0. 使用的环境

- OS：Windows 10 x64
- CPU：Intel Core i5-4210H 2.90GHz
- GPU：Nvidia GEFORCE GTX 860M
- Memory RAM：8GB
- Microsoft Visual Studio Community 2017 version 15.8.9
- Miniconda2 version 5.2 with Python 2.7.15
- Nvidia CUDA 9.0
- cuDNN v7.3.0 for CUDA 9.0
- Git v2.19.1
- OpenCV 3.4.3.18

1. 下载Caffe 下载第三方插件

下载微软版的Caffe Caffe-MS 第三方插件Baidu Yun

把第三方插件解压到caffe目录下面的 `./windows/thirdparty/` . 并且把 `./windows/thirdparty/bins` 加入到环境变量 `PATH`。而后打开VS2017，在这里我发现除了python外的所有解决方案都无法加载，于是重新下载caffe。问题解决。

2. 下载编译需要的VS工具集

使用vs2017的installer程序，在**单个组件**里面装上这两个工具集

windows SDK 10.0.15063：Caffe需要这个版本的SDK，在VS installer里面找到它，有3个G多，下载下来。

用于VS2017的V140版本的C++工具集：其实是用VS2015的C++编译器来编译以前的项目。VS2017本身是V141。

3. CUDA9.0

google搜 `CUDA archive`点进去找到9.0版本的CUDA ，并且把更新包也下下来。

安装时报错。因为VS integration安装失败，先自定义安装，不选VS integration 然后

1. Unzip CUDA installer like an archive into a directory, say `C:\cuda_9.2.88_win10\`
2. Install CUDA without Visual Studio Integration (you need to choose Custom instead of Express installation)
3. Copy `C:\cuda_9.2.88_win10\CUDAVisualStudioIntegration\extras\visual_studio_integration\MSBuildExtensions` to `C:\Program Files (x86)\Microsoft Visual Studio\2017\Community\Common7\IDE\VC\VCTargets\BuildCustomizations` .
4. Copy `C:\cuda_9.2.88_win10\CUDAVisualStudioIntegration\extras\visual_studio_integration\CudaProjectVsWizards` to `C:\Program Files (x86)\Microsoft Visual Studio\2017\Community\Common7\IDE\Extensions\CUDA`
5. Run as administrator: `devenv /setup` from directory `C:\Program Files (x86)\Microsoft Visual Studio\2017\Community\Common7\IDE`
6. Install all the .msi from `C:\cuda_9.2.88_win10\CUDAVisualStudioIntegration`

怎么做第五步：

打开那个文件夹；在资源管理器上面的路径显示框中输入CMD，然后回车，即可在当前文件夹下打开命令提示符，并且路径是当前文件夹；但是这样不是管理员权限。所以还是：

先在资源管理器里面到达指定文件夹，然后复制目录

在windwos图标右键，管理员CMD

cd ctrl+v 然后devenv /setup

4. cuDNN 7.3.0

google 搜cudnn archive 然后点进去下对应的版本 cuDNN 7.3.0 for CUDA 9.0

然后解压把里面的lib include bin三个文件夹扔到CUDA的安装目录

`C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v9.0`

5. 环境变量和其他修改

装完CUDA之后，会有两个系统环境变量

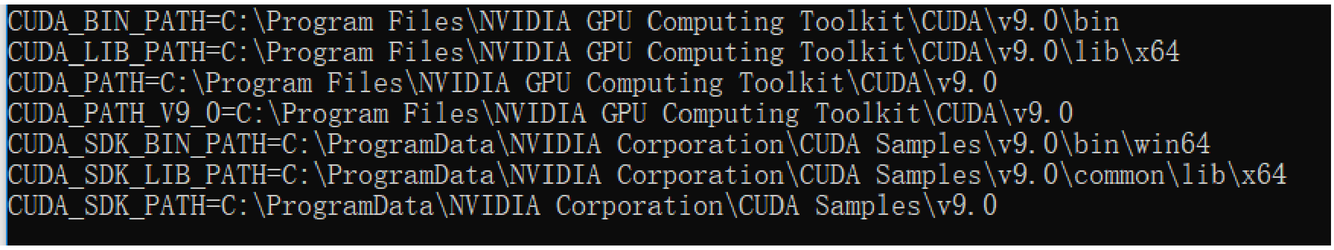
```
CUDA_PATH         C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v9.0
CUDA_PATH_V9_0  C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v9.0
```

