Search
Search

Recent Posts

Solve video tearing on Intel + Ubuntu (XFCE 4)
Overriding mouse/touch pad speed in Ubuntu
2012 in review
Solve video tearing on Intel + Ubuntu (Unity and GNOME Shell)
Installing CUDA 5 on

Installing CUDA on Ubuntu 12.04

Posted on May 11, 2012 by Utkarsh Jaiswal

This post is now outdated. Please refer this link for an updated version.

If you're looking for a quick and easy way to get started with GPGPU computing, you really can't go wrong with nVidia's CUDA. It is a parallel computing architecture that harnesses the power of GPUs in order to achieve significant speedups in problems that would have otherwise taken a *significantly* longer time while executing on the CPU. It is the most mature architecture for GPGPU computing, with a wide number of libraries based around it. This guide is going to cover the installation of the CUDA toolkit and SDK on Ubuntu, along with the necessary development drivers.

NOTE – For CUDA to work, you must have an nVidia GPU which is CUDA capable. If you have an ATI GPU, this guide is not for you. You can, however, look into OpenCL.

If your GPU meets the requirements, head over to the

Ubuntu 12.04

Archives

December 2013
February 2013
December 2012
September 2012
August 2012
May 2012
February 2012
January 2012

Categories

Uncategorized

Meta

Register
Log in
Entries RSS
Comments RSS

CUDA Downloads page and download the toolkit, drivers and SDK from under the Linux section, taking care to choose the 32 or 64-bit version depending on your system. If you're not sure, run

```
uname -m
```

in a terminal. i686 denotes a 32-bit system, and x86_64 denotes a 64-bit one. For the toolkit, I chose the one titled Ubuntu 11.04, although either of the Ubuntu toolkits should work just fine.

Save all three files in an easy to access location, like your Home folder. Do not proceed with this guide until you've either memorized the following steps or printed them for easy reference!

STEP I – Driver installation

Make sure the requisite tools are installed using the following command -

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

Next, blacklist the required modules (so that they don't interfere with the driver installation) -

```
gksu gedit /etc/modprobe.d/blacklist.conf
```

Add the following lines to the end of the file, one per line -

blacklist amd76x_edac
blacklist vga16fb
blacklist nouveau
blacklist rivafb

blacklist nvidiafb
blacklist rivatv

Save the file and exit gedit.

Update – WARNING – Reader Bart points out in the comments that the purge nvidia command resulted in a prompt to remove ubuntu-desktop (for him). If this occurs, do not proceed or it will wreck your GUI. You could reinstall ubuntu-desktop after the entire process but I'm not sure if that'll affect your data (configs for each app).*

In order to get rid of any nVidia residuals, run the following command in a terminal -

```
sudo apt-get remove --purge nvidia*
```

This may take a while, so be patient. Once it's done, reboot your machine. At the login screen, don't login just yet. Press Ctrl+Alt+F1 to switch to a text-based login. Login and switch to the directory which contains the downloaded drivers, toolkit [Follow] Run the following


```
sudo serv
chmod +x
```

where dev ur driver. Next, start

```
sudo ./devdriver*.run
```

Follow the onscreen instructions. If the installer throws up an error about nouveau still running, allow it to create a blacklist for nouveau, quit the installation and

reboot. In that case, run the following commands again -

```
sudo service lightdm stop
sudo ./devdriver*.run
```

The installation should now proceed smoothly. When it asks you if you want the 32-bit libraries and if you want it to edit xorg.conf to use these drivers by default, allow both.

Reboot once the installation completes.

STEP II – CUDA toolkit installation

Next, enter the following in a terminal window (in the directory where the files are stored) -

```
chmod +x cudatoolkit*.run
sudo ./cudatoolkit*.run
```

where cudatoolkit*.run is the full name of the toolkit installer. I recommend leaving the installation path to its default setting (/usr/local/cuda) unless you have a specific reason for not doing so.

