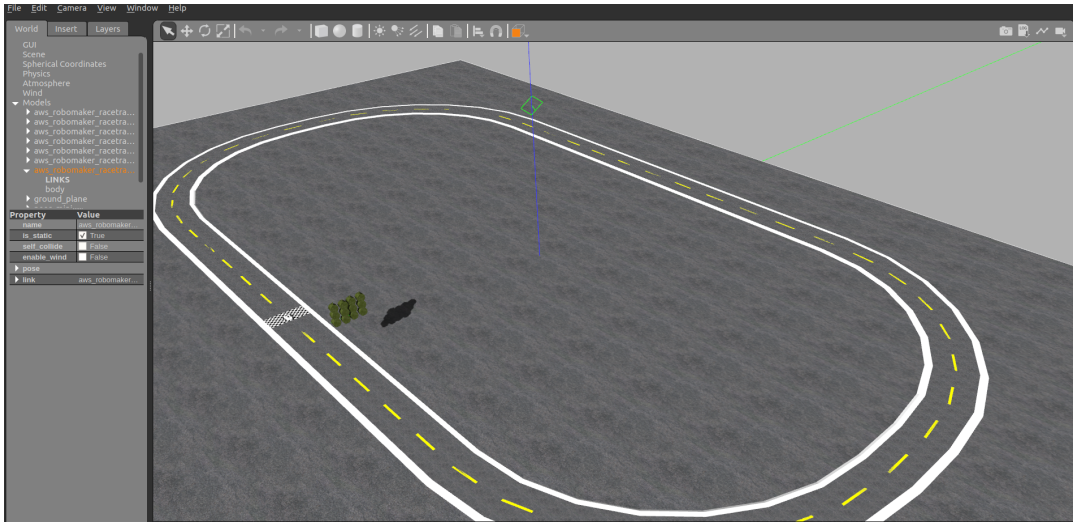
and you will see four new folder in your ./gazebo/models folder as follows:



Step 2 : launch neor_mini_linetrack package launch file

```
# open a Terminal
cd ~/neor_mini/mini_sim18_ws/
catkin_make                                  # compile you all ros node in you workspace
source devel/setup.bash
roslaunch neor_mini_linetrack neor_mini_linetrack_day.launch
```

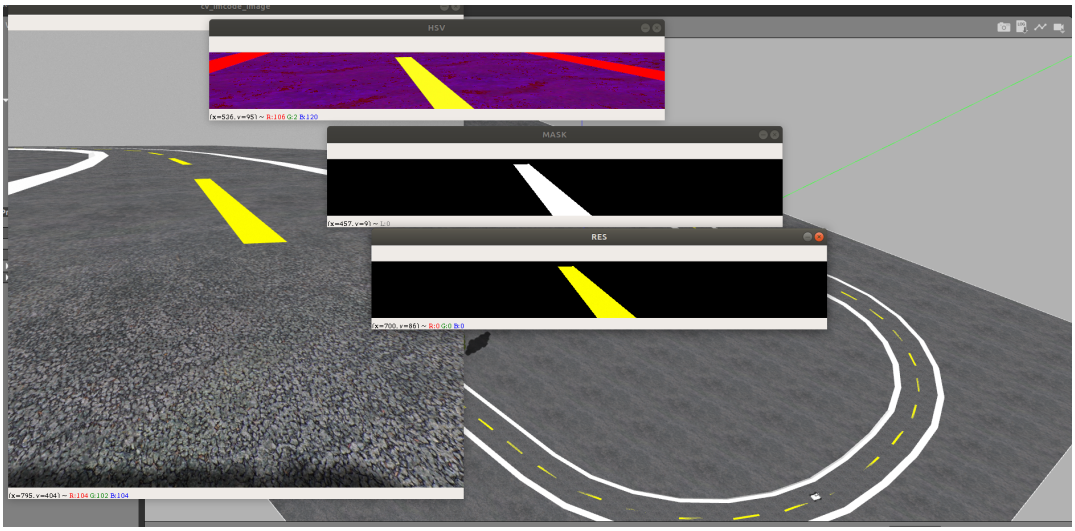
and you will see:



Step 3: rosrun line-tracking python node

```
# open a Terminal
cd ~/neor_mini/mini_sim18_ws/
source devel/setup.bash

# may need to add permission to .py file
sudo chmod +x src/neor_mini_linetrack/scripts/line_follower_object.py     # same as hsv_color_select.py file
rosrun neor_mini_linetrack line_follower_object.py
```

and you will see four new windows ,it's like this:



Congratulations!!! you will see the neor_mini urdf model running automatically using the data from the camera.

Ps. select your own hsv color

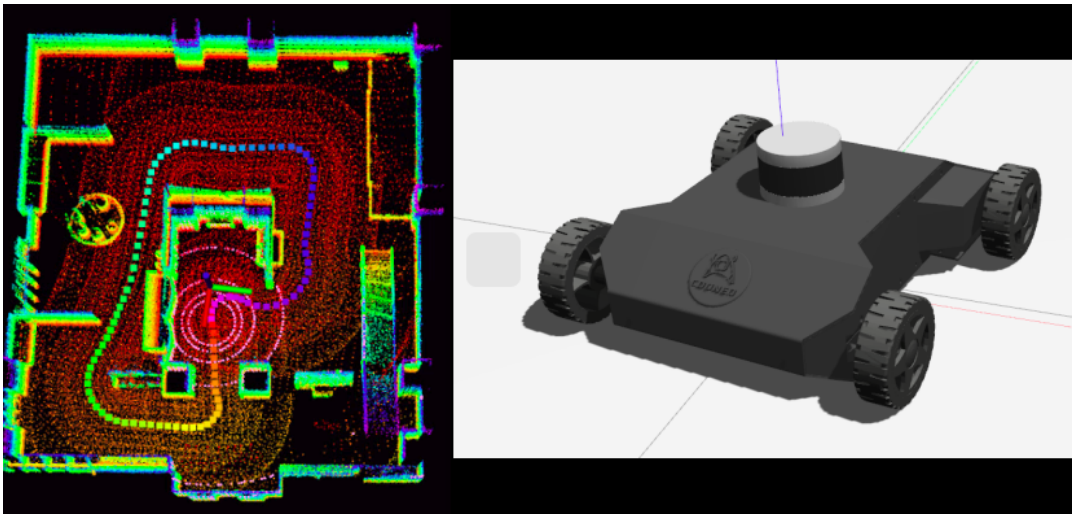
When your running result is different from the above, you can select the hsv color range suitable for you.

