# 树莓派OpenCV开发环境搭建

目录

[树莓派OpenCV开发环境搭建 1](#_Toc25606174)

[概要 2](#_Toc25606175)

[1烧录系统 2](#_Toc25606176)

[2基本设置 2](#_Toc25606177)

[2.1远程桌面 2](#_Toc25606178)

[2.2传输文件 2](#_Toc25606179)

[2.3把目录扩展到整个TF卡 3](#_Toc25606180)

[2.4开启摄像头 4](#_Toc25606181)

[2.5更新软件 6](#_Toc25606182)

[3安装opencv需要的库 8](#_Toc25606183)

[3.1 安装numpy 8](#_Toc25606184)

[3.2 安装OpenCV所需的依赖 8](#_Toc25606185)

[4下载Opencv 8](#_Toc25606186)

[5.安装opencv 9](#_Toc25606187)

[5.1准备阶段 9](#_Toc25606188)

[5.2开始正式编译 10](#_Toc25606189)

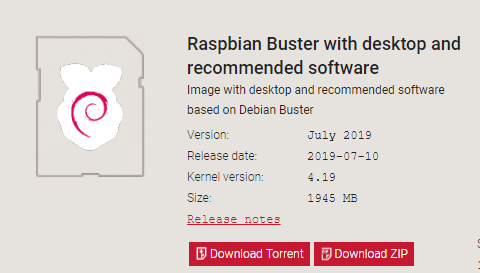
[6测试是否安装成功 11](#_Toc25606190)

## 概要

本章将介绍OpenCV开发环境的搭建，搭建的版本为opencv-3.4.2的Python3环境，在搭建好后，将通过一个Python的Demo打开CSI摄像头来测试开发环境是否搭建完成

## 1烧录系统

本项目程采用的镜像是树莓派官方2019年7月10日发布的发行版镜像，代号Buster，如下图所示：



## 2基本设置

### 2.1远程桌面

树莓派开机（已经经过联网设置，开机自动联网）

在树莓派终端安装xrdp服务器：$ sudo apt-get install xrdp

查看树莓派IP地址：$ ifconfig

打开win10电脑的远程桌面，快捷键为win+R，输入mstsc，在出现远程桌面连接弹窗中，输入树莓派IP地址，连接

在Login to xrdp中填写账号密码，默认用户名为pi，密码为raspberry

连接成功

### 2.2传输文件

电脑端安装FileZilla软件客户端

树莓派安装FTP Server：sudo apt-get install vsftpd

树莓派启动FTP服务：sudo service vsftpd start

电脑端启动FileZilla，填写树莓派IP地址，用户名和密码，选择快速连接

远程站点出现，连接成功

出现问题：可以用ftp协议连接pi，但无法用sftp协议连接，表现为连接超时。

原因：树莓派SSH默认开启反向解析DNS功能，遍历DNS服务器查域名费时，导致FileZilla连接超时

解决：关闭反向解析DNS。sudo nano /etc/ssh/sshd\_config  ->  在文件最后添加：UseDNS=no  ->   保存退出  ->   重启ssh服务：sudo service ssh restart