STEP III – CUDA SDK installation

Update – I've just found out that the SDK must be installed as a regular user (and not as root) according to the "nVidia CUDA C Getting Started Guide for Linux" (refer pg 11). Apparently, this is to prevent access issues with the SDK files.
Also, readers Fernest Hall and Adub have mentioned tips in the comments that might be useful for some readers, although they haven't worked for me. Thanks

guys!

Update 2 – Reader Alan has mentioned a link in the comment section. Although I haven't checked it out yet, it might work for you.

Once the toolkit is installed, enter the following in a terminal -

```
chmod +x gpucomputingsdk*.run
./gpucomputingsdk*.run
```

where gpucomputingsdk*.run is the full name of the SDK installer. Again, follow the instructions onscreen to complete the installation.

You're now ready to journey into the world of CUDA and GPGPU computing. If you're looking for books on the same, check out this page.

Credit for the driver installation goes to this awesome video.

Share this:

Related

Installing CUDA 5 on Ubuntu 12.04
In "12.04"

Get GNOME Shell on ATI + Ubuntu 11.10
In "11.10"

Solve video tearing on nVidia + GNOME 3
In "GNOME 3"

This entry was tagged 12.04, CUDA, driver, GPGPU, guide, how, install, Linux, nVidia, to, Ubuntu. Bookmark the permalink.

← Installing Ubuntu on VirtualBox

Installing Jelly Bean on the Galaxy S →

79 thoughts on “Installing CUDA on Ubuntu 12.04”

avtacha says:
May 13, 2012 at 1:16 am

Thanks for this. Very helpful to me

Reply

Utkarsh Jaiswal says:
May 13, 2012 at 6:15 pm

You're welcome

Reply

mouna says:
January 10, 2013 at 3:38 pm

I tried to install cuda 5 on ubunto 12.04 but i receive the following error :
E: Impossible de trouver le paquet libxll-dev
E: Impossible de trouver le paquet libgll-mesa-glx
E: Impossible de trouver le paquet libglul-mesa
E: Impossible de trouver le paquet libglul-mesa-dev

Utkarsh Jaiswal says:
January 11, 2013 at 9:24 pm

The correct package names are libx11-dev, libgl1-mesa-glx, libglu1-mesa and libglu1-mesa-dev. You seem to have substituted the digits for the letter 'l'. Please copy-paste the command as listed in the guide.

cerx says:

May 15, 2012 at 7:37 pm

Tried your installation method of Ubuntu 12.04 installation since the one in the nVidia documentation did not work for me. Specifically, even though CUDA gets installed and apps like VMD work fine but I can't compile any of the gpucomputing examples – lot's of compiler errors. Strangely, OpenCL examples work just fine. Unfortunately using your instructions did not remedy that. I gues we will have to wait for nVidia to update.

Reply

Utkarsh Jaiswal says:

May 15, 2012 at 10:18 pm

I'm getting errors in the gpucomputing examples too. I thought they were on my part since I'm new to CUDA! Btw, I just found out that gpucomputing must be installed as a regular user (and not as root). Although that didn't help in my case, it might be of some use to you.

Reply

cerx says:

May 16, 2012 at 12:13 am

I tried both, installation of gpucomputing examples as root and regular user – no

difference.

Adub says:

May 23, 2012 at 8:32 am

Indeed, thanks very much! I'm running through this on my Ubuntu setup and everything is going swimingly. I even had the nouveau driver running and everything. Thanks again!

Reply

Adub says:

May 23, 2012 at 8:33 am

Oh, and the make errors for the SDK appear to be a bug on their side.

Reply

Adub says:

May 23, 2012 at 8:53 am

BTW, this seemed to fix any issues: http://troylee2008.blogspot.com/2012/05/linking-error-while-compiling-cuda-sdk.html

Reply

Utkarsh Jaiswal says:

May 23, 2012 at 11:11 am

Thanks a lot Adub, I'll check it out

Reply

Alex says:
May 27, 2012 at 2:37 pm

Thank you so much for posting this article online!