但是还要把下面几个加入环境变量

```
CUDA_SDK_PATH = C:\ProgramData\NVIDIA Corporation\CUDA Samples\v9.0
CUDA_LIB_PATH = %CUDA_PATH%\lib\x64
CUDA_BIN_PATH = %CUDA_PATH%\bin
CUDA_SDK_BIN_PATH = %CUDA_SDK_PATH%\bin\win64
CUDA_SDK_LIB_PATH = %CUDA_SDK_PATH%\common\lib\x64
```

装完看看有没有装成功

打开命令提示符 输入`set cuda`环境变量都出现了，可以。



```
CUDA_BIN_PATH=C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v9.0\bin
CUDA_LIB_PATH=C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v9.0\lib\x64
CUDA_PATH=C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v9.0
CUDA_PATH_V9_0=C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v9.0
CUDA_SDK_BIN_PATH=C:\ProgramData\NVIDIA Corporation\CUDA Samples\v9.0\bin\win64
CUDA_SDK_LIB_PATH=C:\ProgramData\NVIDIA Corporation\CUDA Samples\v9.0\common\lib\x64
CUDA_SDK_PATH=C:\ProgramData\NVIDIA Corporation\CUDA Samples\v9.0
```

然后`cd C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v9.0\extras\demo_suite`

定位到CUDA的demo程序文件夹，运行deviceQuery.exe 和 bandwidthTest.exe两个程序，运行deviceQuery.exe**记一下第一行出来的GPU计算能力值，我这里为5.0** 如果结果Result都是PASS，就没有问题了。

辣鸡英伟达

开始-程序-VS2017tools-x64 prompt命令提示符 `cl /?`查看VS的版本号，前四位数字记下来。比如说19.12就记成1912。找到路径 **C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v9.1\include\crt** 中的 **host_config.h** 文件，右键管理员取得所有权限，打开，找到130行 `#if _MSC_VER < 1600 || _MSC_VER > 1911` 把1911改成1913，反正就是要让1912能通过。这一行限制了VS的版本，如果不解除的话会报错说VS版本不对。

打开项目的属性，为CUDA的Sample选择v140的工具集。

到此为止，VS2017里面CUDA9.0 的sample都可以跑了。。哭了卧槽。。**辣鸡英伟达**

CUDA 的sample在`C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v9.0\doc\pdf`

`C:\ProgramData\NVIDIA Corporation\CUDA Samples\v9.0`

6. Python环境

一定不要通过VS2017安装Anaconda

一定不要通过VS2017安装Anaconda

一定不要通过VS2017安装Anaconda

去清华的开源镜像下自己想要的anaconda版本 地址

Miniconda的环境变量

```
C:\Users\Weijiang\Miniconda2
C:\Users\Weijiang\Miniconda2\Library\mingw-w64\bin
C:\Users\Weijiang\Miniconda2\Library\usr\bin
C:\Users\Weijiang\Miniconda2\Library\bin
C:\Users\Weijiang\Miniconda2\Scripts
```

caffe用到的是anaconda2 with python2.7.15 还要装protobuf。

```
conda install Cython numpy scipy protobuf scikit-image matplotlib ipython h5py leveldb networkx nose pandas pyyaml Pillow six python-gflags pydotplus graphviz python-dateutil>=1.4,<2

--yes
```

报错了记得清理conda缓存

装protobuf的时候报错了,说要升级pip。然后就升级呗,再装就说已满足条件。

因为anaconda自带了很多包。如果装的是miniconda的话就要自己装很多东西。

由于其他项目需求，我还要用到anaconda3，因此需要处理两个环境并存的问题。用conda创建虚拟环境可以方便地在两个环境之间切换。也可以结合VS2017的python环境使用，更爽。

```
PS C:\Users\Weijiang> activate caffetrinken
(caffetrinken) PS C:\Users\Weijiang> python
Python 2.7.15 |Anaconda, Inc.| (default, Nov 13 2018, 17:33:26) [MSC v.1500 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> exit
Use exit() or Ctrl-Z plus Return to exit
>>> ^Z

(caffetrinken) PS C:\Users\Weijiang> activate base
(base) PS C:\Users\Weijiang> python
Python 3.6.5 |Anaconda, Inc.| (default, Mar 29 2018, 13:32:41) [MSC v.1900 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
```

Remark

After you have built solution with Python support, in order to use it you have to either:

- set `PythonPath` environment variable to point to `<caffe_root>\Build\x64\Release\pycaffe`, or
- copy folder `<caffe_root>\Build\x64\Release\pycaffe\caffe` under `<python_root>\lib\site-packages`.