问题2

描述:电脑端无法上传文件，但可以从树莓派端下载文件

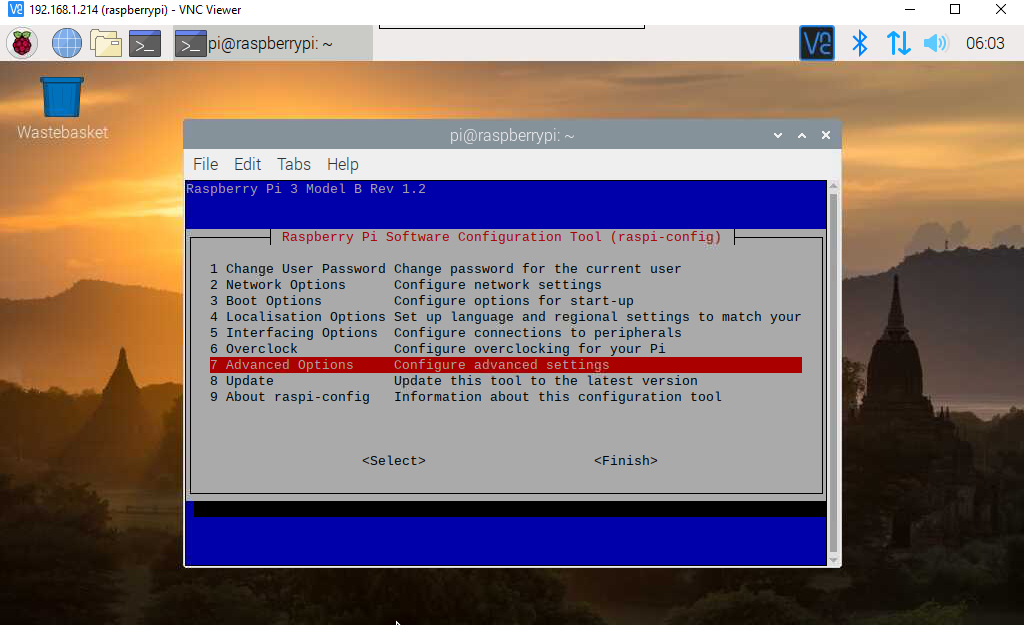
原因：vsftpd默认设置不允许上传文件

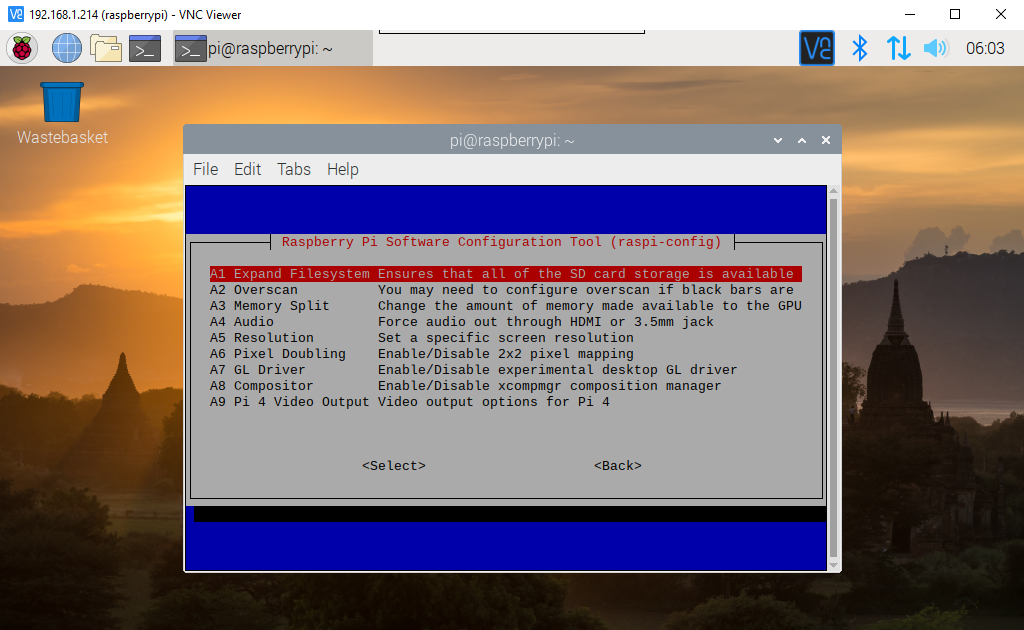
解决：修改配置文件：sudo nano /etc/vsftpd.conf  ->  将 write\_enable=YES 前面的#注释去掉，保存退出  ->  重启vsftpd服务器：service vsftpd restart

### 2.3把目录扩展到整个TF卡

本次搭建OpenCV开发环境，至少选择16G的TF卡，为了充分利用TF的存储空间，需要将TF卡空间扩展到整个TF卡。  
在命令行输入命令，进入树莓派配置界面，用上下左右切换光标位置。

sudo raspi-config

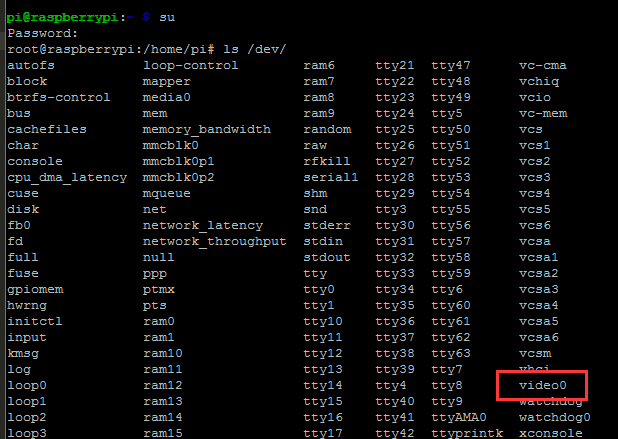




### 2.4开启摄像头

把摄像头的USB口接到树莓派后再开机。一定要先接摄像头再开机，不然树莓派无法识别。

打开终端命令，输入 ls /dev ,如果能够看到video0 就证明已经识别到摄像头模块。(个别情况可能看不到video0也能够使用摄像头)