Chapter 3: Velodyne-16 && Lego_LOAM 3D Mapping

Corresponding WeChat article

手把手教你Velodyne Lidar仿真+3D建图

原创 Lee COONEO 7月30日



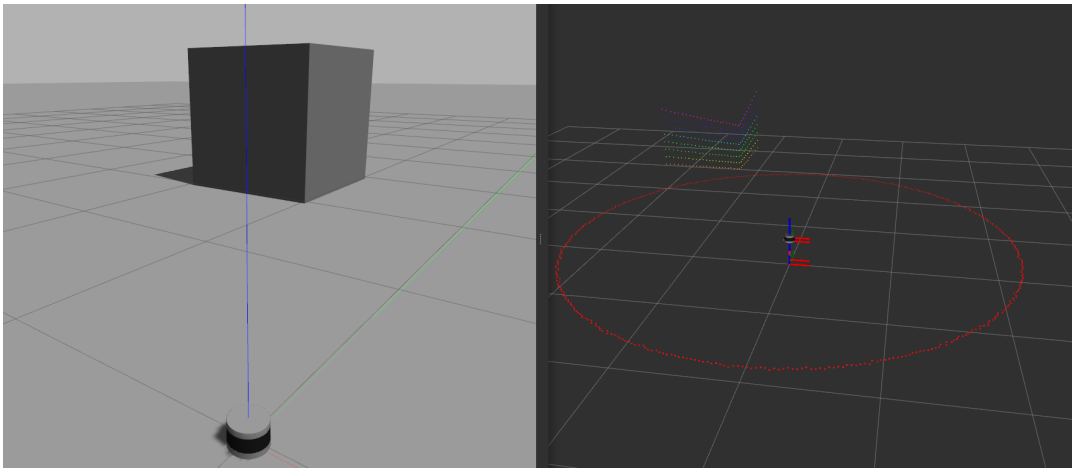
Lego_loam url: https://github.com/RobustFieldAutonomyLab/LeGO-LOAM

install gtsam(Georgia Tech Smoothing and Mapping library, 4.0.0-alpha2)

```
#open a Terminal.
wget -O ~/Downloads/gtsam.zip https://github.com/borglab/gtsam/archive/4.0.0-alpha2.zip
cd ~/Downloads/ && unzip gtsam.zip -d ~/Downloads/
cd ~/Downloads/gtsam-4.0.0-alpha2/
mkdir build && cd build
cmake ..
sudo make install

# copy the libmetis.so library into /usr/lib/ folder.  !!!!!!!
sudo cp /usr/local/lib/libmetis.so /usr/lib
```

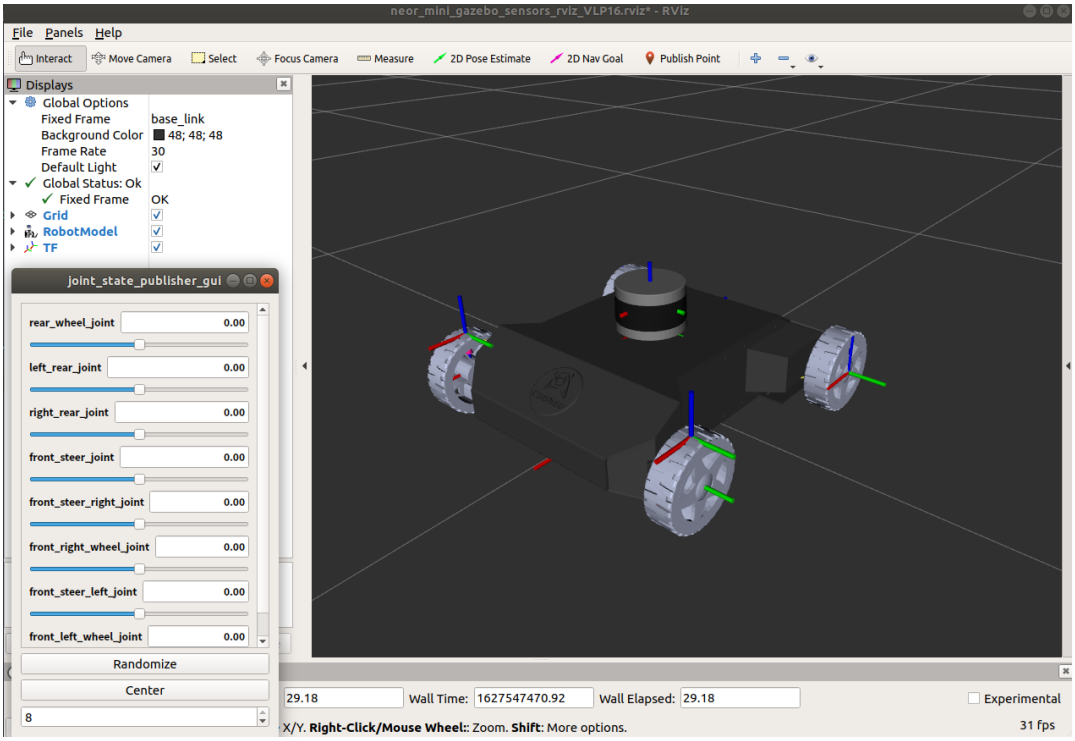
Step 1: launch velodyne_simulator ROS package,and you will see:

```
# open a Terminal
cd ~/neor_mini/mini_sim18_ws/
source devel/setup.bash

# launch VLP-16
roslaunch velodyne_description velodyne_16.launch
```



