

THORMANG3

THORMANG3 Tutorial

Sensor



Table of Contents



1. Introduction

1. Description of the Sensors

2. Sensors

1. HD Web Cam
2. Depth Camera
3. LIDAR
4. FT Sensor
5. IMU

3. Check the Sensors

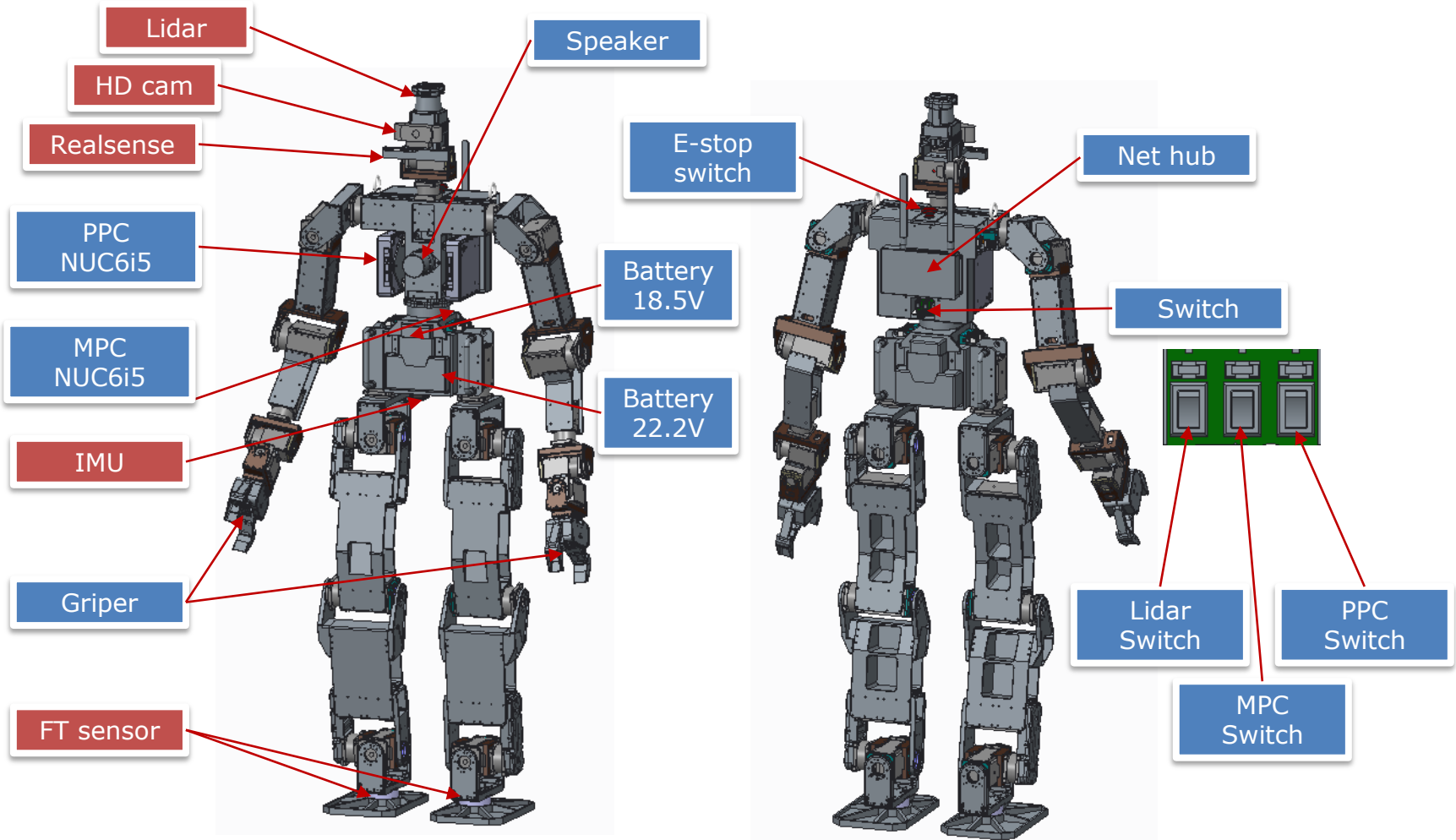
1. Introduction



Description of the Sensors



- Layout of the Sensors



2. Sensors



Sensor – HD Web Cam



1. Overview

1. HW : Logitech C920 HD
2. SW : uvc_camera (ROS package)
 - Wiki site : http://wiki.ros.org/uvc_camera
 - Installation (If user use ros-indigo version)

```
$ sudo apt-get install ros-indigo-uvc-camera
```

- Source : https://github.com/ktossell/camera_umd
- Note
 - HD Web Camera is connected to the PPC(Perception PC). All commands should be typed in PPC.

2. Topic List

	Name	Description
Topic	/robotis/sensor/camera/image_raw	A stream of images from the camera
	/robotis/sensor/camera/camera_info	Camera intrinsics for images



3. How to run individual

```
$ roslaunch thormang3_sensors thormang3_web_cam.launch
```

- thormang3_web_cam.launch

```
<?xml version="1.0"?>

<launch>
  <!-- center camera -->
  <node pkg="uvc_camera" type="uvc_camera_node" name="uvc_camera_center_node" >
    <param name="frame_id" type="string" value="cam_link" />
    <param name="device" type="string" value="/dev/WebCam" />
    <param name="width" type="int" value="640" />
    <param name="height" type="int" value="480" />
    <param name="fps" type="int" value="10" />
    <param name="auto_focus" type="bool" value="False" />
    <param name="focus_absolute" type="int" value="0" />
    <param name="auto_white_balance" value="0" />
    <param name="auto_exposure" value="0" />
    <param name="brightness" value="120" />
    <remap from="/image_raw" to="/robotis/sensor/camera/image_raw"/>
