

0、必要说明

务必根据自己的系统、显卡、使用目的等情况，进行修改相关配置，才能保证环境搭建成功。

由于各种各样的原因，环境搭建过程中会出现各种各样的问题，要有必要的心理准备。

如果出现问题，请使用Baidu/Google进行搜索，尝试解决，一般情况下能解决95%的问题。如果不行，记得重启一下，一般还能解决另外2%的问题。如果还是不行，可以考虑重装一下系统，一般还能解决另外2%的问题。如果仍然不行.....

1、禁用nouveau

在安装NVIDIA驱动以前需要禁止系统自带显卡驱动nouveau：可以先通过指令 **lsmod | grep nouveau** 查看nouveau驱动的启用情况，如果有输出表示nouveau驱动正在工作，如果没有内容输出则表示已经禁用了nouveau。首先，打开禁用设置文件

```
1 sudo gedit /etc/modprobe.d/blacklist-nouveau.conf
```

然后，将下面的内容复制粘贴到打开的blacklist-nouveau.conf中

```
1 blacklist nouveau
2 options nouveau modeset=0
```

保存并退出文件。使用命令更新配置，使之生效。

```
1 sudo update-initramfs -u
```

2、安装显卡驱动

首先，使用ubuntu-drivers devices查看推荐驱动版本（PS：不同的机器、显卡、系统会出现推荐结果不同的情况）。

```
(base) vcisl@vcisl:~$ ubuntu-drivers devices
== /sys/devices/pci0000:b2/0000:b2:00.0/0000:b3:00.0 ==
modalias : pci:v000010DEd00002204sv00001458sd00004043bc03sc00i00
vendor    : NVIDIA Corporation
manual install: True
driver    : nvidia-driver-510-server - distro non-free recommended
driver    : nvidia-driver-470-server - distro non-free
driver    : nvidia-driver-510 - distro non-free
driver    : nvidia-driver-470 - distro non-free
driver    : xserver-xorg-video-nouveau - distro free builtin
```

通过上述命令也可能返回为空什么都不显示，添加官方ppa的源，在更新以下就好了。

```
1 sudo add-apt-repository ppa:graphics-drivers/ppa
```

然后，再安装推荐的驱动版本。

```
(base) vcisl@vcisl:~$ sudo apt install nvidia-driver-510
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
clamav-base clamav-freshclam gyp libclamav9 libhttp-parser2.7.1 libjs-async li
libtfml libuv1-dev node-abbrev node-ansi node-ansi-color-table node-archy node
node-builtin-modules node-combined-stream node-concat-map node-cookie-jar node
node-fstream node-fstream-ignore node-github-url-from-git node-glob node-grace
node-is-builtin-module node-isexe node-json-stringify-safe node-lockfile node-
node-node-uuid node-nopt node-normalize-package-data node-npmlog node-once nod
node-read-package-json node-request node-retry node-rimraf node-semver node-sh
node-spx-license-ids node-tar node-tunnel-agent node-underscore node-validate
nvidia-container-runtime shim
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
gcc-8-base:i386 libatomic1:i386 libbsd0:i386 libc6:i386 libdrm-amdgpu1:i386 li
libedit2:i386 libelf1:i386 libexpat1 libexpat1:i386 libexpat1-dev libffi6:i386
libglvnd0:i386 libglx-mesa0:i386 libglx0:i386 libllvm10:i386 libnvidia-cfg1-51
libnvidia-decode-510 libnvidia-decode-510:i386 libnvidia-encode-510 libnvidia-
libnvidia-fbc1-510:i386 libnvidia-gl-510 libnvidia-gl-510:i386 libpciaccess0:i
libx11-xcb1:i386 libxau6:i386 libxcb-dri2-0:i386 libxcb-dri3-0:i386 libxcb-glx
libxdamage1:i386 libxdmcp6:i386 libxext6:i386 libxf86vm3:i386 libxnvctrl0 libx
nvidia-dkms-510 nvidia-kernel-common-510 nvidia-kernel-source-510 nvidia-prime
xserver-xorg-video-nvidia-510 zlib1g zlib1g:i386 zlib1g-dev
Suggested packages:
```

