安装ubuntu16+nvidia gtx1060 驱动+cuda8.0+cudnn5.1+protobuf2.6.1+opencv2.4.13+

anaconda2+matlab2016b+caffe

1、安装ubuntu

/boot 1G 固态

/交换空间 15G 固态

/ 根目录 剩余固态空间

/home 机械硬盘空间600G

开机后，sudo apt install vim 安装vim

sudo vim /etc/default/grub 修改GRUB\_GFXMODE=1920×1080

重启

系统设置-》软件与更新-》使用中科大的源-》关闭 确认更新

terminal中

cd /etc/apt/   
 sudo cp sources.list sources.list.bak   
 sudo vi sources.list

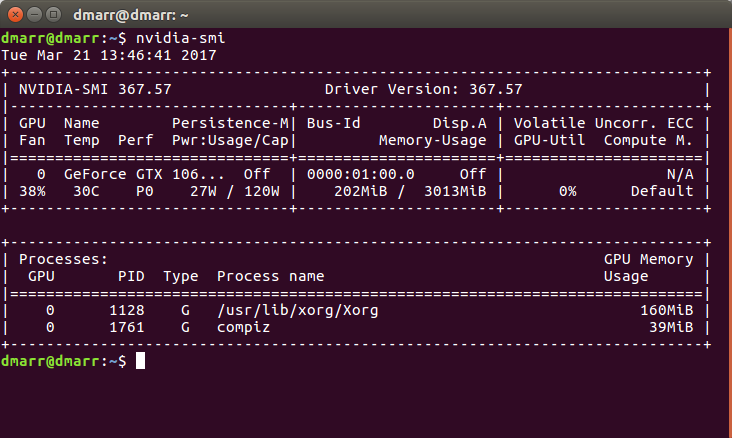
更新Ubuntu16.04源，终端输入   
cd /etc/apt/   
sudo cp sources.list sources.list.bak   
sudo vi sources.list   
把下面的这些源添加到source.list中：   
deb http://mirrors.ustc.edu.cn/ubuntu/ xenial main restricted universe multiverse   
deb http://mirrors.ustc.edu.cn/ubuntu/ xenial-security main restricted universe multiverse   
deb http://mirrors.ustc.edu.cn/ubuntu/ xenial-updates main restricted universe multiverse   
deb http://mirrors.ustc.edu.cn/ubuntu/ xenial-proposed main restricted universe multiverse   
deb http://mirrors.ustc.edu.cn/ubuntu/ xenial-backports main restricted universe multiverse   
deb-src http://mirrors.ustc.edu.cn/ubuntu/ xenial main restricted universe multiverse   
deb-src http://mirrors.ustc.edu.cn/ubuntu/ xenial-security main restricted universe multiverse   
deb-src http://mirrors.ustc.edu.cn/ubuntu/ xenial-updates main restricted universe multiverse   
deb-src http://mirrors.ustc.edu.cn/ubuntu/ xenial-proposed main restricted universe multiverse   
deb-src http://mirrors.ustc.edu.cn/ubuntu/ xenial-backports main restricted universe multiverse   
最后更新源和更新已安装的包：   
terminal输入   
sudo apt-get update   
sudo apt-get upgrade

2、安装nvidia GPU驱动

这里安装nvidia 367.57 驱动版本，

sudo apt-get install nvidia-367   
 sudo apt-get install mesa-common-dev   
 sudo apt-get install freeglut3-dev

重启之后，terminal输入 nvidia-smi得到：

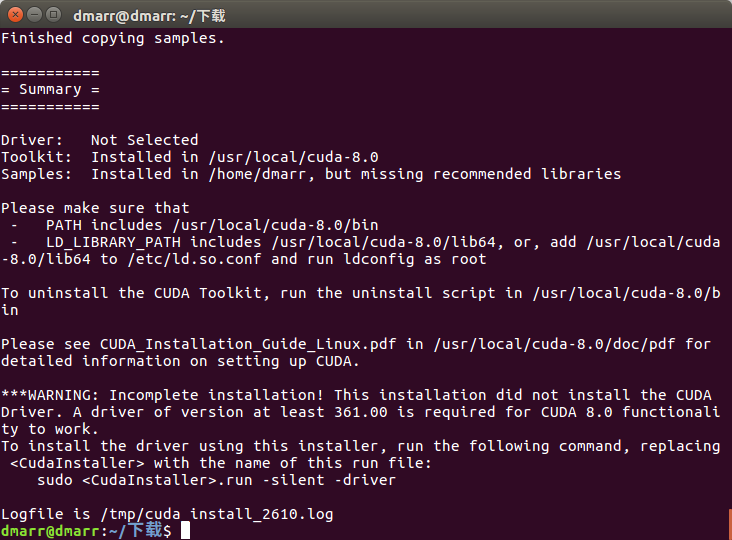
则是 nvidia驱动安装成功

3、安装cuda 8.0/配置cudnn

（1）这里安装已下载的 cuda\_8.0.44\_linux.run，

sh cuda\_8.0.44\_linux.run –override

启动安装程序，一直按空格到最后，输入accept接受条款   
 输入n不安装nvidia图像驱动，已经安装过了   
 输入y安装cuda 8.0工具   
 回车确认cuda默认安装路径：/usr/local/cuda-8.0   
 输入y用sudo权限运行安装，输入密码   
 输入y或者n安装或者不安装指向/usr/local/cuda的符号链接   
 输入y安装CUDA 8.0 Samples，以便后面测试   
 回车确认CUDA 8.0 Samples默认安装路径：/home/dmarr（pawn是我的用户名），该安装路径测试完可以删除   
 安装完显示如下图



