使用说明

**准备工作：**

**（1）**

安装vs2013；

安装Matlab；

安装CUDA；

上面的安装顺序最好不要乱，否则可能出现Matlab找不到vs的情况，在Matlab命令行窗口输入：mbuild -setup，如果出现：



说明Matlab可以找到vs2013。CUDA应在安装vs2013后再安装。

**（2）**

如果你的cuda是6.5，那么，运行一下：

fetch\_data/fetch\_caffe\_mex\_windows\_vs2013\_cuda65.m

（运行代码下载失败的话，用百度云下载：[https://pan.baidu.com/s/1i3m0i0H](https://pan.baidu.com/s/1i3m0i0H%22%20%5Ct%20%22_blank)  ，解压到faster\_rcnn-master下）

得到mex文件。如果不是cuda6.5（如我的是cuda7.5），则需要自己编译mex文件，编译过程参考这里：[Caffe for Faster R-CNN](https://github.com/ShaoqingRen/caffe/tree/faster-R-CNN%22%20%5Ct%20%22_blank)，按步骤做就行了。

也可以下载我编译得到的文件（注意cuda版本）。

下载地址：[Faster-RCNN(Matlab) external文件夹](http://download.csdn.net/detail/sinat_30071459/9501568)

建议还是自己编译，因为版本问题可能会出错。在训练前，可以先下载作者训练好的模型，测试一下，如果可以的话，就不用自己编译了。

**测试过程：**

（1）运行faster\_rcnn-master\faster\_rcnn\_build.m

 （2）运行faster\_rcnn-master\startup.m

（3）运行faster\_rcnn-master\fetch\_data\fetch\_faster\_rcnn\_final\_model.m  下载训练好的模型

（下载失败的话，可以用百度云下载：<https://pan.baidu.com/s/1hsFKmeK> ，解压到faster\_rcnn-master下）

（4）修改faster\_rcnn-master\experiments\script\_faster\_rcnn\_demo.m的model\_dir为你下载的模型，然后运行。

最终得到：



在训练前请确保你的路径faster\_rcnn-master\external\caffe\matlab\caffe\_faster\_rcnn下有以下文件：



（我的opencv版本是2.4.9，cuda版本是7.5，因版本不同上述文件和你的编译结果可能会有差异。+caffe文件夹是从caffe-master或caffe-faster-R-CNN里拷贝过来的。）

如果你没有按上面说的测试过，请先运行：

（1）faster\_rcnn-master\faster\_rcnn\_build.m

（2）faster\_rcnn-master\startup.m

然后再进行下面的修改。

1 、VOCdevkit2007\VOCcode\VOCinit.m的修改

**（1）路径的修改**

VOCopts.annopath=[VOCopts.datadir VOCopts.dataset '/Annotations/%s.xml'];

VOCopts.imgpath=[VOCopts.datadir VOCopts.dataset '/JPEGImages/%s.jpg'];

VOCopts.imgsetpath=[VOCopts.datadir VOCopts.dataset '/ImageSets/Main/%s.txt'];

VOCopts.clsimgsetpath=[VOCopts.datadir VOCopts.dataset '/ImageSets/Main/%s\_%s.txt'];

VOCopts.clsrespath=[VOCopts.resdir 'Main/%s\_cls\_' VOCopts.testset '\_%s.txt'];

VOCopts.detrespath=[VOCopts.resdir 'Main/%s\_det\_' VOCopts.testset '\_%s.txt'];

上面这些路径要正确，第一个是xml标签路径；第二个是图片的路径；第三个是放train.txt、val.txt、test.txt和trainval.txt的路径；第四、五、六个不需要；一般来说这些路径不用修改，你做的数据集格式和VOC2007相同就行。（图片格式默认是jpg，如果是png，修改上面第二行的代码即可。）

**（2）训练集文件夹修改**

VOCopts.dataset = '你的文件夹名';

然后将VOC2007路径注释掉，上面“你的文件夹名”是你放Annotations、ImageSets、JPEGImages文件夹的文件夹名。

**（3）标签的修改**

VOCopts.classes={...

