

windows 下 CAFFE 训练自己的数据

颜廷玉 天津工业大学

张 觅 武汉大学遥感信息工程学院

在 windows 下使用 caffe 训练自己的数据,对刚入门使用 CAFFE 熟悉深度学习的朋友来说,仍会碰到很多问题。本文结合使用 windows 版本 CAFFE 的经验,通过一个实例说明 windows 下使用 CAFFE 的步骤,希望对碰到类似问题的同学能有所帮助。

1. 数据准备:

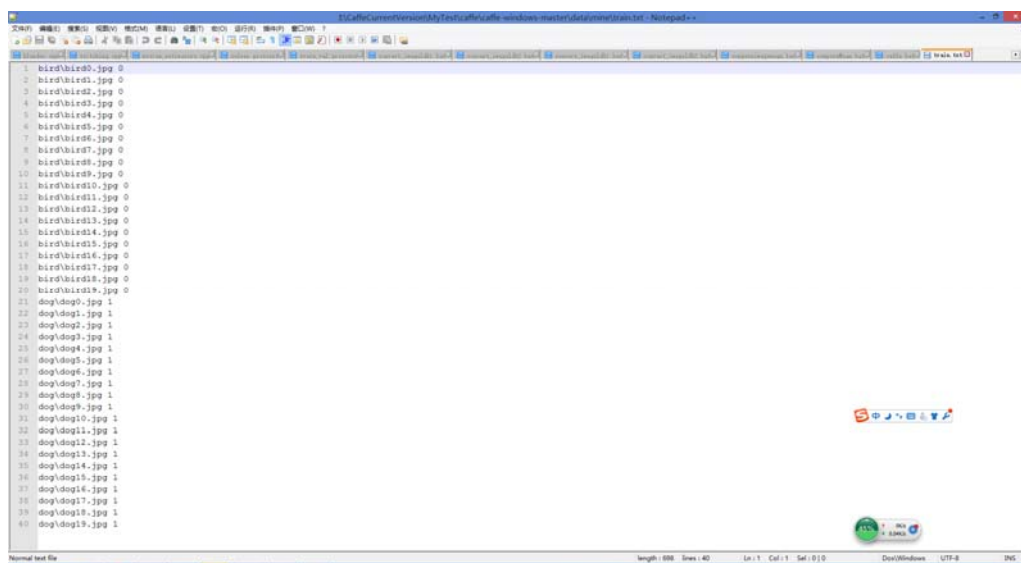
在 caffe 的下面目录中,准备如下数据:



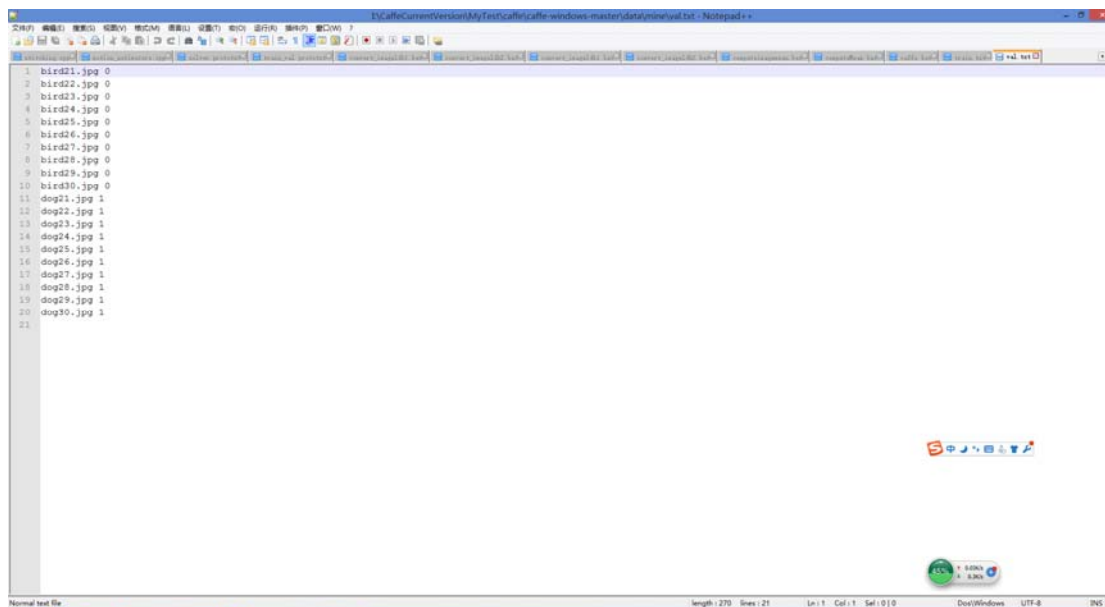
其中, 1) tain 里面放置的是测试数据, 如下:



2) train.txt 为训练数据的一个列表, 如下:

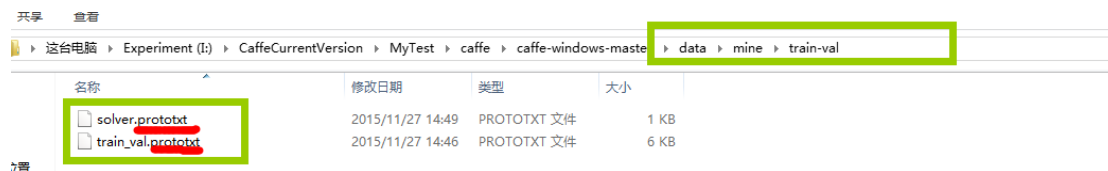


Va.txt 为测试数据的一个列表，如下：



3) 其他的几个文件夹 `mtrainldb`, `mvalldb`, `ming_mean.binaryproto` 均为经过命令处理得到的结果 (`mtrainldb`, `mvalldb` 为 jpg 图像经过格式转换后得到的文件，用于输入到 `caffe` 中；`mean.binaryproto` 为计算均值后生成的文件)

4) `tain-val` 文件夹中为 `caffe` 训练的输入参数



Solver.prototxt 内容如下：（需要修改的地方，就是文件的位置）







train_val.prototxt 内容如下:

```
1 name: "AlexNet"
2 layer {
3   name: "data"
4   type: "Data"
5   top: "data"
6   bottom: "data"
7   include {
8     phase: TRAIN
9   }
10  transform_param {
11    mirror_swap: 0
12    crop_size: 227
13    mean_file: "data/mine/ming_mean.binaryproto"
14  }
15  data_param {
16    source: "data/mine/mtrainldb"
17    batch_size: 20
18    backend: LMDB
19  }
20 }
21 layer {
22   name: "data"
23   type: "Data"
24   top: "data"
25   bottom: "data"
26   include {
27     phase: TEST
28   }
29  transform_param {
30    mirror_swap: 0
31    crop_size: 227
32    mean_file: "data/mine/ming_mean.binaryproto"
33  }
34  data_param {
35    source: "data/mine/mvalldb"
36    batch_size: 20
37    backend: LMDB
38  }
39 }
40 layer {
41   name: "conv1"
42   type: "Convolution"
```

需要注意的是, 他们的结尾都是 prototxt, 是一种 CAFFE 规定的文件输入协议

2. 批处理文件编辑

批处理文件包括下面几个:

 caffe.bat	2015/11/27 14:26	Windows 批处理...	1 KB
 computeMean.bat	2015/11/27 14:26	Windows 批处理...	1 KB
 convert_imageldb1.bat	2015/11/27 14:26	Windows 批处理...	1 KB
 convert_imageldb2.bat	2015/11/27 14:26	Windows 批处理...	1 KB

