



THE GEORGE
WASHINGTON
UNIVERSITY
WASHINGTON, DC



INSTALLATION GUIDE FOR TORCH THEANO, TENSORFLOW, CAFFE AND R WITH CPU AND GPU

Data Science Program, GWU, USA
School of Electrical and Computer Engineering, OSU, USA

✉: martin.t.hagan@okstate.edu
✉: ajafari@gwu.edu
✉: amir.h.jafari@okstate.edu

August 27, 2017

Contents

1	UBUNTU 14.04 DUAL BOOT INSTALLATION	5
2	NVIDIA AND CUDA 7.5 INSTALLATION	6
3	TORCH INSTALLATION GUIDE	8
4	cuDNN INSTALLATION GUIDE	10
5	THEANO INSTALLATION GUIDE	11
6	TENSORFLOW INSTALLATION GUIDE	12
7	CAFFE INSTALLATION GUIDE	13
8	R INSTALLATION GUIDE	15
9	TORCH DEBUGGER (ZEROBRANE)	16
10	PYTHON DEBUGGER (PYCHARM)	17
11	BLENDER (CAFFE)	18
12	TEST TORCH	19
13	TEST THEANO	20
14	TEST TENSORFLOW	21
15	TEST CAFFE	22
	15.1 Initialization	22

15.2 Sample Code	23
16 TEST R	24
17 VNC Installation and Setting	25
18 Teamviewer Installation	27
19 Folder Lock	28
References	29

List of Figures

1	Space allocation	5
2	Automatic partition	5

1 UBUNTU 14.04 DUAL BOOT INSTALLATION

To boot from USB, will have to choose boot from USB option from within Windows itself. Either with PC Setting (bios setting) and change the boot sequence to USB. b

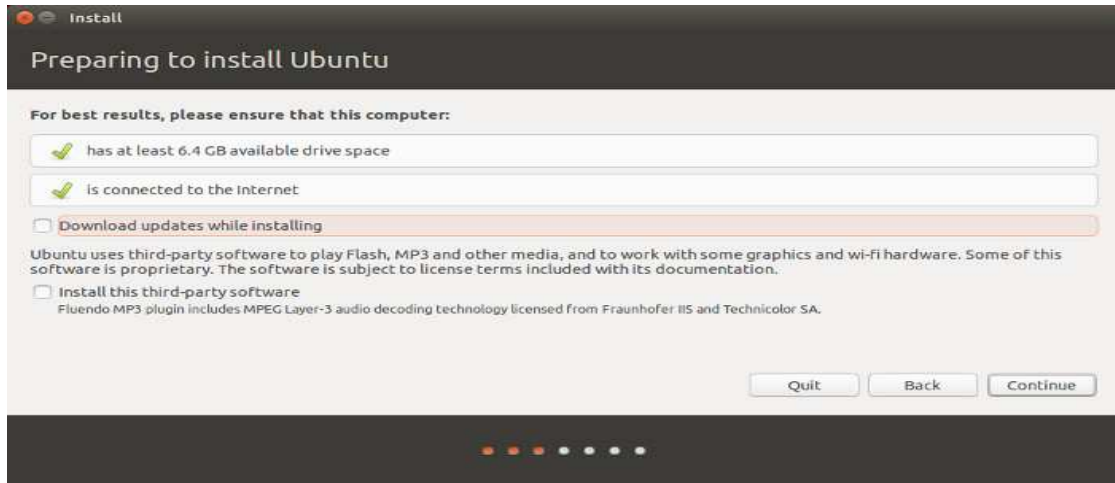


Figure 1: Space allocation

Once you have booted in the live USB, you will be presented with option to try or install Ubuntu. Click on install. You will be presented with few screen options to choose the language. It will then do some checks on available space, power and internet connection etc. **Just click on Continue. Note, do not change the partition size.**