（2）接着安装cudnn v5.1

下载cudnn-8.0-linux-x64-v5.1.tgz

tar zxvf cudnn-8.0-linux-x64-v5.1.tgz   
 解压在下载目录下产生一个cuda目录   
 cd cuda/include/

sudo cp cudnn.h /usr/local/cuda/include/ 复制头文件

cd ../lib64 打开lib64目录

sudo cp lib\* /usr/local/cuda/lib64/ 复制库文件

sudo chmod a+r /usr/local/cuda/include/cudnn.h /usr/local/cuda

/lib64/libcudnn\*给所有用户增加这些文件的读权限

（3）建立软链接

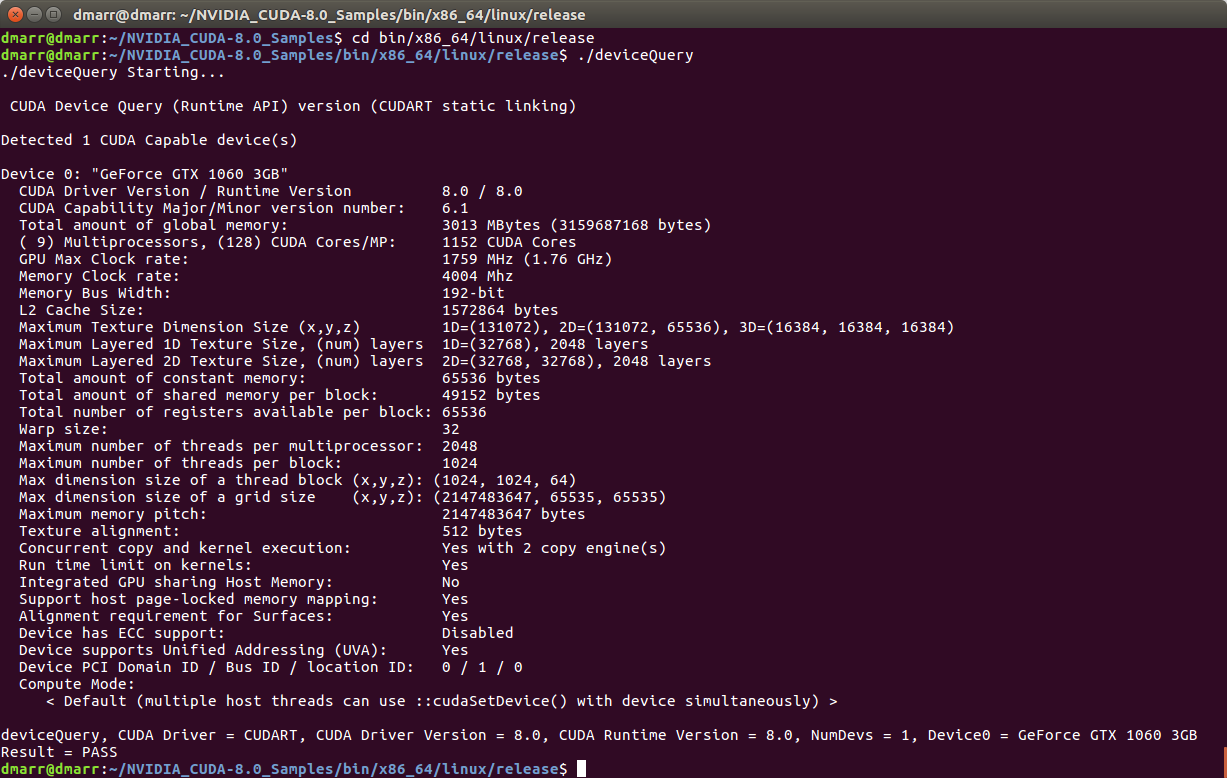
终端输入   
 cd /usr/local/cuda/lib64/   
 sudo rm -rf libcudnn.so libcudnn.so.5   
 sudo ln -s libcudnn.so.5.1.5 libcudnn.so.5   
 sudo ln -s libcudnn.so.5 libcudnn.so   
 设置环境变量，终端输入   
 sudo gedit /etc/profile   
 在末尾加入   
 PATH=/usr/local/cuda/bin:$PATH   
 export PATH   
 保存后，创建链接文件   
 sudo vim /etc/ld.so.conf.d/cuda.conf   
 按a进入插入模式，增加下面一行   
 /usr/local/cuda/lib64   
 按esc退出插入模式，按:wq保存退出   
 最后在终端输入sudo ldconfig使链接生效

