

第 2 课 安装 OpenCV

文中所采用的系统是官方 7-10 发布的 buster，系统内已经安装了 Python3.7，这里就不提及怎么安装了。

1.修改树莓派下载源系统

树莓派默认的软件下载服务器是官方指定的，如果遇到下载文件比较慢或出错的情况，可以前往 [\4.拓展课程\3.拓展课程-树莓派主板基础课程\第 5 课 更换软件下载源的方法](#) 进行学习

2.安装 numpy

Numpy 概述：

每一张图像都有很多个像素点，这也导致了程序中会涉及大量的数组处理。Numpy 是一个 Python 的拓展库，它对多维数组的处理效率比 Python 自带的数组结构强很多，同时可以提高我们代码的可读性。

Numpy 因其强大的多维数组与矩阵计算能力，在机器学习领域被广泛应用。

安装 Numpy：

打开树莓派命令行界面，输入以下命令，安装 Python 科学计算库 numpy。

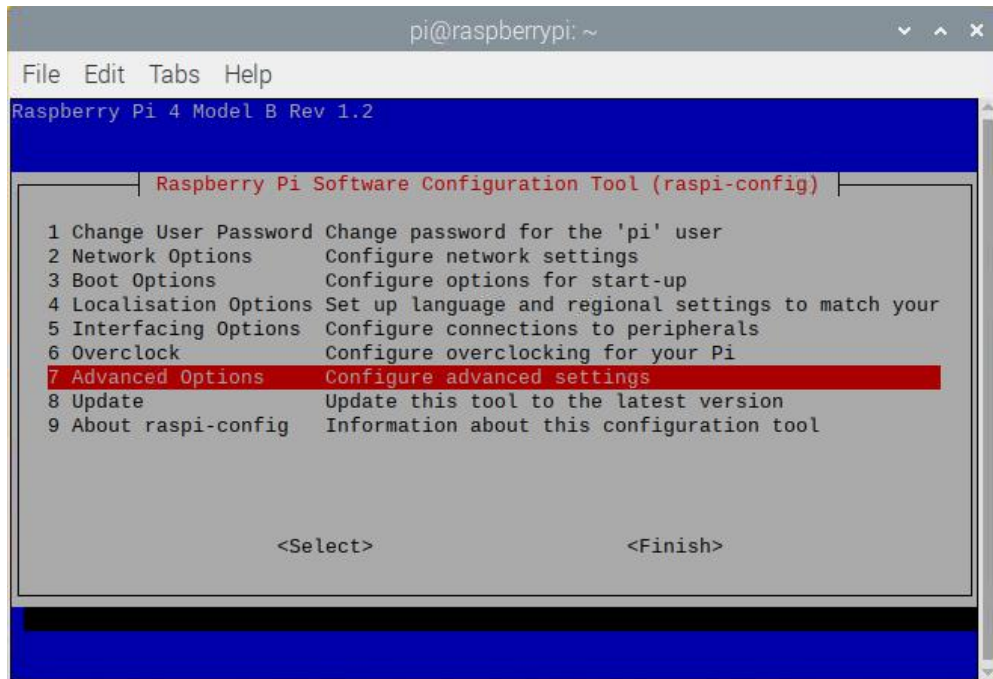
```
sudo pip3 install numpy
```

3.在树莓派设置中把根目录扩大到整个 SD 卡

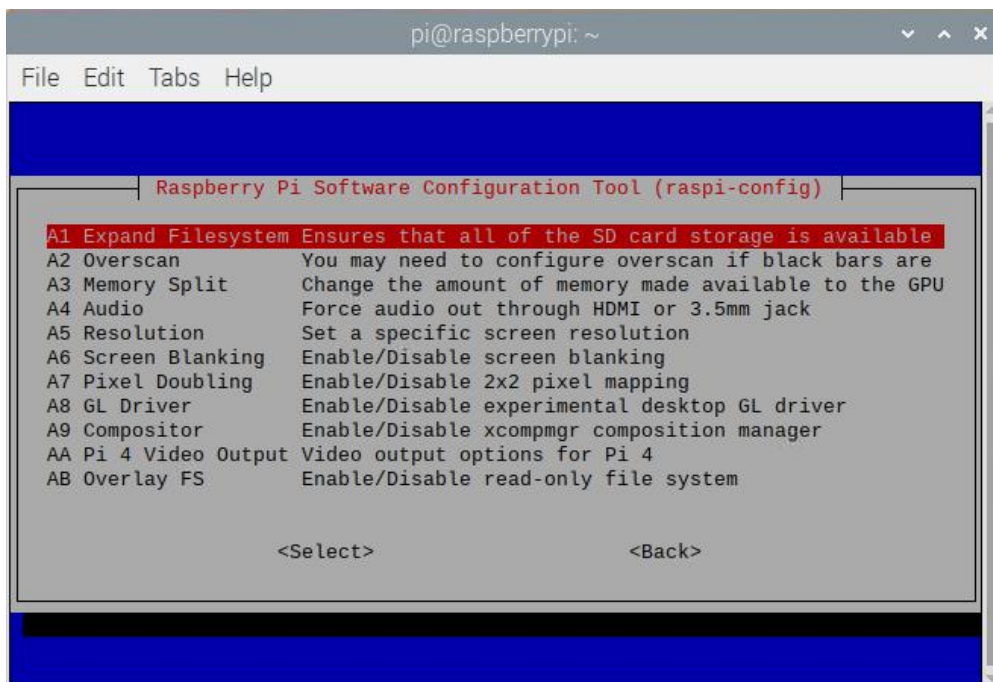
本次搭建 OpenCV 开发环境，至少选择 16G 的 TF 卡，需要充分利用 TF 的存储空间，需要将 TF 卡空间扩展到整个 TF 卡。

1) 命令行输入，出现下面树莓派配置画面，选择 7 Advanced Options:

```
sudo raspi-config
```



2) 按下 enter 键进入后，选择 A1 Expand Filesystem，按下 enter 确认选择，树莓派将自动执行并重启。



4. 安装 OpenCV 所需的库

安装 OpenCV 所需的依赖，依次运行下面八条命令。注意倒数第三条命令中要安装四个 -dev 软件包。

```
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
```

5. 下载 opencv

下载两个压缩包到树莓派的/home/pi/Downloads 目录下。

1) 在命令行输入以下命令，切换到 Downloads 目录：

```
cd /home/pi/Downloads
```

2) 下载第一个安装包：

```
wget https://github.com/Itseez/opencv/archive/3.4.0.zip
```

```
pi@raspberrypi: ~/Downloads
File Edit Tabs Help
pi@raspberrypi:~$ cd /home/pi/Downloads
pi@raspberrypi:~/Downloads$ ls
