

让数字科技助力未来发展



ubuntu安装cuda、cudnn

编辑：杨仁飞 审核：秦银华

上海宽泛科技有限公司
Shanghai KuanFan technology Co.Ltd.

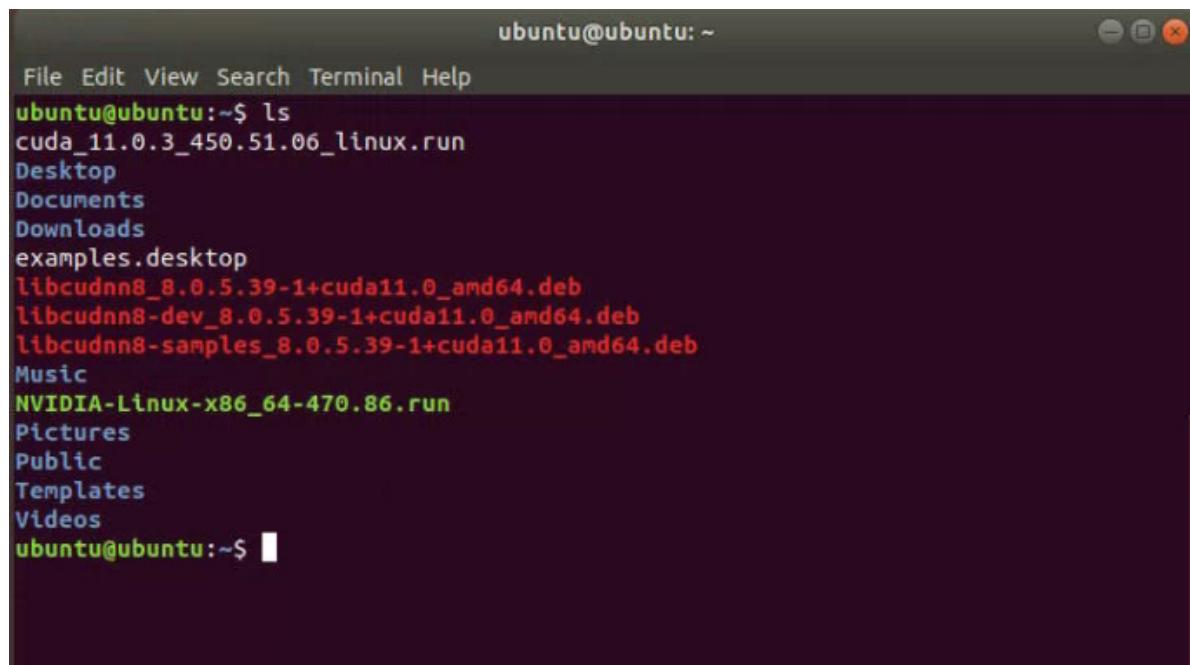
ubuntu安装cuda、cudnn

1、nvidia驱动安装

2、准备cuda、cudnn文件（此文档以cuda11.0, cudnn8.0.5为例）

官方下载链接：

```
cuda: https://developer.nvidia.com/cuda-toolkit-archive  
cudnn: https://developer.nvidia.com/rdp/form/cudnn-download-survey
```



```
ubuntu@ubuntu: ~  
File Edit View Search Terminal Help  
ubuntu@ubuntu:~$ ls  
cuda_11.0.3_450.51.06_linux.run  
Desktop  
Documents  
Downloads  
examples.desktop  
libcuda11.0_amd64.deb  
libcuda11.0_amd64.deb  
libcuda11.0_amd64.deb  
Music  
NVIDIA-Linux-x86_64-470.86.run  
Pictures  
Public  
Templates  
Videos  
ubuntu@ubuntu:~$
```

3、cuda文件赋权

```
sudo chmod +x cuda_11.0.3_450.51.06_linux.run
```

```
ubuntu@ubuntu:~  
File Edit View Search Terminal Help  
ubuntu@ubuntu:~$ sudo chmod +x cuda_11.0.3_450.51.06_linux.run  
[sudo] password for ubuntu:  
ubuntu@ubuntu:~$ ls  
cuda_11.0.3_450.51.06_linux.run  
Desktop  
Documents  
Downloads  
examples.desktop  
libcudnn8_8.0.5.39-1+cuda11.0_amd64.deb  
libcudnn8-dev_8.0.5.39-1+cuda11.0_amd64.deb  
libcudnn8-samples_8.0.5.39-1+cuda11.0_amd64.deb  
Music  
NVIDIA-Linux-x86_64-470.86.run  
Pictures  
Public  
Templates  
Videos  
ubuntu@ubuntu:~$ █
```

4、安装cuda

```
sudo ./cuda_11.0.3_450.51.06_linux.run
```

```
ubuntu@ubuntu:~  
File Edit View Search Terminal Help  
ubuntu@ubuntu:~$ ls  
cuda_11.0.3_450.51.06_linux.run  
Desktop  
Documents  
Downloads  
examples.desktop  
libcudnn8_8.0.5.39-1+cuda11.0_amd64.deb  
libcudnn8-dev_8.0.5.39-1+cuda11.0_amd64.deb  
libcudnn8-samples_8.0.5.39-1+cuda11.0_amd64.deb  
Music  
NVIDIA-Linux-x86_64-470.86.run  
Pictures  
Public  
Templates  
Videos  
ubuntu@ubuntu:~$ sudo ./cuda_11.0.3_450.51.06_linux.run  
█
```