 '你的标签1'

 '你的标签2'

 '你的标签3'

 '你的标签4'};

将其改为你的标签。

2 、VOCdevkit2007\results

results下需要新建一个文件夹，名字是1. (2)中“你的文件夹名”。“你的文件夹名”下新建一个Main文件夹。(因为可能会出现找不到文件夹的错误)

**3 、VOCdevkit2007\local**

local下需要新建一个文件夹，名字是1. (2)中“你的文件夹名”。（同上）

**4 、function\fast\_rcnn\fast\_rcnn\_train.m**

ip.addParamValue('val\_iters', 500, @isscalar);

ip.addParamValue('val\_interval', 2000, @isscalar);

可能在randperm(N,k)出现错误，根据数据集修改。（VOC2007中val有2510张图像，train有2501张，作者将val\_iters设为500，val\_interval设为2000，可以参考作者的设置修改，建议和作者一样val\_iters约为val的1/5，val\_interval不用修改）

5、function\rpn\proposal\_train.m

这里的问题和fast\_rcnn\_train.m一样。

**6.imdb\imdb\_eval\_voc.m**

%do\_eval = (str2num(year) <= 2007) | ~strcmp(test\_set,'test');

do\_eval = 1;

注释掉

do\_eval = (str2num(year) <= 2007) | ~strcmp(test\_set,'test');

并令其为1，否则测试会出现精度全为0的情况

7. imdb\**roidb\_from\_voc.m**

ip.addParamValue('exclude\_difficult\_samples', true, @islogical);

不包括难识别的样本，所以设置为true。(如果有就设置为false)

8.网络模型的修改

**（1） models\ fast\_rcnn\_prototxts\ZF\ train\_val.prototxt**

input: "bbox\_targets"

input\_dim: 1 # to be changed on-the-fly to match num ROIs

input\_dim: 84 # 根据类别数改，该值为（类别数+1）\*4 #################

input\_dim: 1

input\_dim: 1

input: "bbox\_loss\_weights"

input\_dim: 1 # to be changed on-the-fly to match num ROIs

input\_dim: 84 # 根据类别数改，该值为（类别数+1）\*4 ############

input\_dim: 1

input\_dim: 1

layer {

 bottom: "fc7"

 top: "cls\_score"

 name: "cls\_score"

 param {

 lr\_mult: 1.0

 }

 param {

 lr\_mult: 2.0

 }

 type: "InnerProduct"

 inner\_product\_param {

 num\_output: 21 #根据类别数改该值为类别数+1 #########

layer {

 bottom: "fc7"

 top: "bbox\_pred"

 name: "bbox\_pred"

 type: "InnerProduct"

 param {

 lr\_mult: 1.0

 }

 param {

 lr\_mult: 2.0

 }

 inner\_product\_param {

 num\_output: 84 #根据类别数改，该值为（类别数+1）\*4 ##########

**（2） models\ fast\_rcnn\_prototxts\ZF\ test.prototxt**

layer {

 bottom: "fc7"

 top: "cls\_score"

 name: "cls\_score"

 param {

 lr\_mult: 1.0

 }

 param {

 lr\_mult: 2.0

 }

 type: "InnerProduct"

 inner\_product\_param {

 num\_output: 21 #类别数+1 ##########

layer {

 bottom: "fc7"

 top: "bbox\_pred"

 name: "bbox\_pred"

 type: "InnerProduct"

 param {

 lr\_mult: 1.0

 }

 param {

 lr\_mult: 2.0

 }

 inner\_product\_param {

 num\_output: 84 #4\*(类别数+1) ##########

**（3） models\ fast\_rcnn\_prototxts\ZF\_fc6\ train\_val.prototxt**

input: "bbox\_targets"

input\_dim: 1 # to be changed on-the-fly to match num ROIs

input\_dim: 84 # 4\*(类别数+1) ###########

input\_dim: 1

input\_dim: 1

input: "bbox\_loss\_weights"

input\_dim: 1 # to be changed on-the-fly to match num ROIs

input\_dim: 84 # 4\*(类别数+1) ###########

input\_dim: 1

input\_dim: 1

layer {

 bottom: "fc7"

 top: "cls\_score"

 name: "cls\_score"

 param {

 lr\_mult: 1.0

 }

 param {

 lr\_mult: 2.0

 }

 type: "InnerProduct"

 inner\_product\_param {

 num\_output: 21 #类别数+1 ############

layer {

 bottom: "fc7"

 top:"bbox\_pred"

 name:"bbox\_pred"

 type:"InnerProduct"