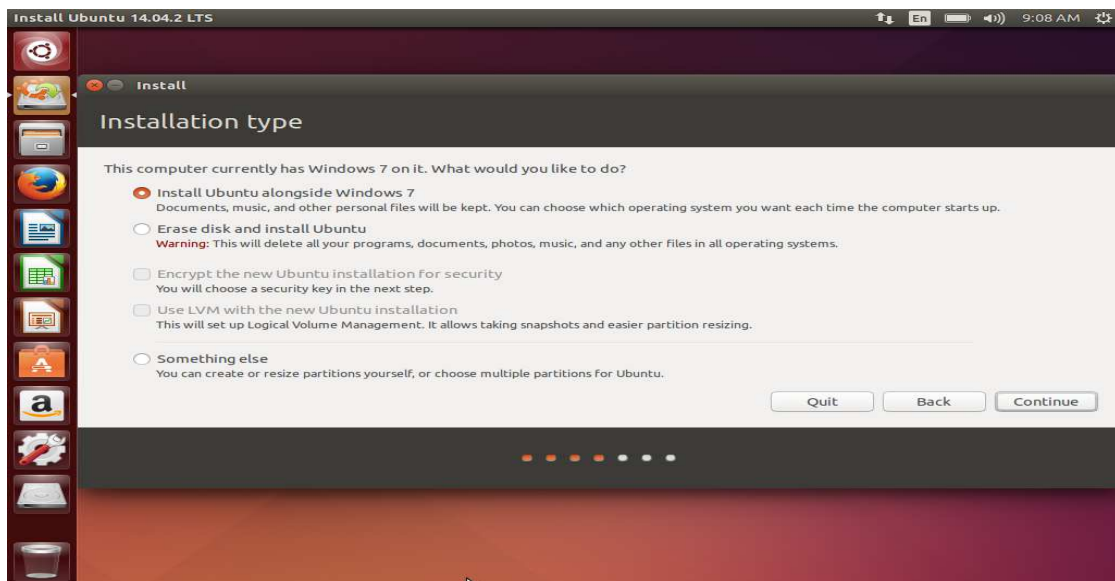


Figure 2: Automatic partition

2 NVIDIA AND CUDA 7.5 INSTALLATION

Installing CUDA Toolkit 7.5 on Ubuntu 14.04 Linux

The following explains how to install CUDA Toolkit 7.5 on 64-bit Ubuntu 14.04 Linux. I have tested it on a self-assembled desktop with NVIDIA GeForce GTX Black Titan graphics card. The instruction assumes you have the necessary CUDA compatible hardware support. Depending on your system configuration, your mileage may vary [1].

CUDA Repository Download

1. Retrieve the CUDA repository package for Ubuntu 14.04 from the provided file.
2. Put run file in the following directory (\home \Downloads)
3. Search for terminal, put on to task bar, and open it (then do the following commands).

```
$ cd \Downloads  
  
$ ls
```

4. You should be able to see cuda_7.5.18_linux.run

CUDA Toolkit and Display Driver Installation

5. Then you have to kill the xserver by entering the following command into terminal.

```
$ sudo service lightdm stop
```

6. The gui will switch off and you will see the black screen.
7. Hold the Ctrl+Alt+F1 key.
8. Type in username and hit enter.
9. Type in password (you will not see what you type in.)
10. Enter the following commands.
11. **Note: Any time you copy and paste underscore go back and replace it with new underscore in terminal.**

```
$ cd \Downloads  
  
$ ls  
  
$ chmod +x cuda_7.5.18_linux.run
```

12. Start the installation by the following command.
13. **Note: Any time you copy and paste underscore go back and replace it with new underscore in terminal.**

```
$. /cuda_7.5.18_linux.run
```

14. Press q key to skip the read me file.
15. accept EULA ——— accept.
16. when terminal prompts for yes or no always yes.
17. when terminal asks for location hit enter (this is the default location)
18. after the the installation finished, we need to reboot. Do the following commands to reboot.

```
$ sudo su  
  
# reboot
```

19. Redo the steps from 9 to 24 again for the full installation **except** for the following command.

```
$ chmod +x cuda_7.5.18_linux.run
```

Now, after the system is rebooted you can search the additional driver in the Linux search box (first item in the taskbar) and you should be able to see the manually install driver checked. Also you can see check this command to be sure the GPU is installed correctly.