有个老哥自己折腾了CUDA9.0的pycaffe包。说是python3.5 3.6 3.7都能用，目前没试

https://github.com/BVLC/caffe/issues/6569

> I have published conda packages. Please install with these commands:
>
> REM Create a new conda environment to install Caffe
> REM You can use python 3.5, 3.6 or 3.7
> conda create -n caffe python=3.7
> REM activate the environment
> REM Depending on your conda version you may have to use activate caffe
> conda activate caffe
> REM add the anaconda channel to resolve all dependencies
> conda config --add channels anaconda
> REM Install caffe with cuda support or install caffe-cpu if you do not want CUDA support
> conda install caffe -c willyd
> Currently only CUDA 9.0 is built but I will merge the files to build the packages once I have confirmation that the packages work.

7. Matlab support

To build Caffe Matlab wrapper set `MatlabSupport` to `true` and `MatlabDir` to the root of your Matlab installation in `.\windows\CommonSettings.props`.

Remark

After you have built solution with Matlab support, in order to use it you have to add the `./matlab` folder to Matlab search path.

MATLAB的build文件在`D:\Caffe\matlab\+caffe\private`

8. CommonSettings.props编辑

原地复制 `D:\Caffe\windows\CommonSettings.props.example`为`CommonSettings.props`

要改动其中的`pythonsupport`、`pythondir`、`matlabsupport`、`matlabdir`以及CPU/GPU运行相关的设置

CUDA版本写`9.0`、cuDNN的路径也要成这样`C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v9.0`

VS2017中视图-属性管理器，为caffe和libcaffe添加`C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v9.0\extras\visual_studio_integration\MSBuildExtensions`下的CUDA 9.0.props

显卡计算能力设置在30几行

```
<!-- Set CUDA architecture suitable for your GPU.
     Setting proper architecture is important to mimize your run and compile time. -->
  <CudaArchitecture>compute_30,sm_30;compute_50,sm_50;compute_52,sm_52;compute_60,sm_60;
</CudaArchitecture>
```

本来`compute_50,sm_50`是没有的，需要加进去。这个是显卡的计算能力设置，由MNIST测试的问题才发现的。

9. 到这一步已经能编译了

不过有很多错误和警告 注意一下VS的python环境改成anaconda2的，对应装上去的python2.7.15。之前装的是anaconda3和python3.6.6。