- Do you accept the above EULA? --accept

```
ubuntu@ubuntu: ~
File Edit View Search Terminal Help

End User License Agreement
-----
NVIDIA Software License Agreement and CUDA Supplement to
Software License Agreement.

Preface
-----
The Software License Agreement in Chapter 1 and the Supplement
in Chapter 2 contain license terms and conditions that govern
the use of NVIDIA software. By accepting this agreement, you
agree to comply with all the terms and conditions applicable
to the product(s) included herein.

NVIDIA Driver

Do you accept the above EULA? (accept/decline/quit):
accept
```

- CUDA Installer取消勾选Driver，其余默认，选择Install

```
ubuntu@ubuntu: ~
File Edit View Search Terminal Help

CUDA Installer
- [ ] 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

Up/Down: Move | Left/Right: Expand | 'Enter': Select | 'A': Advanced options
```

- 安装完毕

```
ubuntu@ubuntu: ~
File Edit View Search Terminal Help
ubuntu@ubuntu:~$ sudo ./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/ubuntu/, 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 functionality 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
ubuntu@ubuntu:~$
```

5、添加环境变量

```
sudo vim ~/.bashrc
```

- 在文档最后添加

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

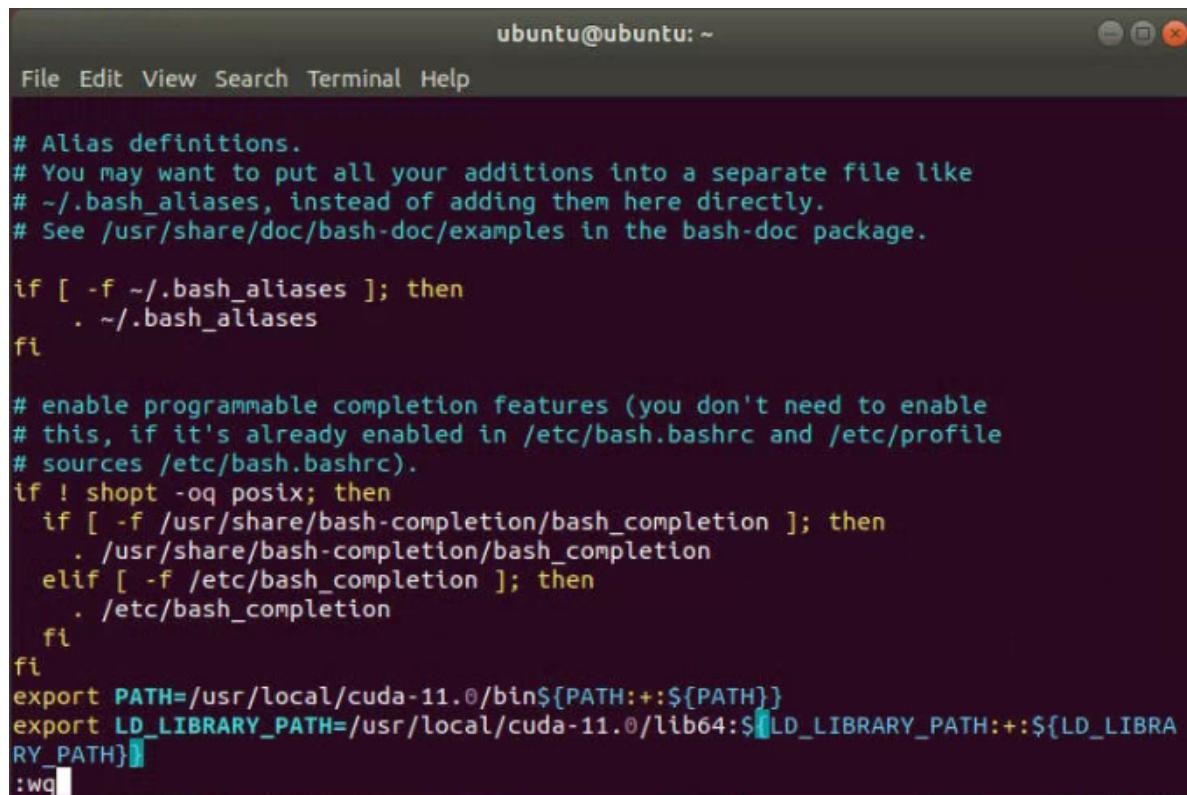
```
ubuntu@ubuntu: ~
File Edit View Search Terminal Help

# Alias definitions.
# You may want to put all your additions into a separate file like
# ~/.bash_aliases, instead of adding them here directly.
# See /usr/share/doc/bash-doc/examples in the bash-doc package.

if [ -f ~/.bash_aliases ]; then
    . ~/.bash_aliases
fi

# enable programmable completion features (you don't need to enable
# this, if it's already enabled in /etc/bash.bashrc and /etc/profile
# sources /etc/bash.bashrc).
if ! shopt -oq posix; then
    if [ -f /usr/share/bash-completion/bash_completion ]; then
        . /usr/share/bash-completion/bash_completion
    elif [ -f /etc/bash_completion ]; then
        . /etc/bash_completion
    fi
fi
export PATH=/usr/local/cuda-11.0/bin${PATH:+:$PATH}
export LD_LIBRARY_PATH=/usr/local/cuda-11.0/lib64:${LD_LIBRARY_PATH:+:$LD_LIBRARY_PATH}
"~/bashrc" 119L, 3916C
```