    <remap from="/camera_info" to="/robotis/sensor/camera/camera_info"/>
  </node>

  <!-- other camera -->
  <!-- <node pkg="uvc_camera" type="uvc_camera_node" name="uvc_camera_second_node" >
    <param name="frame_id" type="string" value="head_right_camera_link" />
    <param name="device" type="string" value="/dev/video2" />
    <remap from="/image_raw" to="/ppc/camera_right/image_raw"/>
    <remap from="/camera_info" to="/ppc/camera_right/camera_info"/>
  </node> -->

</launch>
```



Sensor – Depth camera (1)



1. Overview

1. HW : Intel Realsense R200
2. SW : RealSense_R200 (ROS package)
 - Wiki site : <http://wiki.ros.org/RealSense>
 - Installation : Go to the Wiki
 - Note
 - Depth Camera is connected to the PPC(Perception PC). All commands should be typed in PPC.

2. Topic List

	Name	Description
Topic	/realsense/rgb/image_raw	A stream of color images from the camera
	/realsense/rgb/camera_info	Camera intrinsics for images



Sensor – Depth camera (2)



2. Topic List (cont.)

	Name	Description
Topic	/realsense/depth/image_raw	A stream of depth images from the camera
	/realsense/depth/camera_info	Camera intrinsics for images
	/realsense/depth_registered/points	Registered XYZRGB point cloud

3. How to run individual

```
$ roslaunch thormang3_sensors thormang3_realsense.launch
```



Sensor – Lidar(option)



1. Overview

1. HW : Hokuyo UTM-30LX-EW
2. SW : urg_node (ROS package)
 - Wiki site : http://wiki.ros.org/urg_node
 - Installation (If user use ros-kinetic version)

```
$ sudo apt-get install ros-kinetic-urg-node
```

- Note
 - We assume that you're using ros kinetic.
 - Lidar is connected to the MPC(Motion PC). All commands should be typed in MPC.
 - Lidar's IP address is **10.17.3.31**

