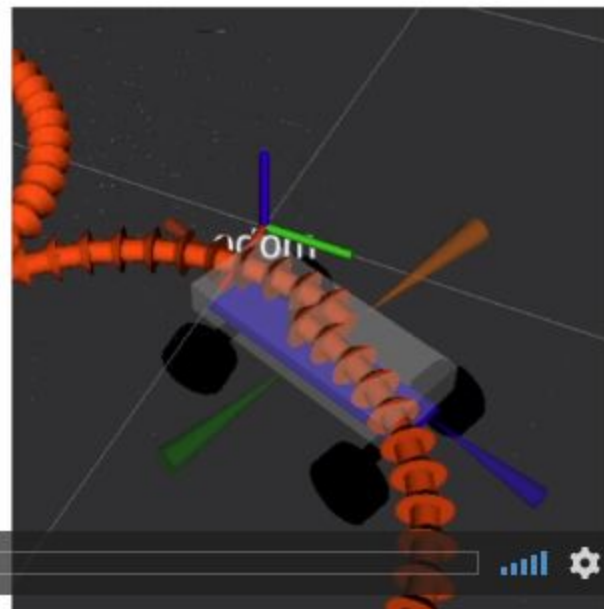
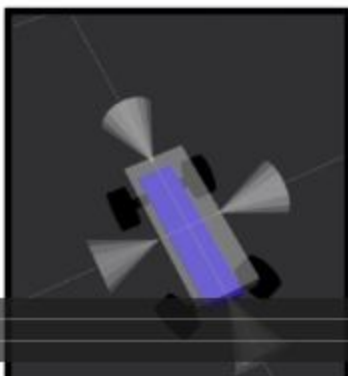




RVIZ 센서 통합

프로그래머스 자율주행 코스 1기 조정민

- RVIZ 가상공간에서 8자 주행을 하는 자이카(3D모델링)에
 - 라이다센서와 IMU센서의 뷰어를 통합해 보자
- 3D 모델링된 차량이 8자 주행을 하면서
 - 주변 장애물까지의 거리값을 Range로 표시하고
 - IMU 센싱값에 따라 차체가 기울어진다



00:30



rviz_all.launch

```
<launch>

  <!-- 박스형상 모델링 -->
  <param name="robot_description" textfile="$(find rviz_all)/urdf/rviz_all.urdf"/>
  <param name="use_gui" value="true"/>
  <!-- Rviz 설정 파일 -->
  <node name="rviz_visualizer" pkg="rviz" type="rviz" required="true"
    |   args="-d $(find rviz_all)/rviz/rviz_all.rviz" />
  <node name="robot_state_publisher" pkg="robot_state_publisher" type="state_publisher"/>

  <!-- 자동차 8자 주행 -->
  <node name="driver" pkg="rviz_xycar" type="odom_8_drive.py" />
  <node name="converter" pkg="rviz_xycar" type="converter.py" />
  <node name="rviz_odom" pkg="rviz_all" type="odom_imu.py" />

  <!-- 라이다 토픽 발행 -->
  <node name="rosbag_play" pkg="rosbag" type="play" output="screen" required="true"
    |   args="$(find rviz_lidar)/src/lidar_topic.bag"/>
  <node name="lidar_converter" pkg="rviz_lidar" type="lidar_urdf.py"/>

  <!-- IMU 토픽 발행 -->
  <node name="imu_generator" pkg="rviz_imu" type="imu_generator.py" output="screen"/>
```

```
/launch
```

1. imu 토픽을 받는 서브스크라이버 선언

```
36 rospy.init_node('rviz_odom')
37
38 odom_pub = rospy.Publisher("odom", Odometry, queue_size=50)
39 rospy.Subscriber('joint_states', JointState, callback_motor)
40 rospy.Subscriber('imu', Imu, callback_imu)
41 odom_broadcaster = tf.TransformBroadcaster()
42 last_time = rospy.Time.now()
43 rate = rospy.Rate(10)
```

2.callback_imu 함수 생성, 쿼터니언 값을 오일러 값으로 변환

```
29 def callback_imu(imu_data):
30     global imu_euler
31     euler = tf.transformations.euler_from_quaternion((imu_data.orientation.x, \
32                                                         imu_data.orientation.y, \
33                                                         imu_data.orientation.z, \
34                                                         imu_data.orientation.w))
35     imu_euler = [euler[0],euler[1],euler[2]]
```



odom_imu.py

3. odom_quat 변수에 각도값을 추가

```
# Euler value -> quaternion value  
odom_quat = tf.transformations.quaternion_from_euler(imu_euler[0] , imu_euler[1], th + imu_euler[2])
```



rviz_all.urdf

xycar_3d.urdf

+

lidar_urdf.urdf



실행

```
roslaunch rviz_all rivz_all.launch
```


실행 화면

