

前面[ROS机器人底盘\(30\)-laser_filters的使用（1）](#)讲了如何使用laser_filters,本文在hades上试下如何使用laser_filters屏蔽车身的摄像头支架

1.车体支架干扰

首先启动建图

```
pibot_gmapping
```

虚拟机启动RVIZ

```
pibot_view
```



可以看到受到摄像头支架干扰，雷达在车体上检测到障碍，这会直接干扰到建图和导航

2.屏蔽支架干扰

2.1添加laser_filters

- 添加[ROS机器人底盘\(30\)-laser_filters的使用（1）](#)这里的2个文件放置下面目录

```
pibot_bringup/launch/box_filter_example.launch  
pibot_bringup/params/box_filter.yaml
```

根据车上宽度(200mm)和长度(300mm)分别修改box_filter.yaml中的参数

设置稍微超出一点 x方向我们设置-0.15至0.15, y方向设置-0.2至0.2

```
scan_filter_chain:
- name: box_filter
  type: laser_filters/LaserScanBoxFilter
  params:
    box_frame: laser_link
    min_x: -0.15
    max_x: 0.15
    min_y: -0.20
    max_y: 0.20
    min_z: -0.1
    max_z: 0.1
```

这里z值忽略

- 添加该launch文件到robot.launch

```
<launch>
  <arg name="lidar" default="$(env PIBOT_LIDAR)" doc="lidar type [rplidar, eai-x4, eai-g4]"/>

  <include file="$(find pibot_bringup)/launch/bringup.launch"/>
  <include file="$(find pibot_bringup)/launch/model.launch"/>

  <include file="$(find pibot_bringup)/launch/$(arg lidar).launch"/>

  <include file="$(find pibot_bringup)/launch/box_filter_example.launch"/>
</launch>
```

- 查看laser_filter发出topic rosnode info laser_filter可以找到发出的scan_filtered的topic

```
pibot@pibot-desktop:~$ rosnode info laser_filter
```

```
-----
Node [/laser_filter]
Publications:
* /rosout [rosgraph_msgs/Log]
* /scan_filtered [sensor_msgs/LaserScan]

Subscriptions:
* /scan [sensor_msgs/LaserScan]
* /tf [tf2_msgs/TFMessage]
* /tf_static [tf2_msgs/TFMessage]

Services:
* /laser_filter/get_loggers
* /laser_filter/set_logger_level
```