- shift+: (冒号) ,输入wq保存退出



```

ubuntu@ubuntu: ~

File Edit View Search Terminal Help

# Alias definitions.
# You may want to put all your additions into a separate file like
# ~/.bash_aliases, instead of adding them here directly.
# See /usr/share/doc/bash-doc/examples in the bash-doc package.

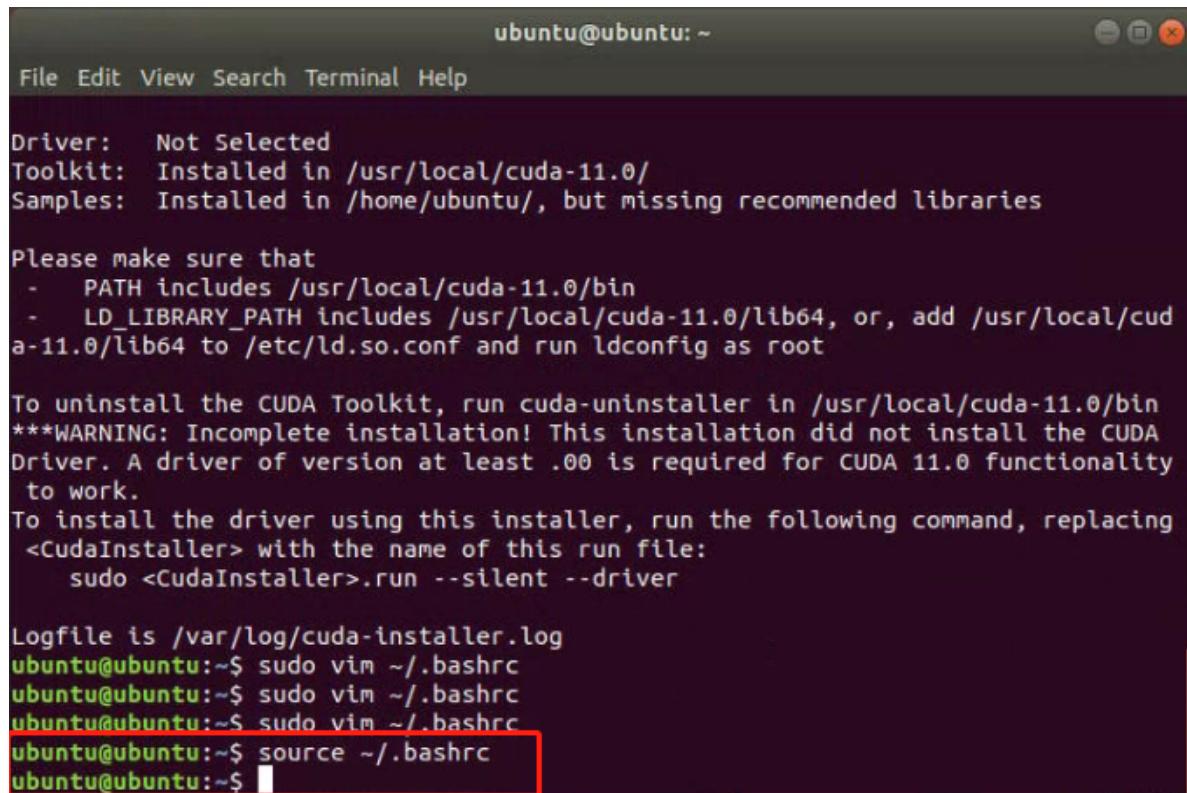
if [ -f ~/.bash_aliases ]; then
    . ~/.bash_aliases
fi

# enable programmable completion features (you don't need to enable
# this, if it's already enabled in /etc/bash.bashrc and /etc/profile
# sources /etc/bash.bashrc).
if ! shopt -oq posix; then
    if [ -f /usr/share/bash-completion/bash_completion ]; then
        . /usr/share/bash-completion/bash_completion
    elif [ -f /etc/bash_completion ]; then
        . /etc/bash_completion
    fi
fi
export PATH=/usr/local/cuda-11.0/bin${PATH:+:${PATH}}
export LD_LIBRARY_PATH=/usr/local/cuda-11.0/lib64:${LD_LIBRARY_PATH:+:${LD_LIBRARY_PATH}}
:wq