pi@raspberrypi:~/Downloads$ wget https://github.com/Itseez/opencv/archive/3.4.0.zip
--2020-09-14 17:09:56-- https://github.com/Itseez/opencv/archive/3.4.0.zip
Resolving github.com (github.com)... 13.229.188.59
Connecting to github.com (github.com)|13.229.188.59|:443... connected.
HTTP request sent, awaiting response... 301 Moved Permanently
Location: https://github.com/opencv/opencv/archive/3.4.0.zip [following]
--2020-09-14 17:09:57-- https://github.com/opencv/opencv/archive/3.4.0.zip
Reusing existing connection to github.com:443.
HTTP request sent, awaiting response... 302 Found
Location: https://codeload.github.com/opencv/opencv/zip/3.4.0 [following]
--2020-09-14 17:09:58-- https://codeload.github.com/opencv/opencv/zip/3.4.0
Resolving codeload.github.com (codeload.github.com)... 13.229.189.0
Connecting to codeload.github.com (codeload.github.com)|13.229.189.0|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: unspecified [application/zip]
Saving to: '3.4.0.zip'

3.4.0.zip          [          <=>          ] 10.59M  69.2KB/s   in 4m 58s

2020-09-14 17:14:58 (36.4 KB/s) - '3.4.0.zip' saved [11102206]

pi@raspberrypi:~/Downloads$ ls
3.4.0.zip
pi@raspberrypi:~/Downloads$
```

3) 下载第二个安装包:

```
wget https://github.com/Itseez/opencv_contrib/archive/3.4.0.zip
```