（4）运行cuda Samples，测试验证cuda安装成功

打开CUDA 8.0 Samples默认安装路径，终端输入   
 cd /home/dmarr/NVIDIA\_CUDA-8.0\_Samples   
 sudo make all -j4 （4核）

cd bin/x86\_64/linux/release   
 ./deviceQuery

得到以下显卡的信息获得通过，则是cuda安装成功。



4、Opencv安装

这里安装opencv\_2.4.13.zip

（1）首先安装必要的依赖库

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

（2）解压下载的Opencv2.4.13.zip，之后进入安装的文件目录：

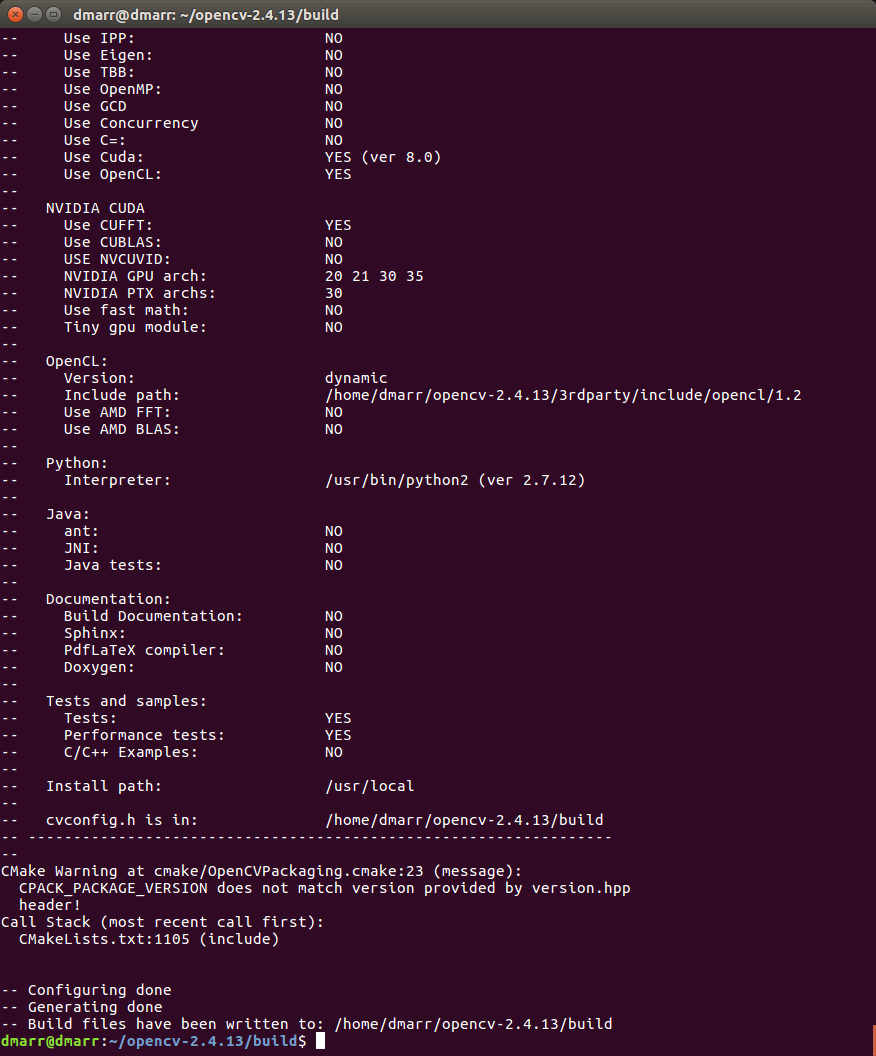
cd opencv-3.1.0

mkdir build #创建build文件夹

cd opencv-3.1.0/build

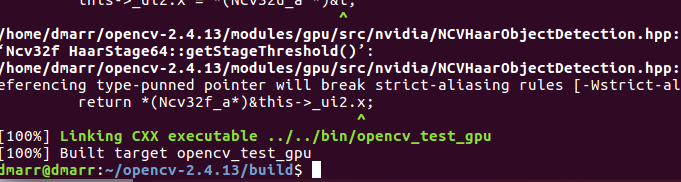
cmake -D CMAKE\_BUILD\_TYPE=Release -D CMAKE\_INSTALL\_PREFIX=/usr/local ..