1) 数据格式转换

Conver_imageldb1.bat 用于转换训练数据格式

SET GLOG_logtostderr=1

bin\convert_imageset.exe (可执行文件名) **I:\CaffeCurrentVersion\MyTest\caffe\caffe-windows-master\data\mine\train** (训练数据所在文件夹) **I:\CaffeCurrentVersion\MyTest\caffe\caffe-windows-master\data\mine\train.txt** (训练数据列表) **I:\CaffeCurrentVersion\MyTest\caffe\caffe-windows-master\data\mine\mtrainldb** (待转换数据列表) **0**

pause

需要注意的地方是, 如果使用 leveldb 格式, 可以加上如下内容: **--backend=leveldb**, 变为:

bin\convert_imageset.exe (可执行文件名) **--backend=leveldb (输入格式控制)** **I:\CaffeCurrentVersion\MyTest\caffe\caffe-windows-master\data\mine\train** (训练数据所在文件夹) **I:\CaffeCurrentVersion\MyTest\caffe\caffe-windows-master\data\mine\train.txt** (训练数据列表) **I:\CaffeCurrentVersion\MyTest\caffe\caffe-windows-master\data\mine\mtrainldb** (待转换数据列表) **0**

相应的 train-val 文件夹下的 train_val.prototxt 文件需要修改一个地方:

```

name: "AlexNet"
layer {
  name: "data"
  type: "Data"
  top: "data"
  top: "label"
  include {
    phase: TRAIN
  }
  transform_param {
    mirror: true
    crop_size: 227
    mean_file: "data/mine/ming_mean.binaryproto"
  }
  data_param {
    source: "data/mine/mtrainldb"
    batch_size: 20
    backend: LMDB
  }
}
layer {
  name: "data"
  type: "Data"
  top: "data"
  top: "label"
  include {
    phase: TEST
  }
  transform_param {
    mirror: false
    crop_size: 227
    mean_file: "data/mine/ming_mean.binaryproto"
  }
  data_param {
    source: "data/mine/mvalldb"
    batch_size: 20
    backend: LMDB
  }
}
layer {
  name: "conv1"

```

LEVELDB

Convert_imageldb2.bat 用于转换测试数据格式，用法与转换训练数据用法相同

```
SET GLOG_logtostderr=1
```

```
bin\convert_imageset.exe
```

```
I:\CaffeCurrentVersion\MyTest\caffe\caffe-windows-
```

```
master\data\mine\val\
```

```
I:\CaffeCurrentVersion\MyTest\caffe\caffe-windows-
```

```
master\data\mine\val.txt
```

```
I:\CaffeCurrentVersion\MyTest\caffe\caffe-windows-
```

```
master\data\mine\mvalldb
```

```
pause
```

2) 均值计算

ComputeMean.bat 内容如下:

```
SET GLOG_logtostderr=1
```

```
bin\compute_image_mean.exe (可执行文件, 在 bin 目录下)
```

```
I:\CaffeCurrentVersion\MyTest\caffe\caffe-windows-master\data\mine\mtrainldb (上一步生成的格式转换后的数据文件夹) I:\CaffeCurrentVersion\MyTest\caffe\caffe-windows-master\data\mine\ming_mean.binaryproto (输出的均值文件)
```

```
pause
```

3) caffe 训练测试

Caffe.bat 内容如下:

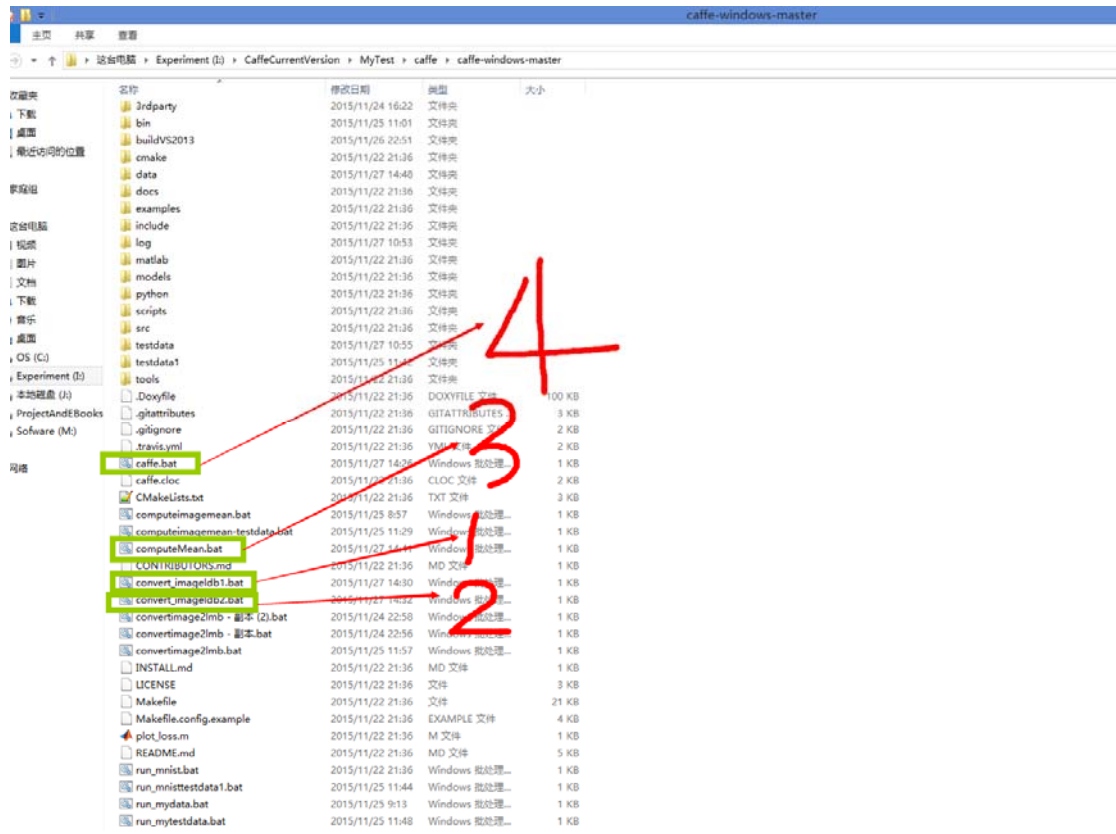
SET GLOG_logtostderr=1

bin\caffe.exe (可执行文件名) train --solver data\mine\train-val\solver.prototxt (指定 solver 文件所在位置)

pause

3. 批处理文件执行

按照顺序，依次执行 convert_imageldb1.bat，convert_imageldb2.bat，computeMean.bat,caffe.bat。其中，1,2 步为数据格式转换步骤，3 为均值计算步骤，4 为训练评价步骤