2. Topic List

	Name	Description
Topic	/robotis/sensor/scan	single return output

3. How to run individual

```
$ roslaunch thormang3_description thor_laserscan.launch
```



Sensor – FT sensor



1. Overview

1. HW : ATI Mini58
2. SW : thormang3_manager
 - thormang3 manager publishes the ft sensors output on the feet

2. Topic List

	Name	Description
Topic	/robotis/sensor/ft_right_foot/raw	raw out put from ft sensor on right foot
	/robotis/sensor/ft_right_foot/scaled	scaled out put from ft sensor on right foot
	/robotis/sensor/ft_left_foot/raw	raw out put from ft sensor on left foot
	/robotis/sensor/ft_left_foot/scaled	scaled out put from ft sensor on left foot



1. Overview

1. HW : MicroStrain 3DM-GX4-25
2. SW : imu_3dm_gx4
 - Wiki site : http://wiki.ros.org/imu_3dm_gx4
 - Git : https://github.com/KumarRobotics/imu_3dm_gx4
 - Installation : Please refer to above links.
 - Note
 - This package should be located on MPC

2. Topic List

	Name	Description
Topic	/robotis/sensor/imu/imu	Gyro and Acceleration
	/robotis/sensor/imu/filter	Orientation and Gyro Bias

3. How to run individual

```
$ roslaunch thormang3_imu_3dm_gx4 imu.launch
```

3. Check the Sensors



Check in terminal



- The Sensors in MPC (IMU, FT, Lidar)
 - **IMU** : type below ros command and check the messages

```
$ rostopic echo /robotis/sensor/imu/imu
```

```
robotis@mpc:~$ rostopic echo /robotis/sensor/imu/imu
header:
  seq: 218798
  stamp:
    secs: 1456227058
    nsecs: 657393447
  frame_id: imu
orientation:
  x: 0.551555931568
  y: 0.831686019897
  z: -0.0350182652473
  w: 0.0534624755383
orientation_covariance: [-1.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0]
angular_velocity:
  x: 0.00529906712472
  y: 0.0256397109479
  z: 0.108881101012
angular_velocity_covariance: [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0]
linear_acceleration:
  x: 1.40463167375
  y: 0.0995190255708
  z: 9.7412729802
linear_acceleration_covariance: [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0]
```



Check in terminal



- **The Sensors in MPC (IMU, FT, Lidar)**
 - **FT sensor** : type below ros command and check the messages

```
$ rostopic echo /robotis/sensor/ft_right_foot/raw  
$ rostopic echo /robotis/sensor/ft_left_foot/raw
```

```
robotis@mpc:~$ rostopic echo /robotis/sensor/ft_right_foot/raw  
header:  
  seq: 110484  
  stamp:  
    secs: 1456227469  
    nsecs: 577939051  
  frame_id: r_leg_foot_link  
wrench:  
  force:  
    x: 5175.13464253  
    y: 12796.9383106  
    z: 9230.67538064  
  torque:  
    x: -134.298186423  
    y: -234.033554692  
    z: -267.399062596
```

```
robotis@mpc:~$ rostopic echo /robotis/sensor/ft_left_foot/raw  
header:  
  seq: 113669  
  stamp:  
    secs: 1456227506  
    nsecs: 383800602  
  frame_id: l_leg_foot_link  
wrench:  
  force:  
    x: 9699.47838375  
    y: 5305.99705173  
    z: 10538.1527615  
  torque:  
    x: -243.133327243  
    y: -325.658292323  
    z: -202.514837491
```



Check in terminal



- The Sensors in MPC (IMU, FT, Lidar)
 - **LIDAR** : type below ros command and check the messages

```
$ rostopic echo /robotis/sensor/scan --noarr
```

- *--noarr : no array option*

```
robotis@mpc:~$ rostopic echo /robotis/sensor/scan --noarr
header:
  seq: 403902
  stamp:
    secs: 1456227605
    nsecs: 320116247
  frame_id: lidar_link
angle_min: -2.35619449615
angle_max: 2.35619449615
angle_increment: 0.00436332309619
time_increment: 1.73611151695e-05
scan_time: 0.0250000003725
range_min: 0.0230000000447
range_max: 60.0
---
header:
  seq: 403903
  stamp:
    secs: 1456227605
    nsecs: 350245326
  frame_id: lidar_link
angle_min: -2.35619449615
angle_max: 2.35619449615
angle_increment: 0.00436332309619
time_increment: 1.73611151695e-05
scan_time: 0.0250000003725
range_min: 0.0230000000447
range_max: 60.0
```