This following command checks to see if the installation worked properly.

```
$ nvidia-smi
```

3 TORCH INSTALLATION GUIDE

1. Torch can be installed to your home folder in ~/torch by running these three commands [2]:
2. Open terminal and enter the following commands.
3. **Note: Any time you see yes or no in the installation type yes.**

```
$ sudo apt-get install git  
  
$ git clone https://github.com/torch/distro.git ~/torch --recursive  
  
$ cd ~/torch; bash install-deps;  
  
$ cd ~/torch  
  
$ ./install.sh
```

4. The first script installs the basic package dependencies that LuaJIT and Torch require. The second script installs LuaJIT, LuaRocks, and then uses LuaRocks (the lua package manager) to install core packages like torch, nn and paths, as well as a few other packages.
5. The script adds torch to your PATH variable. You just have to source it once to refresh your env variables. The installation script will detect what is your current shell and modify the path in the correct configuration file.
6. Enter the following command.

```
$ cd ..  
  
$ source ~/.bashrc
```

7. New packages can be installed using Luarocks from the command-line:

```
$ sudo apt-get install luarocks  
  
$ luarocks install image  
  
$ luarocks list
```



```
$ luarocks install nngraph  
  
$ luarocks install optim  
  
$ luarocks install nn  
  
$ luarocks install cutorch  
  
$ luarocks install cunn
```

*Now, it is time to test **TORCH** and check the installation, follow the instruction on [Section 12](#).*

4 cuDNN INSTALLATION GUIDE

1. Retrieve the cuDNN repository package for Ubuntu 14.04 from the provided file.
2. Copy the file to Downloads folder. Then open terminal
3. **Note: If the command wraps to another line highlight the first line then paste to terminal and then repeat for the next line then hit enter.**

Uncompress and copy the cuDNN files into the toolkit directory. Assuming the toolkit is installed in /usr/local/cuda, run the following commands (edited to reflect the cuDNN version you downloaded) [3]:

```
$ cd \Downloads

$ tar xvzf cudnn-7.5-linux-x64-v5.0-ga.tgz

$ sudo cp cuda/include/cudnn.h /usr/local/cuda/include

$ sudo cp cuda/lib64/libcudnn* /usr/local/cuda/lib64

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

5 THENAO INSTALLATION GUIDE

1. For installing Theano and Phyton, open terminal and run the following commands.
2. **Note: the first command there is a space between first line and second line.**
3. **Note: Any time you see yes or no in the installation type yes.**

```
$ sudo apt-get install python-numpy python-scipy python-dev python-pip  
python-nose g++ libopenblas-dev git  
  
$ sudo pip install Theano
```

*Now, it is time to test **THEANO** and check the installation, follow the instruction on [Section 13](#).*

6 TENSORFLOW INSTALLATION GUIDE

Pip Installation

1. Pip is a package management system used to install and manage software packages written in Python. The packages that will be installed or upgraded during the pip install are listed in the REQUIRED_PACKAGES section of setup.py [4].
2. Open terminal and try the following commands to install TensorFlow.
3. **Note: Any time you copy and paste underscore go back and replace it with new underscore in terminal (the last command contains underscores).**

```
$ sudo apt-get install python-pip python-dev

$ export TF_BINARY_URL=https://storage.googleapis.com/tensorflow/linux/
  cpu/tensorflow-0.10.0rc0-cp27-none-linux_x86_64.whl

$ export TF_BINARY_URL=https://storage.googleapis.com/tensorflow/linux/
  gpu/tensorflow-0.10.0rc0-cp27-none-linux_x86_64.whl

