# ubuntu16.04 GTX1080Ti tensorflow安装方法

## 1.安装ubuntu16.04

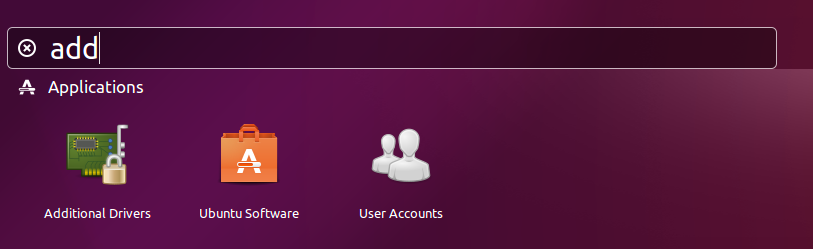
首先将ubuntu系统安装好

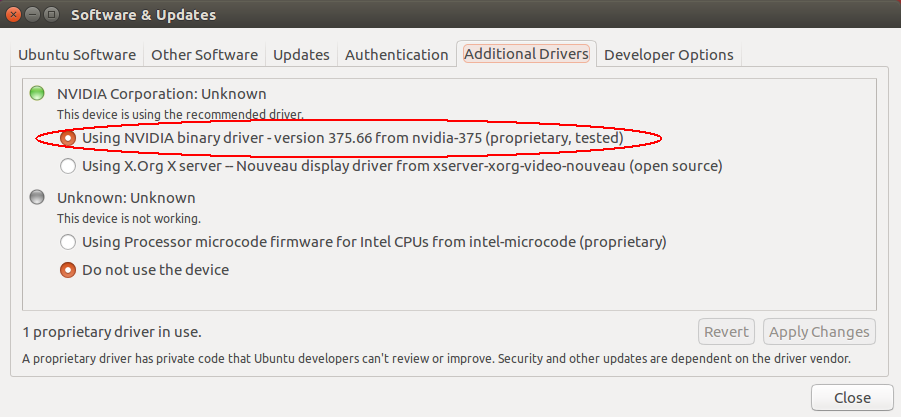
## 2.显卡驱动安装

安装完系统再次进入ubuntu会发现卡在紫色屏，原因是ubuntu自带的显卡驱动不匹配。所以先进入bios将视频输出设置为集成显卡输出，再次启动发现可以进入ubuntu。