Reply

Utkarsh Jaiswal says:
May 27, 2012 at 4:04 pm

Always a pleasure

Reply

Fernest Hall says:
June 1, 2012 at 5:59 pm

I had to do a bit more stuff to compile the CUDA SDK.

If you want the SDK to compile, but get errors:
In the Terminal:

1.) sudo gedit /etc/ld.so.conf
and add this lines at the end:
/usr/local/cuda/lib64
/usr/local/cuda/lib

2.) ldconfig

Assuming you installed to /usr/local/cuda:
3.) export LD_LIBRARY_PATH=/usr/local/cuda/lib:$LD_LIBRARY_PATH
4.) export LD_LIBRARY_PATH=/usr/local/cuda/lib64:$LD_LIBRARY_PATH

If the SDK still does not compile, but aborts during the compiling of nbody, do following:
gedit ~/NVIDIA_GPU_Computing_SDK/C/common/common.mk
find all the lines which begin with “LIB += ” and contain ${OPENGLLIB} as well as $(RENDERCHECKGLLIB)
(In my version there were 3 of them.)
Switch in those lines the position of ${OPENGLLIB} and $(RENDERCHECKGLLIB)

(For me this did the trick. If you find this helpfull, feel free to add it to your post)
Cheers.

Reply

Utkarsh Jaiswal says:
June 1, 2012 at 7:43 pm

Thank you, I’ll check it out

UPDATE – It managed to get farther, but the SDK still won’t compile entirely. Thanks anyway, I’ve mentioned your comment in the article

Reply

Vivek KAul says:
July 28, 2012 at 8:13 am

Hi Fernest,
I had to do what you said to remove compile errors. I do get some warnings though like '-Wimplicit does not work'. Is that fine? Do you use gcc 4.6.3? But thanks you saved my day. I am using GTX 690 so got nervous about it.
Regards
Vivek

Reply

Pandu Aji Wirawan says:
August 31, 2012 at 5:43 am

Hey! Your tips worked for me. thank you

Reply

Will says:
June 2, 2012 at 9:40 pm

Will this work if I have hybrid graphics (Intel integrated and NVIDIA discrete)? I recently tried installing the NVIDIA driver for a 560M card and I guess Ubuntu always uses the Intel card by default. I also had a problem where the resolution changed to the lowest setting, so I reinstalled Ubuntu. From everything else I've seen online I

might need bumblebee instead of the NVIDIA driver. You have a pretty solid guide here either way.

TL;DR What assumptions are you making in this tutorial, are you using hybrid graphics or just NVIDIA?

Reply

Utkarsh Jaiswal says:
June 3, 2012 at 1:38 pm

Thanks for your kind words Will I don't own a machine with hybrid graphics, so I've tested it on ones with just nVidia graphics – a laptop with an 8600M GT, and a desktop with a GTX 260. Sorry but I'm not really sure about your case.

Reply

Will says:
June 3, 2012 at 6:09 pm

I actually figured out my situation, so I'll post here for others to see: basically if you have hybrid graphics, do not install the NVIDIA driver from the website. Download the 4.2 Toolkit and the SDK, and install the bumblebee package (can easily be found via Google). If this is working then you may run the command "optirun glxspheres" or "optirun glxgears" (from the package

mesa-apps) to test out a program running on the NVIDIA card.

Then of course, make sure your PATH and LD_LIBRARY_PATH variables are updated in .bashrc for the CUDA bin and lib respectively (don't forget lib64 for x64!). Reboot or logout to save changes.

Now install the Toolkit and SDK as usual.

The last thing that might cause issues is the version of gcc and g++. Long story short, make sure the pointers gcc and g++ in /usr/bin (and subsequently /usr/local/cuda/bin) are pointing to gcc-4.5 and g++-4.5 (can get these with apt-get) since they are the most recent versions supported by nvcc. Use the soft-link command to achieve this.