```

• 启用环境变量

```
source ~/.bashrc
```



```

ubuntu@ubuntu: ~

File Edit View Search Terminal Help

Driver: Not Selected
Toolkit: Installed in /usr/local/cuda-11.0/
Samples: Installed in /home/ubuntu/, 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

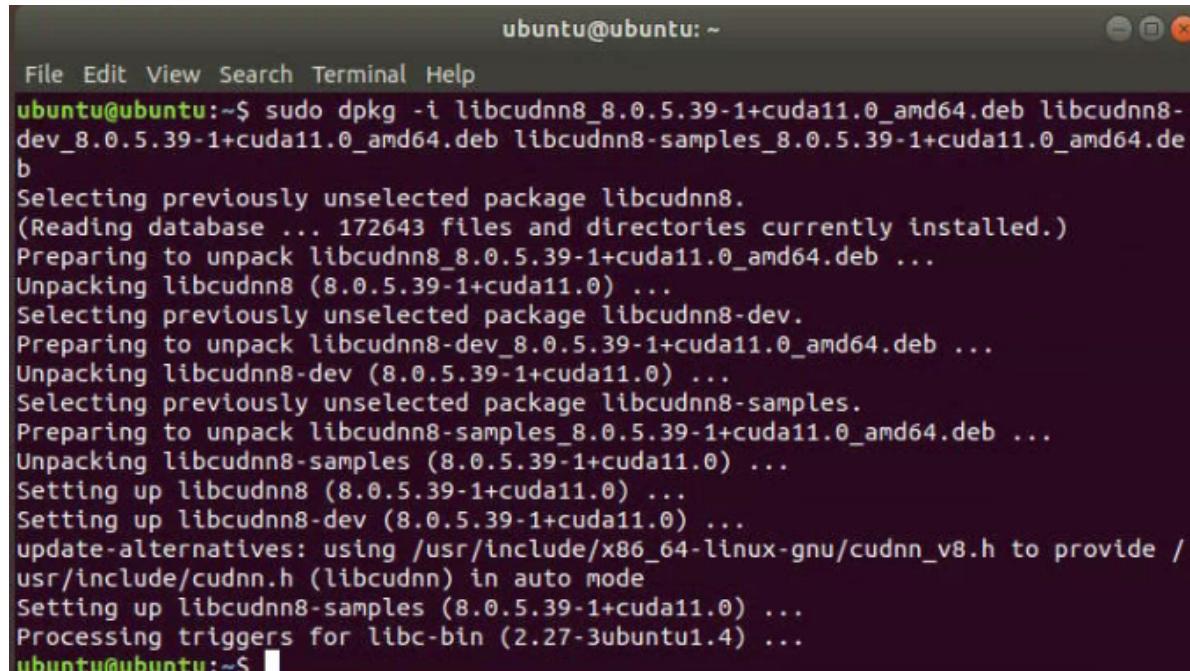
To uninstall the CUDA Toolkit, run cuda-uninstaller in /usr/local/cuda-11.0/bin
***WARNING: Incomplete installation! This installation did not install the CUDA Driver. A driver of version at least .00 is required for CUDA 11.0 functionality to work.
To install the driver using this installer, run the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

Logfile is /var/log/cuda-installer.log
ubuntu@ubuntu:~$ sudo vim ~/.bashrc
ubuntu@ubuntu:~$ sudo vim ~/.bashrc
ubuntu@ubuntu:~$ sudo vim ~/.bashrc
ubuntu@ubuntu:~$ source ~/.bashrc
ubuntu@ubuntu:~$ 

```

6、安装cudnn

```
sudo dpkg -i libcudnn8_8.0.5.39-1+cuda11.0_amd64.deb  
sudo dpkg -i libcudnn8-dev_8.0.5.39-1+cuda11.0_amd64.deb  
sudo dpkg -i libcudnn8-samples_8.0.5.39-1+cuda11.0_amd64.deb  
此三条命令可合成一条: sudo dpkg -i libcudnn8_8.0.5.39-1+cuda11.0_amd64.deb  
libcudnn8-dev_8.0.5.39-1+cuda11.0_amd64.deb libcudnn8-samples_8.0.5.39-  
1+cuda11.0_amd64.deb
```



```
ubuntu@ubuntu: ~  
File Edit View Search Terminal Help  
ubuntu@ubuntu:~$ sudo dpkg -i libcudnn8_8.0.5.39-1+cuda11.0_amd64.deb libcudnn8-  
dev_8.0.5.39-1+cuda11.0_amd64.deb libcudnn8-samples_8.0.5.39-1+cuda11.0_amd64.de  
b  
Selecting previously unselected package libcudnn8.  
(Reading database ... 172643 files and directories currently installed.)  
Preparing to unpack libcudnn8_8.0.5.39-1+cuda11.0_amd64.deb ...  
Unpacking libcudnn8 (8.0.5.39-1+cuda11.0) ...  
Selecting previously unselected package libcudnn8-dev.  
Preparing to unpack libcudnn8-dev_8.0.5.39-1+cuda11.0_amd64.deb ...  
Unpacking libcudnn8-dev (8.0.5.39-1+cuda11.0) ...  
Selecting previously unselected package libcudnn8-samples.  
Preparing to unpack libcudnn8-samples_8.0.5.39-1+cuda11.0_amd64.deb ...  
Unpacking libcudnn8-samples (8.0.5.39-1+cuda11.0) ...  
Setting up libcudnn8 (8.0.5.39-1+cuda11.0) ...  
Setting up libcudnn8-dev (8.0.5.39-1+cuda11.0) ...  
update-alternatives: using /usr/include/x86_64-linux-gnu/cudnn_v8.h to provide /  
usr/include/cudnn.h (libcudnn) in auto mode  
Setting up libcudnn8-samples (8.0.5.39-1+cuda11.0) ...  
Processing triggers for libc-bin (2.27-3ubuntu1.4) ...  
ubuntu@ubuntu:~$
```