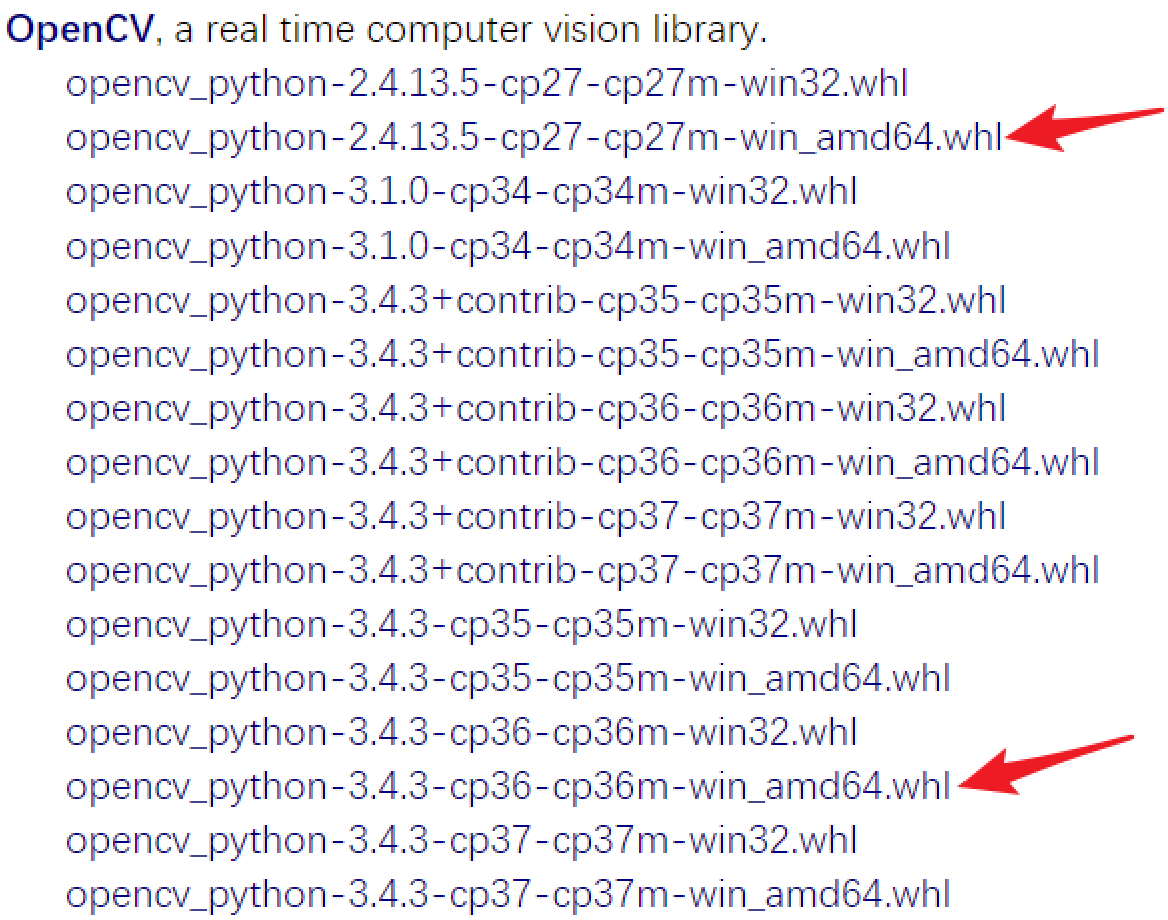
- 首先编译`libcaffe` `caffe`。。在`pycaffe`时出了问题，说无法找到 `python27_d.dll`。但是`C:\Users\weijiang\Miniconda2\libs`只有`python27.dll`这个是debug模式的Python文件，但是下载到的caffe是release版本的。所以要把VS最上头的debug改成release。。或者是，原地复制`python27.dll`并且重命名成为`python27_d.dll`
- 第二种方法会产生假的debug环境, 参考`https://blog.csdn.net/junparadox/article/details/52704287`
- 改成release模式重新生成`pycaffe`，说无法找到libcaffe.lib，因为刚才生成的libcaffe也是debug模式的。换成release模式重新生成一遍。

编译好的文件全部在`D:\Caffe\Build\x64\Release`里面

10. 安装opencv

由于有anaconda2和anaconda3环境同时存在，因此需要分别安装。直接在anaconda2里面安装报错了。。说和一个包不兼容。 anaconda2用的是python2.7，根据官方的说法貌似不能装opencv3，但是我用PIP暴力装也装上去了。

在`https://www.lfd.uci.edu/~gohlke/pythonlibs/#opencv`下载两个包



OpenCV, a real time computer vision library.

- opencv_python-2.4.13.5-cp27-cp27m-win32.whl
- opencv_python-2.4.13.5-cp27-cp27m-win_amd64.whl
- opencv_python-3.1.0-cp34-cp34m-win32.whl
- opencv_python-3.1.0-cp34-cp34m-win_amd64.whl
- opencv_python-3.4.3+contrib-cp35-cp35m-win32.whl
- opencv_python-3.4.3+contrib-cp35-cp35m-win_amd64.whl
- opencv_python-3.4.3+contrib-cp36-cp36m-win32.whl
- opencv_python-3.4.3+contrib-cp36-cp36m-win_amd64.whl
- opencv_python-3.4.3+contrib-cp37-cp37m-win32.whl
- opencv_python-3.4.3+contrib-cp37-cp37m-win_amd64.whl
- opencv_python-3.4.3-cp35-cp35m-win32.whl
- opencv_python-3.4.3-cp35-cp35m-win_amd64.whl
- opencv_python-3.4.3-cp36-cp36m-win32.whl
- opencv_python-3.4.3-cp36-cp36m-win_amd64.whl
- opencv_python-3.4.3-cp37-cp37m-win32.whl
- opencv_python-3.4.3-cp37-cp37m-win_amd64.whl