安装fswebcam，在终端输入：sudo apt-get install fswebcam 提示安装就按Y，然后按回车。

安装mplayer，在终端输入：sudo apt-get install mplayer提示安装就按Y，然后按回车。

调用摄像头查看，在终端输入：sudo mplayer tv://

截取一张图片试试，终端输入

fswebcam -d /dev/video0 --no-banner -r 320x240 /home/pi/image.jpg

截取的图片需要到/home/pi/路径下查看

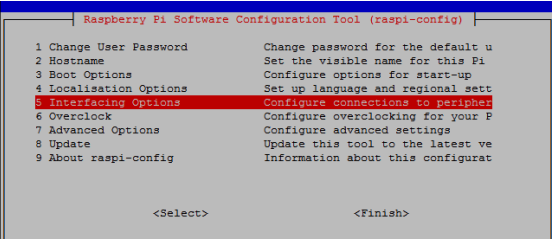
其中-d是配置使用哪个摄像设备，–no-banner应该是水印相关，不加的话，可能会报字体问题， -r后的是图片的宽与高，最后的是待保存的图片路径。

树莓派官方系统需要先使能摄像头驱动才可以，操作如下：

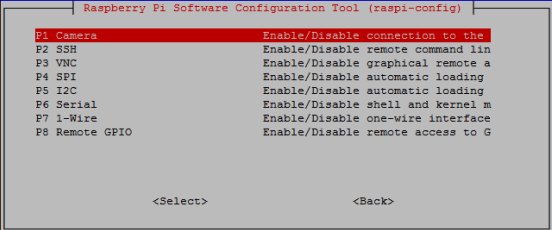
sudo raspi-config进入如下界面：

选择第5个，按enter键进入配置

点击 YES，接下来重启树莓派。



接下来选择第一个Camera：使能摄像头



## 2.5更新软件

鉴于国内网络环境下载各大镜像，软件包速度慢的问题，需要更换软件源，以防下载慢，且在本教程中，统一更换为清华源。

#### 2.5.1更新树莓派软件源

包括系统源与系统更新源，操作方法：

# 编辑 `/etc/apt/sources.list` 文件，

sudo nano /etc/apt/sources.list

删除原文件所有内容，用以下内容取代：

deb http://mirrors.tuna.tsinghua.edu.cn/raspbian/raspbian/ buster main non-free contrib

deb-src http://mirrors.tuna.tsinghua.edu.cn/raspbian/raspbian/ buster main non-free contrib

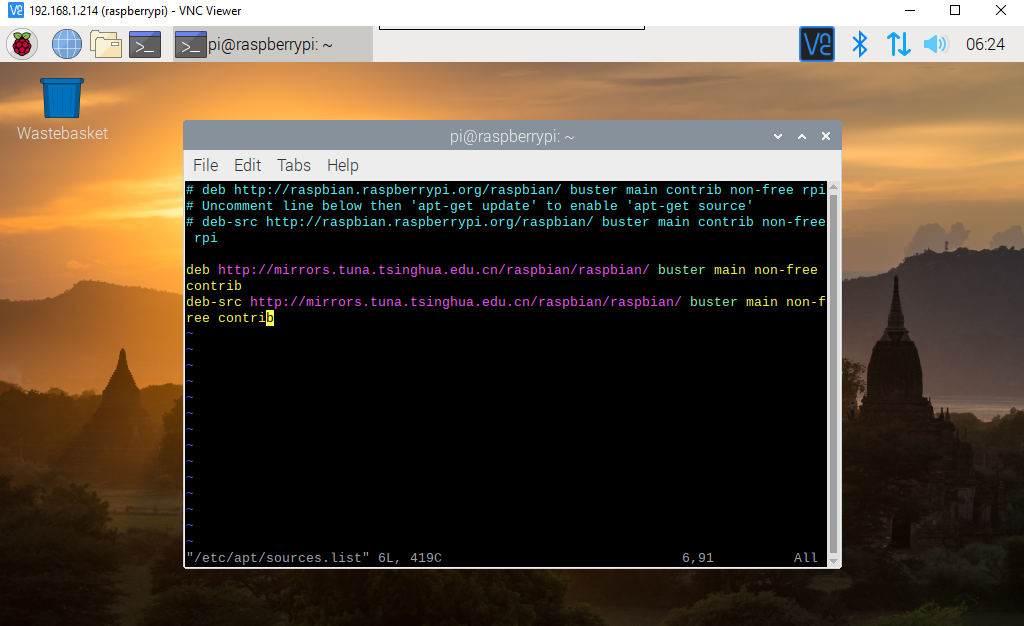
# 编辑 `/etc/apt/sources.list.d/raspi.list` 文件，

