Use USB camera

The `usb_camera_fcos.py` program is installed on the development board to test the data path of the USB camera. This example will collect image data from the USB camera in real time, then run the target detection algorithm, and finally output the image data and detection results through the HDMI interface after fusion.

1. Environment preparation

- Connect the power supply
- Connect the USB camera to the development board and confirm that the `/dev/videoX` device node is generated. `X` represents a number, such as `/dev/video0`
- Connect the development board and the display via an HDMI cable
- Log in to RDK X5 via MobaXterm or other ssh tools

2. Run mode

Execute the program according to the following command

Turn off the desktop display

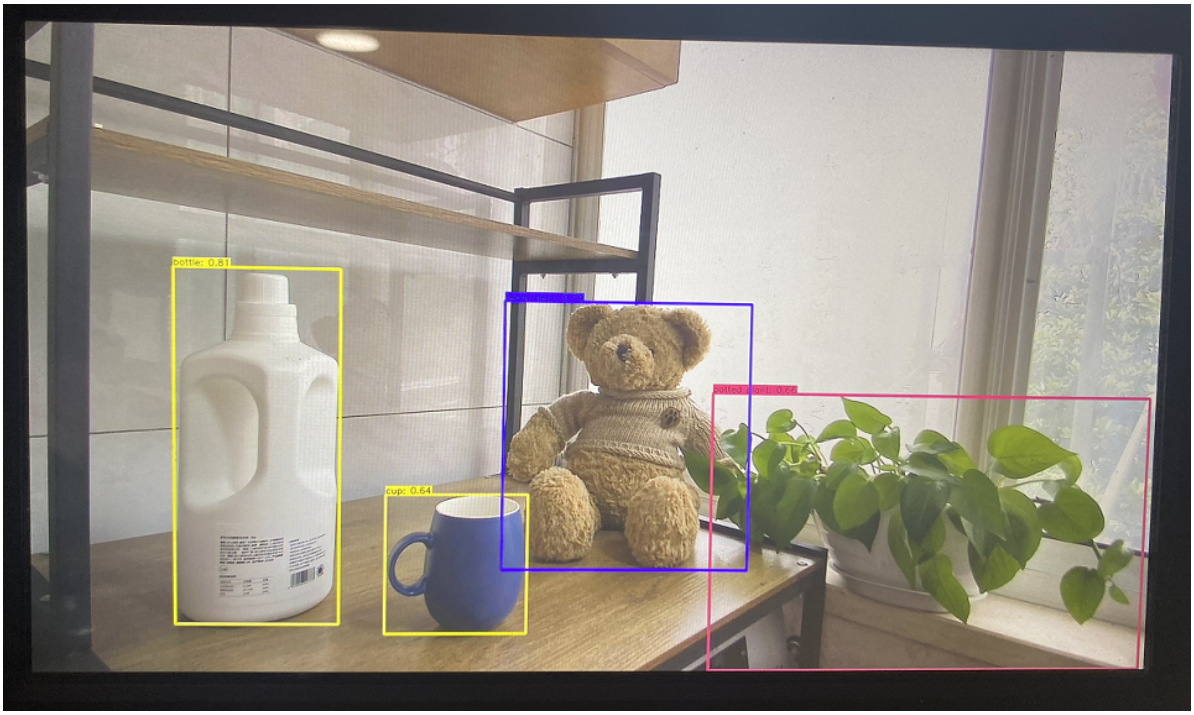
```
sudo systemctl stop lightdm
```

Run USB example

```
sunrise@ubuntu:~$ cd /app/pydev_demo/02_usb_camera_sample/
sunrise@ubuntu:/app/pydev_demo/02_usb_camera_sample$ sudo python3
./usb_camera_fcos.py
```

3. Expected results

After the program is executed, the display will show the camera image and the results of the target detection algorithm (target type, confidence) in real time, as shown below:



Note: When using the 02 and 03 examples of /app/pydev_demo of RDK X5, there is no screen display, only a black window.

(1) First, you need to perform apt upgrade on the premise that apt update has the sweet potato apt source, and upgrade all packages starting with hobot* to the latest version.

(2) The significance of this example is to directly output nv12 data from HDMI hardware through BT1120 instead of displaying it on the xfce desktop. The gray window is not produced by cv2.imshow.

(3) When running the Desktop system, if you run an example that requires an HDMI preview screen, you need to first execute sudo systemctl stop lightdm to close the graphical interface, otherwise an error will be reported and HDMI will not display the expected image.