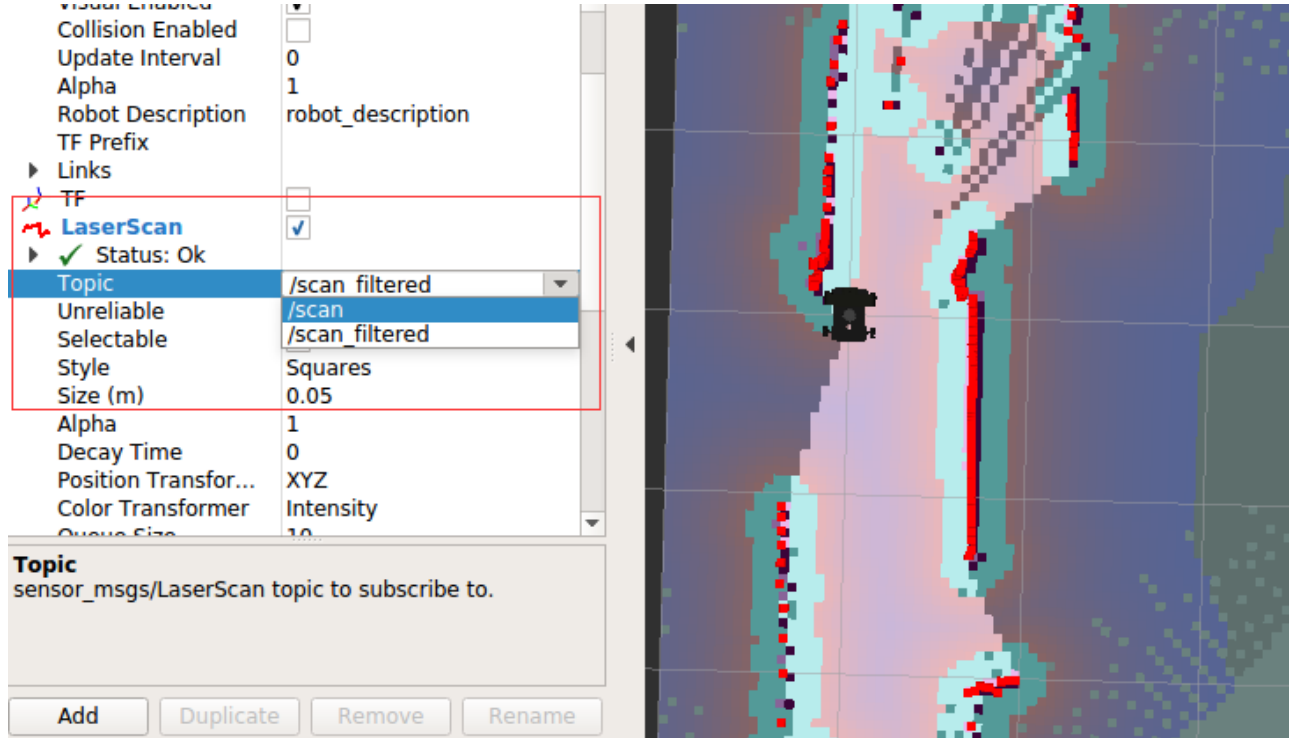
2.2测试添加效果

- 重新启动pibot_gmapping

如果提示下图错误需要安装laser_filters包, 使用pibot_install_ros.sh或者sudo apt-get install ros-kinetic-laser-filters

```
process[robot_state_publisher-4]: started with pid [4913]
process[rplidarNode-5]: started with pid [4926]
ERROR: cannot launch node of type [laser_filters/scan_to_scan_filter_chain]: laser_filters
ROS path [0]=/opt/ros/kinetic/share/ros
ROS path [1]=/home/pibot/pibot_ros/ros_ws/src
ROS path [2]=/opt/ros/kinetic/share
process[move_base-7]: started with pid [4945]
```

- 重新打开pibot_view,如果不出意外应该跟上面一样, 原因这里我们修改添加laser_filters发出来的topic还未使用, 修改rviz中scan topic



修改后可以看到车体再无干扰点出现

2.3修改订阅参数

刚才修改的只是Rviz的订阅参数, 真正导航和建图还是用的之前就的scan

可以通过`rostopic info xxx`查看订阅node订阅topic情况,如`rostopic info xxx`

- 查看下scan topic的信息

```
rostopic info /scan
```

```
pibot@pibot-desktop:~$ rostopic info /scan
Type: sensor_msgs/LaserScan

Publishers:
* /rplidarNode (http://192.168.2.239:32915/)

Subscribers:
* /laser_filter (http://192.168.2.239:33329/)
* /slam_gmapping (http://192.168.2.239:40815/)
* /move_base (http://192.168.2.239:36227/)
* /rviz (http://192.168.2.163:38857/)
```

可以看到除了被rviz订阅外还被

laser_filter,move_base和slam_gmapping订阅, 我们只需要把move_base和slam_gmapping中订阅scan的

topic改为laser_filter发出的topic

- 替换scan为scan_filtered 具体都是配置不知道我们可以查找下

```
roscd piBOT_navigation
grep -rn scan
```

```
piBOT@piBOT-desktop:~/piBOT_ros/ros_ws/src/piBOT_navigation$
piBOT@piBOT-desktop:~/piBOT_ros/ros_ws/src/piBOT_navigation$ grep -rn scan
params/costmap_common_params_zeus.yaml:26: observation_sources: scan
params/costmap_common_params_zeus.yaml:27: scan:
params/costmap_common_params_zeus.yaml:29: topic: scan
params/costmap_common_params_apollo.yaml:26: observation_sources: scan
params/costmap_common_params_apollo.yaml:27: scan:
params/costmap_common_params_apollo.yaml:29: topic: scan
params/costmap_common_params_hadesX.yaml:26: observation_sources: scan
params/costmap_common_params_hadesX.yaml:27: scan:
params/costmap_common_params_hadesX.yaml:29: topic: scan
params/costmap_common_params_hera.yaml:26: observation_sources: scan
params/costmap_common_params_hera.yaml:27: scan:
params/costmap_common_params_hera.yaml:29: topic: scan
params/costmap_common_params_apolloXL.yaml:26: observation_sources: scan
params/costmap_common_params_apolloXL.yaml:27: scan:
params/costmap_common_params_apolloXL.yaml:29: topic: scan
params/costmap_common_params_apolloX.yaml:26: observation_sources: scan
params/costmap_common_params_apolloX.yaml:27: scan:
params/costmap_common_params_apolloX.yaml:29: topic: scan
params/costmap_common_params_hades.yaml:26: observation_sources: scan
params/costmap_common_params_hades.yaml:27: scan:
params/costmap_common_params_hades.yaml:29: topic: scan
launch/hector_mapping_without_odom.launch:11: <arg name="trajectory_source_frame_name" value="scanmatcher_frame"/>
launch/include/karto_mapping.launch.xml:3: <remap from="scan" to="scan"/>
launch/include/rgbd_gmapping.launch.xml:2: <arg name="scan_topic" default="scan" />
launch/include/rgbd_gmapping.launch.xml:41: <remap from="scan" to="$(arg scan_topic)"/>
launch/include/amcl.launch.xml:4: <arg name="scan_topic" default="scan"/>
launch/include/amcl.launch.xml:8: <!-- Publish scans from best pose at a max of 10 Hz -->
launch/include/amcl.launch.xml:40: <remap from="scan" to="$(arg scan_topic)"/>
launch/include/rgbd_amcl.launch.xml:4: <arg name="scan_topic" default="scan"/>
launch/include/rgbd_amcl.launch.xml:8: <!-- Publish scans from best pose at a max of 10 Hz -->
launch/include/rgbd_amcl.launch.xml:40: <remap from="scan" to="$(arg scan_topic)"/>
launch/include/gmapping.launch.xml:2: <arg name="scan_topic" default="scan" />
```

修改相应文件并重启piBOT_gmapping即可