

1.Environmental construction

This course takes Ubuntu 20.04+ros foxy as an example to illustrate how to build an environment using the DaiBai dcw2 camera in the ROS2 environment.

2.Install related dependency libraries

Input following command:

```
sudo apt install libgflags-dev nlohmann-json3-dev libgoogle-glog-dev ros-foxy-image-transport ros-foxy-image-publisher
```

Here, foxy is modified based on the actual version of ROS2 . If it is galactic, it can be replaced with galactic.

3.Compile Feature Pack

1) Create a workspace

To create a workspace named orbbec in the ~directory_ Taking ws as an example,

```
mkdir orbbec_ws  
cd orbbec_ws  
mkdir src
```

2)Copy feature packs to workspace

Unzip the file, copy and paste the folder (feature pack) under src into the ~/orbbec you just created_ Under the ws/src directory.

3)Compile

Input following command:

```
cd ~/orbbec_ws  
colcon build
```

4)Add environment variables

Input following command:

```
echo "source ~/orbbec_ws/install/setup.bash" >> ~/.bashrc
```

4.Install udev rules

Input following command:

```
cd ~/orbbec_ws/src/OrbbecSDK_ROS2/orbbec_camera/scripts  
sudo bash install_udev_rules.sh
```

After connecting the camera, enter the following command to check if the camera is detected.

```
lsusb
```

```
yahboom@VM:~$ lsusb
Bus 004 Device 001: ID 1d6b:0003 Linux Foundation 3.0 root hub
Bus 003 Device 010: ID 2bc5:0561 VIA Labs, Inc. USB Billboard Device
Bus 003 Device 011: ID 2bc5:06a0 VMware VMware Virtual USB Mouse
Bus 003 Device 005: ID 0e0f:0002 VMware, Inc. Virtual USB Hub
Bus 003 Device 004: ID 0e0f:0002 VMware, Inc. Virtual USB Hub
Bus 003 Device 007: ID 2109:8817 VIA Labs, Inc. USB Billboard Device
Bus 003 Device 002: ID 0e0f:0003 VMware, Inc. Virtual Mouse
Bus 003 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
Bus 002 Device 002: ID 0e0f:0002 VMware, Inc. Virtual USB Hub
Bus 002 Device 001: ID 1d6b:0001 Linux Foundation 1.1 root hub
yahboom@VM:~$
```

If the above image appears, it indicates that the device has successfully connected.

Then, enter the following command to check if the rule file was successfully loaded and the camera was bound.

```
ll /dev/dabai_*
```

```
yahboom@VM:~$ ll /dev/dabai_*
lrwxrwxrwx 1 root root 15 10月 18 14:57 /dev/dabai_dcw2 -> bus/usb/003/011
lrwxrwxrwx 1 root root 15 10月 18 14:57 /dev/dabai_dw2_rgb -> bus/usb/003/010
yahboom@VM:~$
```

If the above image appears, it indicates success.

5.Run the camera and view the image

Input following command:

```
ros2 launch orbbec_camera dabai_dcw2.launch.py
```

Input following command to view topic information:

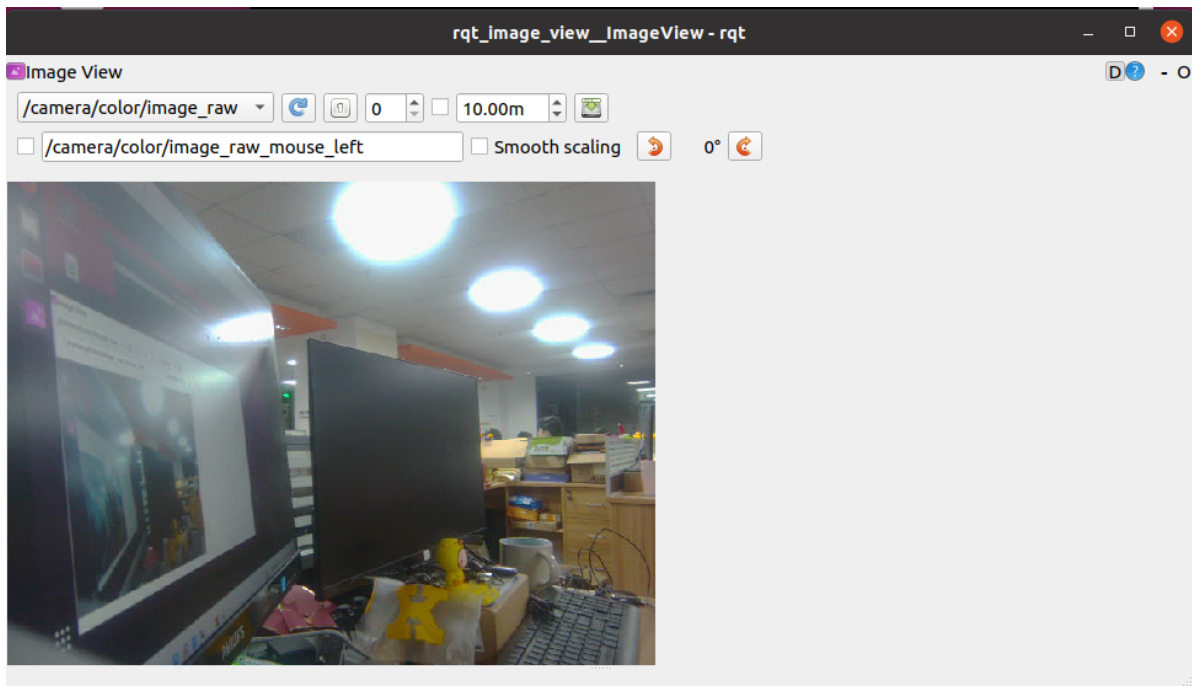
```
ros2 topic list
```

```
yahboom@VM:~/orbbec_ws$ ros2 topic list
/camera/color/camera_info
/camera/color/image_raw
/camera/depth/camera_info
/camera/depth/image_raw
/camera/depth/points
/camera/ir/camera_info
/camera/ir/image_raw
/parameter_events
/rosout
/tf
/tf_static
```

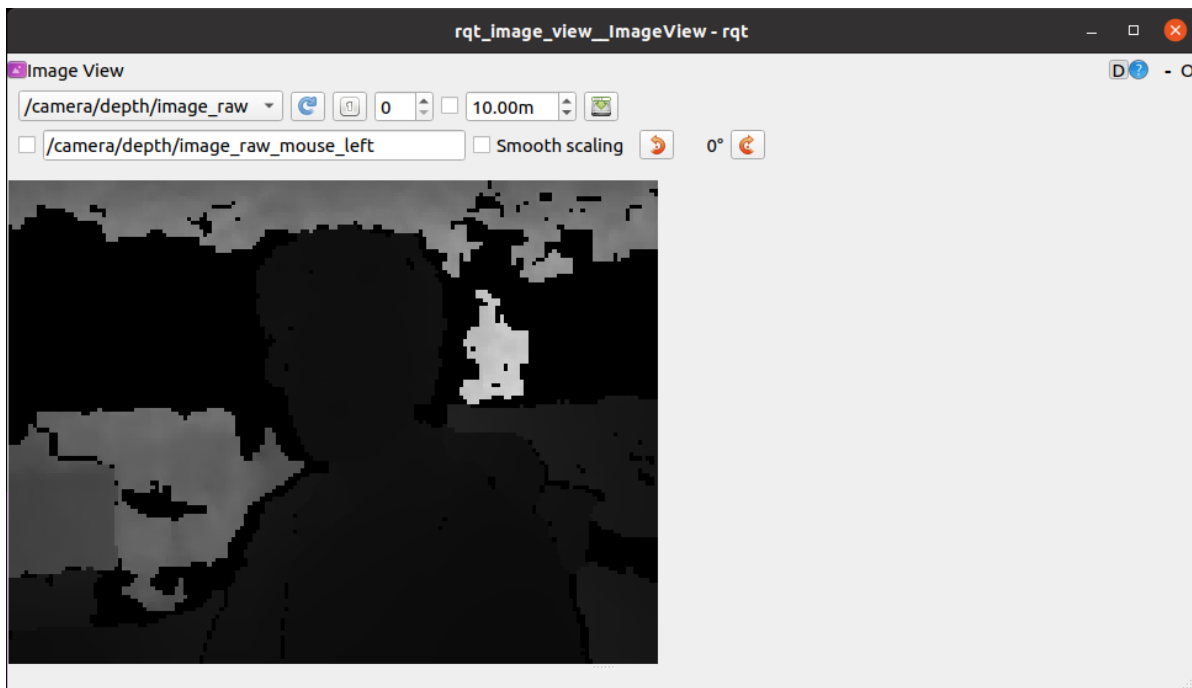
Using rqt_Image_View tool to view images. Input following command:

```
ros2 run rqt_image_view rqt_image_view
```