最终结果如下：

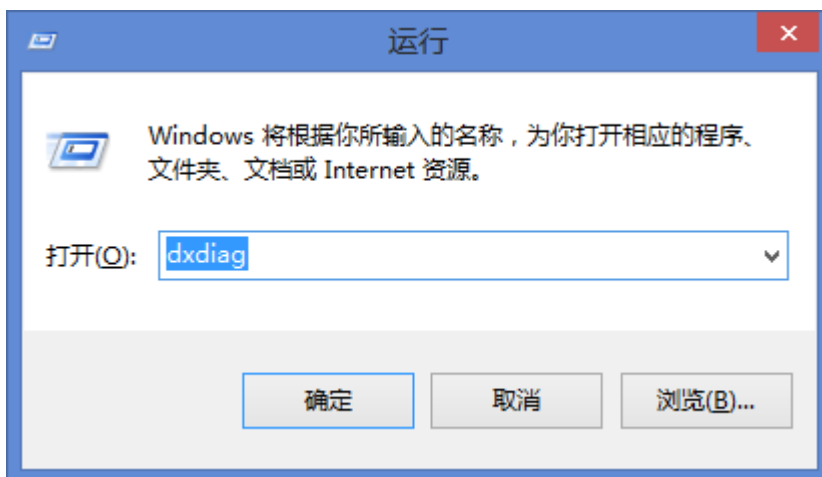
```
C:\Windows\system32\cmd.exe

0162382 (* 1 = 0.000162382 loss)
11127 15:39:21.111706 15912 sgd_solver.cpp:106] Iteration 1720, lr = 0.0001
11127 15:39:56.669242 15912 solver.cpp:236] Iteration 1740, loss = 7.2139e-005
11127 15:39:56.669242 15912 solver.cpp:252] Train net output #0: loss = 7.20
084e-005 (* 1 = 7.20084e-005 loss)
11127 15:39:56.670245 15912 sgd_solver.cpp:106] Iteration 1740, lr = 0.0001
11127 15:40:32.142776 15912 solver.cpp:236] Iteration 1760, loss = 5.87735e-005
11127 15:40:32.142776 15912 solver.cpp:252] Train net output #0: loss = 5.86
429e-005 (* 1 = 5.86429e-005 loss)
11127 15:40:32.143777 15912 sgd_solver.cpp:106] Iteration 1760, lr = 0.0001
11127 15:41:07.114763 15912 solver.cpp:236] Iteration 1780, loss = 4.4421e-005
11127 15:41:07.115736 15912 solver.cpp:252] Train net output #0: loss = 4.42
904e-005 (* 1 = 4.42904e-005 loss)
11127 15:41:07.115736 15912 sgd_solver.cpp:106] Iteration 1780, lr = 0.0001
11127 15:41:40.541934 15912 solver.cpp:340] Iteration 1800, Testing net (#0)
11127 15:41:47.080360 15912 solver.cpp:408] Test net output #0: accuracy = 0
.95
11127 15:41:47.080360 15912 solver.cpp:408] Test net output #1: loss = 0.093
6632 (* 1 = 0.0936632 loss)
11127 15:41:48.667851 15912 solver.cpp:236] Iteration 1800, loss = 4.41137e-005
11127 15:41:48.668853 15912 solver.cpp:252] Train net output #0: loss = 4.39
831e-005 (* 1 = 4.39831e-005 loss)
11127 15:41:48.668853 15912 sgd_solver.cpp:106] Iteration 1800, lr = 0.0001

搜狗拼音输入法 全：
```

