由于实验室服务器连接github连接不上，所以所有在需要从github clone的东西都先在自己的windows笔记本上面下载好，然后通过WinSCP传过去。

首先是代码下载：

git clone [https:*//github.com/jwyang/faster-rcnn.pytorch.git*](https://github.com/jwyang/faster-rcnn.pytorch.git)

若要使用pytorch-1.0的版本，则切到分支：

git checkout pytorch-1.0

然后把整个文件夹传到服务器：/home/ydc/下

随后在faster-rcnn.pytorch下面创建新文件夹data:  
cd faster-rcnn.pytorch && mkdir data

并在data文件夹下创建一个pretrained\_model，用来存放之后的预训练模型

cd data && mkdir pretrained\_model

回到faster-rcnn.pytorch文件夹，安装requirements.txt里面的东西：  
pip install -r requirements.txt

由于此处我用的是conda环境，所以就是：conda install -r requirements.txt

但是由于是用的清华镜像网，所以requirements.txt里面有一些包找不到，于是就一个一个手动安装

随后回到c，执行：  
python setup.py build develop

然后是预训练模型的下载，有两个：VGG16 和 ResNet101

<https://filebox.ece.vt.edu/~jw2yang/faster-rcnn/pretrained-base-models/vgg16_caffe.pth>

<https://filebox.ece.vt.edu/~jw2yang/faster-rcnn/pretrained-base-models/resnet101_caffe.pth>

下载完毕后，放到data/pretrained\_model下面

然后是数据集的下载，我使用的是Pascal\_VOC数据集，可在这三个网站下载：

wget http://host.robots.ox.ac.uk/pascal/VOC/voc2007/VOCtrainval\_06-Nov-2007.tar

wget http://host.robots.ox.ac.uk/pascal/VOC/voc2007/VOCtest\_06-Nov-2007.tar

wget http://host.robots.ox.ac.uk/pascal/VOC/voc2007/VOCdevkit\_08-Jun-2007.tar

由于服务器和我的笔记本连接这三个网站都很慢，所以找到了镜像网站：

<https://pjreddie.com/projects/pascal-voc-dataset-mirror/>

下载好之后，用tar xvf指令，解压到data/VOCdevkit里面：

tar xvf VOCtrainval\_06-Nov-2007.tar

tar xvf VOCtest\_06-Nov-2007.tar

tar xvf VOCdevkit\_08-Jun-2007.tar

数据集格式大致如下：  
VOCdevkit

└── VOC2007

├── Annotations

├── ImageSets

│ └── Main

│ ├── test.txt

│ ├── train.txt

│ ├── trainval.txt

│ └── val.txt

└── JPEGImages

*#VOC数据集的文件夹tree结构：其中Annotations内放xml标注文件，JPEGImages内放图片，*

*#ImageSets/Main/内的四个txt文件分别是测试集、训练集、训练验证集、验证集。*

将这个VOCdevkit创建一个软链接，cd到VOCdevkit上级目录：

ln -s $VOCdevkit \home\VOCdevkit2007

创建软链接相当于创建一个快捷方式，可以把数据集放在其他地方，只需要在faster-rcnn.pytorch/data/VOCdevkit链接到外面的数据集即可

然后开始训练：

配置参数：

CUDA\_VISIBLE\_DEVICES=0 python trainval\_net.py --dataset pascal\_voc --net res101 --epochs 20 --bs 1 --nw 4 --lr 1e-2 --lr\_decay\_step 8 --use\_tfb --mGPUs --cuda

参数含义如下：

CUDA\_VISIBLE\_DEVICES=0 使用哪一个GPU

python trainval\_net.py 运行trainval\_net.py文件

--dataset pascal\_voc 使用哪一个数据集

--net res101 使用哪一个预训练模型

–epochs 20：20个epoch，每个epoch都过一遍全部的训练图。  
–bs 1 batch\_size=1

–nw 4 num\_works=4，4线程进行读取，如果显存小的话就设为1

–lr 1e-2 初始学习率是0.01；可选参数

–lr\_decay\_step 8 每几个epoch学习率衰减一次（默认衰减一次0.1，通过decay\_gamma可调）；可选参数–use\_tfb： 使用tensorboardX实现记录和可视化，不用就不写；可选参数

–mGPUs 多GPU训练，不用就不写该命令；可选参数

–cuda 使用cuda

训练时遇到了很多报错，逐一解决：

报错1：

ImportError: cannot import name \_mask

根据这篇文章

<https://blog.csdn.net/weixin_42279610/article/details/103086379>所说，是因为coco出了问题，所以重新编译coco，解决方法：

下载cocoapi，我放在了faster-rcnn.pytorch/lib目录下，并运行cocoapi/PythonAPI/setup.py相当于就是编译了一次coco

删掉faster-rcnn.pytorch/lib里面的pycocotool文件夹

重新运行训练指令，发现没有这个问题了

报错2：

AssertionError: Path does not exist: /home/ydc/faster-rcnn.pytorch/data/VOCdevkit2007/VOC2007/ImageSets/Main/trainval.txt

此处是因为软链接设置的有问题，后来我直接把数据集放在data下面了，并按照以上报错信息修改了文件名，重新训练，没有报错

报错3：

Connection reset by 101.6.68.185 port 22

长时间不操作，然后自动断开了，由于我是用的MobaXterm软件，所以可以直接设置ssh keepAlive，详见：

<https://www.cnblogs.com/Tom-Ren/p/10024299.html>

训练完成后，会出现保存模型的语句：

save model: models/res101/pascal\_voc/faster\_rcnn\_1\_10\_10021.pth

测试模型的总体效果：

CUDA\_VISIBLE\_DEVICES=1 python test\_net.py --dataset pascal\_voc --net res101 --checksession 1 --checkepoch 10 --checkpoint 10021—cuda

此处的checksession,checkepoch,checkpoint根据自己的参数灵活设置

此处的CUDA\_VISIBLE\_DEVICES我选择了1，因为发现服务器id为0的GPU内存被占用了

执行上述语句后，会把测试数据集里面的几千张照片全部测试一次，给出每个类别的分数和总体平均精确度

若想批量测试图片

python demo.py --dataset pascal\_voc --net res101 --cfg cfgs/res101.yml --load\_dir models --checksession 1 --checkepoch 28 --checkpoint 2813 --image\_dir images –cuda

同样，此处的checksession,checkepoch,checkpoint根据自己的参数灵活设置

此处的--image\_dir images设置为自己保存图片的地方

执行上述语句后，会在images文件夹下面对每一张图片进行目标检测，并生成包含检测框的新图片