

[🏠](#) > [综合](#) > 使用FCN做图像语义分割

使用FCN做图像语义分割

搜索 🔍

[📖 综合](#)[分享到](#)[百度一下](#)

👁 88°C

💬 0评论

[目录\(?\)](#)[\[+\]](#)转载自：http://blog.csdn.net/gavin__zhou/article/details/52142696

FCN原理

原理我已经在上篇博客说过,大家可以参考[FCN原理篇](#)

代码

[FCN](#) 有官方的代码,具体地址是[FCN官方代码](#)不过我用的不是这个代码,我用的是别人修改官方的版本的代码,使用 [Chainer](#) 框架实现的, [Chainer](#) 的源码链接:[Chainer框架源码](#),如果大家使用过 [Keras](#) 的话,应该对它不会感到特别的陌生, [Chainer: a neural network framework](#)好了,我使用的代码是FCN的Chainer implementation, 具体地址是[FCN Chainer implementation](#)

安装

安装很简单,直接 [pip](#) 或者源码安装都可以,但是我在我的机器上装过几次,发现使用 [pip](#) 的方式最后 [fcn.data_dir](#) 这个变量的值会指向到你系统的这个目录需要root权限,所以不推荐使用 [pip](#) 直接安装的方式; 关于此问题的说明见:[fcn.data_dir的问题](#)所以我最后使用的是源码安装的方式,这里推荐使用 [virtualenv](#) 工具建立虚拟环境,实践中发现这是最不会出错的方式,推荐使用!

IT问道推荐

[关闭](#)

银行贷款频频被拒？

“Dr信用牛牛”让你远离信用污点 国内首家信用健康管理平台免费为你提供信用修复方案

clone代码

```
Git clone https://github.com/wkentaro/fcn.git --recursive
```

使用virtualenv安装

```
sudo pip install virtualenv #安装virtualenv  
创建虚拟目录  
virtualenv test-fcn  
cd test-fcn  
激活虚拟环境  
source ./bin/activate  
克隆fcn代码  
git clone https://github.com/wkentaro/fcn.git --recursive  
cd fcn  
安装fcn  
python setup.py develop
```

demo

下载 `VOC2012` 数据集,放入fcn-data-pascal-VOC2012路径下

1. 转换caffe model为Chainer model

```
./scripts/caffe_to_chainermodel.py
```

2. load model,进行分割

```
./scripts/fcn_forward.py --img-files data/pascal/VOC2012/JPEGImages/2007_000129.jpg
```

训练自己的数据

这个前后搞了快一个月,才把最终的训练搞定,其中艰辛很多,在这里写出来供大家参考

准备自己的数据集

IT问道推荐

[关闭](#)

银行贷款频频被拒？

“Dr信用牛牛”让你远离信用污点 国内首家信用健康管理平台免费为你提供信用修复方案

数据集做成 VOC2012 的 `segmentClass` 的样子,下图是示例,上面一张是原图,下面一张是分割图



但是每一种label指定的物体都有对应的具体的颜色,这个我们犯了很多错,最后跟踪代码找出来的,具体的每一类的RGB值如下:

Index	RGB值
0	(0,0,0)

IT问道推荐

[关闭](#)



银行贷款频频被拒？

“Dr信用牛牛”让你远离信用污点 国内首家信用健康管理平台免费为你提供信用修复方案

Index	RGB值
1	(0,128,0)
2	(128,128,0)
3	(0,0,128)
4	(128,0,128)
5	(0,128,128)
6	(128,128,128)
7	(64,0,0)
8	(192,0,0)
9	(62,128,0)
10	(192,128,0)

这里只列出10类的值,更多类的可以看下面这段代码:

```

1 def bitget(byteval, idx):
2     return ((byteval & (1 << idx)) != 0)
3
4
5 def labelcolormap(N=256):
6     cmap = np.zeros((N, 3)) #N是类别数目
7     for i in xrange(0, N):
8         id = i
9         r, g, b = 0, 0, 0
10        for j in xrange(0, 8):
11            r = np.bitwise_or(r, (bitget(id, 0) << 7-j))
12            g = np.bitwise_or(g, (bitget(id, 1) << 7-j))
13            b = np.bitwise_or(b, (bitget(id, 2) << 7-j))
14            id = (id >> 3)
15        cmap[i, 0] = r
16        cmap[i, 1] = g
17        cmap[i, 2] = b