 param {

 lr\_mult:1.0

 }

 param {

 lr\_mult:2.0

 }

 inner\_product\_param{

 num\_output: 84 #4\*（类别数+1） ###########

**（4） models\ fast\_rcnn\_prototxts\ZF\_fc6\ test.prototxt**

layer {

 bottom: "fc7"

 top: "cls\_score"

 name: "cls\_score"

 param {

 lr\_mult: 1.0

 }

 param {

 lr\_mult: 2.0

 }

 type: "InnerProduct"

 inner\_product\_param {

 num\_output: 21 类别数+1 #######

layer {

 bottom: "fc7"

 top: "bbox\_pred"

 name: "bbox\_pred"

 type: "InnerProduct"

 param {

 lr\_mult: 1.0

 }

 param {

 lr\_mult: 2.0

 }

 inner\_product\_param {

 num\_output: 84 #4\*（类别数+1） ##########

9.solver的修改

solver文件有3个，默认使用的solver是solver\_30k40k.prototxt，如下stage 1 rpn，可以在faster\_rcnn-master\experiments\+Model\ZF\_for\_Faster\_RCNN\_VOC2007.m中更换。

model.stage1\_rpn.solver\_def\_file = fullfile(pwd, 'models', 'rpn\_prototxts', 'ZF', 'solver\_30k40k.prototxt');%solver\_60k80k.prototxt

model.stage1\_rpn.test\_net\_def\_file = fullfile(pwd, 'models', 'rpn\_prototxts', 'ZF', 'test.prototxt');

model.stage1\_rpn.init\_net\_file = model.pre\_trained\_net\_file;

**！！！为防止与之前的模型搞混,训练前把output文件夹删除（或改个其他名），还要把imdb\cache中的文件删除（如果有的话）**

**更为简便的方法是直接用你的数据集的Annotations、ImageSets、JPEGImages文件夹替换VOC2007对应文件夹，那么上面只需进行1.（3）、4、5、7、8的修改。**

**10.开始训练**

（1）.下载预训练的ZF模型：fetch\_data/fetch\_model\_ZF.m

（下载失败的话用百度云下载：[https://pan.baidu.com/s/1o6zipPS](https://pan.baidu.com/s/1o6zipPS%22%20%5Ct%20%22_blank) ，解压到faster\_rcnn-master下，预训练模型参数用于初始化）

（2）.运行：

experiments/script\_faster\_rcnn\_VOC2007\_ZF.m

经过一会的准备工作，就进入迭代了：



**11.训练完后**

训练完后，不要急着马上测试，先打开output/faster\_rcnn\_final/faster\_rcnn\_VOC2007\_ZF文件夹，打开detection\_test.prototxt，作如下修改：

将relu5（包括relu5）前的层删除，并将roi\_pool5的bottom改为data和rois。并且前面input: "data"下的input\_dim:分别改为1,256,50,50(如果是VGG就是1,512,50,50,其他修改基本一样)，具体如下

input: "data"

input\_dim: 1

input\_dim: 256

input\_dim: 50

input\_dim: 50

# ------------------------ layer 1 -----------------------------

layer {

 bottom: "data"

 bottom: "rois"

 top: "pool5"

 name: "roi\_pool5"

 type: "ROIPooling"

 roi\_pooling\_param {

 pooled\_w: 6

 pooled\_h: 6

 spatial\_scale: 0.0625 # (1/16)

 }

}

12.测试

训练完成后，打开\experiments\script\_faster\_rcnn\_demo.m，将模型路径改成训练得到的模型路径：

model\_dir = fullfile(pwd, 'output', 'faster\_rcnn\_final', 'faster\_rcnn\_VOC2007\_ZF')

将测试图片改成你的图片：

im\_names = {'001.jpg', '002.jpg', '003.jpg'};

注意：

        如果你的数据集类别比voc2007数据集多，把script\_faster\_rcnn\_demo.m中的showboxes(im, boxes\_cell, classes, 'voc')作如下修改：

改为：

showboxes(im, boxes\_cell, classes);

或者：

showboxes(im, boxes\_cell, classes, 'default');

即去掉‘voc’或将其改为‘default’。

如果测试发现出现的框很多，且这些框没有目标，可以将阈值设高一些（默认是0.6）：

thres = 0.9;

结果如下：



﻿﻿

﻿﻿