4) 下载之后, 把第一个压缩包重新命名为 opencv-3.4.0.zip, 把第二个压缩包重新命名为 opencv_contrib-3.4.0.zip

```
sudo mv 3.4.0.zip opencv-3.4.0.zip
sudo mv 3.4.0.zip.1 opencv_contrib-3.4.0.zip
```

```
pi@raspberrypi: ~/Downloads
File Edit Tabs Help
pi@raspberrypi:~$ cd Downloads/
pi@raspberrypi:~/Downloads$ ls
3.4.0.zip 3.4.0.zip.1
pi@raspberrypi:~/Downloads$ sudo mv 3.4.0.zip opencv-3.4.0.zip
pi@raspberrypi:~/Downloads$ ls
3.4.0.zip.1 opencv-3.4.0.zip
pi@raspberrypi:~/Downloads$ sudo mv 3.4.0.zip.1 opencv_contrib-3.4.0.zip
pi@raspberrypi:~/Downloads$ ls
opencv-3.4.0.zip opencv_contrib-3.4.0.zip
pi@raspberrypi:~/Downloads$
```

5) 解压这两个压缩包

```
cd /home/pi/Downloads
```

```
unzip opencv-3.4.0.zip
unzip opencv_contrib-3.4.0.zip
```

6. 设置编译参数

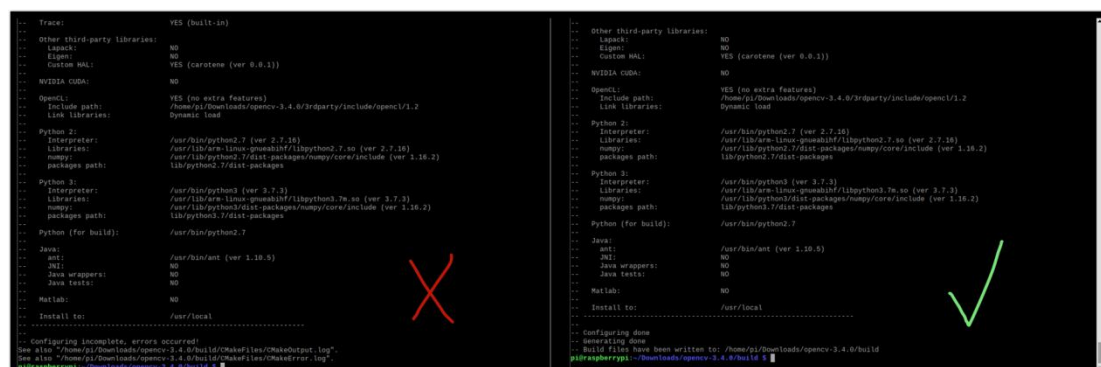
- 1) 输入下面命令行，创建 build 文件夹存放编译文件。

```
cd /home/pi/Downloads/opencv-3.4.0
mkdir build
cd build
```

- 2) 设置 CMAKE 参数，注意，下面这是一行命令（包括最后那两个点儿），需要耐心等待十五分钟左右：

```
cmake -D CMAKE_BUILD_TYPE=RELEASE -D CMAKE_INSTALL_PREFIX=/usr/local -D
INSTALL_C_EXAMPLES=ON -D INSTALL_PYTHON_EXAMPLES=ON -D OPENCV_EXTRA_MODU
LES_PATH=/home/pi/Downloads/opencv_contrib-3.4.3/modules -D BUILD_EXAMPL
ES=ON -D WITH_LIBV4L=ON PYTHON3_EXECUTABLE=/usr/bin/python3.7 PYTHON_INC
LUDE_DIR=/usr/include/python3.7 PYTHON_LIBRARY=/usr/lib/arm-linux-gnueab
ihf/libpython3.7m.so PYTHON3_NUMPY_INCLUDE_DIRS=/usr/lib/python3/dist-pa
ckages/numpy/core/include ..
```

