This is a guide to setting up Caffe in an Ubuntu 14.04 virtual machine with CUDA 6.5 and the system Python for getting started with examples and PyCaffe

# Part I Guide

* Download newest Ubuntu release. I use 14.04 Ubuntu(64 bit) server
* Start VirtualBox, create a new virtual machine (Linux/Ubuntu/64bit/DynamicHD/8GbRAM/…). Then start the VM from the downloaded .iso (do this from inside VirtualBox).
* Install system updates (3.13.0-32 -> 3.13.0-36)
* Install build essentials:
  + sudo apt-get install build-essential
* Install latest version of kernel headers:
  + sudo apt-get install linux-headers-`uname -r`
* Install VBox Additions:
  + Click the USB icon in the task-bar (lower right of screen)
  + Left click the VBOXADDITIONS\_4.3.16\_95972
  + Select “Open with File Manager” (this will mount the virtual CD in the virtual CD drive…). Once the file manager window has opened, you can kill it. This step is only needed to mount the CD...
  + cd /media/<USER>/VBOXADDITIONS\_4.3.16\_95972 (where <USER> is your user name)
  + sudo ./VBoxLinuxAdditions.run
* Activate bidirectional clip-board:
  + In Virtual Box Manager, click Settings, then General | Advanced | Shared Clipboard
* Install curl (for the CUDA download):
  + sudo apt-get install curl
* Download CUDA.
  + cd ~/Downloads/
  + curl -O "http://developer.download.nvidia.com/compute/cuda/6\_5/rel/installers/cuda\_6.5.14\_linux\_64.run"
* Make the downloaded installer file runnable:
  + chmod +x cuda\_6.5.14\_linux\_64.run
* Run the CUDA installer:
  + sudo ./cuda\_6.5.14\_linux\_64.run --kernel-source-path=/usr/src/linux-headers-`uname -r`/
    - Accept the EULA
    - Do **NOT** install the graphics card drivers (since we are in a virtual machine)
    - Install the toolkit (leave path at default)
    - Install symbolic link
    - Install samples (leave path at default)
* Update the library path
  + echo 'export LD\_LIBRARY\_PATH=$LD\_LIBRARY\_PATH:/usr/local/cuda/lib64:/usr/local/lib' >> ~/.bashrc
  + source ~/.bashrc
* Install dependencies:

sudo apt-get install -y libprotobuf-dev libleveldb-dev libsnappy-dev libopencv-dev libboost-all-dev libhdf5-serial-dev protobuf-compiler gfortran libjpeg62 libfreeimage-dev libatlas-base-dev git python-dev python-pip libgoogle-glog-dev libbz2-dev libxml2-dev libxslt-dev libffi-dev libssl-dev libgflags-dev liblmdb-dev python-yaml gcc-4.6 g++-4.6 gcc-4.6-multilib g++-4.6-multilib

* + sudo easy\_install pillow
* Change compiler to g++4.6:
  + sudo update-alternatives --install /usr/bin/cc cc /usr/bin/gcc-4.6 30
  + sudo update-alternatives --install /usr/bin/c++ c++ /usr/bin/g++-4.6 30
* Download Caffe:
  + cd ~
  + git clone https://github.com/BVLC/caffe.git
* Install python dependencies for Caffe:
  + cd caffe
  + cat python/requirements.txt | xargs -L 1 sudo pip install
* Google’s logging framework isn’t available through a repository, so you have to build that from source:
  + wget https://google-glog.googlecode.com/files/glog-0.3.3.tar.gz  
    tar zxvf glog-0.3.3.tar.gz  
    cd glog-0.3.3  
    ./configure  
    make  
    sudo make install  
    cd ..
* Add a couple of symbolic links for some reason:
  + sudo ln -s /usr/include/python2.7/ /usr/local/include/python2.7
  + sudo ln -s /usr/local/lib/python2.7/dist-packages/numpy/core/include/numpy/ /usr/local/include/python2.7/numpy
* Create a Makefile.config from the example:
  + cp Makefile.config.example Makefile.config
  + nano Makefile.config
    - Uncomment the line # CPU\_ONLY := 1 (In a virtual machine we do not have access to the the GPU)
    - Under PYTHON\_INCLUDE, replace /usr/lib/python2.7/dist-packages/numpy/core/include with /usr/local/lib/python2.7/dist-packages/numpy/core/include (i.e. add /local)
* Compile Caffe:
  + make pycaffe
  + make all
  + make test
* Download the ImageNet Caffe model and labels:
  + ./scripts/download\_model\_binary.py models/bvlc\_reference\_caffenet
  + ./data/ilsvrc12/get\_ilsvrc\_aux.sh
* Modify python/classify.py to add the --print\_results option
  + Compare https://github.com/jetpacapp/caffe/blob/master/python/classify.py(version from 2014-07-18) to the current version of classify.py in the official Caffe distributionhttps://github.com/BVLC/caffe/blob/master/python/classify.py
* Test your installation by running the ImageNet model on an image of a kitten:
  + cd ~/caffe (or whatever you called your Caffe directory)
  + python python/classify.py --print\_results examples/images/cat.jpg foo
  + Expected result: [('tabby', '0.27933'), ('tiger cat', '0.21915'), ('Egyptian cat', '0.16064'), ('lynx', '0.12844'), ('kit fox', '0.05155')]
* Test your installation by training a net on the MNIST dataset of handwritten digits:
  + cd ~/caffe (or whatever you called your Caffe directory)
  + ./data/mnist/get\_mnist.sh
  + ./examples/mnist/create\_mnist.sh
  + ./examples/mnist/train\_lenet.sh
  + See <http://caffe.berkeleyvision.org/gathered/examples/mnist.html> for more information...

# Part II Solutions for some common bugs

#### After 'make all', you may face two errors. Here are the solutions:

Error One:

./include/caffe/common.hpp:5:27: fatal error: gflags/gflags.h: No such file or directory compilation terminated.

Solutions:

wget https://github.com/schuhschuh/gflags/archive/master.zip unzip master.zip cd gflags-master mkdir build && cd build export CXXFLAGS="-fPIC" && cmake .. && make VERBOSE=1 make sudo make install

Error Two:

./include/caffe/data\_layers.hpp:11:18: fatal error: lmdb.h: No such file or directory

Solution:

git clone git://gitorious.org/mdb/mdb.git cd mdb/libraries/liblmdb make sudo make install

#### If when you run 'make pycaffe', you see this error

UnicodeDecodeError: 'ascii' codec can't decode byte 0xe2 in position 54: ordinal not in range(128)

you could fix it by running:

sudo apt-get install libevent-dev python-dev

If you get any other bugs, StackOverflow and Github can always help you!