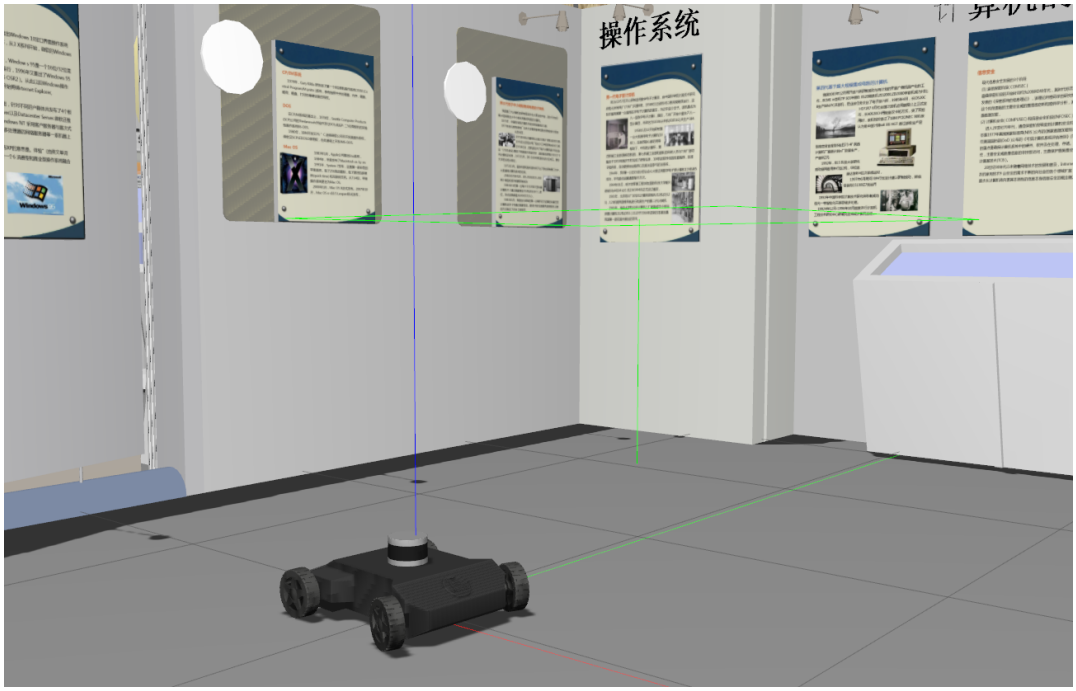
Step 2: Install the Velodyne-16 on the neor_mini

```
# open a Terminal
cd ~/neor_mini/mini_sim18_ws/
source devel/setup.bash

# start launch file and see neor_mini + vlp-16 in Rviz
roslaunch neor_mini display_gazebo_sensors_VLP16.launch
```



Step 3: launch the neor_mini+vlp16 model in Gazebo

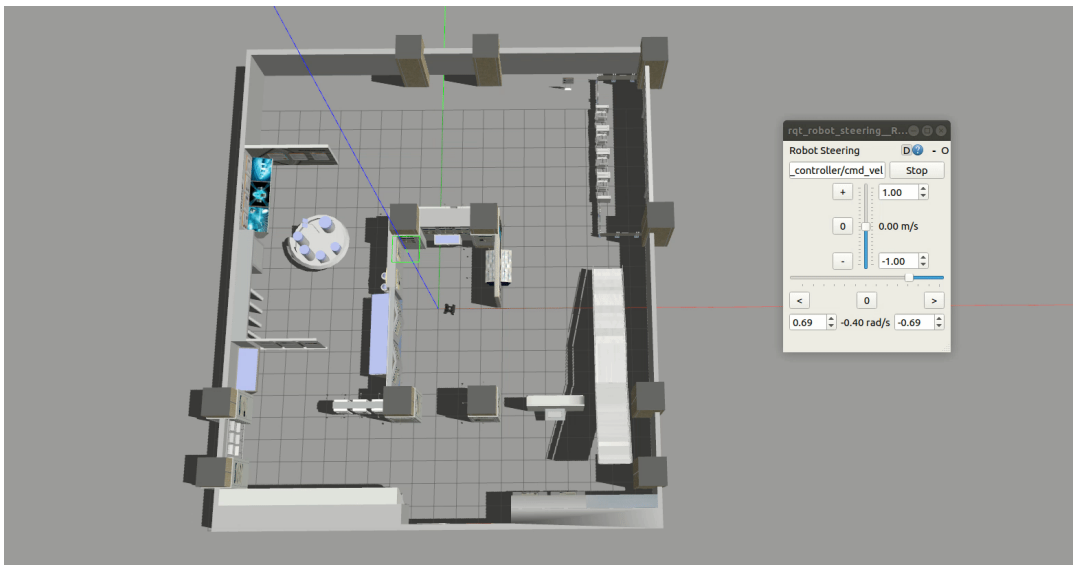
```
# open a Terminal and add the museum model world into Gazebo's model folder
cp -r ~/neor_mini/mini_sim18_ws/src/neor_mini/models/* ~/.gazebo/models/

# open a Terminal
cd ~/neor_mini/mini_sim18_ws/
source devel/setup.bash

# start launch file and see neor_mini + vlp-16 in Gazebo
roslaunch steer_mini_gazebo steer_mini_sim_sensors_VLP16.launch
```



(Gazebo from: iscas_museum.world)