Check in terminal



- **The Sensors in PPC (HD Web Cam, Depth camera)**
 - **HD Web Cam** : type below ros command and check the messages

```
$ rostopic echo /robotis/sensor/camera/image_raw --noarr
```

- *--noarr : no array option*

```
robotis@ppc:~$ rostopic echo /robotis/sensor/camera/image_raw --noarr
header:
  seq: 23
  stamp:
    secs: 1456228631
    nsecs: 803098057
  frame_id: head_p_link
height: 480
width: 640
encoding: rgb8
is_bigendian: 0
step: 1920
---
header:
  seq: 24
  stamp:
    secs: 1456228631
    nsecs: 903091604
  frame_id: head_p_link
height: 480
width: 640
encoding: rgb8
is_bigendian: 0
step: 1920
```



Check in terminal



- **The Sensors in PPC (HD Web Cam, Depth camera)**
 - **Depth Camera** : type below ros command and check the messages

```
$ rostopic echo /realsense/depth_registered/points --noarr
```

- *--noarr : no array option*

```
robotis@ppc:~$ rostopic echo /realsense/depth_registered/points --noarr
header:
  seq: 321
  stamp:
    secs: 1489128097
    nsecs: 394884544
  frame_id: realsense_rgb_optical_frame
height: 480
width: 640
is_bigendian: False
point_step: 32
row_step: 20480
is_dense: False
---
header:
  seq: 322
  stamp:
    secs: 1489128097
    nsecs: 428217877
  frame_id: realsense_rgb_optical_frame
height: 480
width: 640
is_bigendian: False
point_step: 32
row_step: 20480
is_dense: False
```