$ sudo pip install --upgrade $TF_BINARY_URL
```

*Now, it is time to test **TENSORFLOW** and check the installation, follow the instruction on Section [14](#).*

7 CAFFE INSTALLATION GUIDE

1. First, we need to install general dependencies by running the following commands [5].
2. Open terminal.
3. **Note: Any time you see yes or no in the installation type yes.**
4. **Note: the first command there is a space between first line and second line.**

```
$ sudo apt-get install libprotobuf-dev libleveldb-dev libsnappy-dev  
libopencv-dev libhdf5-serial-dev protobuf-compiler  
  
$ sudo apt-get install --no-install-recommends libboost-all-dev  
  
$ sudo apt-get install libgflags-dev libgoogle-glog-dev liblmdb-dev
```

5. We can download caffe from [Here](#). Click on the Clone or download button. Unzip the caffe-master.zip, then rename the result to caffe or do the following commands to download caffe and installation

```
$ git clone git://github.com/BVLC/caffe.git  
  
$ cd caffe  
  
$ cp Makefile.config.example Makefile.config
```

6. In the Linux GUI open up the file folder (looks like filing cabinet) and open up the caffe directory then double click on Makefile.config and do the following change.
7. Change **BLAS := atlas** to **BLAS := open**.

```
$ make all  
  
$ make test  
  
$ export Caffe_ROOT=/home/martin/caffe  
  
$ export LD_LIBRARY_PATH=/usr/local/cuda/lib64:$LD_LIBRARY_PATH  
  
$ make runtest
```

8. Now, caffe should be installed, also for further information check [\[6\]](#).
9. Now make the python connections.
10. **Note: Any time you see yes or no in the installation type yes.**

```
$ sudo apt-get install python-pip  
  
$ export PYTHONPATH=/path/to/caffe/python  
  
$ export PYTHONPATH=~/.technologies/caffe/python/:$PYTHONPATH  
  
$ make pycaffe  
  
$ cd ..  
  
$ source ~/.bashrc  
  
$ sudo apt-get install python-skimage  
  
$ sudo apt-get install python-pydot
```

Now, it is time to test CAFFE and check the installation, follow the instruction on [Section 15](#).

8 R INSTALLATION GUIDE

1. Open terminal.
2. **Note: Any time you see yes or no in the installation type yes.**

```
$ sudo apt-get install r-base
```

```
$ sudo apt-get update
```

3. Retrieve the Rstudio repository package for Ubuntu 14.04 from the provided file and save to Downloads folder.
4. Look for *rstudio-0.99.903-amd64.deb*.
5. Double click on the deb file to open it with ubuntu software center and install it.
6. Now you can search for R studio in the search box and start working with R.

Now, it is time to test R and check the installation, follow the instruction on [Section 16](#).

9 TORCH DEBUGGER (ZEROBRANE)

1. Retrieve the Zerobrane repository package for Ubuntu 14.04 from the provided file and save to Downloads folder.
2. Open terminal.

```
$ cd Downloads  
  
$ chmod +x ZeroBraneStudioEduPack-1.40-linux.sh  
  
$ ./ZeroBraneStudioEduPack-1.40-linux.sh
```

3. Now you can search for Zerobrane in the search box and start working with Torch Debugger.

10 PYTHON DEBUGGER (PYCHARM)

1. Open terminal.
2. Enter the following commands.
3. **Note: Any time you see yes or no in the installation type yes.**
4. **Note: Any time you see enter in the installation type enter.**

