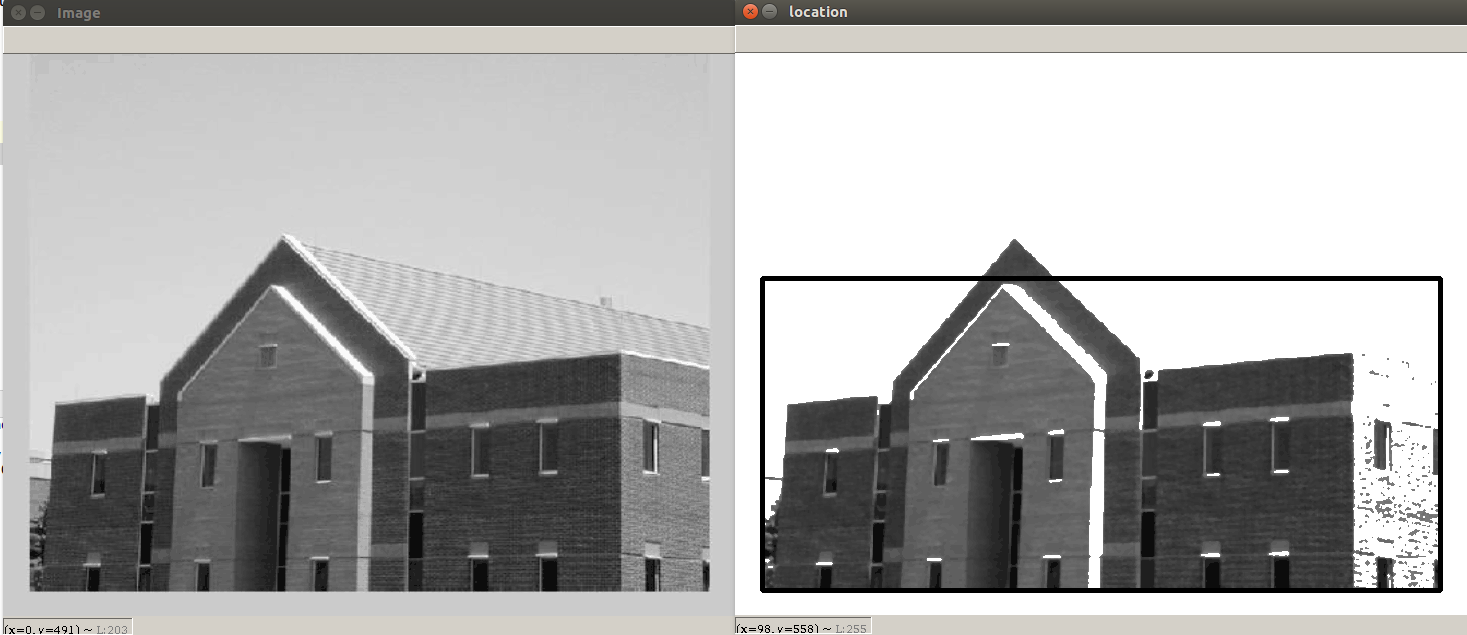
# 目标检测方法探究

目标检测旨在从图片中识别出是否含有我们指定的目标，并且标记出其位置。针对该问题的方法很多，从传统的基于像素点的图像处理方法到如今基于深度学习的各类目标检测算法。当然，相比于一般的图像处理方法，基于深度学习的目标检测算法具有强大的分析能力，能够识别出复杂情况下的多种目标。

## 一、基于阈值的图像分割方法

在工业生产中，有时我们只是想确认目标点的位置，而且背景区域简单，此时图像分割方法可以帮助我们定位相关物体的位置。图像分割方法一般包括基于阈值的分割方法、基于区域的分割方法、基于边缘的分割方法、基于数学形态的分割方法以及基于特定理论的分割方法等，在一些简单的场景中，如当背景和待检测物体具有一定的差异性时，基于阈值的图像分割算法可以取得很好的结果。