得到以下

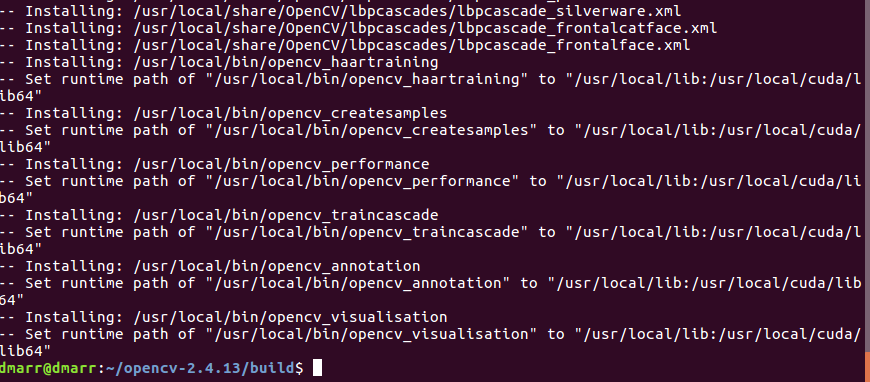
#生成的文件在bulid中，而生成的CMakeList.txt文件在上一级文件中

之后：

make -j4



sudo make install



（3）添加相关路径

sudo gedit /etc/ld.so.conf.d/opencv.conf

加入/usr/local/lib

更新库目录 sudo ldconfig

sudo gedit /etc/bash.bashrc

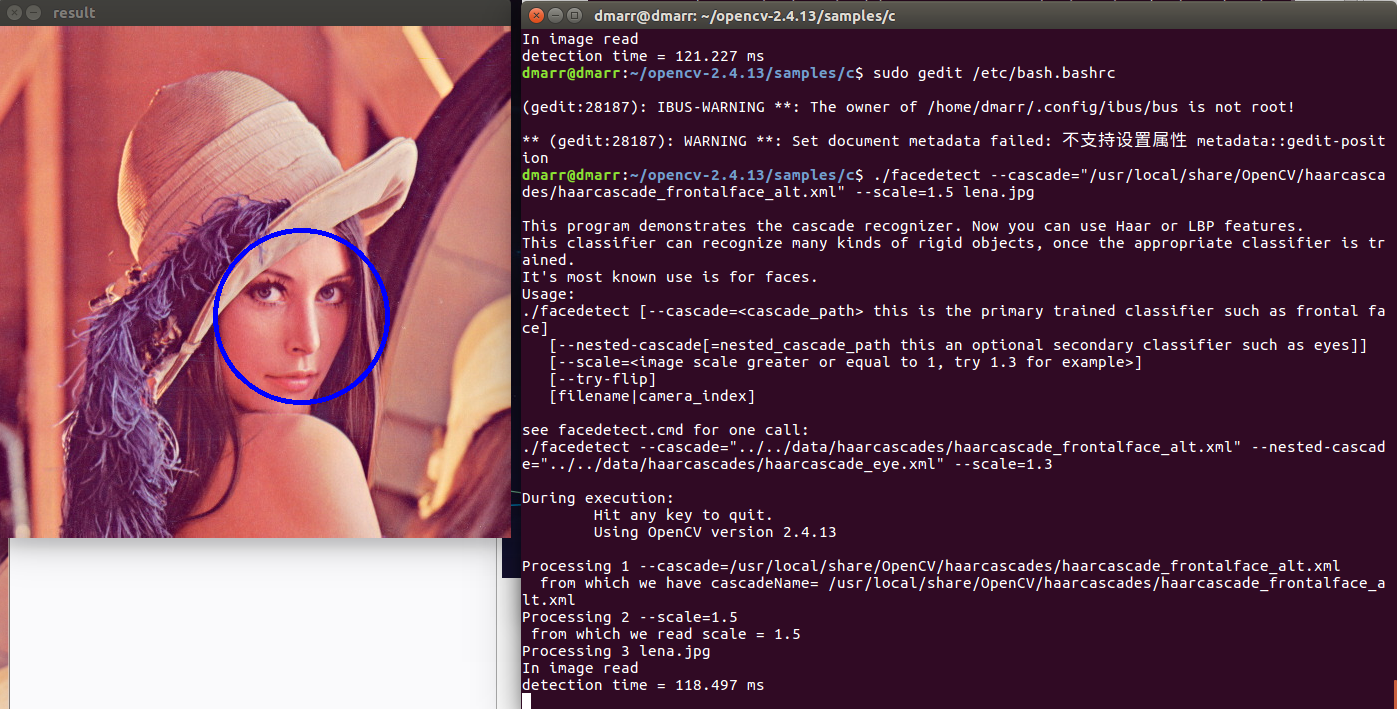
最后一行加入：

export PKG\_CONFIG\_PATH=/usr/local/lib/pkgconfig:$PKG\_CONFIG\_PATH

（4）这样配置就完成了，测试一个程序：

在opencv/samples/c中 执行./build\_all.sh

编译完成后，执行./facedetect --cascade="/usr/local/share/OpenCV/haarcascades/haarcascade\_frontalface\_alt.xml" --scale=1.5 lena.jpg