```
$ sudo add-apt-repository ppa:mystic-mirage/pycharm  
  
$ sudo apt-get update  
  
$ sudo apt-get install pycharm-community
```

5. Now you can search for Pycharm in the search box and start working with Python Debugger.

11 BLENDER (CAFFE)

This a node based tool for creating caffe networks. It works inside the graphics application 'blender' as a plugin. The reason for this is blender's highly stable, and universally compatible node editor [7].

1. Retrieve the Blender repository package for Ubuntu 14.04 from the provided file and save to Downloads folder (one tar one zip).
2. In Linux GUI open documents folder and create a new folder, rename it xyz.
3. Inside the xyz folder create a new folder and rename it addons.
4. Go back to Downloads folder, right click *caffe-gui-tool-master.zip* and extract here.
5. Now, right click *blender-2.77a-linux-glibc211-x86_64.tar.bz2* and extract here.
6. Rename *caffe-gui-tool-master* to *caffe-gui-tool*.
7. Move *caffe-gui-tool* to the new addons folder created.
8. Go back to Downloads folder, open blender which is inside *blender-2.77a-linux-glibc211-x86_64*.
9. Follow the installation from [HERE](#).
10. **Note: from the instruction from the link from the step before the path should look as follows */home/martin/Documents/xyz* in other words stop at xyz.**
11. **Note: when you are at check box step if the box won't check close out blender and try again. Also, don't click on it once it is checked**
12. save user setting.
13. Go to this [LINK](#) and follow the set up.
14. Go to this [LINK](#) and get started.
15. Go to this [LINK](#) and create the Conv. networks.

12 TEST TORCH

1. Retrieve the Torch Test file from the provided file and save to Desktop folder.
2. Save this file as gputest.lua
3. Open terminal.

```
$ cd Desktop  
$ th gputest.lua
```

13 TEST THEANO

1. Retrieve the Theano Test file from the provided file and save to Desktop folder.
2. Save this file as check1.py.
3. **Note: When you copy the commands the dollar signs (inside the commands) will not be copied, you need to type it manually**
4. Open terminal.

```
$ cd Desktop  
  
$ export LD_LIBRARY_PATH=/usr/local/cuda-7.5/lib64:$LD_LIBRARY_PATH  
  
$ export PATH=/usr/local/cuda-7.5/bin:$PATH
```

5. For CPU test enter the following command.

```
$ THEANO_FLAGS=mode=FAST_RUN,device=cpu,floatX=float32 python check1.py
```

6. For GPU test enter the following command.

```
$ THEANO_FLAGS=mode=FAST_RUN,device=gpu,floatX=float32 python check1.py
```

14 TEST TENSORFLOW

1. Retrieve the Tensorflow Test file from the provided file and save to Desktop folder.
2. Save this file as **Tensorflow_test.py**
3. Open terminal.

```
$ cd Desktop  
$ python Tensorflow\_test.py
```

15 TEST CAFFE

15.1 Initialization

First Method Initialization

1. Open terminal and enter the following commands.
2. Remember the user for this computer is *martin*, just for different computers you need to modify it to other users or else it will not work.
3. **Note: When you copy the commands the dollar signs (inside the commands) will not be copied, you need to type it manually**

```
$ export CAFFE_ROOT=/home/martin/caffe

$ export LD_LIBRARY_PATH=/usr/local/cuda/lib64:$LD_LIBRARY_PATH

$ export PYTHONPATH=/home/martin/caffe/python

$ export PYTHONPATH=${PYTHONPATH}:/home/martin/caffe/distribute/python

$ cd caffe

$ source ~/.bashrc

$ python

>>> import caffe
```

Second Method Initialization

1. Save the following commands in a new text document file (right click and open up a new document file) saved as **script**.

```
$ export CAFFE_ROOT=/home/martin/caffe

$ export LD_LIBRARY_PATH=/usr/local/cuda/lib64:$LD_LIBRARY_PATH

$ export PYTHONPATH=/home/martin/caffe/python

$ export PYTHONPATH=${PYTHONPATH}:/home/martin/caffe/distribute/python

$ source ~/.bashrc
```

2. Then change the directory that you save the file.
3. Then enter the following command to save time.

