

## NVIDIA GPUs, CUDA, cuDNN and OpenCV

2021年2月17日 22:11

### Step 1 To see the recommended drivers

```
$ ubuntu-drivers devices or $ sudo lshw -C display
```

### Step 2 To install NVIDIA drivers

```
$ sudo apt install nvidia-driver-460
```

### Step 3 To install CUDA toolkit

```
$ sudo sh cuda_11.2.1_460.32.03_linux.run --silent --override --toolkit
```

### Step 4 Add CUDA to bashrc environment

```
$ gedit ~/.bashrc
export PATH=$PATH:/usr/local/cuda-11.2/bin
export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/usr/local/cuda-11.2/lib64
$ source ~/.bashrc
```

### Step 5 Verify NVIDIA & CUDA Toolkit version

```
$ nvidia-smi
$ nvcc -V
```

### Step 6 To install cuDNN

```
$ tar -xzf cudnn-11.1-linux-x64-v8.0.5.39.tgz
$ cd cuda
$ sudo cp -P lib64/* /usr/local/cuda/lib64/
$ sudo cp -P include/* /usr/local/cuda/include/
```

### Step 7 OpenCV and "dnn" GPU dependencies

```
$ sudo apt-get update
$ sudo apt-get upgrade
$ sudo apt-get install build-essential cmake unzip pkg-config
$ sudo apt-get install libjpeg-dev libpng-dev libtiff-dev
$ sudo apt-get install libavcodec-dev libavformat-dev libswscale-dev
$ sudo apt-get install libv4l-dev libxvidcore-dev libx264-dev
$ sudo apt-get install libgtk-3-dev
$ sudo apt-get install libatlas-base-dev gfortran
$ sudo apt-get install python3-dev
```

### Step 8 Create conda environment

```
$ conda create -n gpu_opencv_4_5_1 python=3.8
$ conda activate gpu_opencv_4_5_1
$ conda install numpy
```

### Step 9 Install OpenCV for NVIDIA GPU support

```
$ wget https://github.com/opencv/opencv/archive/3.4.5.zip -O opencv.zip
$ wget https://github.com/opencv/opencv_contrib/archive/3.4.5.zip -O opencv_contrib.zip
$ unzip opencv.zip
$ unzip opencv_contrib.zip
$ cd opencv-4.5.1/
$ mkdir build
$ cd build
$ cmake -D CMAKE_BUILD_TYPE=RELEASE \
-D CMAKE_INSTALL_PREFIX=/usr/local \
-D WITH_CUDA=ON \
-D ENABLE_FAST_MATH=1 \
-D CUDA_FAST_MATH=1 \
-D WITH_CUBLAS=1 \
-D INSTALL_PYTHON_EXAMPLES=ON \
-D OPENCV_EXTRA_MODULES_PATH=/home/dragon/Systems/cv2_4.5.1/opencv_contrib-4.5.1/modules \
-D PYTHON_EXECUTABLE=/home/dragon/miniconda3/envs/gpu_opencv_4_5_1/bin/python \
-D PYTHON_DEFAULT_EXECUTABLE=/home/dragon/miniconda3/envs/gpu_opencv_4_5_1/bin/python3.8 \
-D PYTHON_INCLUDE_DIRS=/home/dragon/miniconda3/envs/gpu_opencv_4_5_1/include/python3.8 \
```

```
-D PYTHON_LIBRARY=/home/dragon/miniconda3/envs/gpu_opencv_4_5_1/lib \
-D BUILD_EXAMPLES=ON ..
$ make -j8
$ sudo make install
$ sudo ldconfig
```

Note:

```
$ echo `which python` # PYTHON_EXECUTABLE
$ echo `python -c "from distutils.sysconfig import get_python_inc; print(get_python_inc())"` # PYTHON_INCLUDE_DIRS
$ echo `python -c "import distutils.sysconfig as sysconfig; print(sysconfig.get_config_var('LIBDIR'))"` # PYTHON_LIBRARY
$ echo `which python3.8` # PYTHON_DEFAULT_EXECUTABLE
# OPENCV_EXTRA_MODULES_PATH opencv_contrib-4.5.1/modules
```

References

1. How to use OpenCV's "dnn" module with NVIDIA GPUs, CUDA, and cuDNN (PyImageSearch)
2. Install OpenCV with CUDA for Conda (DanielHavir)