进入系统后找到Additional Drivers安装好Nvidia官方驱动。

安装好驱动后再次进入bios将视频输出设置为独立显卡输出。

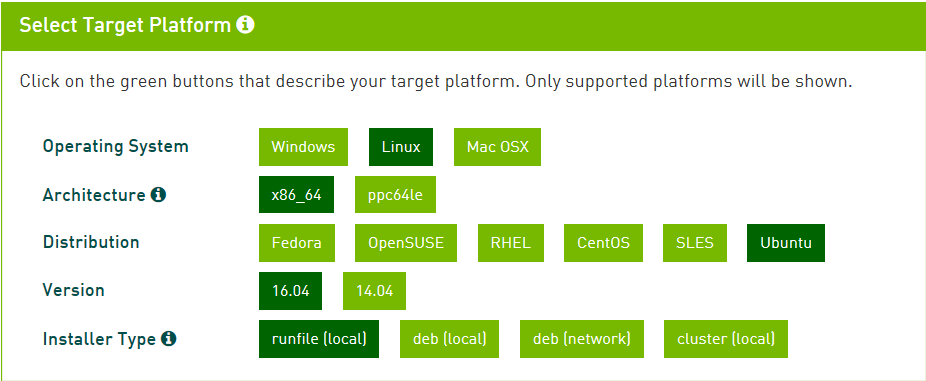


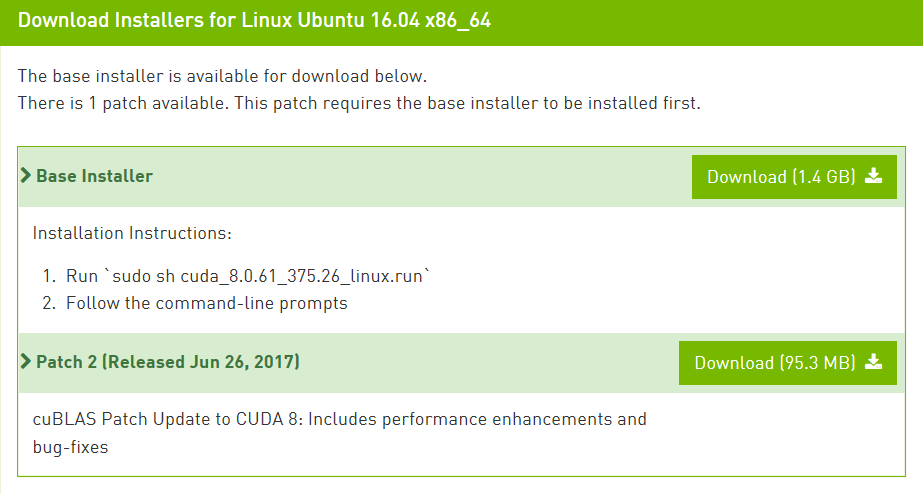


## 3.cuda8.0安装

<https://developer.nvidia.com/cuda-downloads>（下载地址）

首先下载ubuntu安装文件。





下载后敲入如下命令进行安装。

$sudo chmod **+**x cuda\_8.0.61\_375.26\_linux.run

$sudo sh cuda\_8.0.61\_375.26\_linux.run

$sudo sh cuda\_8.0.61.2\_linux.run

安装过程中要注意一点，安装cuda时会自带一个显卡驱动，不要安装这个驱动。

Install NVIDIA Accelerated Graphics Driver **for** Linux**-**x86\_64 361.62**?**

这里需要选择N，其他默认yes。

最后，配置环境变量，直接放在系统配置文件profile里面:

$sudo gedit /etc/profile

在最后面加入两行代码．

export PATH=/usr/local/cuda-8.0/bin:$PATH

export LD\_LIBRARY\_PATH=/usr/local/cuda-8.0/lib64:$LD\_LIBRARY\_PATH

安装完测试cuda是否安装成功，进入NVIDIA\_CUDA-8.0\_Samples，编译deviceQuery并尝试是否运行成功。

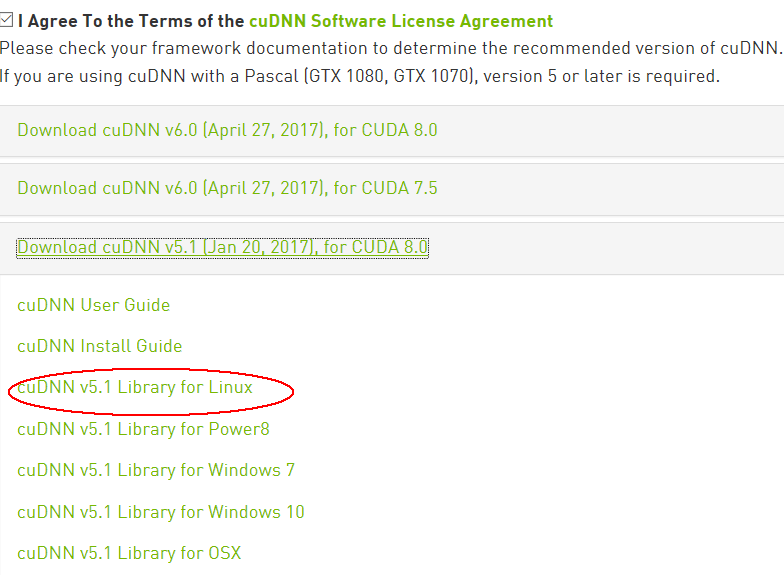
cd 1\_Utilities/deviceQuery

make

## 4.cuDNN安装

下载地址： <https://developer.nvidia.com/cudnn>（需要登录）

下载需要填写一个调查问卷，填写完毕点击 I Agree To 前面的小方框，出现如下：



下载完成后，打开terminal依次输入以下指令：

cd /home/\*\*\*(自己的用户名)/Desktop/###(这个命令意思是找到刚刚我们用U盘传过来的文件)