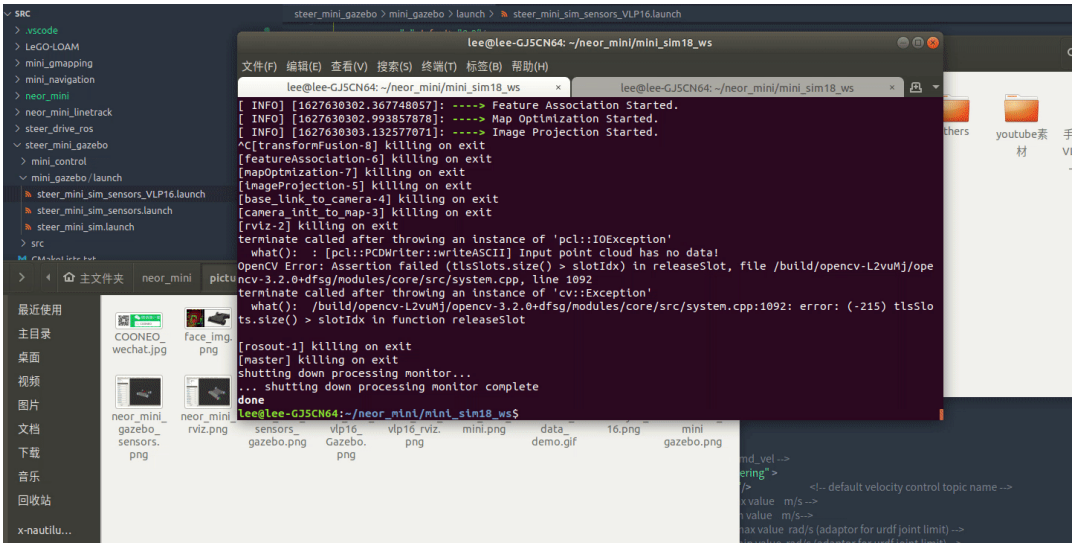
Step 4: Record the data collected by VLP-16 in Gazebo

```
# open a Terminal
cd ~/neor_mini/mini_sim18_ws/
source devel/setup.bash
roslaunch steer_mini_gazebo steer_mini_sim_sensors_VLP16.launch

# open another Terminal and record VLP-16 data By rosbag
cd ~/neor_mini/mini_sim18_ws/
rosbag record -a

# and then, control the neor_mini traverses the iscas_museum world
```



Step 5: Play the Record data && Mapping by Lego_Loam

```
# open a Terminal
cd ~/neor_mini/mini_sim18_ws/
source devel/setup.bash
 roslaunch lego_loam run.launch

 # open another Terminal and "cd" in your bag file folder
 cd ~/neor_mini/mini_sim18_ws/
source devel/setup.bash
rosbag play  ####.bag       # replace with your own or current bag file's name

# and then,you will see mapping by Leog_loam
```



Congratulations!!! You can run Velodyne-16 Lidar in Gazebo with neor_mini and Mapping by Lego_Loam.

The Lego_loam ROS Package from The RobustFieldAutonomyLab.

```
@inproceedings{legoloam2018,
 title={LeGO-LOAM: Lightweight and Ground-Optimized Lidar Odometry and Mapping on Variable Terrain},
 author={Shan, Tixiao and Englot, Brendan},
 booktitle={IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)},
 pages={4758-4765},
 year={2018},
 organization={IEEE}
}
```

Chapter 4: Simulation of Automatic Obstacle Avoidance Based on Three-channel Ultrasonic

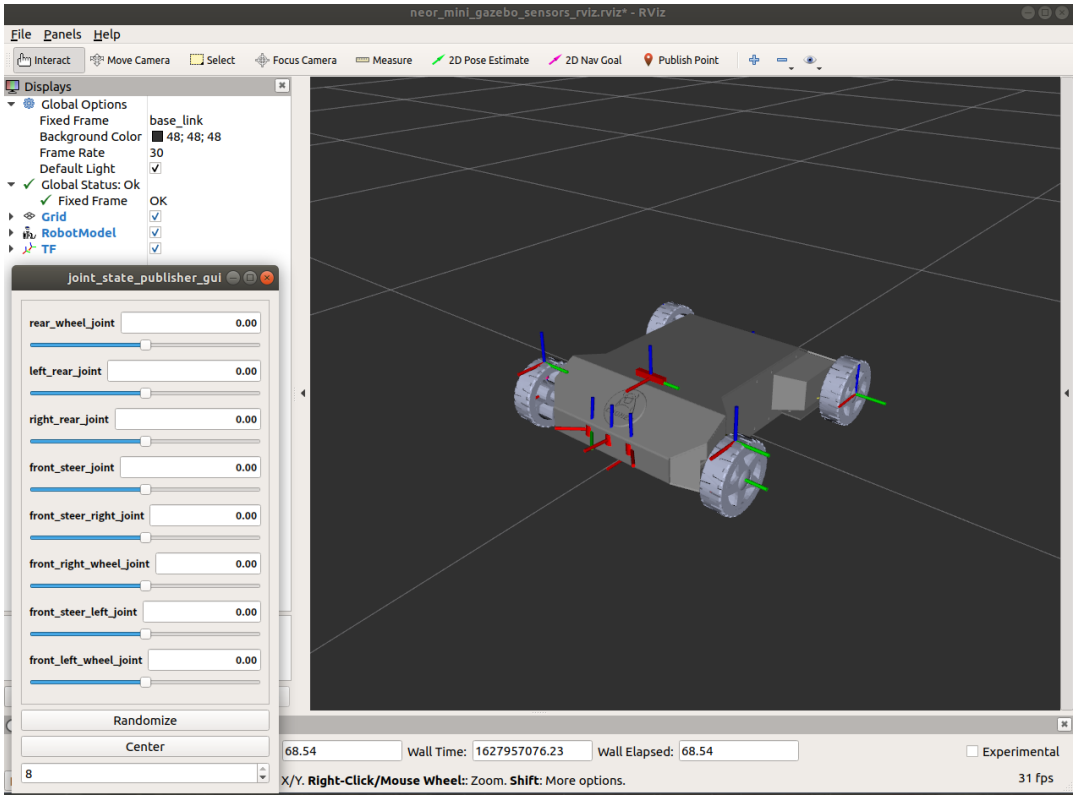
Corresponding WeChat article

开源！手把手教你多路超声波传感器避障仿真

原创 Lee COONEO 8月6日

step 1: Visualization of neor_mini after adding an ultrasonic sensor.