可对丽娜图进行人脸检测：

5、matlab2016b安装

（1）参考方法：http://blog.csdn.net/generallc/article/details/52793820

下载相关安装包1 2,和破解包Crack

（2）挂载文件夹/home/dmarr/matlab2016，先挂载dvd1,dvd2先不进行挂载

mkdir /home/dmarr/matlab2016 #挂载目录

sudo mount -t auto -o loop /home/dmarr/下载/matlab/R2016b\_glnxa64\_dvd1.iso /home/dmarr/matlab2016

（3）安装

进入到挂载目录，有install文件，返回前一层进行安装

cd /home/dmarr/matlab2016                         #进入到挂载目录下，可以看到install文件

cd ..

sudo /home/dmarr/matlab2016/install     # 进行安装、安装密钥在Crack文件中的readme.txt中，输入时选第一个序列号

注意：设置matlab安装目录，选择为/home/dmarr/MATLAB

（4）在安装到80%时，需要换dvd2进行安装

另外打开一个terminal，输入：

sudo mount -t auto -o loop /home/dmarr/下载/matlab2016b/R2016b\_glnxa64\_dvd2.iso /home/dmarr/matlab2016

返回安装界面，点确定继续安装

（5）激活

本文激活过程中，出现了权限不足问题，采用chomd命令提高对license\_standalone.lic文件和安装目录R2016b的权限

cd /home/dmarr/下载/matlab2016b/           #进入下载目录

sudo chmod 777 license\_standalone.lic      #提高对license\_standalone.lic文件的操作权限

cd/home/dmarr/MATLAB     #进入到matlab安装文件中

sudo chmod -R 777 R2016b/      #提高对安装目录R2016b的操作权限

之后，将/home/dmarr/下载/matlab2016b/R2016b/bin/glnxa64内的文件复制并替换到/home/dmarr/MATLAB/R2016b/bin/glnxa64

进行激活操作：

cd/home/generallc/MATLAB/R2016b/bin   #进入到matlab安装文件中

./matlab                                                     #运行matlab，弹出激活对话框，选择用不联网的方法进行激活，加载license\_standalone.lic文件

matlab安装成功

（6）命令行启动matlab

在目录/usr/local/bin里面创建一个指向Matlab安装目录/home/dmarr/MATLAB/R2016b/bin的符号链接：

sudo ln -s /home/dmarr/MATLAB/R2016b/bin/matlab /usr/local/bin/matlab

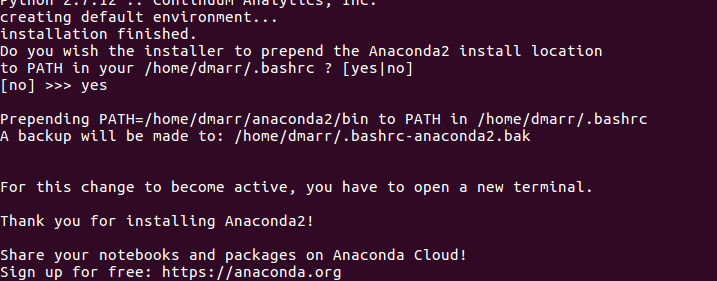
这样就可以在命令行任意文件下启动matlab，terminal： matlab

6、python安装

在下载目录，

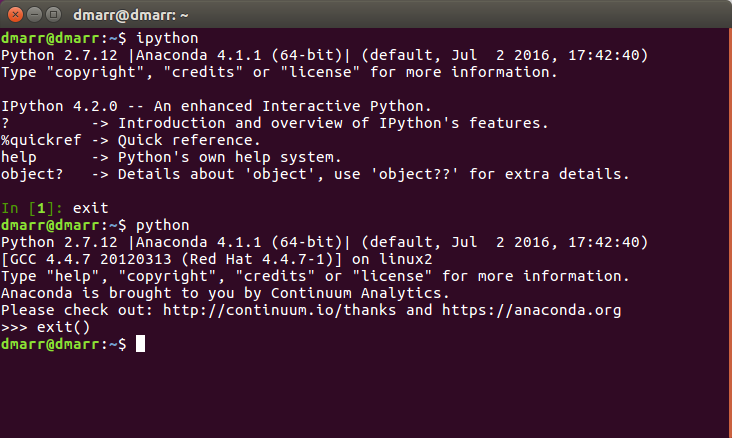
bash Anaconda2-4.1.1-Linux-x86\_64.sh 确认后安装，

切忌不要按多余的回车键，在最后，输入yes，指明PATH



重启

分别输入ipython、python可得



7、安装BLAS

sudo apt-get install libatlas-base-dev

8、caffe的配置

（1）安装protobuf

由于之前的经验，使用别的流程默认安装proto，会安装3.2版本，所以手动安装2.6.1版本：下载链接，  <https://github.com/google/protobuf/releases/download/v2.6.1/protobuf-2.6.1.tar.gz>

tar -zxvf protobuf-2.6.1.tar.gz

cd protobuf-2.6.1/

./configure

make

make check

sudo make install

检查是否安装成功，protoc --version

在这里，有时会报错：

这是由于protobuf的默认安装路径是/usr/local/lib,而/usr/local/lib不在ubuntu体系默认的LD\_LIBRARY\_PATH里,所以就找不到lib，这样解决：