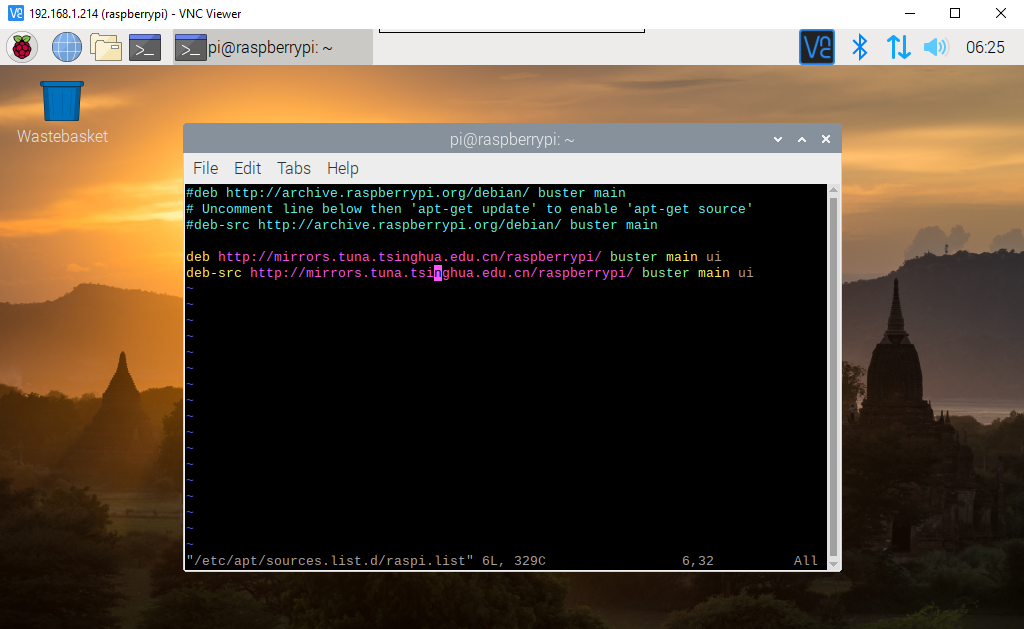
sudo nano /etc/apt/sources.list.d/raspi.list