`opencv_python-2.4.13.5-cp27-cp27m-win_amd64.whl`放在anaconda2的目录下面`C:\Program Files (x86)\Microsoft Visual Studio\Shared\Anaconda2_64\Lib\site-packages`，然后开管理员CMD，打cd 然后复制路径，enter，定位到这个文件夹，然后复制文件全名，打上pip install opencv_python-2.4.13.5-cp27-cp27m-win_amd64.whl

这个时候已经可以在C:\Program Files (x86)\Microsoft Visual Studio\Shared\Anaconda2_64下面开cmd，然后import cv2了

`opencv_python-3.4.3-cp36-cp36m-win_amd64.whl`装在anaconda3的目录下面`C:\Program Files (x86)\Microsoft Visual Studio\Shared\Anaconda3_64\Lib\site-packages`

同样的套路可以用在anaconda3.但是anaconda3的opencv装起来要简单多了。打开anaconda navigator，注意是anaconda3的。然后在environment里面切到anaconda3，上方的installed 切换为all，然后搜索opencv，然后勾选，然后点右下角的apply等待解算环境并安装。下载的时候进度条好像是假的，只有下载完了才会提示。

区分两个python环境，可以用开始-程序里面的anaconda2和anaconda3下面的命令提示符快捷程序Anaconda Prompt。用2的启动就对应python2，用3的启动就对应python3.

也可以，用anaconda2的时候把2的环境变量放在3的上面，用3的时候就把3的环境变量上移。但是，这样不能从上面说的文件夹启动，要从初始文件夹启动。

装完之后，先后打开两个命令提示符，用conda list可以看到`opencv-python 2.4.13.5 <pip>`表明已经安装了opencv，但是是通过PIP的方式安装的，不是conda安装的。这种情况下，再到conda里面安装opencv，虽然会下载，但是是装不上去的。import cv2也是可以用的。

注意，这样安装opencv的话，VS2017里面，包(conda)是看不到opencv的，要切成包（pypi）

上面的protobuf也是用pip安装的，也不能在conda包里面看到。

管理员权限真的是个非常神奇的东西，以管理员权限运行VS和Anaconda就不会报错说两个包冲突了。MMP

坑 numpy版本和opencv版本要对应。。艹。具体是怎么样的不知道，反正opencv装最新的。。不要装4.0..

11. 编译 Caffe

先编译libcaffe，然后caffe，这是主程序。pycaffe和matcaffe分别是python和matlab的接口.

pythondir没写对会报错，说找不到pyconfig.h

pycaffe 编译报错。PythonPostBuild.cmd 出错。。MSB3073，在pycaffe上右键属性-生成事件-后期生成事件-在生成中使用改成否

12. MNIST和Cifar10测试

12.1 MNIST

去http://yann.lecun.com/exdb/mnist/下载四个压缩包，包括两个test两个train

把t10k开头的test数据解压到`D:\Caffe\data\mnist\mnist_test_lmdb`

把train的数据解压到`D:\Caffe\data\mnist\mnist_train_lmdb`

在caffe根目录下编写bat文件create_mnist.bat

```
.\Build\x64\Release\convert_mnist_data.exe .\data\mnist\mnist_train_lmdb\train-images.idx3-ubyte
.\data\mnist\mnist_train_lmdb\train-labels.idx1-ubyte .\examples\mnist\mnist_train_lmdb
echo.
.\Build\x64\Release\convert_mnist_data.exe .\data\mnist\mnist_test_lmdb\t10k-images.idx3-ubyte
.\data\mnist\mnist_test_lmdb\t10k-labels.idx1-ubyte .\examples\mnist\mnist_test_lmdb
pause
```

运行`create_mnist.bat`产生imdb数据文件在`D:\Caffe\examples\mnist`再次运行会报错，因为文件已经存在。

修改D:\Caffe\examples\mnist\lenet_solver.prototxt，将最后一行改为`solver_mode:CPU`，修改D:\Caffe\examples\mnist\lenet_train_test.prototxt，把两个source的路径，换成