The easiest program to make and run in the SDK is vectorAdd since it has one .cu file and no other .cpp or .h files. Make it and go to the bottom of the /C/bin folder and run it via "optirun ./vectoradd". It should run perfectly. Now the next step would be to run matrixMul, since this uses .cpp files and will test your gcc/g++ dependencies. If you have the correct version (and both versions match) and all soft-links are done correctly, it should compile and run via "optirun ./matrixMul". Then you can move on and address the other examples in the SDK, but this is at least enough to compile and run your own

programs if you know how to write a makefile.

This should address most common issues users will find and sums up what seemed like dozens of other posts I read. Best of luck!

apostl3pol says:
October 2, 2012 at 12:55 pm

Will's tips on using Bumblebee gcc/g++-4.5 worked for me. Thanks Will!

apostl3pol says:
October 2, 2012 at 1:46 pm

A couple of extra comments. The SDK would not compile until I followed ALL of the steps at

http://fcns.eu/2012/05/24/cuda-4-2-9-sdk-compile-on-ubuntu-12-04-lts-fails-with-undefined-reference-to-gluerrorstring/

and ran the command "source .profile" after adding the line

export LIBRARY_PATH="$LIBRARY_PATH:/usr/lib/nvidia-current"

to .profile in my home directory (which is

really where the other exports that Will mentions should go.) I would also reiterate that ANY cuda file (or anything else you want to run on the graphics card) has to be run with the “optirun” prefix if you’re using Bumblebee on an optimus-enabled machine.

Fredrik says:
June 5, 2012 at 6:23 pm

Thank you! Excellent instructions! Very useful. It worked straight away, without any problems at all.

Reply

Pingback: 如何在Ubuntu 12.04上配置CUDA 4.2 | bfcat-计算机视觉博客

Nick says:
June 15, 2012 at 1:03 am

Thank you for the article. it was helpful and it was presented very professionally.

I do have a few questions regarding errors that appeared at the end of compiling the SDK file.

collect2: ld returned 1 exit status
make[2]: *** [../../bin/linux/release/randomFog] Error 1
make[2]: Leaving directory `/root/NVIDIA_GPU_Computing_SDK/CUDALibr

aries/src/randomFog'
make[1]: *** [src/randomFog/Makefile.ph_build] Error 2
make[1]: Leaving directory `/root/NVIDIA_GPU_Computing_SDK/CUDALibraries'
make: *** [all] Error 2

Do you know why these error messages are here? could they cause future problems?
we are using cuda for important research so we want to make sure everything is working without errors.

One important detail is that we accidentally installed SDK from root. What problems will this cause?

————My System Information————
-Ubuntu 12.04
-Nvidia driver and the toolbar has been fully installed
-driver version 295.41
-gcc 4.6.3
-I have already made the suggested changes to common.mk, common_cudalib.mk along with the changes to boxFilterNPP, imageSegmentationNPP, freeImageInteropNPP, histEqualizationNPP, and randomFog Makefiles. these changes were suggested at http://forums.nvidia.com/index.php?showtopic=231150
-everything we installed was from cuda 4.2

Reply

Utkarsh Jaiswal says:
June 15, 2012 at 6:33 pm

Hey Nick,

Thanks for your kind words The SDK merely includes examples and is as such not necessary for CUDA development. As far as I know, these errors seem to be pretty common (I still have them) and should not cause any issues in the future. They have something to do with the SDK referencing a different gcc version than the one you have installed, if I recall correctly. I'm not exactly sure how to fix these linking errors and I haven't had the time to work them all out, so I'm afraid you'll have to look elsewhere. If you do find a solution, please let me know so I can add it to the guide!

And I'd accidentally installed the SDK as root too, all you have to do is delete the folder NVIDIA_GPU_Computing_SDK and reinstall it as a regular user. The CUDA getting started guide recommends this to avoid access issues, although this seems to be a moot point since the SDK won't compile anyway.

Reply

Nick says:
June 15, 2012 at 11:16 pm

thank you very much for the quick

responce. luckily i did successfully compile sdk earlier this morning. without any errors. everything is running smothly now. thank you for the tutorial tho.