图：图像分割确定目标位置的测试图

## 二、基于Tensorflow训练自己的目标检测模型

### 1、环境配置

安装python，推荐使用Anaconda包管理器

安装tensorflow: pip install tensorflow

下载安装TensorFlow Model模型库，[安装步骤](https://github.com/tensorflow/models/blob/master/research/object_detection/g3doc/installation.md)：

git clone https://github.com/tensorflow/models.git

### 2、构建自己的模型

#### 2.1、收集训练数据集

git clone https://github.com/datitran/raccoon\_dataset.git

#### 2.2、使用[labelImg](https://github.com/tzutalin/labelImg)对数据进行打标

#### 2.3、将图片和标注转换为TFRecord格式

Tensorflow目标检测API接受的输入是TFRecord格式的文件，用户必须将训练数据和测试数据都转为相应的TFRecord文件。此外，用户还应该定义一个包含类别编号和类别名字的映射标签。

输入命令：

* python create\_pascal\_tf\_record4raccoon.py --data\_dir=raccoon\_data\_me/images --set=raccoon\_data\_me/train.txt --output\_path=raccoon\_data\_me/train.record --label\_map\_path=raccoon\_data\_me/raccoon\_label\_map.pbtxt --annotations\_dir=raccoon\_data\_me/annotations
* python generate\_tfrecord.py --data\_dir=images/ --set=train.txt --ouput\_path=train.record --label\_map\_path=raccoon\_label\_map.pb --annotations\_dir=annotation
* python create\_pascal\_tf\_record4raccoon.py --data\_dir=raccoon\_data\_me/images --set=raccoon\_data\_me/val.txt --output\_path=raccoon\_data\_me/val.record --label\_map\_path=raccoon\_data\_me/raccoon\_label\_map.pbtxt --annotations\_dir=raccoon\_data\_me/annotations
* python mycreate\_tf\_record.py --data\_dir=images --set=val.txt --output\_path=val.record --label\_map\_path=raccoon\_label\_map.pbtxt --annotations\_dir=annotations

#### 2.4、修改训练配置文件

Tensorflow目标检测API使用protobuf来进行配置训练和评估过程。在较高级别，配置文件分为5个部分：

* model配置 定义了被训练模型的类别
* train\_config 定义训练模型的参数
* eval\_config 定义评估的指标
* train\_input\_config 定义被训练的数据集
* eval\_input\_config 定义被测试的数据集

结构如下：

model {

(... Add model config here...)

}

train\_config : {

(... Add train\_config here...)

}

train\_input\_reader: {

(... Add train\_input configuration here...)

}

eval\_config: {

}

eval\_input\_reader: {

(... Add eval\_input configuration here...)

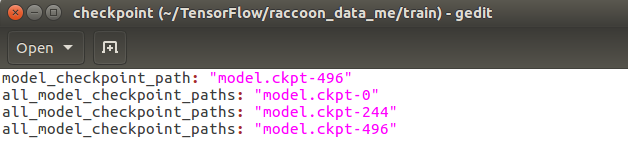
}

#### 2.5、进行训练

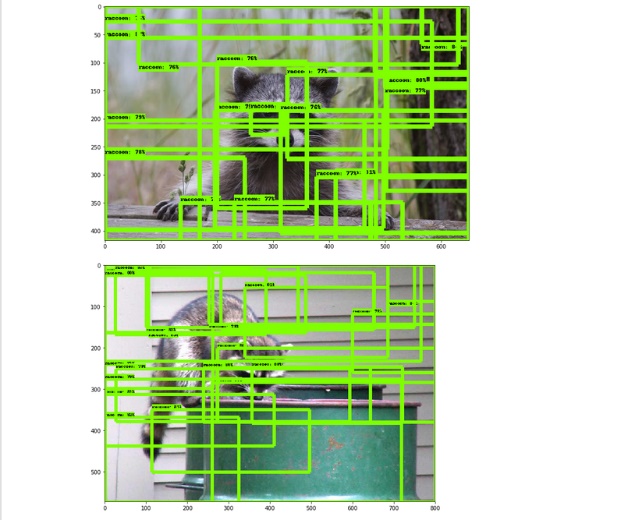
在进行模型训练时，由于我们已有的数据很少，不可能从头开始训练，所以我们可以采用一种策略：在预训练模型的基础上，使用自己的数据对模型进行训练和调优。在这里我们选用的是ssd\_mobilenet\_v1\_coco\_2017\_11\_17模型。

python /home/siemens/TensorFlow/object\_detection/model\_main.py --logtostderr --pipeline\_config\_path=/home/siemens/TensorFlow/raccoon\_data\_me/ssd\_mobilenet\_v1\_raccoon.config --train\_dir=/home/siemens/TensorFlow/raccoon\_data\_me/train