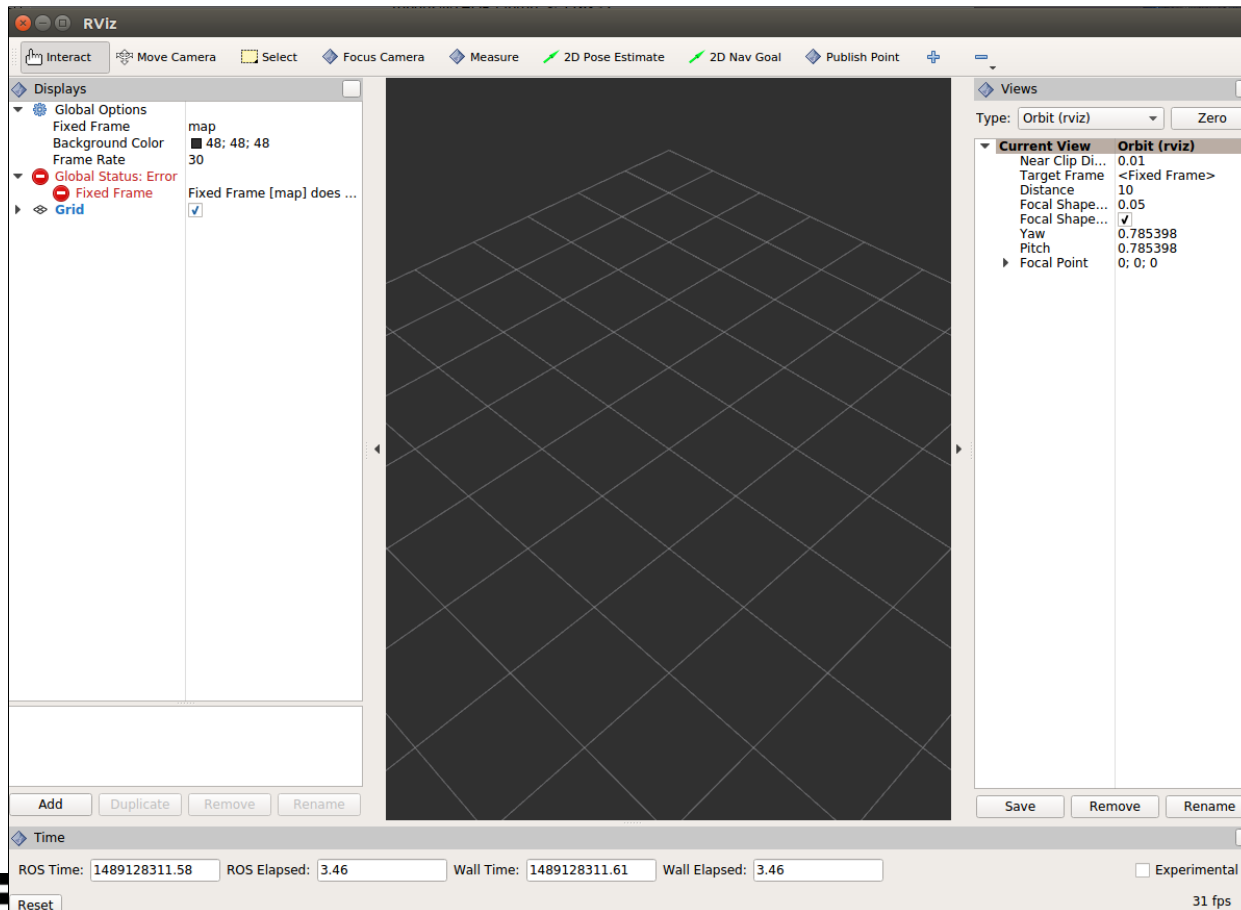


Check in visualization tool(Rviz)



- Open visualization tool in OPC
 - Run rviz : type below command

```
$ rviz
```