Hank says:
August 4, 2012 at 9:40 am

I am having the same problem as you, how did you fixed your errors?
the link http://forums.nvidia.com/index.php?showtopic=231150 seems to be down

Reply

Nav says:
July 13, 2012 at 2:33 pm

Hi
First of all thank you for the post, it was very useful. i am having a compilation error which is the following:
navish@navish-MacBookPro:~/cuda/test$ nvcc helloworld1.cu
helloworld1.cu:2:28: fatal error: ../common/book.h: No such file or directory
compilation terminated.
trying to find out what the problem is and would really appreciate any help of yours.
Regards

Reply

Utkarsh Jaiswal says:
July 13, 2012 at 3:53 pm

Hi, thanks for your feedback First off, your compile command is correct. I'm guessing you're using the hello world example from the "CUDA By Example" book. Unless you have specified the correct path to the book.h header, the nvcc compiler will output an error. If you've not received any source files with the book, you will not have the book.h header since it is a custom header developed by the authors of the book, and is not a part of the standard CUDA library. You can replace it with , and use the corresponding C headers for and C library functions your program may use.

Reply

Nav says:
July 13, 2012 at 4:55 pm

thanks again.yes i am starting off with the book and did not receive any source files. i will replace it and shall let you know about the output.
Regards

Nikhil says:
July 17, 2012 at 2:29 pm

Getting stuck while rebooting, after purging for any

nVidia residuals. Can't i simply install the drivers making the run file executable and then following the other steps of Installing CUDA n SDK? what difference will it make? Pls help. Need Cuda for blender cycles.

Reply

Utkarsh Jaiswal says:
July 17, 2012 at 2:57 pm

I'm not sure if it makes a difference, sorry. What error are you receiving?

Reply

Nikhil says:
July 18, 2012 at 2:08 pm

The machine is not able to show show the login screen. Btw i'm using Ubuntu Studio 12.04, the start up screen is where the machine gets stuck after the initial animation of the curved streaks going in circles stops.
Sys config: i5 2nd generation, 16 gb Ram, nVidia 550 Ti

Utkarsh Jaiswal says:
July 18, 2012 at 8:34 pm

I'm sorry but I don't really know how to

proceed here. Perhaps you could try the Ubuntu forums?

Cesar A. Simon says:
July 25, 2012 at 6:48 am

What happen if I have a second GPU from AMD and I wan to use it for my diplay. I should not allow the Nvidia driver change my xorg.conf?

Reply

Utkarsh Jaiswal says:
July 25, 2012 at 10:47 am

That’s a pretty rare case, how’d you manage to get the two working together?!

Reply

Erin Hodgess says:
July 25, 2012 at 12:48 pm

Hi! When I run the programs from NVIDIA, I get a no CUDA-capable device is detected

Have you run ino this please?

Reply

Utkarsh Jaiswal says:
July 25, 2012 at 5:42 pm

No, I haven't. Which GPU are you using? Please post the output of 'lshw -C display' (run it as a super-user)

Reply

Erin Hodgess says:
July 28, 2012 at 11:26 am

Hi!
I have an ASUS A53S Laptop with Nvidia 610M. The programs run if I have just run the sh /usr/bin/nvidia-bug-report but not without that.

Pingback: Update: CUDA computing « Entertaining Research

ss4 says:
August 9, 2012 at 2:54 am

All working fine for me,no problems

Reply

Yiannis says:
August 11, 2012 at 12:05 am

Hi,

Thank you for this post. It was very helpful for my installation of CUDA 4.2 on Lubuntu 12.04. Much appreciated!

Reply

Utkarsh Jaiswal says:
August 11, 2012 at 1:08 am

You're welcome!