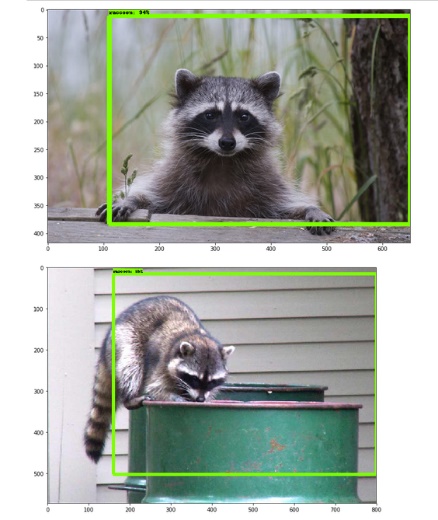
已经生成的节点信息文件集合：



模型训练过程输出的结果：

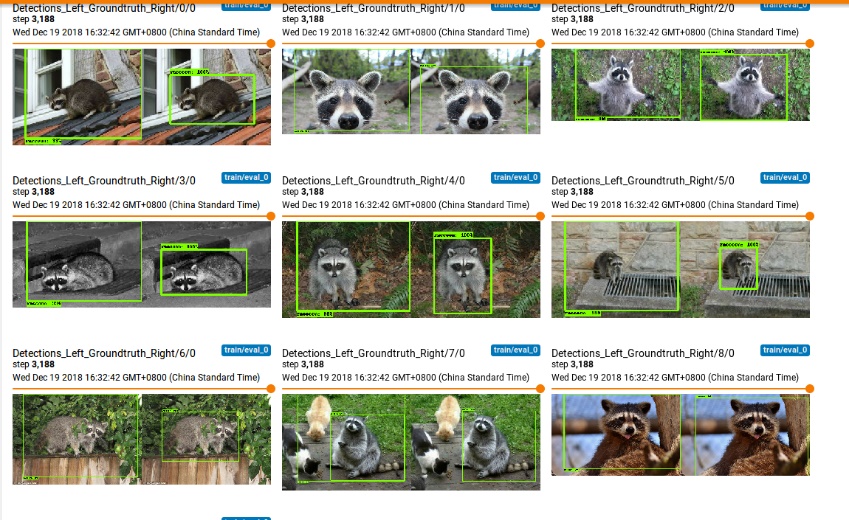


图：迭代次数为0时的检测结果



图：迭代次数为905时的检测结果

Batch=4时的输出：



通过分析可以看出，随着训练次数的增加，结果会明显提升；此外，通过改变batch\_size的大小可以有效提升模型质量。

#### 2.6、将检查点文件导出为冻结的模型文件

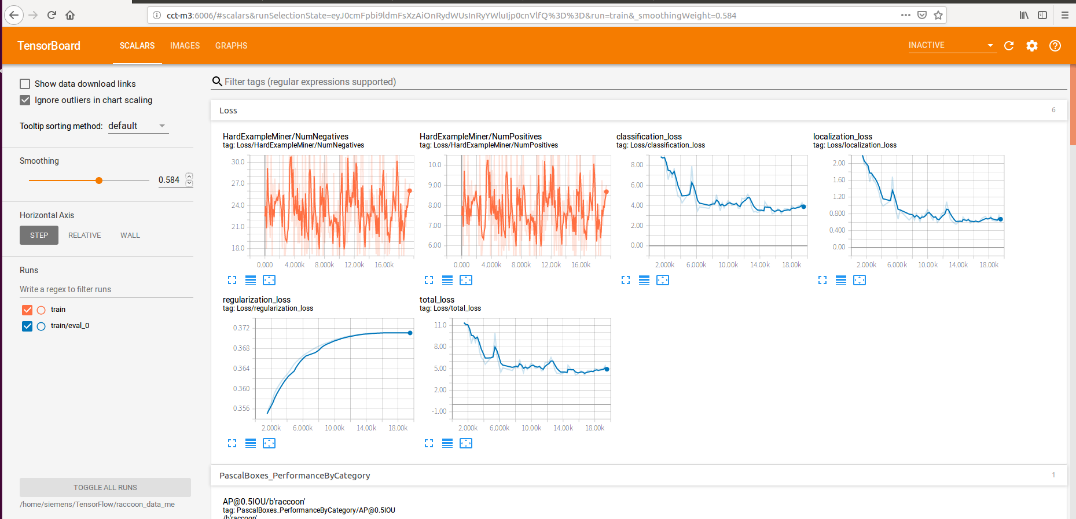
sudo python export\_inference\_graph.py --pipeline\_config\_path=/home/siemens/TensorFlow/raccoon\_data\_me/ssd\_mobilenet\_v1\_raccoon.config --trained\_checkpoint\_prefix=/home/siemens/TensorFlow/raccoon\_data\_me/train/model.ckpt-1798 --output\_directory=/home/siemens/TensorFlow/raccoon\_data\_me/train