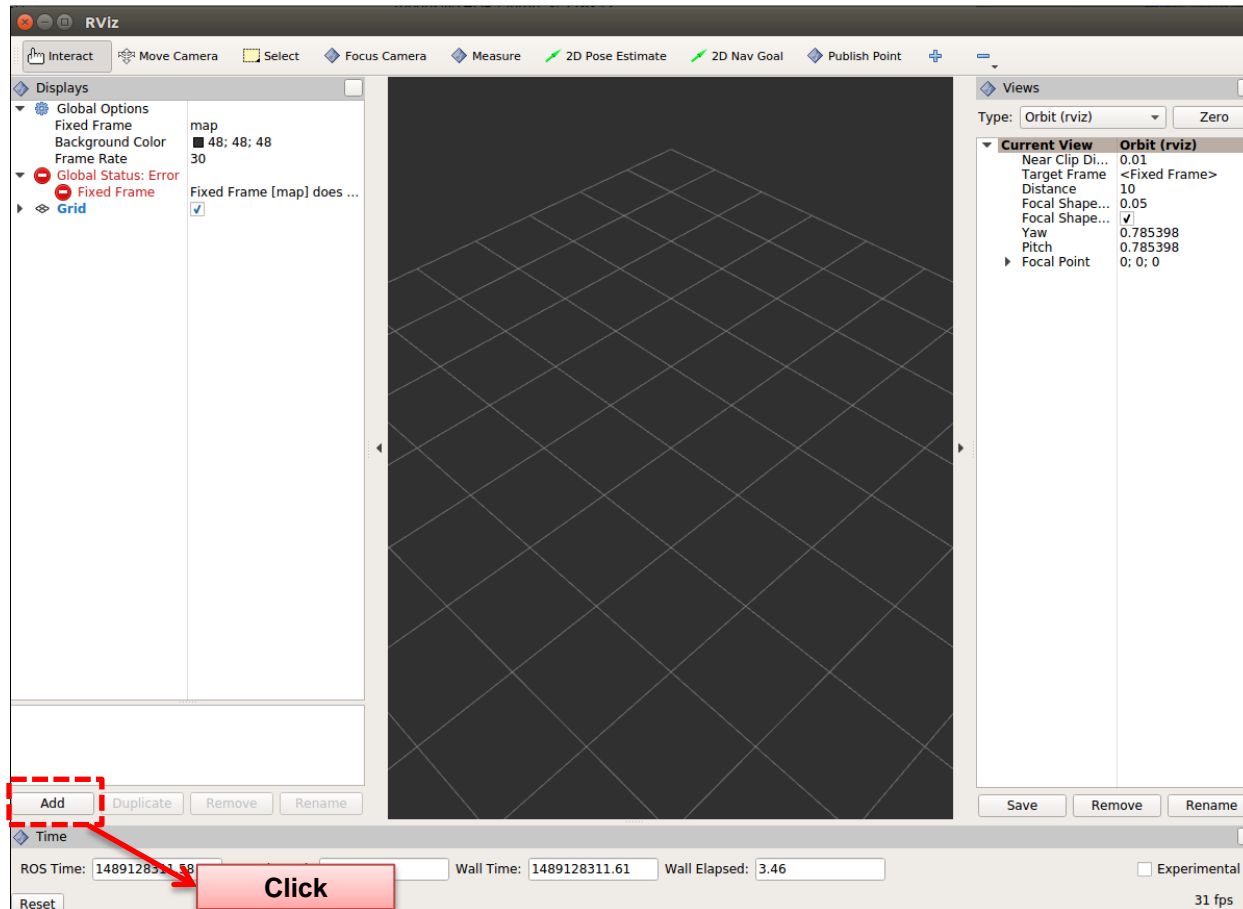




Check in visualization tool(Rviz)



- Open visualization tool in OPC
 - Add sensor topic and change 'Fixed Frame' to name of sensor frame

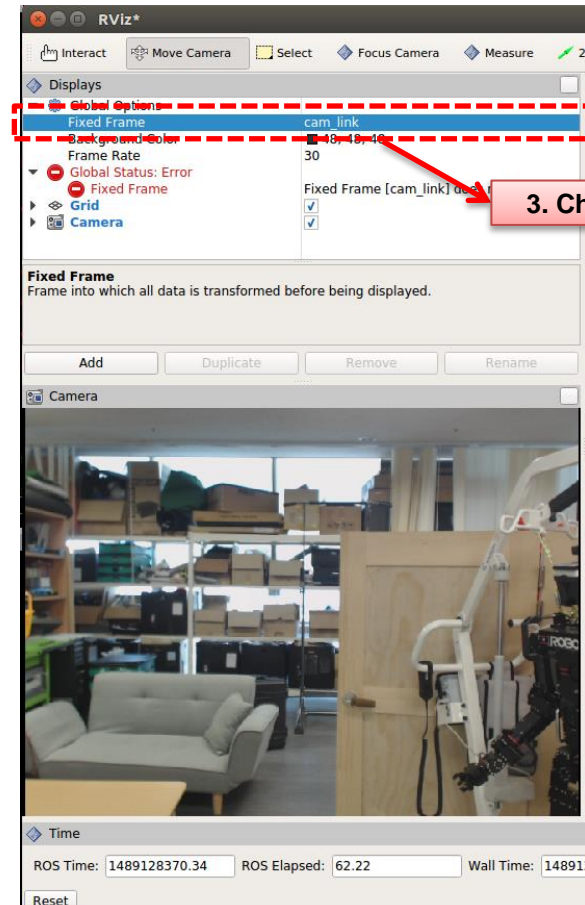
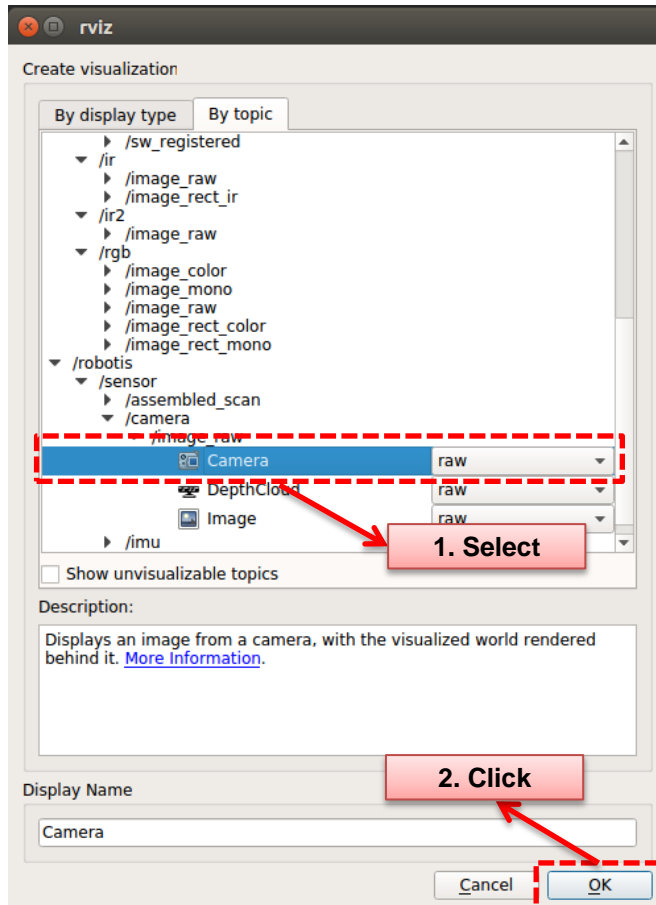




