

RVIZ 라이다 센서 과제

조정민

lidar_urdf.launch

```
<launch>

  <!-- 박스형상 모델링 -->
  <param name="robot_description" textfile="$(find
rviz_lidar)/urdf/lidar_urdf.urdf"/>
  <param name="use_gui" value="true"/>
  <!-- Rviz 설정 파일 -->
  <node name="rviz_visualizer" pkg="rviz" type="rviz" required="true"
args="-d $(find rviz_lidar)/rviz/lidar_urdf.rviz" />
  <node name="robot_state_publisher" pkg="robot_state_publisher"
type="state_publisher"/>
  <!-- 라이다 토픽 발행 -->
  <node name="rosbag_play" pkg="rosbag" type="play" output="screen"
required="true" args="$(find rviz_lidar)/src/lidar_topic.bag">
    <param name="~frame_id" value="base_link" />
  </node>
  <!-- 토픽 변환 -->
  <node name="lidar_converter" pkg="rviz_lidar" type="lidar_urdf.py"/>

</launch>
```

lidar_urdf.urdf

- 원래는 기반 링크 이름을 base_link 로 하였으나, slam에서 보내는 laserScan 메시지를 함께 보고 싶었기 때문에 fixed framed 이 laser로 통일 시켜주었습니다.

```
<?xml version="1.0" ?>
<robot name="xycar" xmlns:xacro="http://www.ros.org/wiki/xacro">

  <link name="laser"/>
  <link name="baseplate">
    <visual>
      <material name="black"/>
      <origin rpy="0 0 0" xyz="0 0 0"/>
      <geometry>
        <box size="0.1 0.1 0.07"/>
      </geometry>
    </link>
  </robot>
```

```

    </visual>
</link>

<joint name="laser_to_baseplate" type="fixed">
  <parent link="laser"/>
  <child link="baseplate"/>
  <origin rpy="0 0 0" xyz="0 0 0"/>
</joint>

<link name="front"/>
<joint name="baseplate_to_front" type="fixed">
  <parent link="baseplate"/>
  <child link="front"/>
  <origin rpy="0 0 0" xyz="0.1 0 0"/>
</joint>

<link name="back"/>
<joint name="baseplate_to_back" type="fixed">
  <parent link="baseplate"/>
  <child link="back"/>
  <origin rpy="0 0 -3.14" xyz="-0.1 0 0"/>
</joint>

<link name="left"/>
<joint name="baseplate_to_left" type="fixed">
  <parent link="baseplate"/>
  <child link="left"/>
  <origin rpy="0 0 1.57" xyz="0 0.1 0"/>
</joint>

<link name="right"/>
<joint name="baseplate_to_right" type="fixed">
  <parent link="baseplate"/>
  <child link="right"/>
  <origin rpy="0 0 -1.57" xyz="0 -0.1 0"/>
</joint>

<material name="black">
  <color rgba="0.0 0.0 0.0 1.0"/>
</material>
</robot>

```

lidar_urdf.py

```
#!/usr/bin/env python

import serial, time, rospy
from sensor_msgs.msg import LaserScan
from sensor_msgs.msg import Range
from std_msgs.msg import Header
# frame_id 값을 담고 있는 리스트 생성
frame = ["back", "right", "front", "left"]
def lidar_callback(data):
    rg = Range()
    rg.header = Header()
    rg.radiation_type = Range().ULTRASOUND
    rg.min_range = 0.00 # 2cm
    rg.max_range = 13.0 # 2m
    for i in range(4):
        # frame_id 에 알맞는 range값 할당 및 발행
        rg.header.frame_id = frame[i]
        rg.header.stamp = rospy.Time.now()
        rg.field_of_view = (20.0/180.0) * 3.14
        rg.range = data.ranges[90 * i]
        pub[i].publish(rg)

rospy.init_node('lidar')

rospy.Subscriber('scan', LaserScan, lidar_callback)

pub = [None, None, None, None]

# 4개의 퍼블리셔 노드들을 for문 이용해서 생성
for i in range(4):
    name = 'scan'+str(i+1)
    pub[i] = rospy.Publisher(name, Range, queue_size=1)

rospy.spin()
```

실행화면

```
$ roslaunch rviz_lidar lidar_urdf.launch
```