#### 2.7、用模型进行浣熊识别

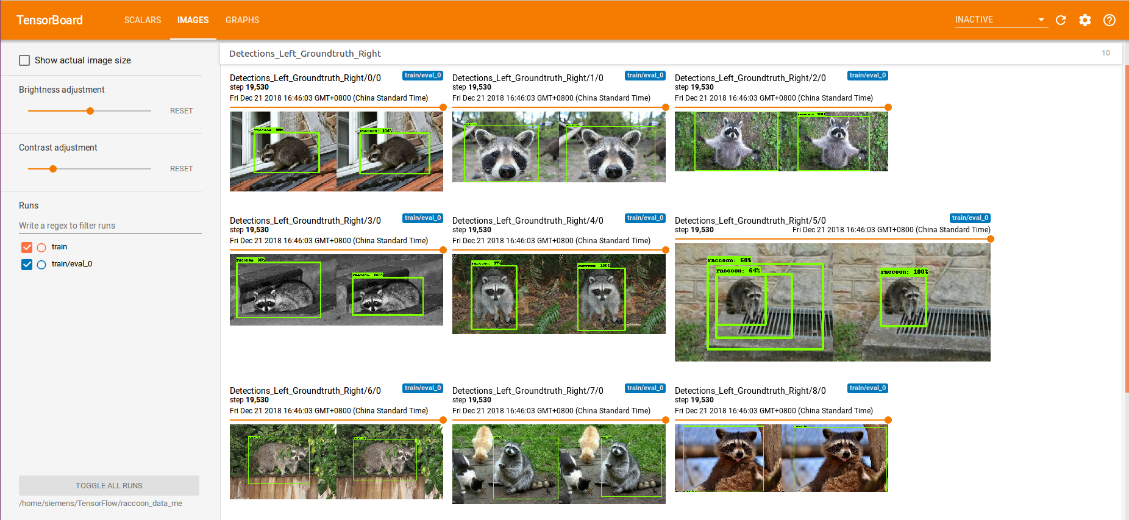
### 3、使用TensorBoard查看训练进程

TensorBoard是一款用于展现TensorFlow程序训练过程的可视化工具，具有强大的数据展示功能。使用方法为：

在Linux终端输入tensorboard --logdir=<file\_path>,其中<file\_path>是你的训练数据所在的文件位置。



图：TensorBoard指标图



图：TensorBoard测试集结果

sudo pip install protobuf

sudo apt-get install python3-tk

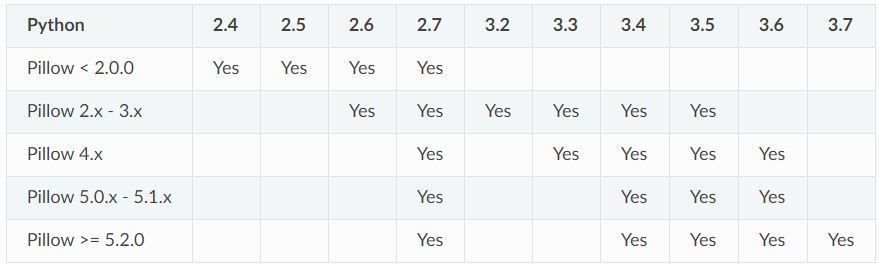
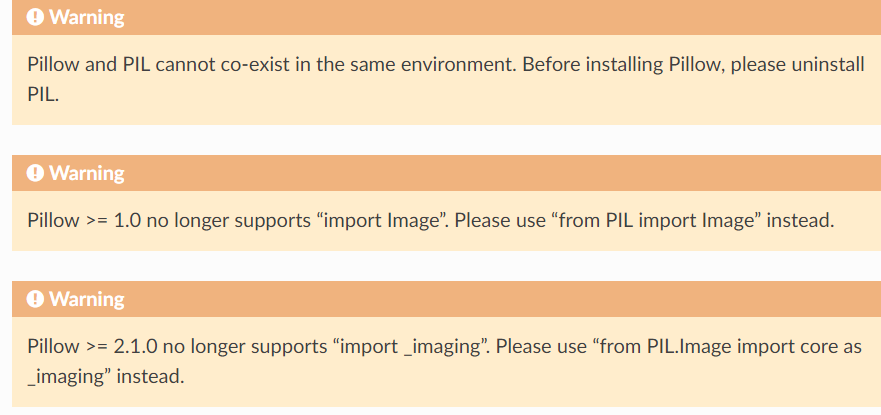
sudo pip install Pillow