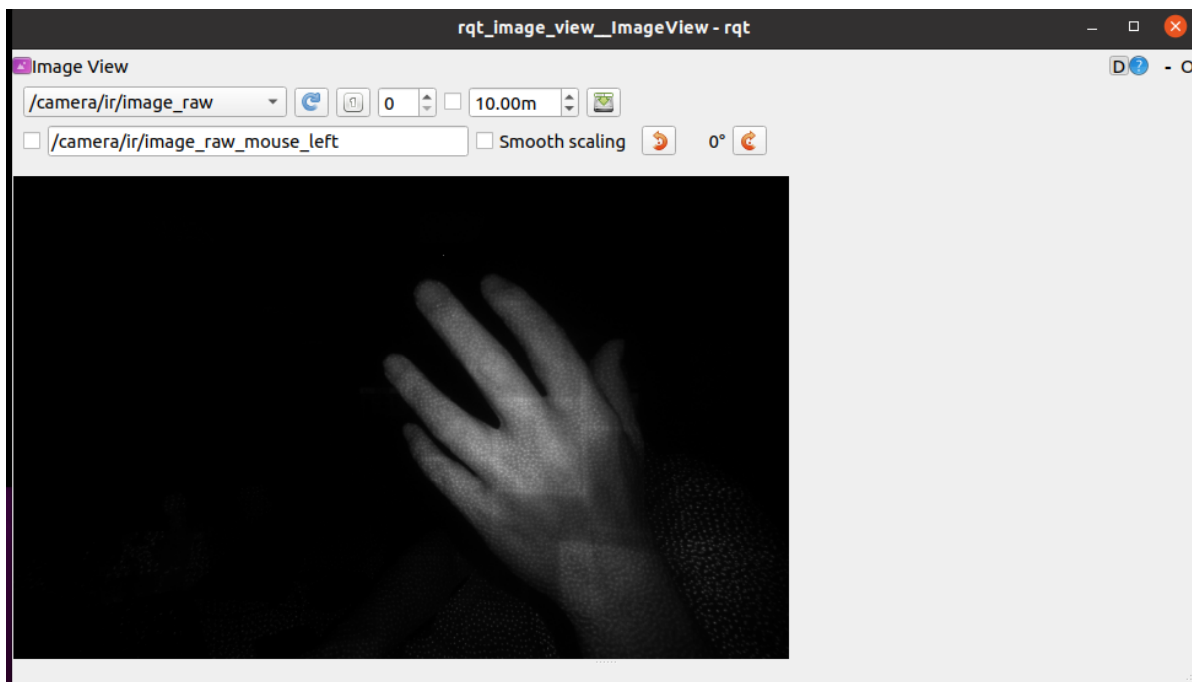
Color Map



Depth map



Infrared IR image



Select the topic in the upper left corner.

6. Run the camera and view the point cloud image

Input following command:

```
ros2 launch orbbec_camera dabai_dcw2.launch.py
```

You can see the point cloud data published by the camera in rviz.

Input following command:

```
rviz2
```