等待安装完毕后，使用 `nvidia_smi` 查看显卡和驱动信息。

```
(base) vcisl@vcisl:~$ nvidia-smi
Thu Apr 28 17:02:29 2022
+-----+
| NVIDIA-SMI 460.67          Driver Version: 460.67          CUDA Version: 11.2          |
+-----+-----+
| GPU   Name               Persistence-M| Bus-Id        Disp.A | Volatile Uncorr. ECC |
| Fan   Temp   Perf      Pwr:Usage/Cap|      Memory-Usage | GPU-Util  Compute M. |
|                               |                      |              MIG M. |
+-----+-----+
|  0    GeForce RTX 3090      Off      | 00000000:65:00:0 Off |                  N/A |
| 34%    45C    P8         32W / 350W | 19MiB / 24265MiB |      0%      Default |
|                               |                      |                  N/A |
+-----+-----+
|  1    GeForce RTX 3090      Off      | 00000000:B3:00:0 Off |                  N/A |
| 59%    52C    P2        132W / 370W | 2298MiB / 24268MiB |     17%      Default |
|                               |                      |                  N/A |
+-----+-----+

+-----+
| Processes: |
| GPU   GI    CI          PID    Type    Process name                  GPU Memory |
|       ID    ID              |              |           Usage              |
+-----+-----+
|  0     N/A  N/A         1250      G   /usr/lib/xorg/Xorg              14MiB |
|  1     N/A  N/A        31980      C      python                    2295MiB |
+-----+-----+
```

3、安装CUDA

首先，查看nvidia驱动对应的CUDA版本（链接：<https://docs.nvidia.com/cuda/cuda-toolkit-release-notes/>）。

Table 3. CUDA Toolkit and Corresponding Driver Versions

CUDA Toolkit	Toolkit Driver Version	
	Linux x86_64 Driver Version	Windows x86_64 Driver Version
CUDA 11.6 Update 2	>=510.47.03	>=511.65
CUDA 11.6 Update 1	>=510.47.03	>=511.65
CUDA 11.6 GA	>=510.39.01	>=511.23
CUDA 11.5 Update 2	>=495.29.05	>=496.13
CUDA 11.5 Update 1	>=495.29.05	>=496.13
CUDA 11.5 GA	>=495.29.05	>=496.04
CUDA 11.4 Update 4	>=470.82.01	>=472.50
CUDA 11.4 Update 3	>=470.82.01	>=472.50
CUDA 11.4 Update 2	>=470.57.02	>=471.41
CUDA 11.4 Update 1	>=470.57.02	>=471.41
CUDA 11.4.0 GA	>=470.42.01	>=471.11
CUDA 11.3.1 Update 1	>=465.19.01	>=465.89
CUDA 11.3.0 GA	>=465.19.01	>=465.89
CUDA 11.2.2 Update 2	>=460.32.03	>=461.33
CUDA 11.2.1 Update 1	>=460.32.03	>=461.09
CUDA 11.2.0 GA	>=460.27.03	>=460.82
CUDA 11.1.1 Update 1	>=455.32	>=456.81
CUDA 11.1 GA	>=455.23	>=456.38
CUDA 11.0.3 Update 1	>= 450.51.06	>= 451.82
CUDA 11.0.2 GA	>= 450.51.05	>= 451.48
CUDA 11.0.1 RC	>= 450.36.06	>= 451.22
CUDA 10.2.89	>= 440.33	>= 441.22
CUDA 10.1 (10.1.105 general release, and updates)	>= 418.39	>= 418.96
CUDA 10.0.130	>= 410.48	>= 411.31