Check in visualization tool(Rviz)



- **Open visualization tool in OPC**
 - Add sensor topic and change 'Fixed Frame' to name of sensor frame





Check in visualization tool(Rviz)



- **Open visualization tool in OPC**
 - Topic name and frame id(for Fixed Frame)

Sensor	Topic name	Frame ID
HD Web Cam	/robotis/sensor/camera/image_raw	cam_link
Depth Camera	/realsense/depth_registered/points	realsense_rgb_optical_frame
LIDAR	/robotis/sensor/scan	lidar_link
FT Sensor	/robotis/sensor/ft_right_foot/raw	r_foot_ft_link
IMU	/robotis/sensor/imu/imu	imu_link



Check in visualization tool(Rviz)



- Open visualization tool in OPC

The screenshot displays the Rviz (Robot Visualization) interface. The main 3D view shows a robot model with several sensor data overlays:

- Depth Camera:** A green dashed circle highlights the depth camera's field of view, showing a point cloud of the environment.
- IMU:** A yellow dashed box highlights the Inertial Measurement Unit data, showing a purple vertical plane.
- Lidar:** A white dashed rectangle highlights the Lidar sensor's field of view, showing a point cloud of the environment.
- FT Sensor:** A red dashed circle highlights the Force/Torque sensor data, showing a red vertical line.
- Web Camera:** A blue dashed box highlights the web camera feed, showing a live video stream of the robot in a real-world environment.

The left sidebar contains the 'Displays' panel, which lists the following displays and their properties:

- WrenchStamped**
 - Status: Ok
 - Topic: /robotis/sensor/ft_right_foot/scaled
 - Force Color: 204; 51; 51
 - Torque Color: 204; 204; 51
 - Alpha: 1
 - Arrow Scale: 0.005
 - Arrow Width: 0.4
 - History Length: 10
- Imu**
 - Status: Ok
 - Topic: /robotis/sensor/imu/imu
 - Color: 204; 51; 204
 - Alpha: 0.2
 - History Length: 1

The bottom status bar shows the following information:

- Time: ROS Time: 1456231070.07, ROS Elapsed: 37.27, Wall Time: 1456231070.11, Wall Elapsed: 37.24
- Reset: Left-Click: Rotate. Middle-Click: Move X/Y. Right-Click/Mouse Wheel: Zoom. Shift: More options.
- Experimental: ☐ Experimental
- 30 fps