```
$ source ./script
```

15.2 Sample Code

1. Retrieve the Caffe Test file from the provided file and save to Desktop folder.
2. Save this file as testcaffe.py.
3. Save as the conv.prototxt file from the Caffe folder in the example folder .
4. Open terminal.
5. Then enter the following command to save time.

```
$ python testcaffe.py
```

16 TEST R

1. Retrieve the R Test file from the provided file and save to Desktop folder.
2. Search R-studio software in the search box and go to file menu and open the RNN.R.
3. Go to tools and install neural and neuralnet packages.
4. Open the saved file named RNN.R in R.
5. Press Ctrl+A and select all and click the bottom run.

17 VNC Installation and Setting

1. Open terminal

```
$ esudo apt-get install -y x11vnc  
  
$ sudo x11vnc -storepasswd /etc/x11vnc.pass
```

2. Type in the password for the VNC.
3. Verify the password (Note:yes)

```
$ hostname -I
```

4. Write down the IP address

```
$ vino-preferences
```

5. Check allow sharing.
6. Check require the user to enter password and type in the same VNC password in the box.
7. Uncheck must confirm each access to this machine.
8. Close the preferences

```
$ sudo apt-get install dconf-tools
```

9. Then search for dconf in the search bar:
10. Click on it.
11. On the left pane Go to
12. org -> gnome -> desktop -> remote access

13. on the right window, uncheck the encryption.

```
$ sudo apt-get update  
$ sudo apt-get install xrdp  
$ sudo apt-get install xfce4 xfce4-terminal  
$ echo xfce4-session > ~/.xsession  
$ sudo gedit /etc/xrdp/startwm.sh
```

14. The window will open up

15. change . /etc/x11/Xsession to . /etc/x11/startxfce4

16. Save it on the top menu.

```
$ sudo service xrdp restart  
$ sudo gedit /etc/init/x11vnc.conf
```

17. The window will open up copy the following in to the window and save it.

18. copy it line by line after each line there is sapce.

```
start on login-session-start  
  
script  
  
/usr/bin/x11vnc -xkb -auth /var/run/lightdm/root/:0 -noxrecord -noxfixes  
-noxdamage -rfbauth /etc/x11vnc.pass -forever -bg -rfbport 5900 -o  
/var/log/x11vnc.log  
  
end script
```

19. Restart your computer.

20. To vnc put in the ip address and add :5900 to the end.

21. Type in the password

18 Teamviewer Installation

1. Double click on the provided deb file (teamviewer_12.0.71510_i386.deb)
2. A screen will pop up and click install.
3. After installation is done you should be able to open it in the search box in a launcher.

19 Folder Lock

1. Open terminal.

```
$ sudo apt-get install cryptkeeper
```

19. Search for cryptkeeper in the search box launcher and run the program.
20. There should be a key icon in the top bar now (the most right top corner where the gearbox, time, date located).
21. Note: Keep pressing retry if you get a problem or restarting it (take a little while).

References

- [1] Nvidia driver. [Online]. Available: <http://developer.download.nvidia.com/compute/cuda/7.5/Prod/docs/sidebar.html>
- [2] Getting started with torch. [Online]. Available: <http://torch.ch/docs/getting-started.html>
- [3] cudnn ubuntu installation. [Online]. Available: https://www.tensorflow.org/versions/r0.10/get_started/os_setup.html#optional-install-cuda-gpus-on-linux
- [4] Download and setup of tensorflow. [Online]. Available: https://github.com/tensorflow/tensorflow/blob/master/tensorflow/g3doc/get_started/os_setup.md
- [5] Caffe ubuntu installation. [Online]. Available: <http://caffe.berkeleyvision.org/install apt.html>
- [6] Caffe website just for checking. [Online]. Available: <http://sunshineatnoon.github.io/How-to-install-caffe/>
- [7] Caff gui tool. [Online]. Available: <https://github.com/Chasvortex/caffe-gui-tool>