Reply

Alan says:
August 13, 2012 at 7:20 am

This is really good, but to complete the task you must follow the instructions on this website http://fcns.eu/2012/05/24/cuda-4-2-9-sdk-compile-on-ubuntu-12-04-lts-fails-with-undefined-reference-to-gluerrorstring/ and then go to NVIDIA_GPU_Computing_SDK and type "make" to build the test programs. Finally go to NVIDIA_GPU_Computing_SDK/C/bin/linux and type "deviceQuery" to see if your CUDA card is operating properly. There are lots of cool demos in the same directory. I like smokeParticles.

Reply

Utkarsh Jaiswal says:
August 13, 2012 at 8:57 am

Thank you so much! I don't have a CUDA device with me now, but I'll update the post nonetheless.

Reply

Pingback: Installing CUDA on Ubuntu 12.04 |

Pingback: Gone Linux | Modelling Acoustics

Bart says:
September 6, 2012 at 1:03 am

When I run the command sudo apt-get remove –purge nvidia*, it tells me:
The following packages will be REMOVED:
nvidia-common* nvidia-current* nvidia-settings* ubuntu-desktop*
I'm especially concerned about the last package (although it only seems to be a meta package), is it really safe to remove this?
Thanks!

Reply

Utkarsh Jaiswal says:
September 6, 2012 at 9:33 am

From this link, removing ubuntu-desktop seems to be inviting trouble. I'm not sure about it, but I'd advise you to hold off for now. Can you post the details of lsmod on pastebin and link to it here? And try asking on the Ubuntu forums as well, a lot of knowledgeable folk out

there let's see if we can get to the bottom of this!

Reply

Bart says:
September 6, 2012 at 12:09 pm

Thanks for your fast response. I was in a brave (but stupid) mood yesterday and went ahead, and it is indeed not healthy to remove ubuntu-desktop But since I was already having some problems with Ubuntu (mainly on memory usage when running code parallel with MPI), I'm migrating back to good old OpenSUSE. I've been working with that for years, and it suits me better than Ubuntu…
So sorry, can't help you (or myself) get to the bottom of this, but you might want to warn about this in your post. I was using a relatively clean and up-to-date Ubuntu (12.04) installation, so others might run into the same problem.
Cheers, Bart

Utkarsh Jaiswal says:
September 6, 2012 at 9:51 pm

Good luck to you! I hadn't anticipated this problem though. I'll update the post, thanks!

Pingback: Installing CUDA on Ubuntu 12.04 | Peter Luk's Blog

Edgar says:

September 27, 2012 at 5:33 am

Hi all,
Thanks for this post. I am also having problems when trying to compile the SDK examples. When I go:
make
in the ~/NVIDIA_GPU_Computing_SDK/ I get:
make[2]: Entering directory `/home/edgar/NVIDIA_GPU_Computing_SDK/C/src/SobolQRNG'
/usr/local/cuda/bin/nvcc: 1: /usr/local/cuda/bin/nvcc: Syntax error: ")" unexpected
make[2]: *** [obj/i386/release/sobol_gpu.cu.o] Error 2
make[2]: Leaving directory `/home/edgar/NVIDIA_GPU_Computing_SDK/C/src/SobolQRNG'
make[1]: *** [src/SobolQRNG/Makefile.ph_build] Error 2
make[1]: Leaving directory `/home/edgar/NVIDIA_GPU_Computing_SDK/C'
make: *** [all] Error 2

so it seems that something is wrong with nvcc. I get the same error when I try to compile any cuda file: /usr/local/cuda/bin/nvcc: 1: /usr/local/cuda/bin/nvcc: Syntax error: ")" unexpected. Do you guys have any idea about how to go about it?

Cheers,
Edgar

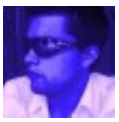
Reply

Utkarsh Jaiswal says:
September 27, 2012 at 10:08 am

Hey, I've never quite managed to fix those errors and I don't have access to any CUDA environment right now, but you could probably check this link out -

http://bit.ly/NHI2cE

Reply



Edgar says:
September 27, 2012 at 11:00 pm

Hi again,
Just wanted to say thanks for posting this and your help. Finally it worked for me!
What I did was to change my kernel from 32 bits to 64 bits and followed the steps here again and BUM!