然后，选择对应版本的CUDA（链接：<https://developer.nvidia.com/cuda-toolkit-archive>）。

[CUDA Toolkit 11.4.0 \(June 2021\), Versioned Online Documentation](#)
[CUDA Toolkit 11.3.1 \(May 2021\), Versioned Online Documentation](#)
[CUDA Toolkit 11.3.0 \(April 2021\), Versioned Online Documentation](#)
[CUDA Toolkit 11.2.2 \(March 2021\), Versioned Online Documentation](#)
[CUDA Toolkit 11.2.1 \(February 2021\), Versioned Online Documentation](#)
[CUDA Toolkit 11.2.0 \(December 2020\), Versioned Online Documentation](#)
[CUDA Toolkit 11.1.1 \(October 2020\), Versioned Online Documentation](#)
[CUDA Toolkit 11.1.0 \(September 2020\), Versioned Online Documentation](#)
[CUDA Toolkit 11.0.3 \(August 2020\), Versioned Online Documentation](#)
[CUDA Toolkit 11.0.2 \(July 2020\), Versioned Online Documentation](#)
[CUDA Toolkit 11.0.1 \(June 2020\), Versioned Online Documentation](#)
[CUDA Toolkit 11.0.0 \(March 2020\), Versioned Online Documentation](#)
[CUDA Toolkit 10.2 \(Nov 2019\), Versioned Online Documentation](#)
[CUDA Toolkit 10.1 update2 \(Aug 2019\), Versioned Online Documentation](#)
[CUDA Toolkit 10.1 update1 \(May 2019\), Versioned Online Documentation](#)
[CUDA Toolkit 10.1 \(Feb 2019\), Online Documentation](#)
[CUDA Toolkit 10.0 \(Sept 2018\), Online Documentation](#)

根据自己的系统情况，进行下载。

Select Target Platform

Click on the green buttons that describe your target platform. Only supported platforms will be shown. By downloading and using the software, you agree to fully comply with the terms and conditions of the [CUDA EULA](#).

Operating System	Linux	Windows			
Architecture	x86_64	ppc64le	arm64-sbsa		
Distribution	CentOS	OpenSUSE	RHEL	SLES	Ubuntu
Version	16.04	18.04	20.04		
Installer Type	deb (local)	deb (network)	runfile (local)		

Download Installer for Linux Ubuntu 18.04 x86_64

The base installer is available for download below.

Base Installer

Installation Instructions:

```
$ wget https://developer.download.nvidia.com/compute/cuda/11.0.3/local_installers/cuda_11.0.3_450.51.06_linux.run
$ sudo sh cuda_11.0.3_450.51.06_linux.run
```

如果使用wget下载困难，将下载链接复制出来，使用其他下载器下载。

在安装CUDA之前需要首先安装一些相互依赖的库文件。

```
1 sudo apt-get install freeglut3-dev build-essential libx11-dev libxmu-dev libxi-dev
libgl1-mesa-glx libglu1-mesa libglu1-mesa-dev
```

然后，执行以下命令来运行下载的安装包，按照提示进行安装。

```
1 sudo sh cuda_11.0.3_450.51.06_linux.run
```

注意！ 询问你是否安装Driver的时候的选择no！ 驱动前面已经安装过了，而且此时 安装显卡驱动非常容易出错。 Toolkit必选，其他的随意。

```
CUDA Installer se Agreement
- [ ] Driver
  [ ] 450.51.06
+ [X] CUDA Toolkit 11.0
  [X] CUDA Samples 11.0
  [X] CUDA Demo Suite 11.0
  [X] CUDA Documentation 11.0
Options
Install
```