`source: "./examples/mnist/mnist_train_lmdb"`，其实就是example前面加`./`

尝试改用GPU模式时报错，错误内容为

`Check failed: error == cudaSuccess (48 vs. 0) no kernel image is available for execution on the device`

因为显卡计算能力与定义不符。

在`https://developer.nvidia.com/cuda-gpus`查询显卡的计算能力，GTX860M的Kepler架构计算能力为3.0，Maxwell架构为5.0或者5.2.下载Nvidiainspector，我的显卡核心为`GM107`，因此是Maxwell架构的GTX860M。所以第8节的显卡计算能力，也是因为这个才要设置的。

对D:\Caffe\cmake\Cuda.cmake修改 86-100行注释掉，

```
if(${CUDA_ARCH_NAME} STREQUAL "Fermi")
    set(__cuda_arch_bin "20 21(20)")
  elseif(${CUDA_ARCH_NAME} STREQUAL "Kepler")
    set(__cuda_arch_bin "30 35")
  elseif(${CUDA_ARCH_NAME} STREQUAL "Maxwell")
    set(__cuda_arch_bin "52")
  elseif(${CUDA_ARCH_NAME} STREQUAL "Pascal")
```

```
    set(__cuda_arch_bin "60 61")
  elseif(${CUDA_ARCH_NAME} STREQUAL "All")
    set(__cuda_arch_bin ${Caffe_known_gpu_archs})
  elseif(${CUDA_ARCH_NAME} STREQUAL "Auto")
    caffe_detect_installed_gpus(__cuda_arch_bin)
  else()  # (${CUDA_ARCH_NAME} STREQUAL "Manual")
    set(__cuda_arch_bin ${CUDA_ARCH_BIN})
  endif()
```

在101行插入

```
set(__cuda_arch_bin "50")
```

当然， 也可以吧Maxwell下面的52改成50，因为已经知道了显卡的架构，效果就是一样的。最上面还有一行 `set(Caffe_known_gpu_archs "20 21(20) 30 35 52 60 61")` 也要把52改过来。

12.2 cifar10测试

- 获取数据集，需要编译出convert_cifar_data.exe

在http://www.cs.toronto.edu/~kriz/cifar-10-binary.tar.gz下载cifar10数据集，压缩包放到 `D:\Caffe\data\cifar10`

然后运行 `get_cifar10.sh` 或者直接解压就可以得到一堆bin文件

在 `D:\Caffe\Build\x64\Release` 新建一个input文件夹和一个output文件夹，把那一堆bin文件复制到input里面

在 `D:\Caffe\Build\x64\Release` 打开cmd

```
D:\Caffe\Build\x64\Release>convert_cifar_data.exe input output leveldb
I1128 13:46:26.378101  8680 db_leveldb.cpp:18] Opened leveldb output/cifar10_train_leveldb
I1128 13:46:26.379825  8680 convert_cifar_data.cpp:52] Writing Training data
I1128 13:46:26.380331  8680 convert_cifar_data.cpp:55] Training Batch 1
I1128 13:46:26.454551  8680 convert_cifar_data.cpp:55] Training Batch 2
I1128 13:46:26.536113  8680 convert_cifar_data.cpp:55] Training Batch 3
I1128 13:46:26.614226  8680 convert_cifar_data.cpp:55] Training Batch 4
I1128 13:46:26.707296  8680 convert_cifar_data.cpp:55] Training Batch 5
I1128 13:46:27.262730  8680 convert_cifar_data.cpp:73] Writing Testing data
I1128 13:46:27.288148  8680 db_leveldb.cpp:18] Opened leveldb output/cifar10_test_leveldb
```

output里面就有了caffe能用的leveldb格式数据

- 求数据集平均值需要编译出comput_image_mean.exe

在 `D:\Caffe\Build\x64\Release` 打开cmd，注意一定强调**后端是leveldb**格式！否则默认lmdb会出错

`compute_image_mean.exe --backend=leveldb output/cifar10_train_leveldb mean.binaryproto`

在D:\Caffe\Build\x64\Release下得到mean.binaryproto

把meanfile和leveldb的文件夹复制到D:\Caffe\examples\cifar10

- 训练网络

把cifar10_quick_train_test.prototxt中关于路径和格式的部分修改,backend一定要**大写**

如果选择的网络是cifar10_full_train_test，那么你的solver应为cifar10_full_solver

```
    mean_file: "examples/cifar10/mean.binaryproto"

    source: "examples/cifar10/cifar10_train_leveldb"

    backend: LEVELDB
```

