1. Camera usage instructions

1. Error when using opencv and usb_cam to drive the camera

After using the launch command in the Orbbec_SDK function package to drive the camera, you will find that the camera cannot be driven by usb_cam, and the error shown in the figure below will be reported.

```
process[image_view-3]: started with pid [10766]
[ INFO] [1697529235.315517776]: Initializing nodelet with 4 worker threads.
[ INFO] [1697529235.405725041]: Using transport "raw"
[ INFO] [1697529235.425513328]: using default calibration URL
[ INFO] [1697529235.426148155]: camera calibration URL: file:///home/yahboom/.ro
s/camera_info/head_camera.yaml
[ WARN] [1697529235.426534453]: [head_camera] does not match name narrow_stereo
in file /home/yahboom/.ros/camera_info/head_camera.yaml
[ INFO] [1697529235.426583374]: Starting 'head_camera' (/dev/video0) at 640x480
via mmap (yuyv) at 30 FPS
[ERROR] [1697529235.426640067]: Cannot identify '/dev/video0': 2, No such file o
r directory
[usb_cam-2] process has died [pid 10765, exit code 1, cmd /opt/ros/noetic/lib/us
b_cam/usb_cam_node __name:=usb_cam __log:=/home/yahboom/.ros/log/51dd48e6-6cc2-1
1ee-b19d-65f5d1e636b5/usb_cam-2.log].
log file: /home/yahboom/.ros/log/51dd48e6-6cc2-11ee-b19d-65f5d1e636b5/usb_cam-2*
.log
```

Solution: Re-plug the camera, and then use usb_cam to open the camera normally, including using opencv to open the camera. If the above error occurs, re-plug it. As long as the camera is driven by launch in Orbbec_SDK first and before using opencv to drive the camera, you need to re-plug it.

2. Virtual machine case demonstration

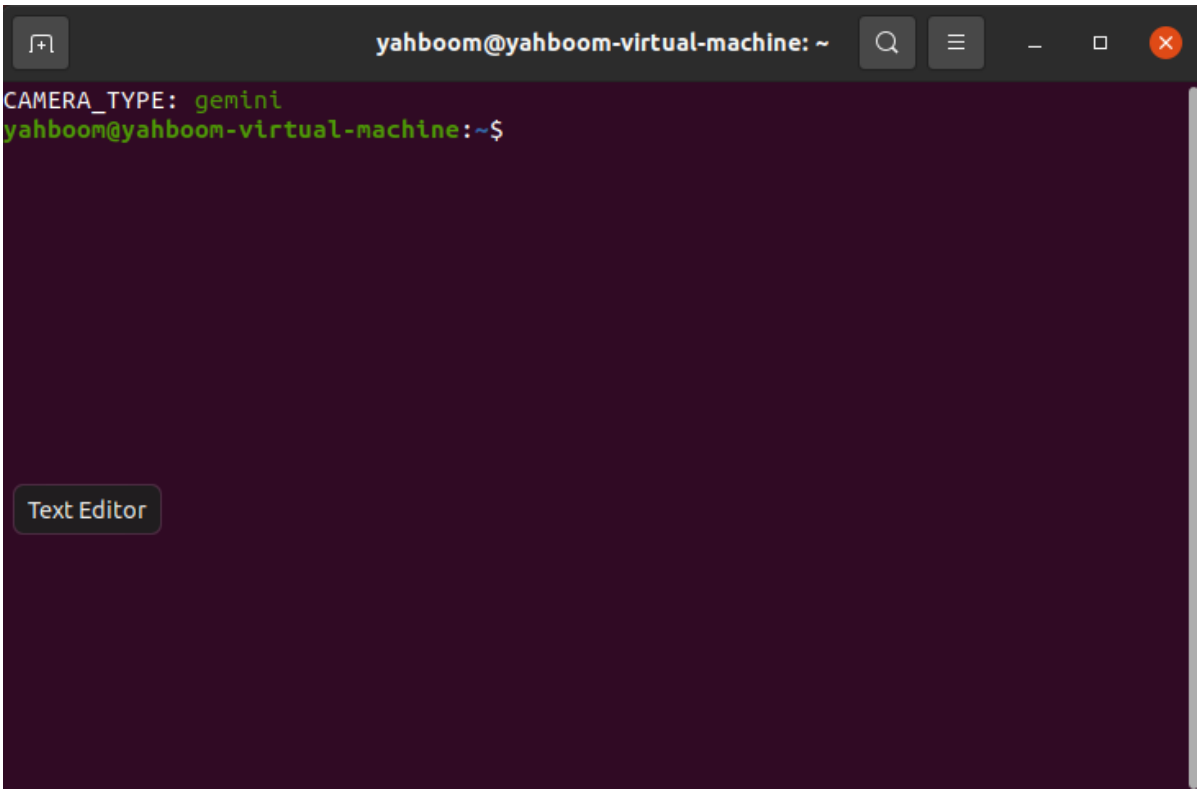
The cases in the virtual machine include SDKs for multiple cameras. Therefore, when matching the cases in the virtual machine experiment tutorial, you need to set the car camera file through the ~/.bashrc file. You need to set [CAMERA_TYPE] to the camera model you purchased. If you purchased a gemini camera, you need to change the value of [CAMERA_TYPE] to gemini. Input in the terminal, (astraproplus camera is set to astraproplus)

```
sudo gedit ~/.bashrc
```

140 lines, [CAMERA_TYPE] is set to gemini.

```
export CAMERA_TYPE=gemini
echo -e "CAMERA_TYPE: \033[32m$CAMERA_TYPE\033[0m"
source ~/ArTrack_ws/devel/setup.bash
source ~/orbbec_ws/devel/setup.bash
source ~/ros_ws/devel/setup.bash
```

Save and exit, restart a terminal, the terminal will print out the set camera type,



3. Source code description

We provide two sets of source code, one is the source code of the camera SDK only, and the other is the camera SDK and some demo codes in the tutorial.

orbbec_ws_src.tar.xz contains the function package for driving the camera and the function package for running demos with the virtual machine.

orbbec-ros-sdk.tar.xz contains only the function package for driving the camera.

opencv.zip contains only opencv related function packages