在/etc/ld.so.conf.d文件下，

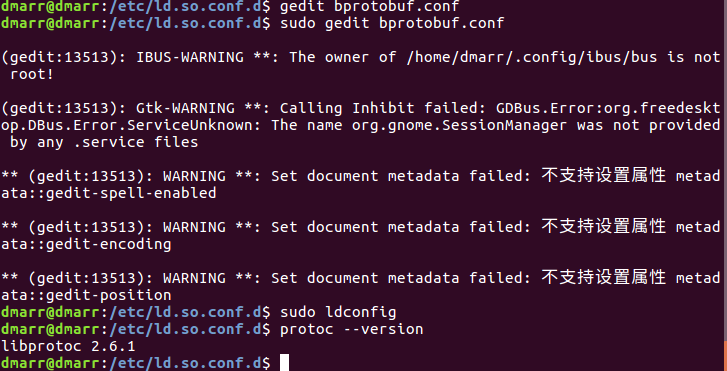
sudo gedit bprotobuf.conf，

输入/usr/local/lib

保存退出，再

sudo ldconfig加载路径。

再检查protobuf版本：protoc --version



这样安装成功。

（2）安装caffe其他依赖

sudo apt-get install libleveldb-dev libsnappy-dev libboost-all-dev libhdf5-serial-dev libgflags-dev libgoogle-glog-dev liblmdb-dev

（3）配置caffe makefile.config

进入caffe目录，cd caffe

sudo cp Makefile.config.example Makefile.config # 备份配置文件

gedit Makefile.config # 修改编译文件

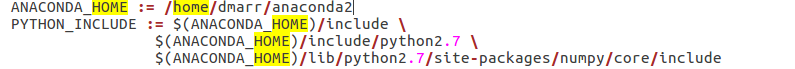
根据我自己的需求，修改了以下内容，

1） USE\_CUDNN := 1 注销了’#‘

2） 关于OPENCV\_VERSION := 3，如果装的是opencv3 ，则注销’#‘，如果是opencv2则不需要注销。

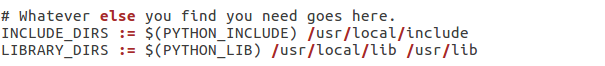
3） 这里注销掉：



4） 这里取消注释，并且注意自己的anaconda位置与名称：

5） 这里取消注释： WITH\_PYTHON\_LAYER := 1

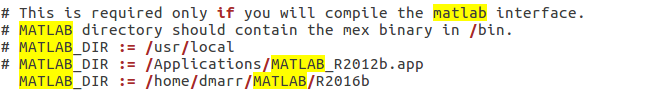
6） 将以下的代码

修改为：

INCLUDE\_DIRS := $(PYTHON\_INCLUDE) /usr/local/include /usr/include/hdf5/serial

LIBRARY\_DIRS := $(PYTHON\_LIB) /usr/local/lib /usr/lib /usr/lib/x86\_64-linux-gnu /usr/lib/x86\_64-linux-gnu/hdf5/serial

7）修改matlab接口，要将MATLAB\_DIR换成自己的MATLAB安装路径



（4）编译

1）在caffe目录下，

make all -j4

报错 没有hdf5,解决方法如下：

cd /usr/lib/x86\_64-linux-gnu

0

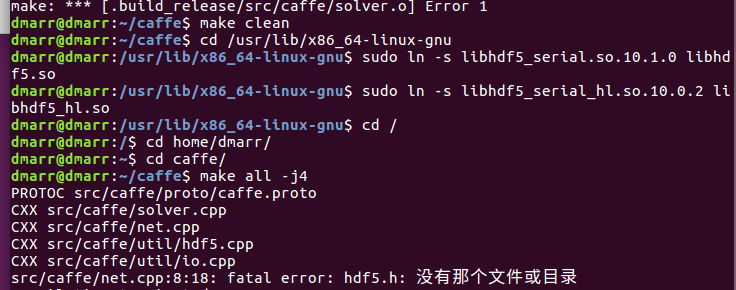
\\然后根据情况执行下面两句：

sudo ln -s libhdf5\_serial.so.10.1.0 libhdf5.so

sudo ln -s libhdf5\_serial\_hl.so.10.0.2 libhdf5\_hl.so

以上这步应配合（3）中6）的操作。接着 sudo ldconfig

在caffe目录下，make clean 清除 编译，make all -j4 再次make all -j4

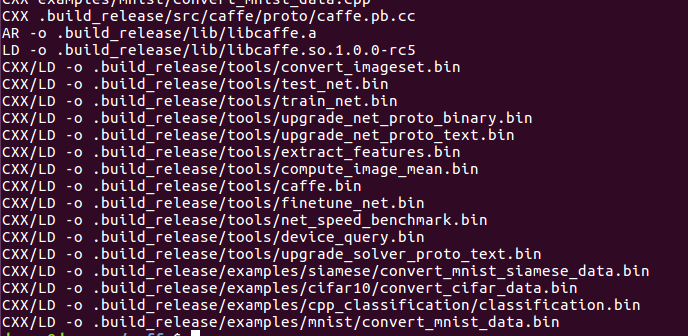


依然报错，在hdf5目录下sudo rm -rf libhdf5.so ，sudo rm -rf libhdf5\_hl.so 删除之前的链接;