经过漫长的等待之后，输出以下界面代表安装完毕（稍微留意一下红框中的信息）。

```
(base) vcisl@vcisl:~/Downloads$ sudo bash cuda_11.0.3_450.51.06_linux.run
=====
= Summary =
=====
Driver:    Not Selected
Toolkit:   Installed in /usr/local/cuda-11.0/
Samples:   Installed in /home/vcisl/, but missing recommended libraries

Please make sure that
- PATH includes /usr/local/cuda-11.0/bin
- LD_LIBRARY_PATH includes /usr/local/cuda-11.0/lib64, or, add /usr/local/cuda-11.0/lib64 to /etc/ld.so.conf and run ldconfig as root

To uninstall the CUDA Toolkit, run cuda-uninstaller in /usr/local/cuda-11.0/bin
***WARNING: Incomplete installation! This installation did not install the CUDA Driver. A driver of version at least .00 is required for CUDA 11.0
to work.
To install the driver using this installer, run the following command, replacing <CudaInstaller> with the name of this run file:
    sudo <CudaInstaller>.run --silent --driver

Logfile is /var/log/cuda-installer.log
```

然后配置CUDA环境变量，打开bashrc文件，并在文件尾部追加内容。

```
1 export PATH=/usr/local/cuda-11.0/bin:${PATH}
2 export LD_LIBRARY_PATH=/usr/local/cuda-11.0/lib64:${LD_LIBRARY_PATH}
3 export CUDA_HOME=/usr/local/cuda-11.0/
4
```

注意！ 务必修改CUDA版本（下图红框选中的内容），按照CUDA安装完成后输出的内容进行适配。

```
1 export PATH=/usr/local/cuda-11.0/bin:${PATH}
2 export LD_LIBRARY_PATH=/usr/local/cuda-11.0/lib64:${LD_LIBRARY_PATH}
```

添加完成后，保存退出文件。

当然还需要 source ~/.bashrc 一下把新添加的环境激活一下。如果激活成功的话继续下一步，如果失败看看是不是自己路径错了通过which或者whereis去查找一下。排除错误。

最后，通过 nvcc -V 判断cuda是否安装成功，如果返回版本信息就是成功了。

```
(base) vcisl@vcisl:~$ nvcc -V
nvcc: NVIDIA (R) Cuda compiler driver
Copyright (c) 2005-2020 NVIDIA Corporation
Built on Wed Jul 22 19:09:09 PDT 2020
Cuda compilation tools, release 11.0, V11.0.221
Build cuda_11.0_bu.TC445_37.28845127_0
```

或者通过编译cuda sample来进行测试

```
1 cd /usr/local/cuda-11.0/samples/1_Uutilities/deviceQuery
2 sudo make
3 sudo ./deviceQuery
```

输出以下内容，也表示CUDA安装成功。

```
Max dimension size of a grid size (x,y,z): (2147483647, 65535, 65535)
Maximum memory pitch: 2147483647 bytes
Texture alignment: 512 bytes
Concurrent copy and kernel execution: Yes with 2 copy engine(s)
Run time limit on kernels: No
Integrated GPU sharing Host Memory: No
Support host page-locked memory mapping: Yes
Alignment requirement for Surfaces: Yes
Device has ECC support: Disabled
Device supports Unified Addressing (UVA): Yes
Device supports Managed Memory: Yes
Device supports Compute Preemption: Yes
Supports Cooperative Kernel Launch: Yes
Supports MultiDevice Co-op Kernel Launch: Yes
Device PCI Domain ID / Bus ID / location ID: 0 / 179 / 0
Compute Mode:
  < Default (multiple host threads can use ::cudaSetDevice() with device simultaneously) >
> Peer access from GeForce RTX 3090 (GPU0) -> GeForce RTX 3090 (GPU1) : No
> Peer access from GeForce RTX 3090 (GPU1) -> GeForce RTX 3090 (GPU0) : No

deviceQuery, CUDA Driver = CUDART, CUDA Driver Version = 11.2, CUDA Runtime Version = 11.0, NumDevs = 2
Result = PASS
```

(PS:有兴趣的可以了解一下CUDA Driver Version 和 CUDA Runtime Version 的区别)