7、验证cuda

- 确保已经安装了make、gcc、g++，若无，请安装

安装make、gcc、g++

```
sudo apt install make gcc g++ -y
```

- 进入bandwidthTest验证

```
cd NVIDIA_CUDA-11.0_Samples/1_Utils/bandwidthTest/  
sudo make
```

```
ubuntu@ubuntu:~/NVIDIA_CUDA-11.0_Samples/1_Utilities/bandwidthTest$ ls
bandwidthTest.cu  Makefile  NsightEclipse.xml  readme.txt
ubuntu@ubuntu:~/NVIDIA_CUDA-11.0_Samples/1_Utilities/bandwidthTest$ sudo make
/usr/local/cuda/bin/nvcc -ccbin g++ -I../../common/inc -m64 -gencode arch=compute_35,code=sm_35 -gencode arch=compute_37,code=sm_37 -gencode arch=compute_50,code=sm_50 -gencode arch=compute_52,code=sm_52 -gencode arch=compute_60,code=sm_60 -gencode arch=compute_61,code=sm_61 -gencode arch=compute_70,code=sm_70 -gencode arch=compute_75,code=sm_75 -gencode arch=compute_80,code=sm_80 -gencode arch=compute_80,code=sm_80 -o bandwidthTest.o -c bandwidthTest.cu
nvcc warning : The 'compute_35', 'compute_37', 'compute_50', 'sm_35', 'sm_37' and 'sm_50' architectures are deprecated, and may be removed in a future release (Use -Wno-deprecated-gpu-targets to suppress warning).
/usr/local/cuda/bin/nvcc -ccbin g++ -m64 -gencode arch=compute_35,code=sm_35 -gencode arch=compute_37,code=sm_37 -gencode arch=compute_50,code=sm_50 -gencode arch=compute_52,code=sm_52 -gencode arch=compute_60,code=sm_60 -gencode arch=compute_61,code=sm_61 -gencode arch=compute_70,code=sm_70 -gencode arch=compute_75,code=sm_75 -gencode arch=compute_80,code=sm_80 -gencode arch=compute_80,code=sm_80 -o bandwidthTest bandwidthTest.o
nvcc warning : The 'compute_35', 'compute_37', 'compute_50', 'sm_35', 'sm_37' and 'sm_50' architectures are deprecated, and may be removed in a future release (Use -Wno-deprecated-gpu-targets to suppress warning).
mkdir -p ../../bin/x86_64/linux/release
cp bandwidthTest ../../bin/x86_64/linux/release
ubuntu@ubuntu:~/NVIDIA_CUDA-11.0_Samples/1_Utilities/bandwidthTest$ ls
bandwidthTest  bandwidthTest.o  NsightEclipse.xml
bandwidthTest.cu  Makefile  readme.txt
ubuntu@ubuntu:~/NVIDIA_CUDA-11.0_Samples/1_Utilities/bandwidthTest$
```

- 执行bandwidthTest

```
./bandwidthTest
```

```
ubuntu@ubuntu:~/NVIDIA_CUDA-11.0_Samples/1_Utilities/bandwidthTest$ ./bandwidthTest
[CUDA Bandwidth Test] - Starting...
Running on...

Device 0: NVIDIA GeForce RTX 3090
Quick Mode

Host to Device Bandwidth, 1 Device(s)
PINNED Memory Transfers
Transfer Size (Bytes)      Bandwidth(GB/s)
32000000                  10.2

Device to Host Bandwidth, 1 Device(s)
PINNED Memory Transfers
Transfer Size (Bytes)      Bandwidth(GB/s)
32000000                  11.2

Device to Device Bandwidth, 1 Device(s)
PINNED Memory Transfers
Transfer Size (Bytes)      Bandwidth(GB/s)
32000000                  786.4

Result = PASS

NOTE: The CUDA Samples are not meant for performance measurements. Results may vary when GPU Boost is enabled.
ubuntu@ubuntu:~/NVIDIA_CUDA-11.0_Samples/1_Utilities/bandwidthTest$
```