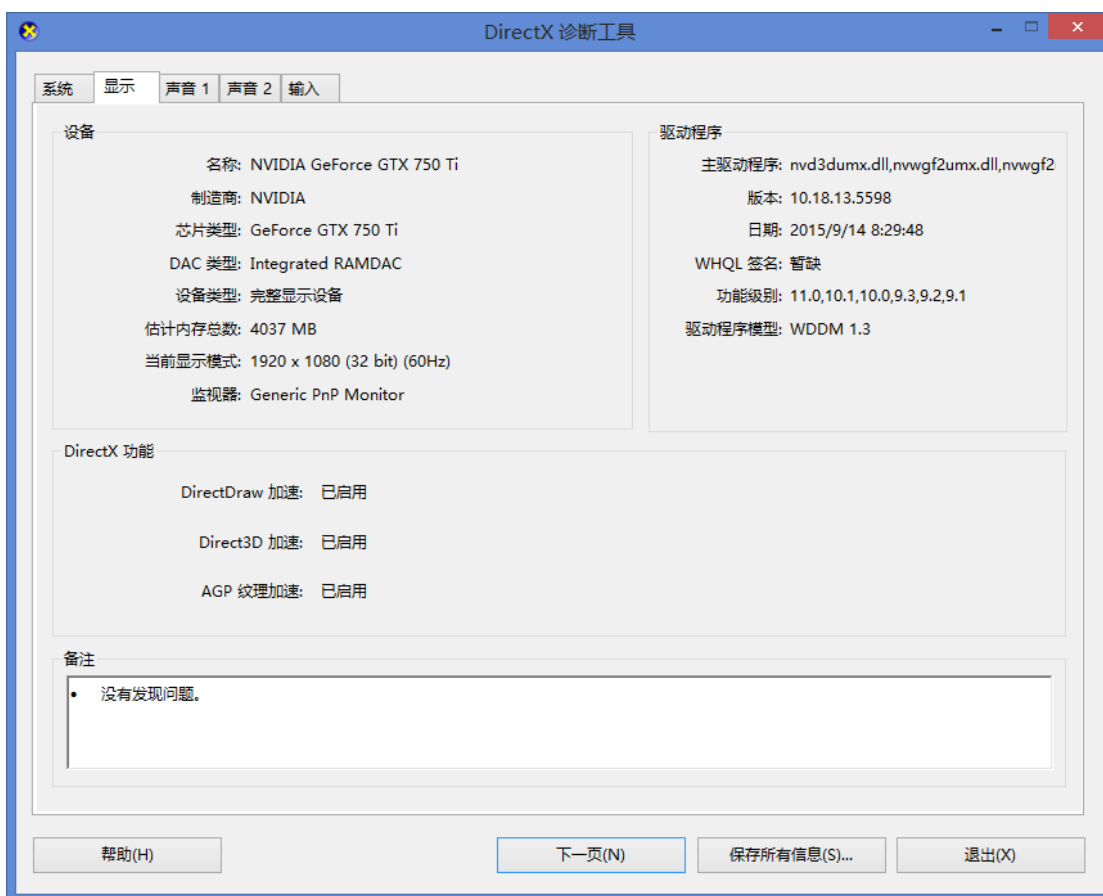
附：Windows 下，如果有 GPU 的话，可以查询一下显卡配置，以便使用 Caffe 的 GPU 性能，方法如下：

1. Ctrl+R 进入运行，然后输入 dxdiag

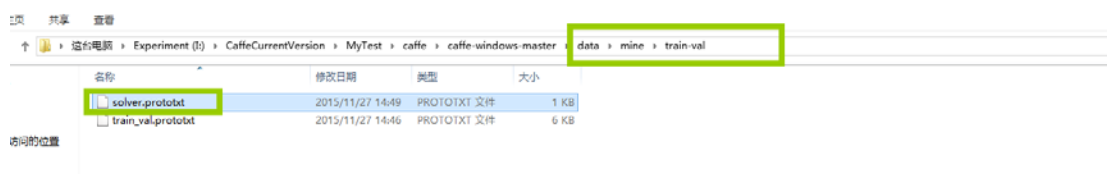


2. 得到如下结果：

在显示标签页上面，就有电脑系统显卡配置的相关信息



4. 修改 train_val 文件夹下的 solver.prototxt 文件，将其中的 solver_mode 修改为 GPU



```
1 net: "data/mine/train-val/train_val.prototxt"
2 test_iter: 10
3 test_interval: 200
4 base_lr: 0.001
5 lr_policy: "step"
6 gamma: 0.1
7 stepsize: 1000
8 display: 20
9 max_iter: 4500
0 momentum: 0.9
1 weight_decay: 0.0005
2 snapshot: 4000
3 snapshot_prefix: "models/bvlc_alexnet/caffe_alexnet_train"
4 solver_mode: GPU
```

特别感谢 CAFFE 的创始人贾扬清, github 社区 windows 版本 CAFFE 的发起者 niuzhiheng, happynear, 以及 CAFFE 线上线下活动组织者和社区的人员。正是因为你们, 我们这些入门者才有机会对深度学习有了感性的认识。

参考资料:

1. <http://blog.csdn.net/u013657981/article/details/49497753>
2. <http://blog.csdn.net/u012878523/article/details/41698209>
3. <https://github.com/niuzhiheng/caffe>
4. <https://github.com/happynear/caffe-windows>