删除原文件所有内容，用以下内容取代：

deb http://mirrors.tuna.tsinghua.edu.cn/raspberrypi/ buster main ui

deb-src http://mirrors.tuna.tsinghua.edu.cn/raspberrypi/ buster main ui

如下图所示：  




#### 2.5.2更换pip软件源

为加速Python pip安装速度，特更改Python软件源，操作方法：

pip install pip -U

pip config set global.index-url <https://pypi.tuna.tsinghua.edu.cn/simple>

#### 2.5.3重启

以上步骤均完成之后，需要进行重启才生效，重启命令如下：

sudo reboot

#### 2.5.4更新系统

//软件源更新

sudo apt-get update

//升级本地所有安装包，最新系统可以不升级，版本过高反而需要降级才能安装

sudo apt-get upgrade

//升级树莓派固件，固件比较新或者是Ubuntu则不用执行

sudo rpi-update

## 3安装opencv需要的库

## 3.1 安装numpy

sudo pip3 install numpy

## 3.2 安装OpenCV所需的依赖

sudo apt-get install build-essential git cmake pkg-config -y

sudo apt-get install libjpeg8-dev -y

sudo apt-get install libtiff5-dev -y

sudo apt-get install libjasper-dev -y

sudo apt-get install libpng12-dev -y

sudo apt-get install libavcodec-dev libavformat-dev libswscale-dev libv4l-dev -y

sudo apt-get install libgtk2.0-dev -y

sudo apt-get install libatlas-base-dev gfortran -y

## 4下载Opencv

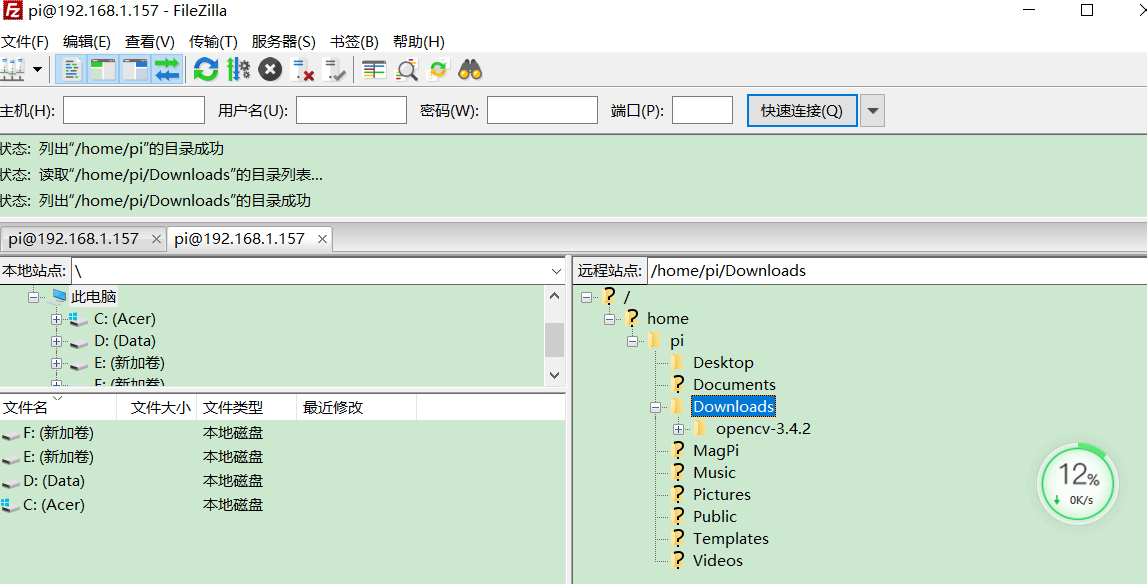
Opencv官网：<https://opencv.org/releases/>

下载相应版本,本次使用的是opencv-3.4.2，电脑现状易出现下载错误和下载速度太慢等问题，建议先用手机下载好后上传至电脑，电脑通过FileZilla传输至树莓派/home/pi/Download/目录下

