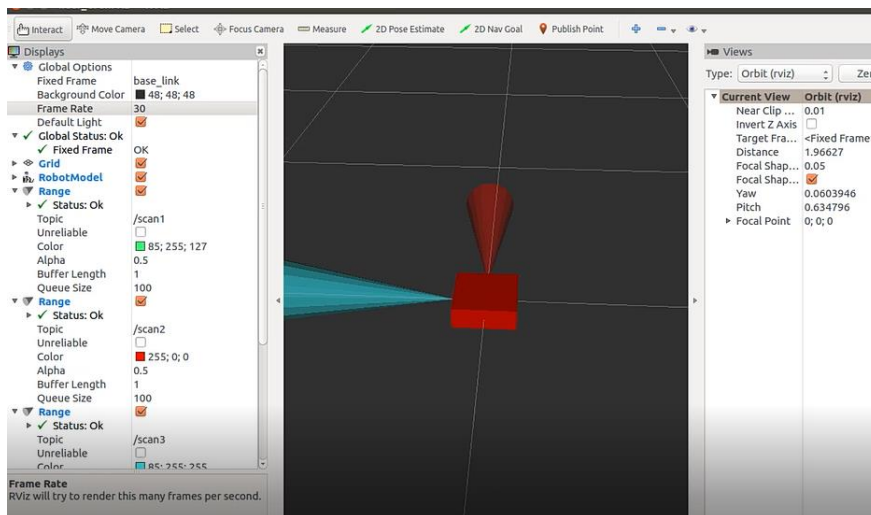


과제 설명

RVIZ에서 라이다 정보를 Range로 표시하기

➔ 라이다 데이터를 Range 데이터로 바꾸어 RVIZ로

➔ lidar_topic.bag(scan토픽 발행) -> lidar_urdf.py -> RVIZ 뷰어



1. lidar_urdf.py

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

```
#필요한 파일 가져오기
```

```
import rospy, time, serial
```

```
from std_msgs.msg import Header
```

```
from sensor_msgs.msg import LaserScan
```

```
from sensor_msgs.msg import Range
```

```
#lidar_topic.bag의 range 값을 가지게 될 변수
```

```
lidar_points = None
```

```
#!/scan으로부터 구독할 때마다 해당 데이터의 range 값 얻기
```

```

def lidar_callback(data):

    global lidar_points

    lidar_points = data.ranges

    # lidar이름으로 노드 생성 및 /scan으로부터 구독하기
    rospy.init_node("lidar")

    rospy.Subscriber("/scan", LaserScan, lidar_callback, queue_size=1)

    #토픽을 발행 할 4개의 노드 생성

    pub1 = rospy.Publisher('scan1', Range, queue_size=1)
    pub2 = rospy.Publisher('scan2', Range, queue_size=1)
    pub3 = rospy.Publisher('scan3', Range, queue_size=1)
    pub4 = rospy.Publisher('scan4', Range, queue_size=1)

    # Range데이터 타입의 메시지 설정

    msg=Range()

    h=Header()

    msg.header=h

    msg.radiation_type = Range().ULTRASOUND

    msg.field_of_view = (20.0/180.0)*3.14

    msg.min_range = 0.2

    msg.max_range = 2.0


    # 멈추기 전까지 계속 진행

    while not rospy.is_shutdown():

        # lidar_points값이 들어올 때까지 기다림

        if lidar_points==None:

            continue

    # scan1~scan4까지 lidar_points로 들어오는 값을 하나씩 Range타입의 range로 설정하여 발행

```

```

for i in range(len(lidar_points)):

    h.frame_id="front"    #각 방향에 맞게 발행 노드에 따른 헤더 값 지정

    msg.range =lidar_points[i] # lidar_points의 값을 하나씩 발행

    pub1.publish(msg)

```

```

for i in range(len(lidar_points)):

    h.frame_id="back"

    msg.range =lidar_points[i]

    pub2.publish(msg)

```

```

for i in range(len(lidar_points)):

    h.frame_id="left"

    msg.range =lidar_points[i]

    pub3.publish(msg)

```

```

for i in range(len(lidar_points)):

    h.frame_id="right"

    msg.range =lidar_points[i]

    pub4.publish(msg)

```

2. lidar_urdf.urdf

```

<?xml version="1.0" ?>

<robot name="xycar" xmlns:xacro="http://www.ros.org/wiki/xacro">

    <link name="base_link"/>

```

```

<link name="baseplate"> # baseplate으로 box 설정

  <visual>

    <material name="red"/> # 색상 red로 설정

    <origin rpy="0 0 0" xyz="0 0 0"/>

    <geometry>

      <box size="0.2 0.2 0.07"/> # box크기 가로 0.2, 세로0.2, 높이 0.07로 설정로 설정

    </geometry>

  </visual>

</link>

<joint name="base_link_to_baseplate" type="fixed">

  <parent link="base_link"/>

  <child link="baseplate"/>

  <origin rpy="0 0 0" xyz="0 0 0"/>

</joint>

<link name="front"/> #앞면의 센서 프레임

<joint name="baseplate_to_front" type="fixed">

  <parent link="baseplate"/> #baseplate에 연결

  <child link="front"/>

  <origin rpy="0 0 0" xyz="0.1 0 0"/> #joint부분을 box의 x축으로 0.1위치에 지정

</joint>

<link name="back"/> #뒷면의 센서 프레임

<joint name="baseplate_to_back" type="fixed">

  <parent link="baseplate"/> #baseplate에 연결

  <child link="back"/>

  <origin rpy="0 0 -3.1" xyz="-0.1 0 0"/> #방향을 z축으로 -3.1만큼 이동 후 joint부분을 box

```

의 -x축으로 0.1위치에 지정

```
</joint>
```

```
<link name="left"/> #왼쪽면의 센서 프레임
```

```
<joint name="baseplate_to_left" type="fixed">
```

```
<parent link="baseplate"/>#baseplate에 연결
```

```
<child link="left"/>
```

```
<origin rpy="0 0 -1.6" xyz="0 -0.1 0"/># 방향을 z축으로 -1.6만큼 이동 후 joint부분을  
box의 -y축으로 0.1위치에 지정
```

```
</joint>
```

```
<link name="right"/> #오른쪽면의 센서 프레임
```

```
<joint name="baseplate_to_right" type="fixed">
```

```
<parent link="baseplate"/>#baseplate에 연결
```

```
<child link="right"/>
```

```
<origin rpy="0 0 1.6" xyz="0 0.1 0 "/># 방향을 z축으로 1.6만큼 이동 후 joint부분을 box  
의 y축으로 0.1위치에 지정
```

```
</joint>
```

#해당 색상지정 관련 material

```
<material name="black">
```

```
<color rgba="0.0 0.0 0.0 1.0"/>
```

```
</material>
```

```
<material name="blue">
```

```
<color rgba="0.0 0.0 0.8 1.0"/>
```

```
</material>
```

```
<material name="green">
```

```
<color rgba="0.0 0.8 0.0 1.0"/>

</material>

<material name="grey">

  <color rgba="0.2 0.2 0.2 1.0"/>

</material>

<material name="orange">

  <color rgba="1.0 0.423529411765 0.0392156862745 1.0"/>

</material>

<material name="brown">

  <color rgba="0.870588235294 0.811764705882 0.764705882353 1.0"/>

</material>

<material name="red">

  <color rgba="0.8 0.0 0.0 1.0"/>

</material>

<material name="white">

  <color rgba="1.0 1.0 1.0 1.0"/>

</material>

<material name="acrylic">

  <color rgba="1.0 1.0 1.0 0.4"/>

</material>

</robot>
```