```

IT问道推荐

关闭



银行贷款频频被拒？

“Dr信用牛牛”让你远离信用污点 国内首家
信用健康管理平台免费为你提供信用修复方案

```
18     cmap = cmap.astype(np.float32) / 255 #获得Cmap的RGB值
19     return cmap
20
21 def _label_rgb_to_32sc1(self, label_rgb):
22     assert label_rgb.dtype == np.uint8
23     label = np.zeros(label_rgb.shape[:2], dtype=np.int32)
24     label.fill(-1)
25     cmap = fcn.util.labelcolormap(len(self.target_names))
26     cmap = (cmap * 255).astype(np.uint8) #转换为整数值
27     for l, rgb in enumerate(cmap):
28         mask = np.all(label_rgb == rgb, axis=-1)
29         label[mask] = l
30     return label
```

按照此颜色表做图就没有问题,代码可以正确的读取分割的ground-truth结果

原始的图像放在 `fcn/data/pascal/VOC2012/JPEGImages`

分割的图像放在 `fcn/data/pascal/VOC2012/SegmentationClass`

之后在 `fcn/data/pascal/VOC2012/ImageSets/Segmentation` 写 `train.txt`, `trainval.txt`, `val.txt`, 写入需要进行相应任务的图片的编号

修改代码

`fcn/scripts/fcn_train.py`

```
1 # setup optimizer
2 optimizer = 0.MomentumSGD(lr=1e-10, momentum=0.99) #这里的lr一定要小, 大的话程序会报错, 我使用的是1e-9
3 optimizer.setup(model)
4
5 # train
6 trainer = fcn.Trainer(
7     dataset=dataset,
8     model=model,
9     optimizer=optimizer,
10    weight_decay=0.0005,
11    test_interval=1000,
12    max_iter=100000,
13    snapshot=4000,
14    gpu=gpu,
15 )
```

`fcn/fcn/pascal.py`

IT问道推荐

[关闭](#)



银行贷款频频被拒？

“Dr信用牛牛”让你远离信用污点 国内首家
信用健康管理平台免费为你提供信用修复方案

```
1 target_names = np.array([
2     'background',
3     'aeroplane',
4     'bicycle',
5     'bird',
6     'boat',
7     'bottle',
8     'bus',
9     'car',
10    'cat',
11    'chair',
12    'cow',
13    'diningtable',
14    'dog',
15    'horse',
16    'motorbike',
17    'person',
18    'potted plant',
19    'sheep',
20    'sofa',
21    'train',
22    'tv/monitor',
23 ]) #修改成自己的,记得按照颜色表写
```

fcf/fcf/util.py

```
1 def resize_img_with_max_size(img, max_size=500*500): #修改max_size,按照实际写
2     """Resize image with max size (height x width)"""
3     from skimage.transform import rescale
4     height, width = img.shape[:2]
5     scale = max_size / (height * width)
6     resizing_scale = 1
7     if scale < 1:
8         resizing_scale = np.sqrt(scale)
9         img = rescale(img, resizing_scale, preserve_range=True)
10        img = img.astype(np.uint8)
11    return img, resizing_scale
```

fcf/fcf/models/fcf32s.py

IT问道推荐

[关闭](#)

银行贷款频频被拒？

“Dr信用牛牛”让你远离信用污点 国内首家
信用健康管理平台免费为你提供信用修复方案

```
1 def __init__(self, n_class=21): #修改类别n_class
2     self.n_class = n_class
3     super(self.__class__, self).__init__(
4         conv1_1=L.Convolution2D(3, 64, 3, stride=1, pad=100),
5         conv1_2=L.Convolution2D(64, 64, 3, stride=1, pad=1),
6
7         conv2_1=L.Convolution2D(64, 128, 3, stride=1, pad=1),
8         conv2_2=L.Convolution2D(128, 128, 3, stride=1, pad=1),
9
10        conv3_1=L.Convolution2D(128, 256, 3, stride=1, pad=1),
11        conv3_2=L.Convolution2D(256, 256, 3, stride=1, pad=1),
12        conv3_3=L.Convolution2D(256, 256, 3, stride=1, pad=1),
13
14        conv4_1=L.Convolution2D(256, 512, 3, stride=1, pad=1),
15        conv4_2=L.Convolution2D(512, 512, 3, stride=1, pad=1),
16        conv4_3=L.Convolution2D(512, 512, 3, stride=1, pad=1),
17
18        conv5_1=L.Convolution2D(512, 512, 3, stride=1, pad=1),
19        conv5_2=L.Convolution2D(512, 512, 3, stride=1, pad=1),
20        conv5_3=L.Convolution2D(512, 512, 3, stride=1, pad=1),
21
22        fc6=L.Convolution2D(512, 4096, 7, stride=1, pad=0),
23        fc7=L.Convolution2D(4096, 4096, 1, stride=1, pad=0),
24
25        score_fr=L.Convolution2D(4096, self.n_class, 1, stride=1, pad=0),
26
27        upscore=L.Deconvolution2D(self.n_class, self.n_class, 64,
28                                stride=32, pad=0),
29    )
30    self.train = False
```