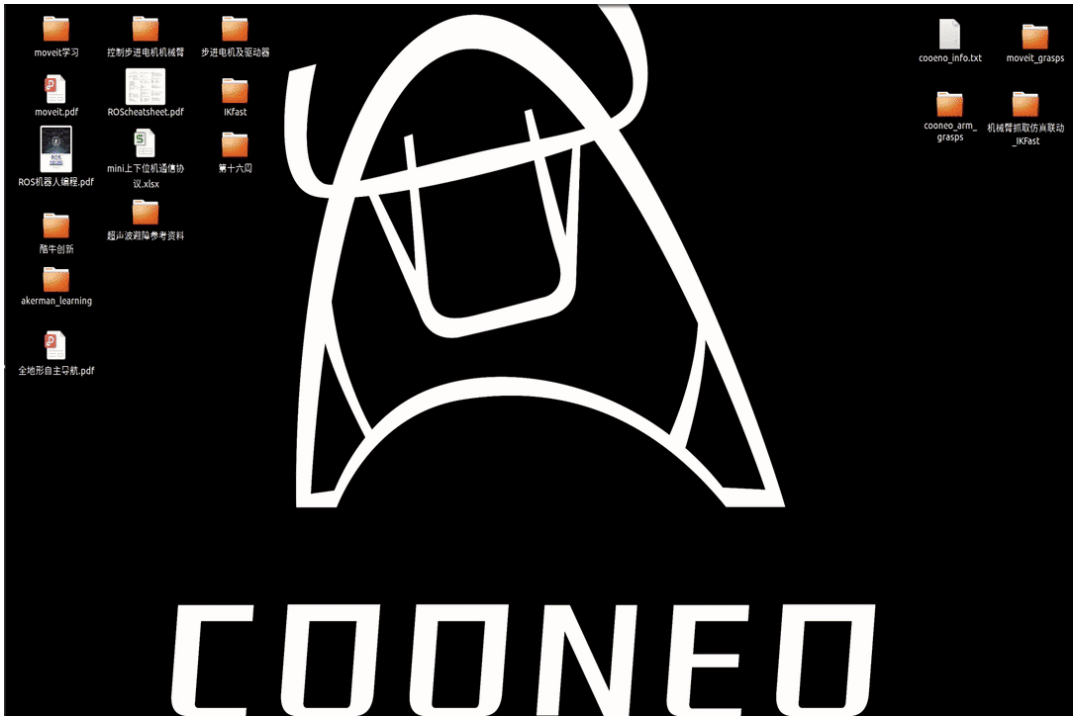
```
# open a Terminal
cd ~/neor_mini/mini_sim18_ws/
source devel/setup.bash        # if failed,please catkin_make all packages

# launchUltrasonars sensors
roslaunch neor_mini display_gazebo_sensors_ultrasonars.launch
```



step 2: Start ultrasonic obstacle avoidance simulation in Gazebo (Stop step one)

```
# open one Terminal
cd ~/neor_mini/mini_sim18_ws/
source devel/setup.bash        # if failed,please catkin_make all packages

roslaunch steer_mini_gazebo steer_mini_sim_sensors_ultrasonars.launch
```


```
# open another Terminal
cd ~/neor_mini/mini_sim18_ws/
source devel/setup.bash        # if failed,please catkin_make all packages

# launch the ultra_avoid_obstacle node
roslaunch ultra_avoid_obstacle ultra_avoid_obstacle_node.launch
```



Congratulations!!!