寻找其他办法，  
在文件Makefile中  
将hdf5\_hl and hdf5 改为 hdf5\_serial\_hl ，hdf5\_serial

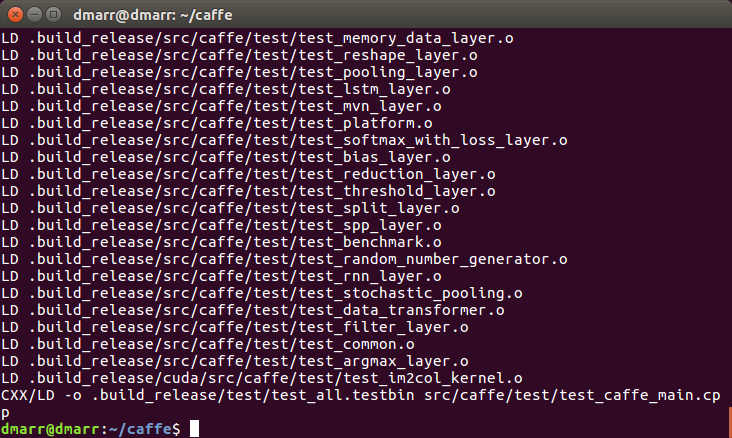
这样也不行，在实在没办法之后，我将网上的

直接粘贴到了 makefile.config中，直接替换了那部分，于是编译通过了：

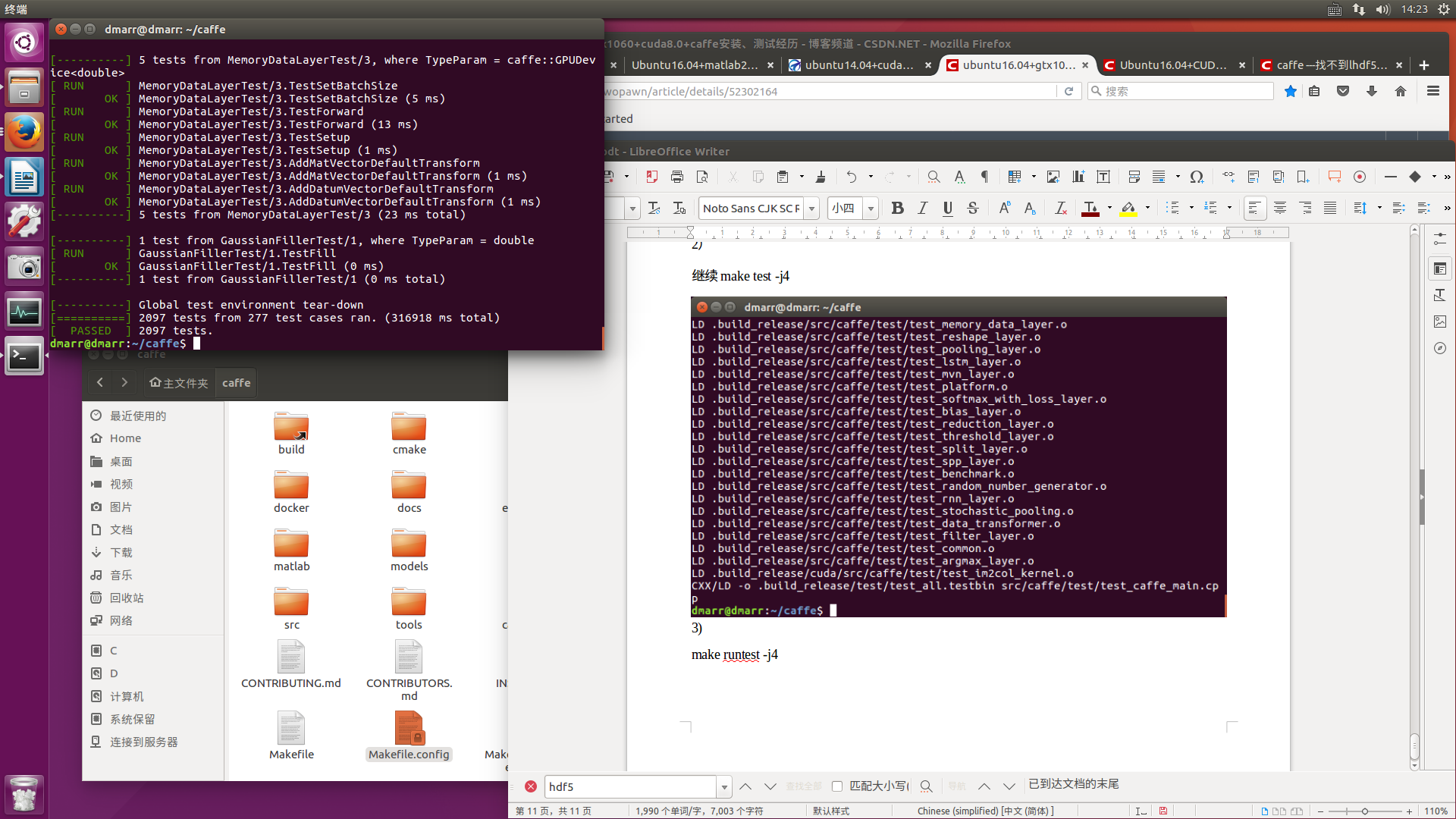


2）

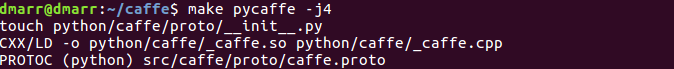
继续make test -j4

3)

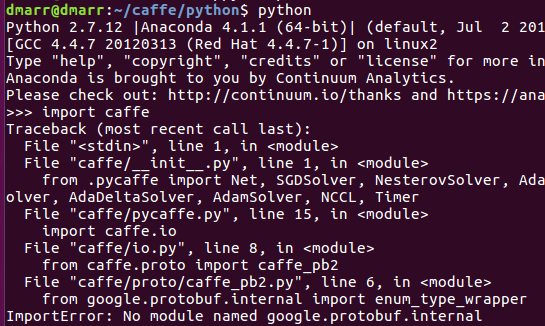
make runtest -j4

4)

make pycaffe -j4

5）

在python中import时会报错



这时去protobuf文档下，在/python/google下创建空文件夹compiler

cd..

再运行：

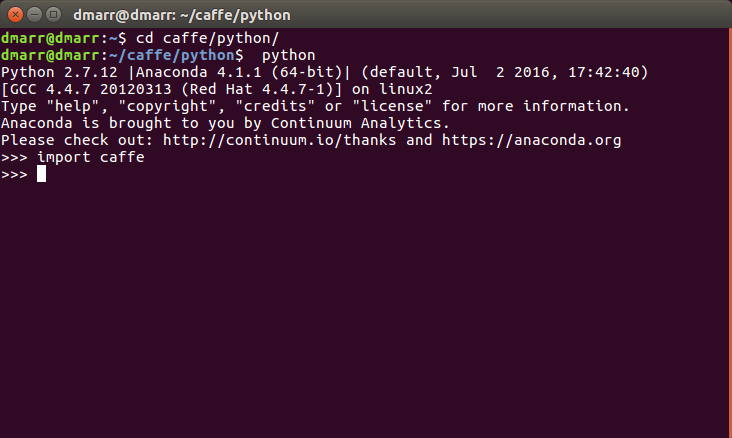
python setup.py install 安装成功

6)再在 caffe/python下

python

import caffe

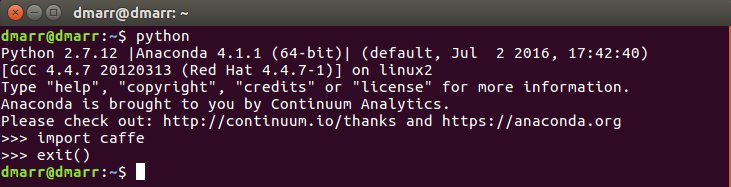
没有报错，则成功

7）为了能在任何目录下python都能import caffe成功

sudo gedit ~/.bashrc

export PYTHONPATH="/home/dmarr/caffe/python:$PYTHONPATH"

保存退出，sudo ldconfig



8)编译matlab接口

terminal中caffe目录下运行：

make matcaffe

9）上一步提到gcc版本的问题，在成功完成以上的所有步骤之后，为了matlab能在相应匹配的gcc版本下运行，这里要对gcc版本进行降级

查看gcc -v版本

安装gcc4.9 g++4.9

sudo apt-get install gcc-4.9

sudo apt-get install g++-4.9

cd /usr/bin 查看目录中已有的gcc版本 ls gcc\*

设置gcc版本号

sudo update-alternatives --install /usr/bin/gcc gcc /usr/bin/gcc-4.9 49

sudo update-alternatives --install /usr/bin/gcc gcc /usr/bin/gcc-5 54

选择版本sudo update-alternatives --config gcc 选择1 ，即是选择了gcc4.9

设置g++版本号

sudo update-alternatives --install /usr/bin/g++ g++ /usr/bin/g++-4.9 49

sudo update-alternatives --install /usr/bin/g++ g++ /usr/bin/g++-5 54

选择版本sudo update-alternatives --config g++ 选择1 ，即是选择了g++4.9

查看gcc -v版本

至此 安装完成。