训练

```
./scripts/fcn_train.py
```

其会在 `fcn/data/` 下创建一个目录叫做 `SegmentationClassDataset_db`, 里面存放训练的图像的pickle数据, 如果需要修改原始的训练图片则需要 `ckle` 数据作为图像的原始数据

会在 `fcn` 下创建 `snapshot` 这个目录, 里面有训练保存的 `model`, 日志文件等, 重新训练的话, 建议删除此目录

使用自己训练的model

IT问道推荐

[关闭](#)



银行贷款频频被拒？

“Dr信用牛牛”让你远离信用污点 国内首家
信用健康管理平台免费为你提供信用修复方案

```
./scripts/fcn_forward.py -c path/to/your/model -i path/to/your/image
```

结果存放在 `fcn/data/forward_out` 下

1 黄金	5 电缆接头	9 百度金融	13 印刷厂	17 黄金	21 股票推荐	25 老房装修	29 无锡婚纱摄影
2 黄金	6 股票推荐	10 老房装修	14 无锡婚纱摄影	18 必胜宅急送	22 汗蒸房造价	26 外贸英语	30 深圳网站建设
3 必胜宅急送	7 汗蒸房造价	11 外贸英语	15 深圳网站建设	19 jj游戏大厅	23 贵阳装修公司	27 绿能宝官网	31 黄金
4 jj游戏大厅	8 贵阳装修公司	12 绿能宝官网	16 黄金	20 电缆接头	24 百度金融	28 印刷厂	32 黄金

相关推荐

- [GitHub 上 57 款最流行的开源深度学习项目](#)
- [ubuntu下安装dlib时 cpu_dlib.cpp:9:0报错](#)
- [matcaffe错误：Invalid MEX-file: caffe.mexa64,x86_64-linux-gnu/libstdc++.so.6](#)
- [SSD的配置及运行](#)
- [使用Faster-Rcnn进行目标检测](#)
- [博弈论游戏 - 信任的进化](#)
- [制作表白动画](#)
- [自定义一个和京东一样的效果的UIAlertView](#)
- [从源码角度分析UIScrollView嵌套ListView显示不全的问题](#)
- [数据集下载](#)
- [Linux下Qt程序的打包发布](#)
- [yolo-darknet配置安装与测试](#)
- [ResNet && DenseNet](#)
- [Recycleview封装Adapter增加头部尾部](#)
- [那些年刷过的题 - Runtime](#)
- [View draw流程分析](#)
- [自定义UIDatePicker，以后不用第三方的，想怎么改就怎么改](#)
- [Kotlin基本语法](#)

IT问道推荐

关闭



银行贷款频频被拒？

“Dr信用牛牛”让你远离信用污点 国内首家信用健康管理平台免费为你提供信用修复方案

[登录](#)

留下您的评论和问题吧

还没有评论，快来抢沙发吧！

IT问道网正在使用畅言



Sublime Text 3最好的功能、插件和设置 - 移动开



热评话题

[Xcode:用于管理多个 target 配置的 XCConfig 文件 - 移动开发 - IT问道](#)[nodejs+redis应用 - 互联网 - IT问道](#)[NodeJS与Django协同应用开发（0）—— node.js 基础知识 - 互联网...](#)[【Java集合源码剖析】ArrayList源码剖析 - 编程语言 - IT问道](#)[Xcode 扩展 - 移动开发 - IT问道](#)[css页面布局补充 - Web前端 - IT问道](#)[Swift 中的尾递归和蹦床 - 移动开发 - IT问道](#)版权所有 © 广州市禄志信息科技有限公司 粤ICP备16049987号 [商务合作QQ：2948173213（请注明来意）](#) 站长统计

银行贷款频频被拒？

“Dr信用牛牛”让你远离信用污点 国内首家
信用健康管理平台免费为你提供信用修复方案