Chapter 5: Send the Goal to navigation through the ROS node

Corresponding WeChat article

搭建ROS机器人之——用程序发布导航目标点

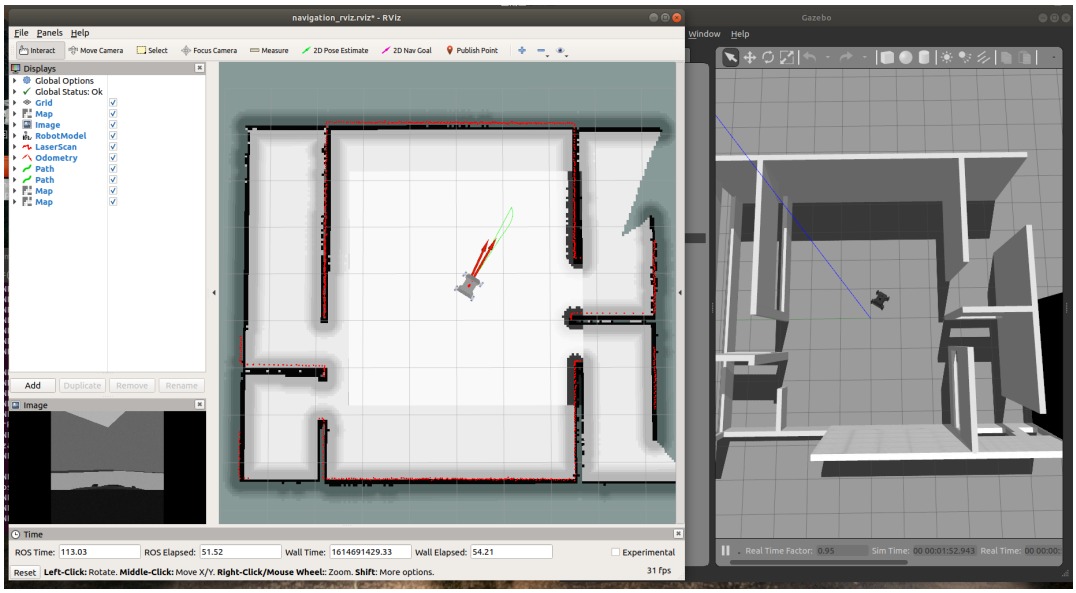
原创：Lee

搭建ROS机器人之——用程序发布导航目标点（Python版本）

原创：Lee

Step 1 : launch mini_navigation launch file.

```
# open a terminal
cd ~/neor_mini/mini_sim18_ws
catkin_make
source devel/setup.bash
roslaunch mini_navigation navigation_steer_mini_sensors.launch
```



Step 2 : rosrun the send_goals ROS node. (**C++ ROS node**)

```
# open another terminal "ctrl + shift +t" (Recommend)
source devel/setup.bash
rosrun send_goals send_goals_node
```

```
lee@lee:~/neor_mini/mini_sim18_ws$ rosrun send_goals send_goals_node
[ INFO] [1632018773.504112868, 82.635000000]:  Init success!!!
[ INFO] [1632018773.506881526, 82.636000000]: Send NO. 0 Goal !!!
[ INFO] [1632018793.874863128, 92.636000000]: The NO. 0 Goal achieved success !!!
[ INFO] [1632018793.874937326, 92.636000000]: Send NO. 1 Goal !!!
[ INFO] [1632018821.371050977, 106.238000000]: The NO. 1 Goal achieved success !!!
[ INFO] [1632018821.371196283, 106.238000000]: Send NO. 2 Goal !!!
[ INFO] [1632018875.159910372, 132.438000000]: The NO. 2 Goal achieved success !!!
[ INFO] [1632018875.160014704, 132.438000000]: Send NO. 3 Goal !!!
[ INFO] [1632018915.563981344, 152.038000000]: The NO. 3 Goal achieved success !!!
```

Step 3 : rosrun the send_goals_python.py file (**Python ROS node**)

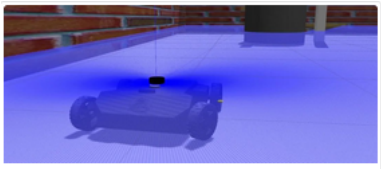
```
# open another terminal "ctrl + shift +t" (Recommend)
source devel/setup.bash
sudo chmod 0777 src/send_goals/scripts/send_goals_python.py
rosrun send_goals send_goals_python.py
```

```
lee@lee:~/neor_mini/mini_sim18_ws$ source devel/setup.bash
lee@lee:~/neor_mini/mini_sim18_ws$ rosrun send_goals send_goals_
send_goals_node       send_goals_python.py
lee@lee:~/neor_mini/mini_sim18_ws$ rosrun send_goals send_goals_python.py
[INFO] [1634729755.633151, 106.857000]: Send NO. 0 Goal !!!
[INFO] [1634729764.446170, 115.088000]: The NO. 0 Goal achieved success !!!
[INFO] [1634729764.447432, 115.089000]: Send NO. 1 Goal !!!
[INFO] [1634729779.169243, 128.899000]: The NO. 1 Goal achieved success !!!
[INFO] [1634729779.170825, 128.900000]: Send NO. 2 Goal !!!
[INFO] [1634729810.961006, 158.901000]: The NO. 2 Goal Planning Failed for some reasons
[INFO] [1634729810.962507, 158.903000]: Send NO. 3 Goal !!!
[INFO] [1634729831.524803, 178.211000]: The NO. 3 Goal achieved success !!!
[INFO] [1634729831.526369, 178.212000]: Mission Finished.
```

Congratulations!!!

Chapter 6: Hector_mapping

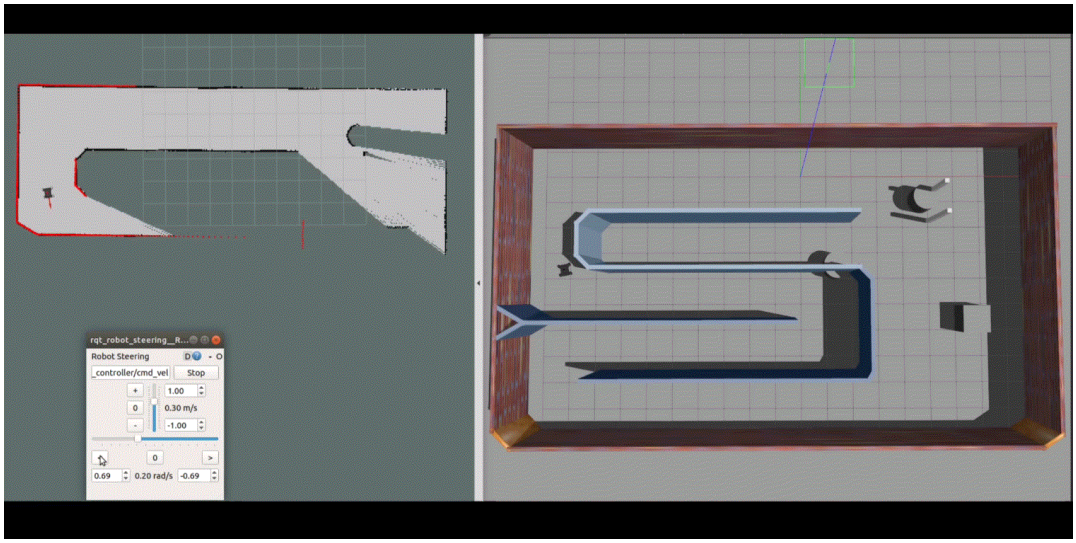
Corresponding WeChat article



搭建ROS机器人之——雷达2D建图方法总览

Step 1 : launch hector_mapping.launch

```
# # open a terminal
cd ~/neor_mini/mini_sim18_ws
rosdep install --from-paths src --ignore-src -r -y    # you need wait a moment
catkin_make
source devel/setup.bash
roslaunch mini_gmapping hector_mapping.launch
```



Congratulations!!!