tar xvzf cudnn-8.0-linux-x64-v5.1-ga.tgz###（解压这个文件）

sudo cp cuda/include/cudnn.h /usr/local/cuda/include###（复制）

sudo cp cuda/lib64/libcudnn\* /usr/local/cuda/lib64###（复制）

sudo chmod a+r /usr/local/cuda/include/cudnn.h /usr/local/cuda/lib64/libcudnn\*

## 5.安装libcupti-dev

$ sudo apt-get install libcupti-dev

## 6.安装tensorflow

首先安装python相关环境。

$ sudo apt**-**get install python**-**pip python**-**dev # **for** Python 2.7

$ sudo apt**-**get install python3**-**pip python3**-**dev # **for** Python 3.n

然后安装tensorflow。

$ pip install tensorflow # Python 2.7**;** CPU support **(**no GPU support**)**

$ pip3 install tensorflow # Python 3.n**;** CPU support **(**no GPU support**)**

$ pip install tensorflow**-**gpu # Python 2.7**;** GPU support

$ pip3 install tensorflow**-**gpu # Python 3.n**;** GPU support

最后验证tensorflow是否安装成功。

$ python

# Python

import tensorflow as tf

hello **=** tf**.**constant**(**'Hello, TensorFlow!'**)**

sess **=** tf**.**Session**()**

print**(**sess**.**run**(**hello**))**

## 7.服务器其他配置

### 7.1 固定IP

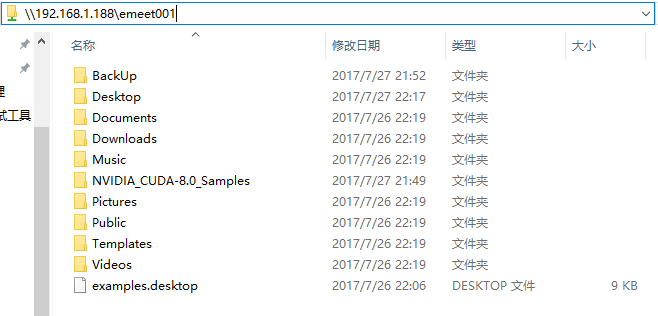
192.168.1.188

### 7.2 多用户

目前eMeet服务器有三个用户emeet001 emeet002 emeet003，其中emeet001为root权限，暂不使用。其余两个账户密码为123456。

