layout: post
title: "RCNN系列文章之RCNN详解"
date: 2020-07-10
description: "目标检测"

tag: 目标检测

RCNN系列的文章主要是RCNN，Fast RCNN， Faster RCNN， Mask RCNN, Cascade RCNN,这一系列的文章是目标检测two-stage算法的代表，这系列的算法精度高，效果好，是一类重要的方法。

论文地址：[Rich feature hierarchies for accurate object detection and semantic segmentation]

主要介绍

用于目标检测与语义分割的丰富特征层次结构

RCNN文章是two-stage算法的开篇之作，奠定了一个基础。在RCNN之前，很多效果好的算法主要是采用融合多个低层次的图像特征与高层次的上下文信息的复杂ensamble（集成）的系统。

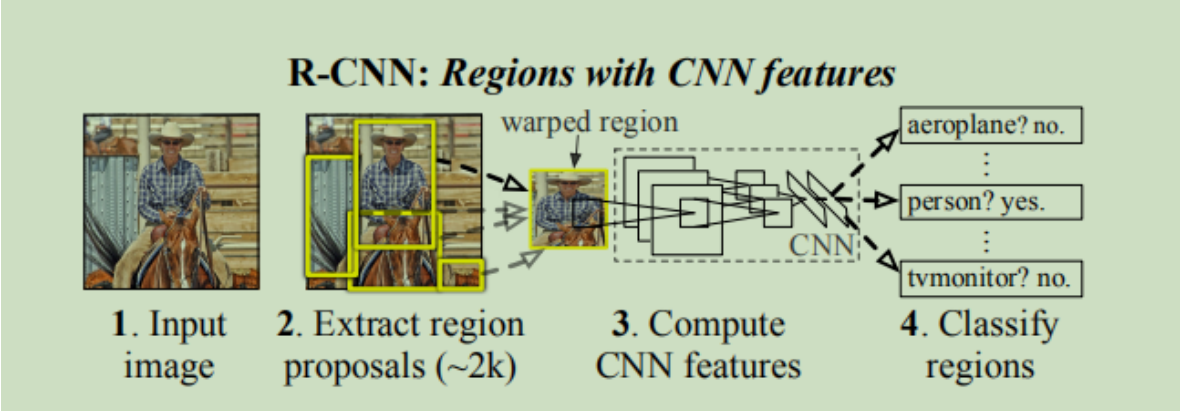
RCNN是一种简单可放缩的目标检测算法，在VOC2012上得到最好的MAP53.3%,算法的两个关键的因素：

1. 使用一个CNN网络来自顶而下的提取候选区域的特征，用于目标检测和分割物体
2. 当数据集的规模较小时，采用其他辅助任务的预训练迁移学习，性能能够获得很大的提升。

RCNN首次在PASCAL VOC数据集上说明了对比基于HOG特征的算法CNN在目标检测上可以获得更高的表现，实现这个工作需要解决两个主要的问题：用深度网络定位目标，利用少量的标注数据训练一个高性能的CNN网络。

CNN实现定位又两种方法第一种是将定位看作一个回归问题，这种方法效果不好精度较差，另一种方法是采用滑窗的方法，但是随着网络的加深，高层的神经元有非常大的感受野，这很难利用滑窗产生精确的定位信息。

RCNN使用区域的方法进行识别解决了CNN定位的问题。测试的时候，RCNN生成2000个类别独立的建议区域（region proposal），每个区域采用cnn提取特征，然后使用线性SVM及逆行分类。不考虑区域的大小，将所有的区域resize到固定尺寸的CNN输入。



由于RCNN在区域上进行操作，因此可以很容易的拓展到语义分割任务中，进行简单的修正之后，RCNN在PASCAL VOC语义分割数据集上获得了最优的结果47.9%.

算法细节介绍

RCNN包含了三个主要的模块：

1. 生成类别独立的regoin proposal（区域建议），为感知器定义候选区域
2. CNN提取固定长度的特征
3. 线性SVM分类器进行类别的分类

Region Proposal

采用selective search(SS,选择性搜索)来进行region proposal的选取。

Feature Extraction

采用AlexNet作为backbone来进行特征提取，从中提取4096维的特征向量，由于需要产生固定尺度的特征向量，需要将每个特征区域都缩放到CNN网络所需要的输入尺度，无论生成的region proposal是什么形状或者长宽比，直接将其缩放到固定的尺寸（227x227），在缩放之前，扩大了被缩放的区域，使得在缩放后，原始区域边界到现有区域边界宽度为p像素，然后直接resize到特定的尺寸。

测试时检测

在测试时，使用selective search在测试图像中生成2000个建议区域，resize这些区域然后输入CNN网络，从特定的层提取特征，对于每一个类利用SVM来给提取得到的特征向量得分。在得到所有的特征区域的得分之后，采用贪婪非极大抑制的方法消除重复的框。

运行时特性分析

有两个特点使得RCNN算法高效：

1. 所有的CNN参数共享
2. CNN得到的特征向量是低维的

训练

先使用ImageNet上的预训练参数模型，然后在VOC数据集上进行微调。

首先，去掉AlexNet的最后的分类层，将1000维的输出更换为21维的输出，然后使用resize后的region proposal作为输入图像进行训练。在每次的SGD迭代中，采样得到32个正例的region proposal（包含所有的类别）和96个背景的regoin（采用128的batch size），由于正例数目相比较背景类的数目相对较少，因此通常对正例样本进行过采样。

分类器， 对于检测汽车的二分类器。一个紧紧包裹着一辆汽车的图像区域就是正例。同样的，没有汽车的就是背景区域，也就是负例。较为不明确的是怎样标注哪些只和汽车部分重叠的区域。可以采用IoU重叠阈值来解决这个问题，低于这个阈值的就是负例。这个阈值我们选择了0.3，这个阈值是在验证集上基于{0, 0.1, ... 0.5}通过**网格搜索**得到的。我们发现认真选择这个阈值很重要。如果设置为0.5，可以降低mAP5个点，设置为0，就会降低4个点。正例就严格的是标注的框。（归结起来就是与ground truth重叠度低于0.3的就是负例，正例就是严格的ground truth）。

得到特征向量之后，就训练线性SVM分类器。由于训练数据太大不能完全的载入内存，采用标准的困难样本挖掘来进行处理，采用hard negative mining算法收敛的非常快。

注意：这里有两个问题：

1. 为什么在训练cnn的时候正例负例定义与训练SVM时不一致？
2. 为什么不直接采用softmax而是采用SVM？

1. 训练CNN的时候，IOU大于0.5标记为正样本，其他的标记为背景，而在训练SVM的时候，IOU小于0.3的标记为负样本，ground truth为正样本，其他的丢弃。在训练CNN的时候对正例样本定义相对宽松，会在一定程度上加大正例的数据量，防止网络的过拟合，而SVM这种算法的机制，适合小样本的训练，因此对真样本限制严格。
2. 作者尝试了采用softmax直接进行训练，但是效果很差，作者认为当IOU大于0.5就认为是正样本会导致定位准确度的下降，而又需要采用IOU阈值0.5来训练CNN，因此采用CNN+SVM结合的方法来完成算法。

可视化分析

RCNN中采用了可视化的方法来查看CNN的特征提取效果以及作用，同时采用一种检测分析方式来理解调参对于各层网络的影响，可参考：

```
D. Hoiem, Y. Chodpathumwan, and Q. Dai. Diagnosing error in object detectors. In *ECCV*. 2012.
```