Details paper:

```
@INPROCEEDINGS{KohlbrecherMeyerStrykKlingaufFlexibleSlamSystem2011,
 author = {S. Kohlbrecher and J. Meyer and O. von Stryk and U. Klingauf},
 title = {A Flexible and Scalable SLAM System with Full 3D Motion Estimation},
 year = {2011},
 month = {November},
 booktitle = {Proc. IEEE International Symposium on Safety, Security and Rescue Robotics (SSRR)},
 organization = {IEEE},
}
```

Chapter 7: rf2o_laser_odometry && gmapping

Corresponding WeChat article

Step 1 : Find neor_mini_gazebo_sensors.urdf file, and replace the lidar scan frequency with 5.

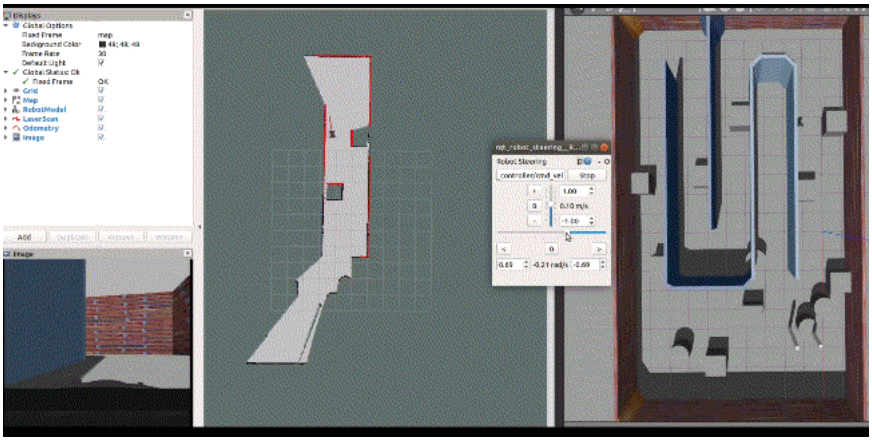
```
# the detiles blow: cd neor_mini/mini_sim18_ws/src/neor_mini/urdf && vim neor_mini_gazebo_sensors.urdf
...
<gazebo reference="laser_link">
 <sensor type="ray" name="Top_laser_sensor">
  <visualize>false</visualize>  <!-- visual or unvisual laser ray :true or false-->
  <update_rate>40</update_rate> <!-- 默认是 40，修改为 5 Hz -->
...
```

Step 2 : Find the steer_mini_sim_sensors_with_other_world.launch , and replace the world file with "cooneo_rooom2.world"

```
# cd neor_mini/mini_sim18_ws/src/steer_mini_gazebo/mini_gazebo/launch
# vim steer_mini_sim_sensors_with_other_world.launch
<!--Load the surrounding environment into Gazebo-->
<include file="$(find gazebo_ros)/launch/empty_world.launch" >
  <arg name="world_name" value="$(find neor_mini)/worlds/cooneo_room2.world"/>
</include>
...
```

Step 3 : launch rf2o_laser_odometry and gmapping node.

```
#open a terminal
cd ~/neor_mini/mini_sim18_ws
rosdep install --from-paths src --ignore-src -r -y    # you need wait a moment
catkin_make
source devel/setup.bash
roslaunch mini_gmapping laser_odom_gmapping.launch
```



Congratulations!!!

Thinks for MAPIRlab/**rf2o_laser_odometry** !!!

Chapter 8: Velodyne-16 && lio_sam 3D Mappig

Step 1 : install the dependence of the lio_sam，follow as below:

· gtsam(Georiga Tech Smoothing and Mapping library)

```
sudo add-apt-repository ppa:borglab/gtsam-release-4.0
sudo apt update
sudo apt install libgtsam-dev libgtsam-unstable-dev
```

or see the lio_sam git.

```
https://github.com/TixiaoShan/LIO-SAM.git
```

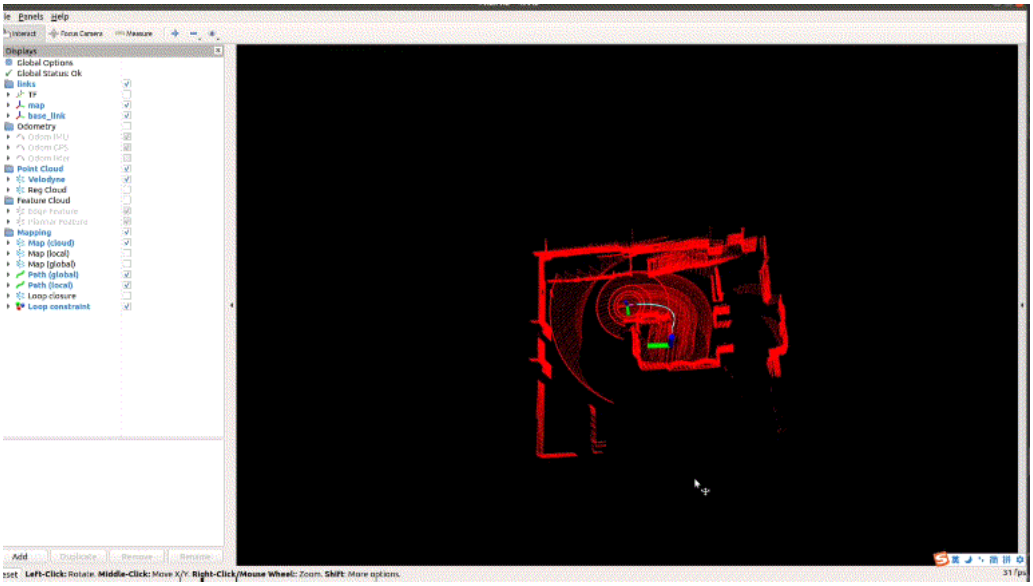
Step 2 : recording the vlp-16 && imu data in Gazebo By rosbag .

```
# open a new Terminal
cd neor_mini/mini_sim18_ws/
catkin_make
source devel/setup.bash
roslaunch steer_mini_gazebo steer_mini_sim_sensors_VLP16_lio_sam.launch

# open another Terminal
rosbag record -a

# moving the neor_mini by rqt_robot_steering plugines.
```