sudo pip install lxml

sudo pip install matplotlib

### 4、注意的问题

#### 4.1、不同版本的python与Pillow的版本兼容问题



#### 4.2、如何在非root用户环境下安装protobuf

解压tar.gz文件

tar -zxvf protobuf-python-3.6.0.tar.gz

进入解压后的目录，运行以下命令

./configure --prefix={your\_path}

make && make install

export PATH={your\_path}/bin:$PATH

查看是否安装成功

protoc --version

#### 4.3、编译protoc文件出错

运行$ protoc object\_detection/protos/\*.proto --python\_out=.（python3）

编译出错了，打印如下：

object\_detection/protos/ssd.proto:87:3: Expected "required", "optional", or "repeated".

object\_detection/protos/ssd.proto:87:12: Expected field name.

object\_detection/protos/model.proto: Import "object\_detection/protos/ssd.proto" was not found or had errors.

object\_detection/protos/model.proto:12:5: "Ssd" is not defined

解决方案：

安装最新版protobuf：<https://blog.csdn.net/rookie_wei/article/details/81143814>

#### 4.4、导入object\_detection失败

<https://blog.csdn.net/weixin_41683218/article/details/81214785>

即在anaconda安装目录下的site-packages文件夹下建立一个tensorflow\_model.pth文件，文件内容写models的路径，以及slim的路径

/home/ai-boy/anaconda3/lib/python3.6/site-packages/tensorflow/models/research

/home/ai-boy/anaconda3/lib/python3.6/site-packages/tensorflow/models/research/slim

#### 4.5、运行model\_main.py时，看不到过程输出？

在import导入各种包之后添加tf.logging.set\_verbosity(tf.logging.INFO)，可以实现打印输出训练信息（每100步一次）。

#### 4.6、迭代收敛慢，效果差？

尝试修改batch\_size，参考资料：

<https://blog.csdn.net/ycheng_sjtu/article/details/49804041>

4.7、ModuleNotFoundError: No module named 'absl'？  
解决：pip install absl-py

#### 4.8、NameError: name 'unicode' is not defined

Python2 的unicode 函数在 Python3 中被命名为 str。在 Python3 中使用 ·str 来代替 Python2 中的 unicode.

## 三、知识点补充：

1、Numpy中的[np.expand\_dims()](https://www.jianshu.com/p/da10840660cb)

使用示例：image\_np\_expanded = np.expand\_dims(image\_np, axis = 0)

## 四、参考文档

使用TensorFlow进行目标检测

<https://blog.csdn.net/chenmaolin88/article/details/79357263>

<https://www.cnblogs.com/zongfa/p/9663649.html>

from PyQt5 import QtCore, QtGui, QtWidgets

ImportError: libGL.so.1: cannot open shared object file: No such file or directory

解决：conda install pyqt=4.11.4

import matplotlib.pyplot as plt

ImportError: No module named matplotlib.pyplot

解决：conda install matplotlib

Python2.7安装matplotlib出现版本不匹配，Matplotlib 3.0+ does not support Python 2.x, 3.0, 3.1, 3.2, 3.3, or 3.4.

Beginning with Matplotlib 3.0, Python 3.5 and above is required.

This may be due to an out of date pip.

Make sure you have pip >= 9.0.1.

----------------------------------------

Command "python setup.py egg\_info" failed with error code 1 in /tmp/pip-install-TYhYq9/matplotlib/

解决：pip install matplotlib==2.0