D:\Caffe下开CMD输入

`.\Build\x64\Release\caffe.exe train --solver=./examples/cifar10/cifar10_quick_solver.prototxt`

出现

```
I1128 14:19:13.352393  8432 solver.cpp:403]     Test net output #0: accuracy = 0.7114
I1128 14:19:13.352393  8432 solver.cpp:403]     Test net output #1: loss = 0.861337 (* 1 = 0.861337 loss)
I1128 14:19:13.352393  8432 solver.cpp:321] Optimization Done.
I1128 14:19:13.352393  8432 caffe.cpp:260] Optimization Done.

D:\Caffe>pause
请按任意键继续. . .
```

练full的时候改成

```
.\Build\x64\Release\caffe.exe train --solver=./examples/cifar10/cifar10_full_solver.prototxt
```

要从中断的地方继续，就要：

```
.\Build\x64\Release\caffe.exe train --solver=./examples/cifar10/cifar10_full_solver.prototxt --snapshot=./examples/cifar10/cifar10_full_iter_60000.solverstate.h5
```

13. pycaffe的使用

把 `D:\Caffe\python\caffe` 下所有文件复制到 `<python_root>\lib\site-packages`

把caffe的release里面编译出来的所有东西(看生成时间)复制到 `<python_root>\lib\site-packages`

在build完毕之后出现

`AttributeError: 'module' object has no attribute 'PYQT_VERSION_STR'`

protobuf的版本要3.2.0的。据说是这个变量用的很少，最新版3.6.1给删了。

但是实际上caffe的thirdparty里面有protobuf这个东西，所以python环境里面直接删了protobuf也不要紧。环境变量里面把thirdparty放到最上面去，这个貌似是这个版本的作者踩坑踩出来的一套可以用的依赖库版本。

修改内容

https://github.com/sergeyk/selective_search_ijcv_with_python下下来放在python2.7的site-package

PYQT5报错

做CNN卷积特征可视化的时候`from PyQt5 import QtGui`报错，说`No module named sip`

安装的PyQt5的版本，与Python版本不匹配，后来用pip3 install PyQt5命令升级到相匹配的版本后，但是sip没有做版本适应。 先升级sip到4.19.3版本，发现from PyQt5 import QtGui会直接让python崩溃，但是升级之后再降级回4.18.1，就能成功import了。真的非常诡异。

caffe的IO和numpy的IO重名

when import numpy

```
  File "/Users//anaconda/lib/python2.7/site-packages/skimage/util/dtype.py", line 8, in <module>
    dtype_range = {np.bool_: (False, True),
AttributeError: 'module' object has no attribute 'bool_'
```

the author Dr Jia says so:

```
I just realized what happened - you are loading from the 'caffe/python' folder, which contains a file "io.py". When numpy is being loaded, it will load this io.py instead of loading numpy's io, and that causes the problem.

As a rule of thumb, do not invoke python in a module's source file folder - instead, go to a different folder like $HOME/research/mycode, and add caffe/python into your PYTHONPATH.
```

do not run python scripts directly from d:\caffe\python\caffe, even opening it with visual studio will results in the same problem. so try creating a new solution for these caffe scripts in a different directory and copy them there.

I have finished this problem. That if you have added caffe/python into your Path, you must take draw_net.py out of this folder, and put it in another folder, and then use it.

If anyone is still struggling to find a solution after following all the above steps (I was), here's a solution that worked for me. Extremely hacky, but it works until someone finds out a better way to do this.