this step is same as the chapter 3 .

Step 3: play your recording bag ,and run lio_sam to 3D-Mapping.

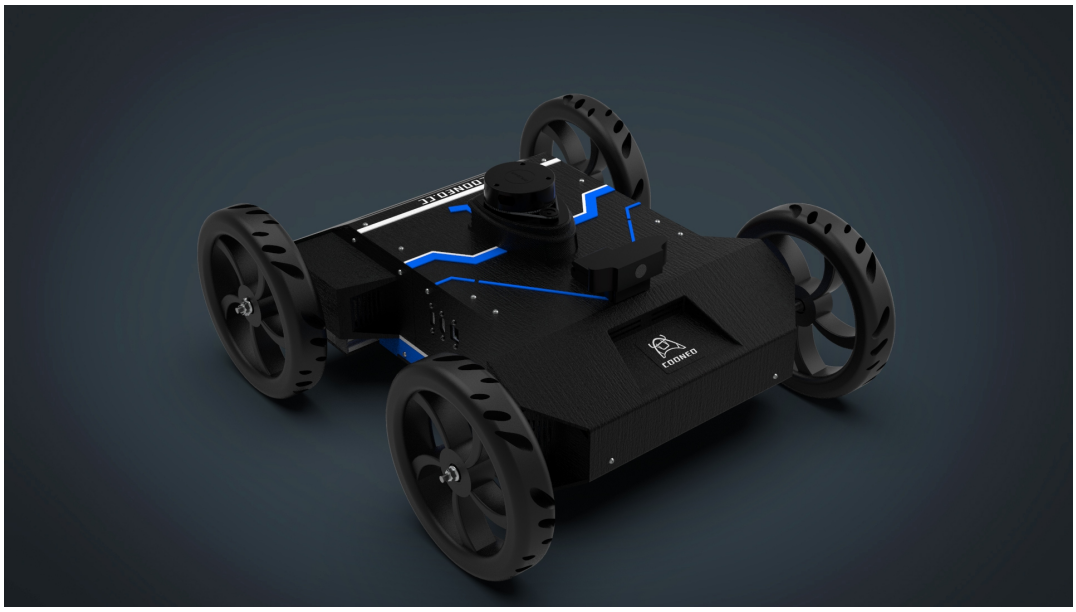
```
# open a new Terminal
cd neor_mini/mini_sim18_ws/
catkin_make
source devel/setup.bash
roslaunch lio_sam run_neor_mini.launch

#open another terminal
cd  your_bag_saved_folder
rosbag play -r 5 yourbag_name.bag
```




```
@inproceedings{liosam2020shan,
title={LIO-SAM: Tightly-coupled Lidar Inertial Odometry via Smoothing and Mapping},
author={Shan, Tixiao and Englot, Brendan and Meyers, Drew and Wang, Wei and Ratti, Carlo and Rus
Daniela},
booktitle={IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)},
pages={5135-5142},
year={2020},
organization={IEEE}
}
```

Thank you very much for Tixiaoshan && iscas_museum.

our new Arduino Ackermann ROS CAR.



2022.04.17

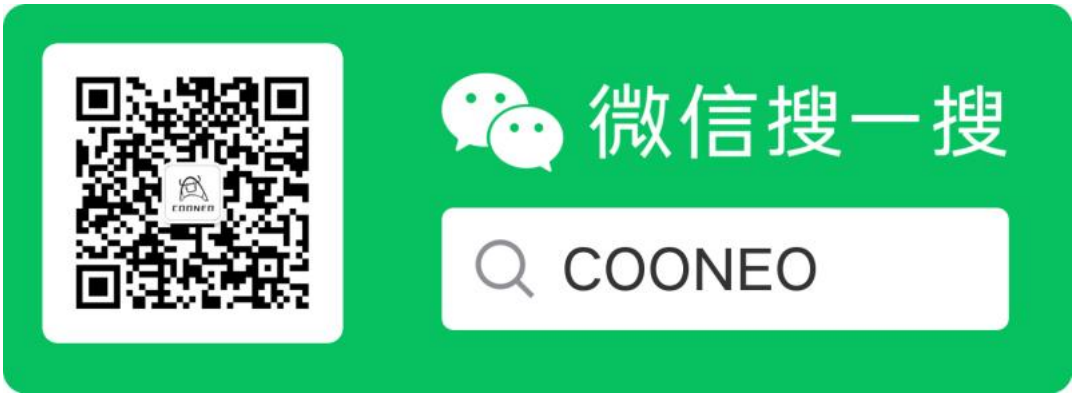
author:ZhaoXiang Li && COONEO

COONEO Co.,Ltd

Web:http://cooneo.cc

E: cooneo@outlook.com

For more details,you can search "COONEO" in your WeChat.



For more tutorials'video, please search "COONEO" in your Bilibili App.

neor_mini