- 进入deviceQueryDrv验证

```
cd NVIDIA_CUDA-11.0_Samples/1_Utils/deviceQueryDrv/  
sudo make  
.deviceQueryDrv
```

```
ubuntu@ubuntu:~/NVIDIA_CUDA-11.0_Samples/1_Utils/deviceQueryDrv$ ls  
deviceQueryDrv    deviceQueryDrv.o  NsightEclipse.xml  
deviceQueryDrv.cpp  Makefile      readme.txt  
ubuntu@ubuntu:~/NVIDIA_CUDA-11.0_Samples/1_Utils/deviceQueryDrv$ ./deviceQueryDrv  
.deviceQueryDrv Starting...  
  
CUDA Device Query (Driver API) statically linked version  
Detected 2 CUDA Capable device(s)  
  
Device 0: "NVIDIA GeForce RTX 3090"  
  CUDA Driver Version:          11.4  
  CUDA Capability Major/Minor version number: 8.6  
  Total amount of global memory: 24265 MBytes (25443893248 bytes)  
MapSMtoCores for SM 8.6 is undefined. Default to use 64 Cores/SM  
MapSMtoCores for SM 8.6 is undefined. Default to use 64 Cores/SM  
  (82) Multiprocessors, ( 64) CUDA Cores/MP:   5248 CUDA Cores  
  GPU Max Clock rate:           1695 MHz (1.70 GHz)  
  Memory Clock rate:           9751 Mhz  
  Memory Bus Width:            384-bit  
  L2 Cache Size:               6291456 bytes  
  Max Texture Dimension Sizes  1D=(131072) 2D=(131072, 65536) 3D=(16384, 16384, 16384)  
  Maximum Layered 1D Texture Size, (num) layers  1D=(32768), 2048 layers  
  Maximum Layered 2D Texture Size, (num) layers  2D=(32768, 32768), 2048 layers  
  Total amount of constant memory:             65536 bytes  
  Total amount of shared memory per block:     49152 bytes  
  Total number of registers available per block: 65536  
  Warp size:                                32  
  Maximum number of threads per multiprocessor: 1536  
  Maximum number of threads per block:         1024  
  Max dimension size of a thread block (x,y,z): (1024, 1024, 64)  
  Max dimension size of a grid size (x,y,z):  (2147483647, 65535, 65535)  
  Texture alignment:                         512 bytes  
  Maximum memory pitch:                     2147483647 bytes  
  Concurrent copy and kernel execution:    Yes with 2 copy engine(s)  
  Run time limit on kernels:                Yes  
  Integrated GPU sharing Host Memory:       No  
  Support host page-locked memory mapping: Yes  
  Concurrent kernel execution:              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 / 2 / 0  
  Compute Mode:  
    < Default (multiple host threads can use ::cudaSetDevice() with device simultaneously) >  
  
Device 1: "NVIDIA GeForce RTX 3090"  
  CUDA Driver Version:          11.4  
  CUDA Capability Major/Minor version number: 8.6  
  Total amount of global memory: 24268 MBytes (25447170048 bytes)  
MapSMtoCores for SM 8.6 is undefined. Default to use 64 Cores/SM
```

```
ubuntu@ubuntu: ~/NVIDIA_CUDA-11.0_Samples/1_Utilities/deviceQueryDrv$ ./deviceQuery
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)
Concurrent kernel execution: 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 / 2 / 0
Compute Mode:
< Default (multiple host threads can use ::cudaSetDevice() with device simultaneously) >

Device 1: "NVIDIA GeForce RTX 3090"
CUDA Driver Version: 11.4
CUDA Capability Major/Minor version number: 8.6
Total amount of global memory: 24268 MBytes (25447170048 bytes)
MapSMtoCores for SM 8.6 is undefined. Default to use 64 Cores/SM
MapSMtoCores for SM 8.6 is undefined. Default to use 64 Cores/SM
(82) Multiprocessors, ( 64) CUDA Cores/MP: 5248 CUDA Cores
GPU Max Clock rate: 1695 MHz (1.70 GHz)
Memory Clock rate: 9751 Mhz
Memory Bus Width: 384-bit
L2 Cache Size: 6291456 bytes
Max Texture Dimension Sizes
Maximum Layered 1D Texture Size, (num) layers 10=(131072) 2D=(131072, 65536) 3D=(16384, 16384, 16384)
Maximum Layered 2D Texture Size, (num) layers 10=(32768), 2048 layers
Maximum Layered 3D Texture Size, (num) layers 2D=(32768, 32768), 2048 layers
Total amount of constant memory: 65536 bytes
Total amount of shared memory per block: 49152 bytes
Total number of registers available per block: 65536
Warp size: 32
Maximum number of threads per multiprocessor: 1536
Maximum number of threads per block: 1024
Max dimension size of a thread block (x,y,z): (1024, 1024, 64)
Max dimension size of a grid size (x,y,z): (2147483647, 65535, 65535)
Texture alignment: 512 bytes
Maximum memory pitch: 2147483647 bytes
Concurrent copy and kernel execution: Yes with 2 copy engine(s)
Run time limit on kernels: Yes
Integrated GPU sharing Host Memory: No
Support host page-locked memory mapping: Yes
Concurrent kernel execution: 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 / 130 / 0
Compute Mode:
< Default (multiple host threads can use ::cudaSetDevice() with device simultaneously) >
> Peer-to-Peer (P2P) access from NVIDIA GeForce RTX 3090 (GPU0) -> NVIDIA GeForce RTX 3090 (GPU1) : No
> Peer-to-Peer (P2P) access from NVIDIA GeForce RTX 3090 (GPU1) -> NVIDIA GeForce RTX 3090 (GPU0) : No
Result = PASS
ubuntu@ubuntu:~/NVIDIA_CUDA-11.0_Samples/1_Utilities/deviceQueryDrv$
```