- 3) 根据下图判断你是否配置成功了 CMAKE。



```
Trace:
--
-- Other third-party libraries:
--   Lapack: NO
--   Eigen: NO
--   Custom HAL: YES (carotene (ver 0.0.1))
--   WIDIA CUDA: NO
--   OpenCL: YES (no extra features)
--   Include path: /home/pi/Downloads/opencv-3.4.0/3rdparty/include/opencl/1.2
--   Link libraries: Dynamic load
--
-- Python 2:
--   Interpreter: /usr/bin/python2.7 (ver 2.7.18)
--   Libraries: /usr/lib/arm-linux-gnueabihf/libpython2.7.so (ver 2.7.18)
--   NumPy: /usr/lib/python2.7/dist-packages/numpy/core/include (ver 1.16.2)
--   packages path: lib/python2.7/dist-packages
--
-- Python 3:
--   Interpreter: /usr/bin/python3 (ver 3.7.3)
--   Libraries: /usr/lib/arm-linux-gnueabihf/libpython3.7m.so (ver 3.7.3)
--   NumPy: /usr/lib/python3/dist-packages/numpy/core/include (ver 1.16.2)
--   packages path: lib/python3.7/dist-packages
--
-- Python (for build): /usr/bin/python2.7
--
-- Java:
--   JNI: /usr/bin/ant (ver 1.10.5)
--   Java wrappers: NO
--   Java tests: NO
--
-- Matlab: NO
--
-- Install to: /usr/local
--
-----
Configuring incomplete, errors occurred!
See also "/home/pi/Downloads/opencv-3.4.0/build/CMakeFiles/CMakeOutput.log".
See also "/home/pi/Downloads/opencv-3.4.0/build/CMakeFiles/CMakeError.log".
CMakeError.log: /home/pi/Downloads/opencv-3.4.0/build/

Other third-party libraries:
--
--   Lapack: NO
--   Eigen: NO
--   Custom HAL: YES (carotene (ver 0.0.1))
--   WIDIA CUDA: NO
--   OpenCL: YES (no extra features)
--   Include path: /home/pi/Downloads/opencv-3.4.0/3rdparty/include/opencl/1.2
--   Link libraries: Dynamic load
--
-- Python 2:
--   Interpreter: /usr/bin/python2.7 (ver 2.7.18)
--   Libraries: /usr/lib/arm-linux-gnueabihf/libpython2.7.so (ver 2.7.18)
--   NumPy: /usr/lib/python2.7/dist-packages/numpy/core/include (ver 1.16.2)
--   packages path: lib/python2.7/dist-packages
--
-- Python 3:
--   Interpreter: /usr/bin/python3 (ver 3.7.3)
--   Libraries: /usr/lib/arm-linux-gnueabihf/libpython3.7m.so (ver 3.7.3)
--   NumPy: /usr/lib/python3/dist-packages/numpy/core/include (ver 1.16.2)
--   packages path: lib/python3.7/dist-packages
--
-- Python (for build): /usr/bin/python2.7
--
-- Java:
--   JNI: /usr/bin/ant (ver 1.10.5)
--   Java wrappers: NO
--   Java tests: NO
--
-- Matlab: NO
--
-- Install to: /usr/local
--
-----
Configuring done
Generating done
Build files have been written to: /home/pi/Downloads/opencv-3.4.0/build
#lsgraph.py: /home/pi/Downloads/opencv-3.4.0/build/
```

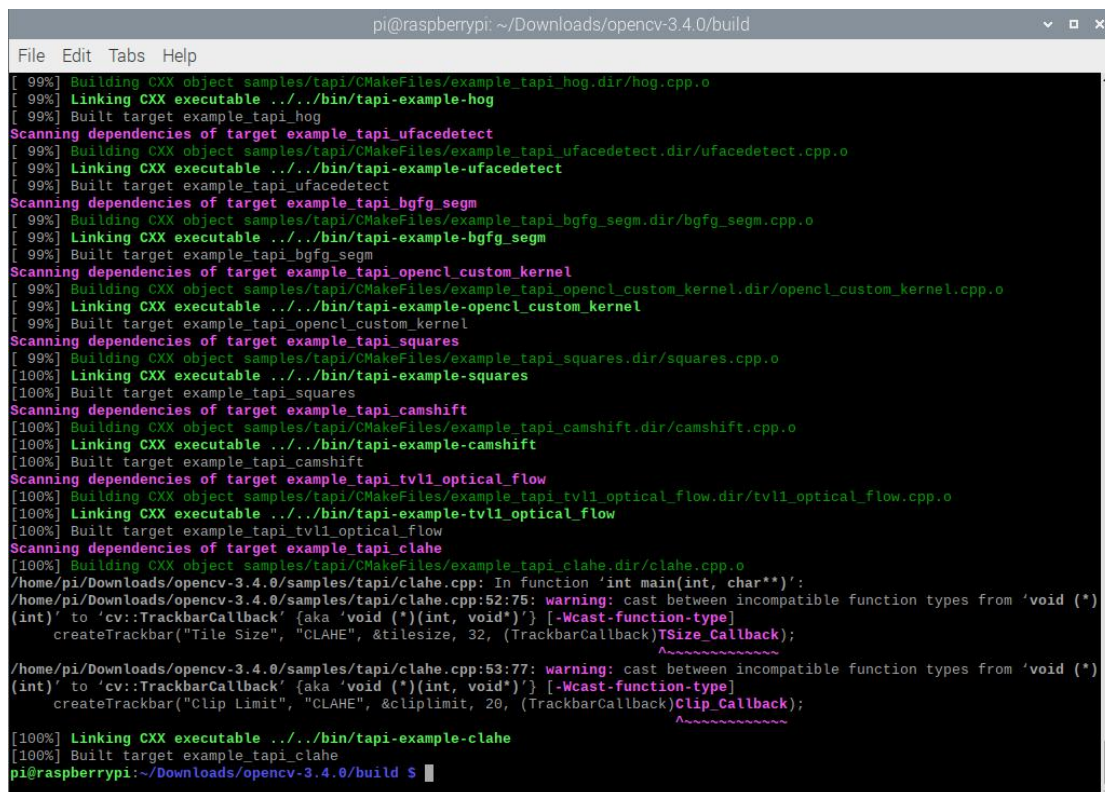
- 4) 如果失败，可能是因为配置里关于 opencv 与 python 的路径不对，需要根据自己的版本适当更改路径。如果成功，就可以开始最重要的编译了。

7.编译 OpenCV

1) 最后一步，也是最重要的一步：编译。保证树莓派有至少 5G 的存储空间，建议本命令用树莓派桌面上的命令行工具运行，而不要使用远程 ssh 连接。因为执行命令时间太长，中途如果 ssh 断线的话无法得知是否已经安装完毕。

```
cd /home/pi/Downloads/opencv-3.4.0/build  
make
```

2) 编译两个小时之后，完成了 40%。静待五个小时的编译。注意，在此期间，树莓派要供电充足，不要运行其它任务，以免因为内存不够报错。

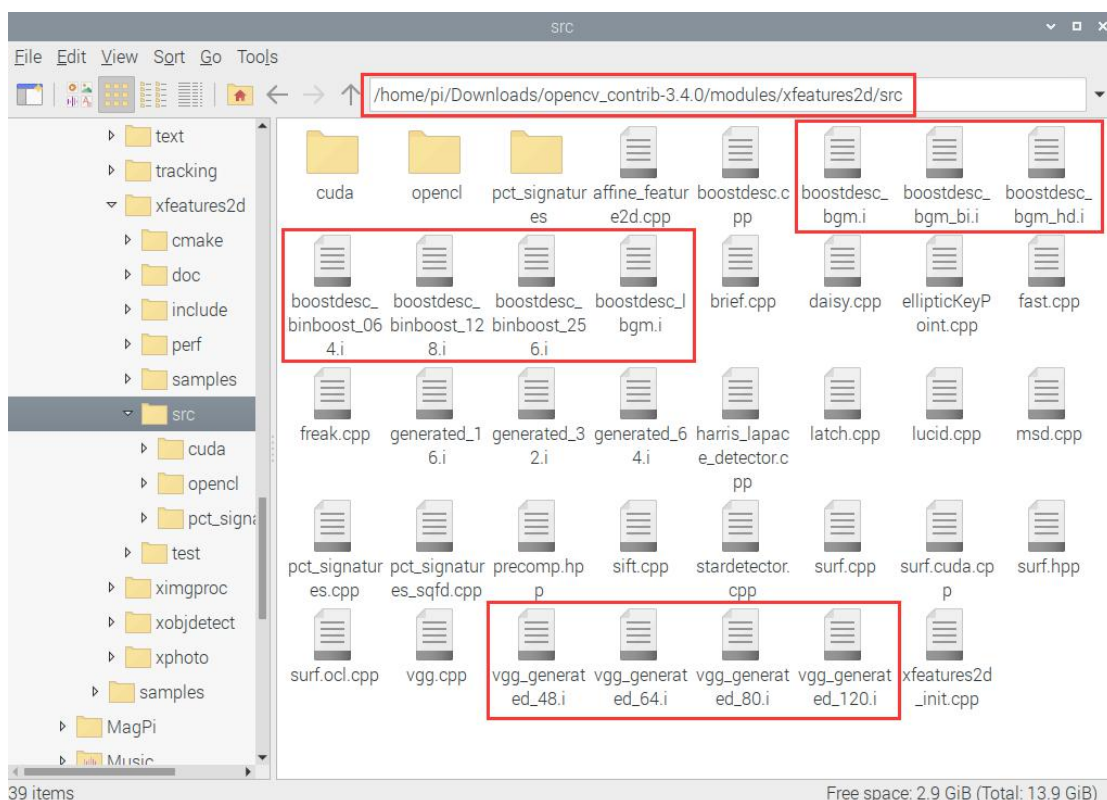


```
pi@raspberrypi: ~/Downloads/opencv-3.4.0/build  
File Edit Tabs Help  
[ 99%] Building CXX object samples/tapi/CMakeFiles/example_tapi_hog.dir/hog.cpp.o  
[ 99%] Linking CXX executable ../../bin/tapi-example-hog  
[ 99%] Built target example_tapi_hog  
Scanning dependencies of target example_tapi_ufaceDetect  
[ 99%] Building CXX object samples/tapi/CMakeFiles/example_tapi_ufaceDetect.dir/ufaceDetect.cpp.o  
[ 99%] Linking CXX executable ../../bin/tapi-example-ufaceDetect  
[ 99%] Built target example_tapi_ufaceDetect  
Scanning dependencies of target example_tapi_bgfg_segm  
[ 99%] Building CXX object samples/tapi/CMakeFiles/example_tapi_bgfg_segm.dir/bgfg_segm.cpp.o  
[ 99%] Linking CXX executable ../../bin/tapi-example-bgfg_segm  
[ 99%] Built target example_tapi_bgfg_segm  
Scanning dependencies of target example_tapi_opencv_custom_kernel  
[ 99%] Building CXX object samples/tapi/CMakeFiles/example_tapi_opencv_custom_kernel.dir/opencv_custom_kernel.cpp.o  
[ 99%] Linking CXX executable ../../bin/tapi-example-opencv-custom_kernel  
[ 99%] Built target example_tapi_opencv_custom_kernel  
Scanning dependencies of target example_tapi_squares  
[ 99%] Building CXX object samples/tapi/CMakeFiles/example_tapi_squares.dir/squares.cpp.o  
[100%] Linking CXX executable ../../bin/tapi-example-squares  
[100%] Built target example_tapi_squares  
Scanning dependencies of target example_tapi_camshift  
[100%] Building CXX object samples/tapi/CMakeFiles/example_tapi_camshift.dir/camshift.cpp.o  
[100%] Linking CXX executable ../../bin/tapi-example-camshift  
[100%] Built target example_tapi_camshift  
Scanning dependencies of target example_tapi_tvll_optical_flow  
[100%] Building CXX object samples/tapi/CMakeFiles/example_tapi_tvll_optical_flow.dir/tvll_optical_flow.cpp.o  
[100%] Linking CXX executable ../../bin/tapi-example-tvll_optical_flow  
[100%] Built target example_tapi_tvll_optical_flow  
Scanning dependencies of target example_tapi_clahe  
[100%] Building CXX object samples/tapi/CMakeFiles/example_tapi_clahe.dir/clahe.cpp.o  
/home/pi/Downloads/opencv-3.4.0/samples/tapi/clahe.cpp: In function 'int main(int, char**)':  
/home/pi/Downloads/opencv-3.4.0/samples/tapi/clahe.cpp:52:75: warning: cast between incompatible function types from 'void (*)  
(int)' to 'cv::TrackbarCallback' {aka 'void (*)(int, void*)'} [-Wcast-function-type]  
    createTrackbar("Title Size", "CLAHE", &titlesize, 32, (TrackbarCallback)TSize_Callback);  
                                                                    ~~~~~  
/home/pi/Downloads/opencv-3.4.0/samples/tapi/clahe.cpp:53:77: warning: cast between incompatible function types from 'void (*)  
(int)' to 'cv::TrackbarCallback' {aka 'void (*)(int, void*)'} [-Wcast-function-type]  
    createTrackbar("Clip Limit", "CLAHE", &cliplimit, 20, (TrackbarCallback)Clip_Callback);  
                                                                    ~~~~~  
[100%] Linking CXX executable ../../bin/tapi-example-clahe  
[100%] Built target example_tapi_clahe  
pi@raspberrypi:~/Downloads/opencv-3.4.0/build $
```

注：如果中途遇到类似下图的错误

```
[ 79%] Building CXX object modules/xfeatures2d/CMakeFiles/opencv_xfeatures2d.dir  
/src/boostdesc.cpp.o  
/home/pi/Downloads/opencv_contrib-3.4.0/modules/xfeatures2d/src/boostdesc.cpp:64  
6:20: fatal error: boostdesc_bgm.i: No such file or directory  
      #include "boostdesc_bgm.i"  
            ^~~~~~  
compilation terminated.
```

可以在当前文档所在的目录下将 `boostdesc_bgm.i....rar` 压缩包解压出来，将里面所有的文件拷贝到 `opencv_contrib/modules/xfeatures2d/src/` 目录下，



再重新编译一次。

如果遇到类似 “**fatal error:** opencv2/xfeatures2d/cuda.hpp: No such file or directory”

```
[ 96%] Building CXX object modules/stitching/CMakeFiles/opencv_stitching_pch_dep
help.dir/opencv_stitching_pch_dephelp.cxx.o
In file included from /home/pi/Downloads/opencv-3.4.0/modules/stitching/include/
opencv2/stitching.hpp:49,
                from /home/pi/Downloads/opencv-3.4.0/modules/stitching/src/prec
omp.hpp:59,
                from /home/pi/Downloads/opencv-3.4.0/build/modules/stitching/op
encv_stitching_pch_dephelp.cxx:1:
/home/pi/Downloads/opencv-3.4.0/modules/stitching/include/opencv2/stitching/deta
il/matchers.hpp:52:12: fatal error: opencv2/xfeatures2d/cuda.hpp: No such file o
r directory
# include "opencv2/xfeatures2d/cuda.hpp"
          ^~~~~~
compilation terminated.
make[2]: *** [modules/stitching/CMakeFiles/opencv_stitching_pch_dephelp.dir/buil
d.make:63: modules/stitching/CMakeFiles/opencv_stitching_pch_dephelp.dir/opencv_
stitching_pch_dephelp.cxx.o] Error 1
make[1]: *** [CMakeFiles/Makefile2:21309: modules/stitching/CMakeFiles/opencv_st
itching_pch_dephelp.dir/all] Error 2
make: *** [Makefile:163: all] Error 2
```

可以在根目录下执行查找文件的指令：

```
find . -name "cuda.hpp"
```

结果发现 cuda.hpp 文件的绝对路径位于：

```
/home/pi/Downloads/opencv_contrib-3.4.0/modules/xfeatures2d/include/opencv
2/xfeatures2d/cuda.hpp
```

所以，根据提示信息我们可以将

```
/home/pi/Downloads/opencv-3.4.0/modules/stitching/include/opencv2/stitchin
g/detail/matchers.hpp 文件中第 52 行的
```

```
#include "opencv2/xfeatures2d/cuda.hpp"
```

修改成以下绝对路径的形式

```
#include "/home/pi/Downloads/opencv_contrib-3.4.0/modules/xfeatures2d/include/opencv
2/xfeatures2d/cuda.hpp"
```

再重新编译

如果还出现类似以下的错误


```
/home/pi/Downloads/opencv-3.4.0/build/modules/python_bindings_generator/pyopencv_generated_ns_reg.h:2779:30: warning: cast between incompatible function types from 'PyObject*' (*)('PyObject*', 'PyObject*', 'PyObject*') {aka '_object*' (*)('object*', '_object*', '_object*')} to 'PyCFunction' {aka '_object*' (*)('object*', '_object*')} [-Wcast-function-type]
    {"inpaint", (PyCFunction)pyopencv_cv_xphoto_inpaint, METH_VARARGS | METH_KEYWORDS, "inpaint(src, mask, dst, algorithmType) -> None\n. @brief The function implements different single-image inpainting algorithms.\n. \n. See the original paper @cite He2012 for details.\n. \n. @param src source image, it could be of any type and any number of channels from 1 to 4. In case of\n. 3- and 4-channels images the function expect them in CIElab colorspace or similar one, where first\n. color component shows intensity, while second and third shows colors. Nonetheless you can try any\n. colorspaces.\n. @param mask mask (CV_8UC1), where non-zero pixels indicate valid image area, while zero pixels\n. indicate area to be inpainted\n. @param dst destination image\n. @param algorithmType see xphoto::InpaintTypes"},
    ~~~~~
make[2]: *** [modules/python3/CMakeFiles/opencv_python3.dir/build.make:63: modules/python3/CMakeFiles/opencv_python3.dir/__src2/cv2.cpp.o] Error 1
make[1]: *** [CMakeFiles/Makefile2:21633: modules/python3/CMakeFiles/opencv_python3.dir/all] Error 2
make: *** [Makefile:163: all] Error 2
```

可以在 `/home/pi/Downloads/opencv-3.4.0/modules/python/src2/cv2.cpp` 文件中的第 885 行附近（如下图所示），在 `"PyString_AsString(obj);"` 前加上 `"(char *)"`

```
879 template<>
880 bool pyopencv_to(PyObject* obj, String& value, const char* name)
881 {
882     (void)name;
883     if(!obj || obj == Py_None)
884         return true;
885     char* str = (char *)PyString_AsString(obj);
886     if(!str)
887         return false;
888     value = String(str);
889     return true;
890 }
891
892 template<>
893 bool pyopencv_to(PyObject* obj, Size& sz, const char* name)
894 {
895     (void)name;
896     if(!obj || obj == Py_None)
897         return true;
898     return PyArg_ParseTuple(obj, "ii", &sz.width, &sz.height) > 0;
899 }
```

保存修改后，再次编译。

3) `make` 编译命令执行完成之后，执行下面的安装命令，执行命令需要一分钟：

```
sudo make install
```

8.在 Python3 上测试 OpenCV

1) 安装好之后，在命令行中输入以下命令：

```
python3
import cv2
cv2.__version__
```

- 2) 如果出现下图的结果, 说明 Python3 环境下的 OpenCV 安装成功。

```
-- Installing: /usr/local/share/OpenCV/samples/python/floodfill.py
-- Installing: /usr/local/share/OpenCV/samples/python/gabor_threads.py
-- Installing: /usr/local/share/OpenCV/samples/python/gaussian_mix.py
-- Installing: /usr/local/share/OpenCV/samples/python/grabcut.py
-- Installing: /usr/local/share/OpenCV/samples/python/hist.py
-- Installing: /usr/local/share/OpenCV/samples/python/houghcircles.py
-- Installing: /usr/local/share/OpenCV/samples/python/houghlines.py
-- Installing: /usr/local/share/OpenCV/samples/python/inpaint.py
-- Installing: /usr/local/share/OpenCV/samples/python/kalman.py
-- Installing: /usr/local/share/OpenCV/samples/python/kmeans.py
-- Installing: /usr/local/share/OpenCV/samples/python/lappyr.py
-- Installing: /usr/local/share/OpenCV/samples/python/letter_recog.py
-- Installing: /usr/local/share/OpenCV/samples/python/lk_homography.py
-- Installing: /usr/local/share/OpenCV/samples/python/lk_track.py
-- Installing: /usr/local/share/OpenCV/samples/python/logpolar.py
-- Installing: /usr/local/share/OpenCV/samples/python/morphology.py
-- Installing: /usr/local/share/OpenCV/samples/python/mosse.py
-- Installing: /usr/local/share/OpenCV/samples/python/mouse_and_match.py
-- Installing: /usr/local/share/OpenCV/samples/python/mser.py
-- Installing: /usr/local/share/OpenCV/samples/python/opencv_version.py
-- Installing: /usr/local/share/OpenCV/samples/python/opt_flow.py
-- Installing: /usr/local/share/OpenCV/samples/python/peopledetect.py
-- Installing: /usr/local/share/OpenCV/samples/python/plane_ar.py
-- Installing: /usr/local/share/OpenCV/samples/python/plane_tracker.py
-- Installing: /usr/local/share/OpenCV/samples/python/squares.py
-- Installing: /usr/local/share/OpenCV/samples/python/stereo_match.py
-- Installing: /usr/local/share/OpenCV/samples/python/texture_flow.py
-- Installing: /usr/local/share/OpenCV/samples/python/tst_scene_render.py
-- Installing: /usr/local/share/OpenCV/samples/python/turing.py
-- Installing: /usr/local/share/OpenCV/samples/python/video.py
-- Installing: /usr/local/share/OpenCV/samples/python/video_threaded.py
-- Installing: /usr/local/share/OpenCV/samples/python/video_v4l2.py
-- Installing: /usr/local/share/OpenCV/samples/python/watershed.py
pi@raspberrypi:~/Downloads/opencv-3.4.0/build $ python3
Python 3.7.3 (default, Apr 3 2019, 05:39:12)
[GCC 8.2.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import cv2
>>> cv2.__version__
'3.4.0'
>>>
```