I still have compilation errors for when building the SDK examples but I am reading http://fcns.eu/2012/05/24/cuda-4-2-9-sdk-compile-on-ubuntu-12-04-lts-fails-with-undefined-reference-to-gluerrorstring/ and it seems like it is an already documented fix.

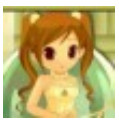
Cheers!
Edgar

Reply

Ankur says:
October 3, 2012 at 11:38 am

Thanks works perfectly!

Reply



hanismile says:
October 3, 2012 at 6:59 pm

Hi there,
Could you please help me with my problem? I've got stuck in it in two weeks already
—
nvcc warning : option 'host-compilation' has been deprecated and is ignored
nvcc fatal : A single input file is required for a non-link phase when an outputfile is specified
CMake Error at cuda_stuff_library_generated_get_feature_value_tex2d.cu.o.cmake:198 (message):
Error generating /home/nana/Desktop/rodrigo/src/applications/objects_detection/CMakeFiles/cuda_stuff_library.dir/__/__/__/src/objects_detection/gpu/./cuda_stuff_library_generated_get_feature_value_tex2d.cu.o

make[2]: *** [CMakeFiles/cuda_stuff_library.dir/__/__/__/src/o

bjects_detection/gpu/./cuda_stuff_library_generated_get_feature_value_tex2d.cu.o] Error 1
make[1]: *** [CMakeFiles/cuda_stuff_library.dir/all] Error 2
make: *** [all] Error 2
_____

And it seems I couldn't use "nvcc -v", it happend that "nvcc: command not found"
I am new in Ubuntu and Cuda and Cmake. I've just worked with them since last week

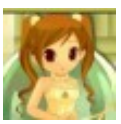
Thanks a lot

Reply

Utkarsh Jaiswal says:
October 3, 2012 at 10:32 pm

Hey, I'd love to help out but I'm sorry I don't have access to a CUDA enabled machine right now – it was just a summer intern. I'm not experienced enough to help offsite, and this guide was just a documentary of my findings as I managed to get CUDA working on Ubuntu 12.04. I'm afraid you'll have to look elsewhere. Hope you manage to solve it soon

Reply



hanismile says:
October 4, 2012 at 7:06 am

Oh, it's ok

Thanks a lot

Pingback: How to install CUDA on hybrid graphics notebooks with Ubuntu 12.04 | Jaideep @ CES

Inphi says:
October 8, 2012 at 12:49 pm

Is there no way of installing the drivers without removing the ubuntu-desktop package? Or rather, is removing just bumblebee-nvidia*, nvidia-common*, nvidia-current* and nvidia-settings* sufficient to install the drivers?

Reply

Utkarsh Jaiswal says:
October 8, 2012 at 1:42 pm

Ideally, the removal of bumblebee-nvidia*, nvidia-common*, nvidia-current* and nvidia-settings* ought to be sufficient to install the drivers. I do not remember getting a prompt to remove ubuntu-desktop while installing the development drivers. Do consult the Ubuntu forums and back your system up before proceeding though.

Reply

Pingback: installing GTX 560 on Windows 7 and Ubuntu 12.04 « Heru Suhartanto

manegomez says:
October 20, 2012 at 7:24 am

hi! im new in ubuntu and im stuck with the devdriver*.run command. the thing is, i dont know the name of my driver. can you help me with this?.. thank you very much!

Reply

Utkarsh Jaiswal says:
October 20, 2012 at 11:02 am

Hey, the devdriver*.run is the development driver downloaded from NVidia's website. It was applicable for CUDA 4.x and earlier. NVidia have started packaging the driver, SDK and toolkit in a single installer now. I'll be updating the guide soon to reflect this.

Reply

manegomez says:
October 20, 2012 at 7:48 pm

thank you very much!