验证结果为*Result = PASS*，则为验证通过

8、验证cudnn

```
cd /usr/src/cudnn_samples_v8/mnistCUDNN/  
sudo make  
../mnistCUDNN
```

```
ubuntu@ubuntu:/usr/src/cudnn_samples_v8/mnistCUDNN
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)
ubuntu@ubuntu:/usr/src/cudnn_samples_v8/mnistCUDNN ./mnistCUDNN
Executing: mnistCUDNN
cudnnGetVersion() : 8005 , CUDNN_VERSION from cudnn.h : 8005 (8.0.5)
Host compiler version : GCC 7.5.0

There are 2 CUDA capable devices on your machine :
device 0 : sms 82 Capabilities 8.0, SmClock 1695.0 Mhz, MemSize (Mb) 24265, MemClock 9751.0 Mhz, Ecc=0, boardGroupID=0
device 1 : sms 82 Capabilities 8.0, SmClock 1695.0 Mhz, MemSize (Mb) 24268, MemClock 9751.0 Mhz, Ecc=0, boardGroupID=1
Using device 0

Testing single precision
Loading binary file data/conv1.bin
Loading binary file data/conv1.bias.bn
Loading binary file data/conv2.bin
Loading binary file data/conv2.bias.bn
Loading binary file data/p1p1.bn
Loading binary file data/p1p1.bias.bn
Loading binary file data/p2p2.bn
Loading binary file data/p2p2.bias.bn
Loading image data/one_28x28.png
Performing forward propagation ...
Testing cudnnGetConvolutionForwardAlgorithm v7 ...
~~~~~ CUDNN_STATUS_SUCCESS For Algo 1: -1.000000 time requiring 0 memory
~~~~~ CUDNN_STATUS_SUCCESS For Algo 0: -1.000000 time requiring 0 memory
~~~~~ CUDNN_STATUS_SUCCESS For Algo 2: -1.000000 time requiring 0 memory
~~~~~ CUDNN_STATUS_SUCCESS For Algo 7: -1.000000 time requiring 2057744 memory
~~~~~ CUDNN_STATUS_SUCCESS For Algo 4: -1.000000 time requiring 184784 memory
~~~~~ CUDNN_STATUS_SUCCESS For Algo 5: -1.000000 time requiring 178432 memory
~~~~~ CUDNN_STATUS_NOT_SUPPORTED For Algo 6: -1.000000 time requiring 0 memory
~~~~~ CUDNN_STATUS_NOT_SUPPORTED For Algo 3: -1.000000 time requiring 0 memory
Testing cudnnFindConvolutionForwardAlgorithm ...
~~~~~ CUDNN_STATUS_SUCCESS For Algo 2: 0.018432 time requiring 0 memory
~~~~~ CUDNN_STATUS_SUCCESS For Algo 1: 0.020480 time requiring 0 memory
~~~~~ CUDNN_STATUS_SUCCESS For Algo 0: 0.021280 time requiring 0 memory
~~~~~ CUDNN_STATUS_SUCCESS For Algo 7: 0.081920 time requiring 2057744 memory
~~~~~ CUDNN_STATUS_SUCCESS For Algo 5: 0.099392 time requiring 178432 memory
~~~~~ CUDNN_STATUS_SUCCESS For Algo 4: 7.784280 time requiring 184784 memory
~~~~~ CUDNN_STATUS_NOT_SUPPORTED For Algo 6: -1.000000 time requiring 0 memory
~~~~~ CUDNN_STATUS_NOT_SUPPORTED For Algo 3: -1.000000 time requiring 0 memory
Testing cudnnGetConvolutionForwardAlgorithm_v7 ...
~~~~~ CUDNN_STATUS_SUCCESS For Algo 1: -1.000000 time requiring 0 memory
~~~~~ CUDNN_STATUS_SUCCESS For Algo 0: -1.000000 time requiring 0 memory
~~~~~ CUDNN_STATUS_SUCCESS For Algo 2: -1.000000 time requiring 0 memory
~~~~~ CUDNN_STATUS_SUCCESS For Algo 5: -1.000000 time requiring 4656640 memory
~~~~~ CUDNN_STATUS_SUCCESS For Algo 4: -1.000000 time requiring 2450080 memory
~~~~~ CUDNN_STATUS_SUCCESS For Algo 7: -1.000000 time requiring 1433120 memory
~~~~~ CUDNN_STATUS_NOT_SUPPORTED For Algo 6: -1.000000 time requiring 0 memory
~~~~~ CUDNN_STATUS_NOT_SUPPORTED For Algo 3: -1.000000 time requiring 0 memory
Resulting weights from Softmax:
0.0000001 0.0000001 0.0000001 0.0000563 0.0000001 0.0000012 0.0000017 0.0000010 0.0000001
Loading image data/three_28x28.png
Performing forward propagation ...
Testing cudnnGetConvolutionForwardAlgorithm_v7 ...
~~~~~ CUDNN_STATUS_SUCCESS For Algo 1: -1.000000 time requiring 0 memory
~~~~~ CUDNN_STATUS_SUCCESS For Algo 0: -1.000000 time requiring 0 memory
~~~~~ CUDNN_STATUS_SUCCESS For Algo 2: -1.000000 time requiring 0 memory
~~~~~ CUDNN_STATUS_SUCCESS For Algo 4: -1.000000 time requiring 184784 memory
~~~~~ CUDNN_STATUS_SUCCESS For Algo 5: -1.000000 time requiring 178432 memory
~~~~~ CUDNN_STATUS_SUCCESS For Algo 7: -1.000000 time requiring 2057744 memory
~~~~~ CUDNN_STATUS_NOT_SUPPORTED For Algo 6: -1.000000 time requiring 0 memory
~~~~~ CUDNN_STATUS_NOT_SUPPORTED For Algo 3: -1.000000 time requiring 0 memory
Testing cudnnFindConvolutionForwardAlgorithm ...
~~~~~ CUDNN_STATUS_SUCCESS For Algo 2: 0.021504 time requiring 0 memory
~~~~~ CUDNN_STATUS_SUCCESS For Algo 0: 0.023552 time requiring 0 memory
~~~~~ CUDNN_STATUS_SUCCESS For Algo 1: 0.025600 time requiring 0 memory
~~~~~ CUDNN_STATUS_SUCCESS For Algo 7: 0.065536 time requiring 2057744 memory
~~~~~ CUDNN_STATUS_SUCCESS For Algo 4: 0.070816 time requiring 184784 memory
~~~~~ CUDNN_STATUS_SUCCESS For Algo 5: 0.073728 time requiring 178432 memory
~~~~~ CUDNN_STATUS_NOT_SUPPORTED For Algo 6: -1.000000 time requiring 0 memory
~~~~~ CUDNN_STATUS_NOT_SUPPORTED For Algo 3: -1.000000 time requiring 0 memory
Testing cudnnGetConvolutionForwardAlgorithm_v7 ...
~~~~~ CUDNN_STATUS_SUCCESS For Algo 1: -1.000000 time requiring 2000 memory
~~~~~ CUDNN_STATUS_SUCCESS For Algo 5: -1.000000 time requiring 4656640 memory
~~~~~ CUDNN_STATUS_SUCCESS For Algo 0: -1.000000 time requiring 0 memory
~~~~~ CUDNN_STATUS_SUCCESS For Algo 2: -1.000000 time requiring 0 memory
~~~~~ CUDNN_STATUS_SUCCESS For Algo 4: -1.000000 time requiring 2450080 memory
~~~~~ CUDNN_STATUS_SUCCESS For Algo 7: -1.000000 time requiring 1433120 memory
~~~~~ CUDNN_STATUS_NOT_SUPPORTED For Algo 6: -1.000000 time requiring 0 memory
~~~~~ CUDNN_STATUS_NOT_SUPPORTED For Algo 3: -1.000000 time requiring 0 memory
Testing cudnnFindConvolutionForwardAlgorithm ...
~~~~~ CUDNN_STATUS_SUCCESS For Algo 0: 0.053248 time requiring 0 memory
~~~~~ CUDNN_STATUS_SUCCESS For Algo 4: 0.076800 time requiring 2450080 memory
~~~~~ CUDNN_STATUS_SUCCESS For Algo 2: 0.086672 time requiring 0 memory
~~~~~ CUDNN_STATUS_SUCCESS For Algo 7: 0.083968 time requiring 1433120 memory
~~~~~ CUDNN_STATUS_SUCCESS For Algo 5: 0.084992 time requiring 4656640 memory
~~~~~ CUDNN_STATUS_SUCCESS For Algo 1: 0.168960 time requiring 2000 memory
~~~~~ CUDNN_STATUS_NOT_SUPPORTED For Algo 6: -1.000000 time requiring 0 memory
~~~~~ CUDNN_STATUS_NOT_SUPPORTED For Algo 3: -1.000000 time requiring 0 memory
Resulting weights from Softmax:
0.0000000 0.0000000 1.0000000 0.0000000 0.0000714 0.0000000 0.0000000 0.0000000 0.0000000
Loading image data/five_28x28.png
Performing forward propagation ...
Resulting weights from Softmax:
0.0000000 0.0000008 0.0000000 0.0000002 0.0000000 1.0000000 0.0000154 0.0000000 0.0000012 0.0000006

Result of classification: 1 3 5

Test passed!
ubuntu@ubuntu:/usr/src/cudnn_samples_v8/mnistCUDNN$
```

