

树莓派 opencv 摄像头操作

一、简介

树莓派自带多个 USB 口，我们可以可以外接 USB 摄像头。如果驱动支持，默认会在系统的 `/dev` 下，直接虚拟出来设备(`video0`、`video1...`)。树莓派 opencv 可以直接这个 `video0` 数据，进行视频显示、处理、录制保存等功能（程序运行过程中，因为程序锁定了虚拟视频设备，拔插摄像头后，这个序列号可能会改变，比如默认的 `video0`，会变成 `video1`）。小 R 科技树莓派镜像系统自带程序已经加入了对摄像头进行识别处理的功能：摄像头循迹。我们可以利用摄像头识别来做出更多的应用案例出来，比如人脸识别、图像识别、二维码识别等等。

接下来我们来学习一下 opencv 对摄像头的基本操作。

PS：因为我们出货固件默认开通了 `mjpeg-streamer` 功能，已经对摄像头进行了占用，所以我们在单独开发 opencv 时，需要手动结束主程序进程。

方法 1：在 `/home/pi/work/wifirobots/startwifirobots.sh` 脚本里面，把启动 `wifirobots.py` 的命令注释掉，重启就不会启动程序。如果需要恢复，就放开注释，重启即可。

方法 2：命令输入 `ps -aux` 显示所有进程，找到 `python wifirobots.py` 和 `mjpeg_streamerxxxxxx` 的进程，记住进程号，两个分别在 650 和 1300 左右。

使用指令 `sudo kill -9 ID1 ID2` 结束对应进程（注意空格）。

二、视频/摄像头的基本操作指令

1、创建 VideoCapture 对象

`cap = cv2.VideoCapture(0)` # 创建一个 VideoCapture 对象，0 是指摄像头

video0，也可以改成文件名来引入视频文件。

2、释放摄像头或视频文件

`cap.release()` #释放对象，再操作完成之后需要释放，否则其他程序无法再次获取摄像头或者视频文件。

3、读取一帧数据

`ret, frame = cap.read()` #capture frame_by_frame 一帧帧读取数据给 frame，并返回一个布尔值（ True/False）并赋值给 ret。如果帧读取的是正确的，就是 True。所以最后你可以通过检查他的返回值来查看视频文件是否已经到了结尾。有时 cap 可能不能成功的初始化摄像头设备。这种情况下上面的代码会报错。你可以使用 `cap.isOpened()`，来检查是否成功初始化了。如果返回值是 True，那就没有问题。否则就要使用函数 `cap.open()`。

4、获取、修改视频参数

`ret = cap.get(propId)` #获取视频对应 ID 的参数

`ret = cap.set(propId,value)` #修改视频对应 ID 的参数

Propid 对应的名称及含义如下表所示。例如，我可以使用 `cap.get(3)` 和 `cap.get(4)` 来查看每一帧的宽和高。默认情况下得到的值是 640X480 但是我可以使用 `ret=cap.set(3,320)` 和 `ret=cap.set(4,240)` 来把宽和高改成 320X240。

Propid	Name	Meanings
0	CV_CAP_PROP_POS_MSEC	Current position of the video file in milliseconds.
1	CV_CAP_PROP_POS_FRAMES	0-based index of the frame to be decoded/captured next.
2	CV_CAP_PROP_POS_AVI_RATIO	Relative position of the video file: 0 - start of the film, 1 - end of the film.
3	CV_CAP_PROP_FRAME_WIDTH	Width of the frames in the video stream.
4	CV_CAP_PROP_FRAME_HEIGHT	Height of the frames in the video stream.
5	CV_CAP_PROP_FPS	Frame rate.
6	CV_CAP_PROP_FOURCC	4-character code of codec.
7	CV_CAP_PROP_FRAME_COUNT	Number of frames in the video file.