### 7.3 samba服务

emeet002和emeet003的samba密码为123456，访问权限设置为各自的home文件夹。



### 7.4 ssh和screen服务

可以通过ssh在windows上远程登录。

使用ssh登录，极有可能在运行tensorflow时意外退出造成程序执行中断，想要继续恢复需要借用screen命令，使用方法如下。

ssh登录后创建一个虚拟终端。

screen **-**S terminal1

这个时候可以断开ssh客户端，并再次ssh登录。

可以调用如下命令显示和恢复上次保存的终端。

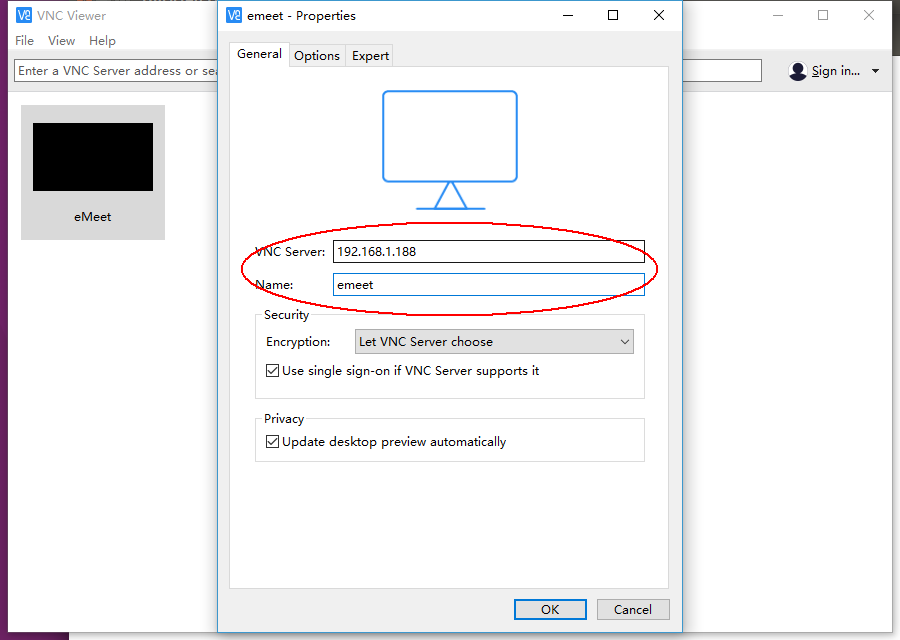
screen **-**ls   列出已经创建的正在后台运行的终端

screen **-**r terminal1

想要退出该终端执行exit可以完全退出。

### 7.5 x11vnc服务

可以通过windows下的realvnc客户端远程登录图形界面，vnc登录密码为123456。



## 8.anaconda安装和使用

python易用，但用好却不易，其中比较头疼的就是包管理和Python不同版本的问题。anaconda则提供了包管理和环境管理，并自带了某个版本的python。如果设置了环境变量，将会使用anaconda自带的python作为系统默认的python。

### 8.1 anaconda安装

bash **~/Downloads/**Anaconda3**-**4.4.0**-**Linux**-**x86\_64**.**sh

### 8.2 包管理

# 安装scipy

conda install scipy

# conda会从从远程搜索scipy的相关信息和依赖项目，对于python 3.4，conda会同时安装numpy和mkl（运算加速的库）

# 查看已经安装的packages

conda list

# 最新版的conda是从site-packages文件夹中搜索已经安装的包，不依赖于pip，因此可以显示出通过各种方式安装的包

### 8.3 虚拟环境

在虚拟环境下可以随意安装python的任何东西而不用担心损坏，即使发生错误直接删除掉该虚拟环境即可。创建虚拟环境可以指定任意的python版本。

# 创建一个名为python34的环境，指定Python版本是3.4（不用管是3.4.x，conda会为我们自动寻找3.4.x中的最新版本）

conda create **--**name python34 python**=**3.4

#如果需要创建带anaconda包的环境,执行如下命令

conda create **-**n python34 python**=**3.4 anaconda

# 安装好后，使用activate激活某个环境

activate python34 # **for** Windows

source activate python34 # **for** Linux **&** Mac

# 激活后，会发现terminal输入的地方多了python34的字样，实际上，此时系统做的事情就是把默认2.7环境从PATH中去除，再把3.4对应的命令加入PATH

# 此时，再次输入

python **--**version

# 可以得到`Python 3.4.5 :: Anaconda 4.1.1 (64-bit)`，即系统已经切换到了3.4的环境

# 如果想返回默认的python 2.7环境，运行

deactivate python34 # **for** Windows

source deactivate python34 # **for** Linux **&** Mac

# 删除一个已有的环境

conda remove **--**name python34 **–**all

#查看当前存在哪些虚拟环境

conda env list 或 conda info **-**e

### 8.4 spyder在虚拟环境中的使用

激活虚拟环境后，直接执行spyder启动图形界面即可。

### 8.5 tensorflow的安装位置

前面是在ubuntu自带的python下安装好了tensorflow，虚拟环境下是否能识别到呢？答案是可以，当然你也可以在虚拟环境中再次安装tensorflow。