结果为*Test passed!* 则通过验证

9、验证过程中出现的问题

```
ubuntu@ubuntu:~/NVIDIA_CUDA-11.0_Samples/1_Utils/bandwidthTest$ sudo make
[sudo] ubuntu 的密码:
/usr/local/cuda/bin/nvcc -ccbin g++ -I../../common/inc -m64 -gencode arch=compute_35,code=sm_35 -gencode arch=compute_37,code=sm_37 -gencode arch=compute_50,code=sm_50 -gencode arch=compute_52,code=sm_52 -gencode arch=compute_60,code=sm_60 -gencode arch=compute_61,code=sm_61 -gencode arch=compute_70,code=sm_70 -gencode arch=compute_75,code=sm_75 -gencode arch=compute_80,code=sm_80 -gencode arch=compute_80,code=compute_80 -o bandwidthTest.o -c bandwidthTest.cu
nvcc warning : The 'compute_35', 'compute_37', 'compute_50', 'sm_35', 'sm_37' and 'sm_50' architectures are deprecated, and may be removed in a future release (Use -Wno-deprecated-gpu-targets to suppress warning).
g++: No such file or directory
nvcc fatal   : Failed to preprocess host compiler properties.
Makefile:309: recipe for target 'bandwidthTest.o' failed
make: *** [bandwidthTest.o] Error 1
```

安装*make*、*gcc*、*g++*即可解决



宽泛科技

专业信息化解决方案供应商



📞 400-180-8812

🌐 WWW.kuanfans.com

📍 上海市闵行区申滨南路1156号