First set PYTHONPATH as /usr/local/lib/python2.7/site-packages/ or /usr/local/lib/python2.7/dist-packages/ Either should work.

```
PYTHONPATH= /usr/local/lib/python2.7/site-packages/
```

Now, open python:

> > import numpy as np import sys sys.path.insert(0,'/home/brt/python/) # the folder where you have the python caffe. Note that this path is without the /caffe at the end. import caffe sys.path.insert(0,'/home/brt/python/caffe') #Note that this has /caffe at the end import _caffe

This should work. Basically what I do is I force python to import numpy from the site-packages and then `caffe` and `_caffe` from the respective Caffe folders. If you have all three paths in your PYTHONPATH simultaneously, Python tries to import numpy from the io.py in the Caffe folders and not site-packages which causes the "no attribute 'bool_' error".

draw.io报错

```
  File "C:\Users\Weijiang\Anaconda2\lib\site-packages\pydotplus\graphviz.py", line 1960, in create
    'GraphViz\'s executables not found')
InvocationException: GraphViz's executables not found
```

因为没装graphviz这个包。。mmp到底有多少依赖关系。。疯了

```
按照caffe/python下面的requirements.text
pip install python-dateutil==1.5
然后报错
anaconda-client 1.6.14 has requirement python-dateutil>=2.6.1, but you'll have python-dateutil 1.5 which is incompatible.
bokeh 0.12.16 has requirement python-dateutil>=2.1, but you'll have python-dateutil 1.5 which is incompatible.
pandas 0.23.0 has requirement python-dateutil>=2.5.0, but you'll have python-dateutil 1.5 which is incompatible.
matplotlib 2.2.2 has requirement python-dateutil>=2.1, but you'll have python-dateutil 1.5 which is incompatible.
jupyter-client 5.2.3 has requirement python-dateutil>=2.1, but you'll have python-dateutil 1.5 which is incompatible.
然后都装16年左右的
pip install anaconda-client==1.2.2
pip install bokeh==0.12.0
pip install pandas==0.18.0
pip install matplotlib==2.0.0
```

14.matcaffe的使用

Using the MATLAB interface

Follow the above procedure and use `-DBUILD_matlab=ON`. Change your current directory in MATLAB to `C:\Projects\caffe\matlab` and run the following command to run the tests:

```
>> caffe.run_tests()
```

If all tests pass you can test if the classification_demo works as well. First, from `C:\Projects\caffe` run `python scripts\download_model_binary.py models\bvlc_reference_caffenet` to download the pre-trained caffemodel from the model zoo. Then change your MATLAB directory to `C:\Projects\caffe\matlab\demo` and run `classification_demo`.

matcaffe运行的时候会报一堆错，一堆警告。。而且

```
caffe.set_mode_gpu();
caffe.set_device(0);
```

这里面，如果device id错了的话，matlab会直接崩溃退出。写0表示第一个GPU。因为集成显卡是不支持CUDA的，不会计算在内，所以0就代表GTX860M。在可以在nvidiainspector里面查看GPU ID。pycaffe 和caffe本体里面设置也一样的，反正用GPU 0就对了。

13. 在caffe中添加normalize layer

到这一步为止已经可以开始happy了，但是我发邮件向作者樊恒博士请教之后，得知还要在caffe中加入normalization layer，这也是SINT的要求。

在https://github.com/taotaoorange/SINT找到SINT的说明，但是作者似乎已经换单位了，原来那个学校找不到他

找个文件夹命名为SINT，gitbash here

```
git clone https://github.com/taotaoorange/SINT.git
cd SINT
git checkout windows
```

however, tmd已经有一个normalize layer了，看代码发现，似乎还是SINT用的版本的改进版，这一部分直接弃了吧。

X. 参考的网站

CAFFE Win10使用VS2017安装Caffe详细总结 https://blog.csdn.net/fengtaoO08/article/details/83023428

Win10 +VS2017+ python3.66 + CUDA9.2 + cuDNNv7.2.1 +微软版pycaffe编译 https://blog.csdn.net/qq_41895190/article/details/82414695

Caffe1.1.0+VS2017+CUDA9.2+cuDNN9.2+OpenCV3.4.3+OpenBLAS+Windows配置教程 https://blog.csdn.net/qq_20226441/article/details/82788799

win10下caffe安装教程 https://blog.csdn.net/u011947630/article/details/81346273

续前：总结本人经历的CAFFE安装配置VS2017的坑 https://blog.csdn.net/qq_28597273/article/details/82787284

CUDA

https://blog.csdn.net/LOVELESSYI/article/details/79219276

https://blog.csdn.net/u013165921/article/details/77891913

windows7+visual studio 2013+CUDA7.5 编译caffe+配置matcaffe+配置pycaffe https://blog.csdn.net/tina_ttl/article/details/51722983

matcaffe

https://blog.csdn.net/zb1165048017/article/details/51702686