ROS2 Package for Intel® RealSense™ Devices

Supported Devices

* Intel® RealSense™ Camera D400-Series
  + Intel® RealSense™ Depth Cameras D415, D435 and D435i
* Intel® RealSense™ Tracking Camera T265

Installation Instructions

The following instructions were verified with ROS2 Foxy on Ubuntu 20.04. The software is verified with ROS2 Dashing on Ubuntu 18.04 also, with the same instructions replacing "foxy" with "dashing".

Install Dependencies

1. Install ROS2 packages [ros-foxy-desktop](https://index.ros.org/doc/ros2/Installation/Linux-Install-Debians/)

2. Install ROS2 dependences

Currently, we support librealsense master branch.

sudo apt-get install ros-foxy-librealsense2

* [Intel® RealSense™ SDK 2.0](https://github.com/IntelRealSense/librealsense.git)

3. Install other non-ROS debian packages

sudo apt-get install -y libssl-dev libusb-1.0-0-dev pkg-config libgtk-3-dev

sudo apt-get install -y libglfw3-dev libgl1-mesa-dev libglu1-mesa-dev

Install ros2\_intel\_realsense From Source

*# fetch code*

mkdir -p ~/ros2\_ws/src

cd ~/ros2\_ws/src

git clone https://github.com/intel/ros2\_intel\_realsense.git

cd ros2\_intel\_realsense

git checkout refactor

*# build*

source /opt/ros/foxy/setup.bash

cd ~/ros2\_ws

colcon build --symlink-install

Usage Instructions

Start Camera Node

Obtain the serial number of your device

rs-enumerate-devices

Change the corresponding yaml file with the specific serial number, e.g. for [d435.yaml](https://github.com/intel/ros2_intel_realsense/blob/refactor/realsense_ros/config/d435.yaml#L3) in line3:

serial\_no: # d435

To start the camera node in ROS2, plug in the camera, then type the following command:

Single camera, taking d435 for example:

source /opt/ros/foxy/setup.bash

source ~/ros2\_ws/install/local\_setup.bash

*## using ros2 run*

cd ~/ros2\_ws

ros2 run realsense\_node realsense\_node \_\_params:**=**`ros2 pkg prefix realsense\_examples`/share/realsense\_ros/config/d435.yaml \_\_ns:**=**/d435

*## using ros2 launch*

ros2 launch realsense\_examples rs\_camera.launch.py

Multi camera, taking d435 and t265 for example:

* Terminal 1:

source /opt/ros/foxy/setup.bash

source ~/ros2\_ws/install/local\_setup.bash

cd ~/ros2\_ws

ros2 run realsense\_node realsense\_node \_\_params:**=**`ros2 pkg prefix realsense\_examples`/share/realsense\_examples/config/d435.yaml \_\_ns:**=**/d435

* Terminal 2:

source /opt/ros/foxy/setup.bash

source ~/ros2\_ws/install/local\_setup.bash

cd ~/ros2\_ws

ros2 run realsense\_node realsense\_node \_\_params:**=**`ros2 pkg prefix realsense\_examples`/share/realsense\_examples/config/t265.yaml \_\_ns:**=**/t265

* Or using ros2 launch :

source /opt/ros/foxy/setup.bash

source ~/ros2\_ws/install/local\_setup.bash

cd ~/ros2\_ws

*## before launch all the cameras, serial number should be set in the launch file*

ros2 launch realsense\_examples rs\_multiple\_devices.launch.py

Multi camera with tf(transform), taking d435 and t265 for example:

source /opt/ros/foxy/setup.bash

source ~/ros2\_ws/install/local\_setup.bash

cd ~/ros2\_ws

*## before launch all the cameras, serial number should be set in the launch file*

ros2 launch realsense\_examples rs\_t265\_and\_d400.launch.py

***for more usage of these launch files in realsense\_examples/launch, please refer to our***[***robot devkit***](https://inte.github.io/robot_devkit)***project, includes SLAM and navigation etc.***

Configure Parameters at Runtime

Currently only support reconfigure parameters by ros2 param at runtime, e.g.

Enable camera stream

ros2 param list

ros2 param get <node\_name> color0.enabled

ros2 param set <node\_name> color0.enabled true

Enable aligned depth

ros2 param set <node\_name> align\_depth true

Enable pointcloud

ros2 param set <node\_name> enable\_pointcloud true

Contribution Instructions

Contirbution to this project is welcomed! For any patches pleaes make sure regression tests passed. For new features added please add test cases also.

Run Colcon test

*# plugin the Realsense device*

*# invoke colcon test*

colcon test --packages-select realsense\_msgs realsense\_node realsense\_ros realsense\_examples

*# check test logs*

vim log/latest\_test/<package name as `realsense\_xxx`**>**/stdout.log