Pingback: Installing CUDA 5 on Ubuntu 12.04 | sn0v

Pingback: SLAM - принципы и ссылки на open source | Мои IT-заметки

Bas Bloemsaat says:
March 17, 2013 at 7:10 pm

I made a writeup for the upcoming Ubuntu 13.04, using much info from this thread: http://blog.bloemsaat.com/2013/03/17/installing-cuda-on-ubuntu-13-04-raring-ringtail/

Reply

Utkarsh Jaiswal says:
March 20, 2013 at 10:18 am

Thanks for the credit!

Reply

Pingback: cuda 5.0 samples doesn't install on ubuntu 12.04

Pingback: SLAM - принципы и ссылки на open source | Мои IT-заметки

shamskabrahams says:
November 21, 2013 at 6:26 am

Hi, is there some one who can hlep me , i started following the instruction above to install cuda but i get this error messages.
shams@ubuntu:~$ sudo apt-get install freeglut3-dev build-essential libx11-dev libxmu-dev libxi-dev libgl1-mesa-glx libglu1-mesa libglu1-mesa-dev
Reading package lists… Done
Building dependency tree

```
Reading state information… Done
libglu1-mesa is already the newest version.
libglu1-mesa set to manually installed.
Some packages could not be installed. This may mean that you have
requested an impossible situation or if you are using the unstable
distribution that some required packages have not yet been created
or been moved out of Incoming.
The following information may help to resolve the situation:

The following packages have unmet dependencies:
freeglut3-dev : Depends: libgl1-mesa-dev but it is not going to be installed or
libgl-dev
Depends: libxt-dev but it is not going to be installed
Depends: libxext-dev but it is not going to be installed
libgl1-mesa-glx : Depends: libglapi-mesa (= 8.0.2-0ubuntu3)
Recommends: libgl1-mesa-dri (>= 7.2)
libglu1-mesa-dev : Depends: libglu1-mesa (= 8.0.2-0ubuntu3) but 8.0.4-0ubuntu0.6 is to be installed
Depends: libgl1-mesa-dev but it is not going to be installed or
libgl-dev
libx11-dev : Depends: libx11-6 (= 2:1.4.99.1-0ubuntu2) but 2:1.4.99.1-0ubuntu2.1 is to be installed
Depends: libxcb1-dev but it is not going to be installed
```

Recommends: libx11-doc but it is not going to be installed
libxi-dev : Depends: libxi6 (= 2:1.6.0-0ubuntu2) but 2:1.6.0-0ubuntu2.1 is to be installed
Depends: libxext-dev but it is not going to be installed
libxmu-dev : Depends: libxext-dev but it is not going to be installed
Depends: libxt-dev but it is not going to be installed
E: Unable to correct problems, you have held broken packages.
THANKS

Reply

Shams says:
November 21, 2013 at 6:52 am

Hello Utkarsh, save me man.

Reply

Utkarsh Jaiswal says:
November 21, 2013 at 6:59 am

Hey Shams,

You seem to have unmet dependencies on your system. Try “sudo apt-get -f install” to see if that fixes things, then rerun the original command

Reply

Shams says:
November 21, 2013 at 7:22 am

Thank you Utkarsh, i will try it now, this whole week i had been trying to install CUDA, did so many fresh install of Ubuntu and follow the instruction on forum. I hop you could help me out with this,
you have mentioned about three files but i found only two on CUDA download page DEB** RUN , is that right ?
here is my system config.
i7 processor with Nvidia Geforce 640
I am trying to run blender under Ubuntu and i want to use GPU option which needs CUDA as far as i guess. Blender runs fine but without GPU option.

Shams says:
November 21, 2013 at 7:54 am

after running given command this is what i got
shams@ubuntu:~$ sudo apt-get -f install
[sudo] password for shams:
Reading package lists… Done
Building dependency tree
Reading state information… Done
0 upgraded, 0 newly installed, 0 to remove and 154 not upgraded.
BTW ITS A FRESH INSTALL.

Reply

Pingback: Cuda4Ubuntu | bittnt's Research Blog

Leave a Reply

Enter your comment here...