8	CV_CAP_PROP_FORMAT	Format of the Mat objects returned by retrieve() .
9	CV_CAP_PROP_MODE	Backend-specific value indicating the current capture mode.
10	CV_CAP_PROP_BRIGHTNESS	Brightness of the image (only for cameras).
11	CV_CAP_PROP_CONTRAST	Contrast of the image (only for cameras).
12	CV_CAP_PROP_SATURATION	Saturation of the image (only for cameras).
13	CV_CAP_PROP_HUE Hue	of the image (only for cameras).
14	CV_CAP_PROP_GAIN	Gain of the image (only for cameras).
15	CV_CAP_PROP_EXPOSURE	Exposure (only for cameras).
16	CV_CAP_PROP_CONVERT_RGB	Boolean flags indicating whether images should be converted to RGB.
17	CV_CAP_PROP_WHITE_BALANCE	Currently unsupported
18	CV_CAP_PROP_RECTIFICATION	Rectification flag for stereo cameras (note: only supported by DC1394 v 2.x backend currently)

5、截取灰度图像

`gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)` #从帧图像 frame 中截取灰度图并赋值给 gray。

6、设置视频格式

`fourcc = cv2.cv.FOURCC(*'MJPG')` #设置编码格式为 MJPG 格式对应如下：

`cv.FOURCC('P','I','M','1')` = MPEG-1 codec
`cv.FOURCC('M','J','P','G')` = motion-jpeg codec (does not work well)
`cv.FOURCC('M', 'P', '4', '2')` = MPEG-4.2 codec
`cv.FOURCC('D', 'I', 'V', '3')` = MPEG-4.3 codec
`cv.FOURCC('D', 'I', 'V', 'X')` = MPEG-4 codec
`cv.FOURCC('U', '2', '6', '3')` = H263 codec
`cv.FOURCC('I', '2', '6', '3')` = H263I codec
`cv.FOURCC('F', 'L', 'V', '1')` = FLV1 codec

`out = cv2.VideoWriter('output.avi', fourcc, 20.0, (640, 480))` #定义输出属性：将一帧图像保存到名为 output.avi 的文件中。

`out.write(frame)` #将帧文件 frame 按照 out 属性保存为视频文件

7、旋转图像

`frame = cv2.flip(frame, 0)` #0:垂直反转；1:水平翻转；-1:水平垂直翻转

三、操作实例

我们接下来做两个窗口，分别显示摄像头灰度视频和彩色视频，并保存彩色

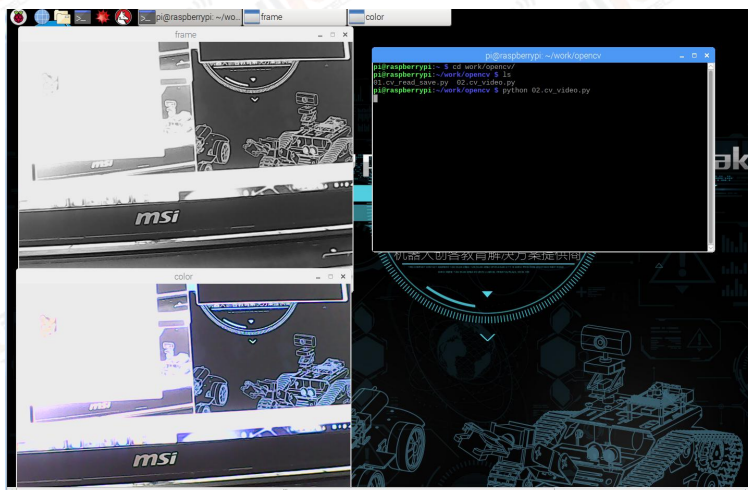
视频文件为 output.avi,

1、编写代码如下：

```
# -*- coding: utf-8 -*-
import cv2
import numpy as np
cap = cv2.VideoCapture(0)
fourcc = cv2.cv.FOURCC(*'WMV1')
out = cv2.VideoWriter('output.avi', fourcc, 20.0, (640, 480))
while (cap.isOpened()):
    ret, frame = cap.read() #capture frame_by_frame
    if ret == True:
        #frame = cv2.flip(frame, 0)
        gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY) #获取灰度视图
        out.write(frame) #保存彩色视图
        cv2.imshow('frame', gray) #显示灰度视图
        cv2.imshow('color', frame) #显示彩色视图
        if cv2.waitKey(1) & 0xFF == ord('q'):
            break
    else:
        break
cap.release()
out.release()
cv2.destroyAllWindows()
```

2、代码保存为 02.cv_video.py，并上传到上一章中建立的文件夹 /work/opencv 中。

3、远程桌面运行此文件，可以看到灰度图窗口和彩色图窗口。并且文件夹内也保存了 output.avi 的视频文件。





官网: www.xiao-r.com

论坛: www.wifi-robots.com

官方商城: wifi-robots.taobao.com

微信公众号:

