# Linux下安装OpenCV-CUDA加速，包括Opencv-Python

## 0 前言

注意在安装前，先安装好显卡驱动、CUDA以及CUDNN。

## 1 安装

### 1. 1安装相关依赖的包

[compiler]

sudo apt-get install build-essential

[required]

sudo apt-get install cmake git libgtk2.0-dev pkg-config libavcodec-dev libavformat-dev libswscale-dev

[optional]

sudo apt-get install python-dev python-numpy libtbb2 libtbb-dev libjpeg-dev libpng-dev libtiff-dev libjasper-dev libdc1394-22-dev

### 1.2 下载OpenCV以及OpenCV\_Contrib源代码

cd ~/<my\_working\_directory>

git clone https://github.com/opencv/opencv.git

git clone https://github.com/opencv/opencv\_contrib.git

### 2.3 编译和安装

#### 2.3.1 创建build文件夹

cd ~/opencv

mkdir build

cd build

#### 2.3.2 使用Cmake配置编译选项

sudo cmake -D CMAKE\_BUILD\_TYPE=RELEASE \

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

-D INSTALL\_C\_EXAMPLES=OFF \

-D WITH\_TBB=ON \

-D BUILD\_opencv\_cudacodec=ON \

-D ENABLE\_FAST\_MATH=1 \

-D CUDA\_FAST\_MATH=1 \

-D WITH\_CUDA=ON \

-D WITH\_CUBLAS=1 \

-D WITH\_FFMPEG=ON \

-D WITH\_V4L=ON \

-D WITH\_QT=OFF \

-D WITH\_OPENGL=ON \

-D WITH\_GSTREAMER=ON \

-D OPENCV\_GENERATE\_PKGCONFIG=ON \

-D OPENCV\_ENABLE\_NONFREE=ON \

-D OPENCV\_PYTHON3\_INSTALL\_PATH=/home/guiyan/miniconda3/envs/iis/lib/python3.6/site-packages \

-D WITH\_NVCUVID=ON \

-D BUILD\_opencv\_python3=yes \

-D OPENCV\_EXTRA\_MODULES\_PATH=~/opencv4/opencv\_contrib/modules \

-D PYTHON3\_EXECUTABLE=/home/guiyan/miniconda3/envs/iis/bin/python \

-D PYTHON\_INCLUDE\_DIR=/home/guiyan/miniconda3/envs/iis/include/python3.6m \

-D PYTHON3\_LIBRARY=/home/guiyan/miniconda3/envs/iis/lib/libpython3.6m.so \

-D PYTHON3\_NUMPY\_INCLUDE\_DIRS=/home/guiyan/miniconda3/envs/iis/lib/python3.6/site-packages/numpy/core/include \

-D BUILD\_EXAMPLES=OFF \

-D BUILD\_TESTS=OFF \

-D WITH\_CUDNN=ON \

-D OPENCV\_DNN\_CUDA=ON \

-D CUDA\_ARCH\_BIN=7.5 ..

注意：

1. 命令最后有两点
2. 编译CUDA加速的模块需要设置OpenCV\_Contrib的路径OPENCV\_EXTRA\_MODULES\_PATH
3. 设置不编译测试相关的代码以及例子BUILD\_TESTS=OFF、BUILD\_EXAMPLES=OFF
4. 设置CUDA加速 BUILD\_opencv\_cudacodec=ON、WITH\_CUDA=ON、WITH\_CUDNN=ON、OPENCV\_DNN\_CUDA=ON、CUDA\_ARCH\_BIN=7.5 这里的ARCH\_BIN需要对应到自己GPU的算力，可以根据自己的显卡型号在NVIDIA官网查看<https://developer.nvidia.com/cuda-gpus>
5. 如果需要OpenCV-Python也可以使用CUDA加速，可以设置python相关的路径，包括OPENCV\_PYTHON3\_INSTALL\_PATH、PYTHON3\_EXECUTABLE、PYTHON\_INCLUDE\_DIR、PYTHON3\_LIBRARY、PYTHON3\_NUMPY\_INCLUDE\_DIRS

#### 2.3.3 编译

make -j8

其中8是线程数，根据自己的机器CPU情况给定，也可直接make

#### 2.3.4 安装并更新动态库

注意：在安装前，为了以免opencv-python冲突，可以先将python环境下的opencv-python、opencv-contrib-python卸载

sudo make install

更新动态库：

sudo ldconfig

## 2 测试是否安装好

### opencv-python

import cv2 as cv

print(cv.cuda.getCudaEnabledDeviceCount() > 1)

输出True，则说明已配置好。

### 重新安装opencv-python

卸载之前的opencv-python

进入到build文件夹中的python\_loader文件夹下

执行命令：/home/Tom/miniconda3/envs/torch/bin/python setup.py install

以上这种情况有时候会有可能会有找不到cv2库的问题（在windows下通过将OpenCV官方下载的库文件<https://github.com/opencv/opencv/releases> 的build文件夹中python/cv2目录下的文件复制到python环境site-packages中的cv2文件夹即可），可以进去到build文件夹下重新用cmake安装一下，见2.3.4

## 3. 相关参考

1) Ubuntu上安装opencv-cuda加速 - <https://blog.csdn.net/tracelessle/article/details/108209710>

2) python+opencv+gpu加速 - 爱梅达斯的文章 - 知乎

https://zhuanlan.zhihu.com/p/345439669