操作如下：

终端输入：ifconfig查询树莓派IP地址，在FileZilla中输入相关信息，上传文件

结果如下图：



## 5.安装opencv

### 5.1准备阶段

1.打开opencv-3.4.2文件夹

cd /Download/ opencv-3.4.2

2. 新建release文件夹

mkdir release

3.进入release文件夹

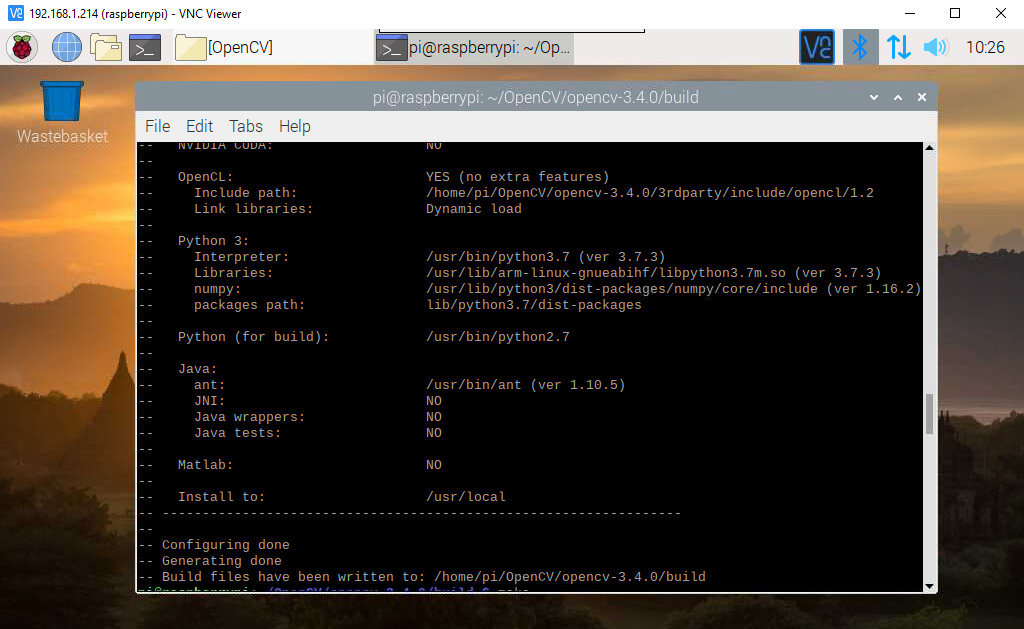
cd release

4.设置cmake编译参数，安装目录默认为/usr/local：

//CMAKE\_BUILD\_TYPE是编译方式，CMAKE\_INSTALL\_PREFIX是安装目录

cmake -D CMAKE\_BUILD\_TYPE=RELEASE \

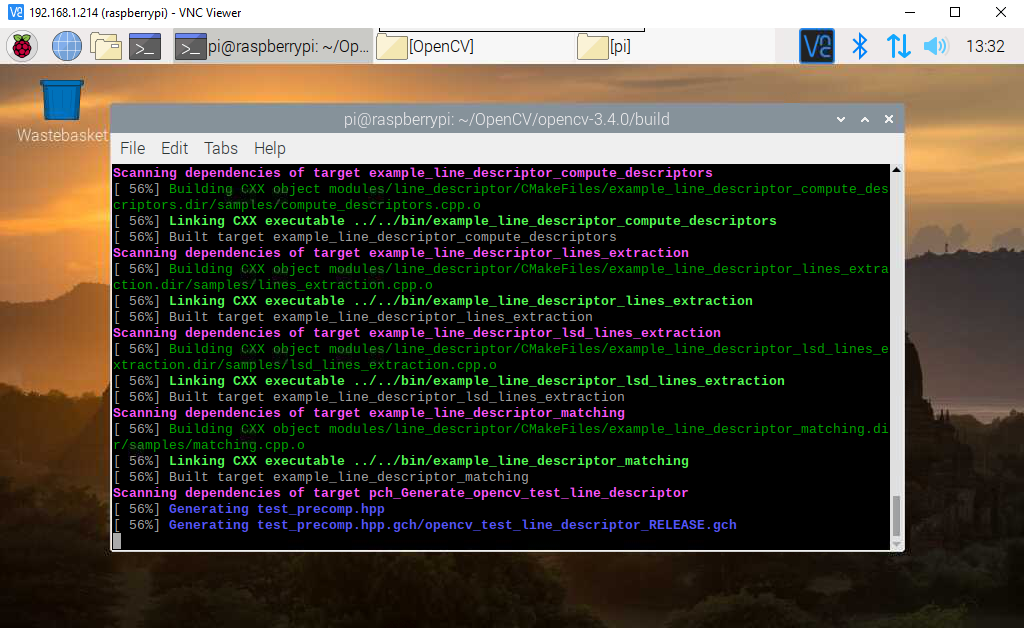
-D CMAKE\_INSTALL\_PREFIX=/usr/local ..



### 5.2开始正式编译

1编译

sudo make



2安装

sudo make install

3更新动态链接库

sudo ldconfig

## 6测试是否安装成功

运行python3如下程序：

import cv2 as cv #导入opencv———cv2 库命名为cv

import numpy as np #导入numpy库，命名为np

def video\_input(): #定义视频函数

capture = cv.VideoCapture(0) #打开摄像头 /参数是数字表示摄像头 也可以是视频路径

cv.namedWindow("video show",cv.WINDOW\_AUTOSIZE) #定义窗口 /以一个参数是窗口名称 //第二个参数是窗口形式

while(True): #循环 /无限循环

ret , frame = capture.read() #获取视频信息 \第一个返回值表示视频状态 \第二个返回值表示每一帧画面

if ret == False:

break

frame = np.flip(frame,1) #对画面进行颠倒 /画面信息 //1：左右，-1上下

cv.imshow("video show",frame) #展示图片信息

c = cv.waitKey(50)

if c == 27:

break

print("\*\*\*\*\*\*\*\*\*\*\*\*\*hello python\*\*\*\*\*\*\*\*\*\*\*\*\*\*")

video\_input()

cv.waitKey(0) #等待 /参数表示时间 0表示一直等待

cv.destroyAllWindows() #销毁所有窗口