Literature Study Report

Object Detection : A Simple Review

冯浩哲

2017 年 11 月 5 日

提纲

Object Detection, Semantic Segmentation, Instance Segmentation是Computer Version的三大主流领域，而Instance Segmentation需要结合前两者的特点，从而同时完成多目标检测与分割任务。

本报告主要从**Region-Based-CNN(R-CNN)**[3],**Fast-RCNN**[2],**Faster-RCNN**[4]三篇文章出发，描述三篇文章的侧重点与两个逐渐形成的主流框架，这两个框架是Deep Learning 应用于 Object Detection的基础框架。该报告主要从以下6个方面展开：

- Object Detection要完成什么样的任务，训练数据集是什么样的
- R-CNN提出了什么样的架构，解决了什么问题
- Fast R-CNN提出了什么架构，在哪些方面改进了R-CNN
- Faster R-CNN提出了什么架构，如何改进R-CNN
- 现在的前沿架构与上述架构的异同
- 3D Object Detection 方向主流架构与困境

Object Detection要完成什么样的任务

Object Detection主要需要完成2个任务:

- 图片分类

  对于给定的图片，我们需要完成对图片中目标进行分类的任务，如某张图片所含有的物体是狗还是马，是汽车还是飞机这种任务。有的图片含有多个目标物体，因此这种分类也可能是多分类任务。
- 目标定位

  目标定位，即给定一张图片，使用Bounding Box来框出目标物体所在的位置。目标问题是目标检测的核心问题，也是深度学习框架所要解决的核心问题。

Object Detection训练数据集是什么样的

我们以PASCAL VOC数据集为例。

PASCAL VOC数据集提供了不同目标检测任务(如Bicycle,Bird,Boat)的训练数据集与测试数据集。每一张图片有一个对应的xml，xml记录了该图片中的目标种类以及每一种目标相应的Bounding Box。 图片都是统一的jpg格式的2D图片，但是大小不一。

R-CNN提出了什么样的架构，解决了什么问题

R-CNN的主要贡献在于提出了Selective Search算法来选取待分类的候选区域，并确定了深度学习用于Object Detection的主流框架。

- Selective Search

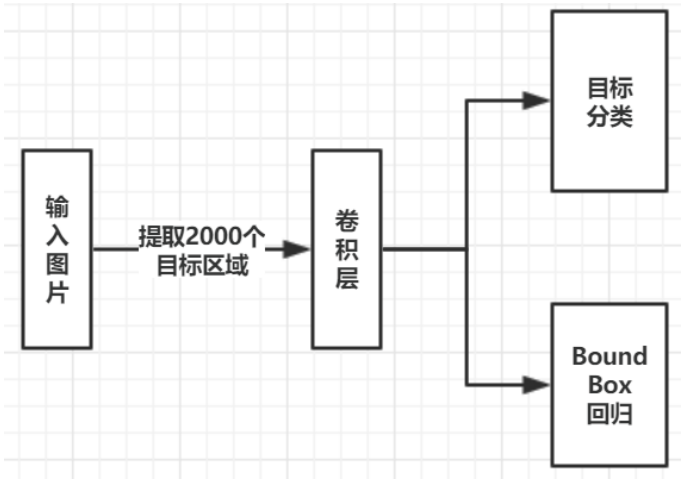
  如何选取待分类的候选区域一直是Object Detection的核心问题。传统做法一般是使用由小到大的滑窗对每一张图片提取相应大小的patch，并对每一个patch进行预测。这意味着一张图需要生成10^4量级的候选区域，因此这种做法是极其浪费计算时间的。

  R-CNN提出了一种Selective Search的做法，能够在一张图上有效寻找出覆盖所有目标的2000个区域，它的具体过程是这样的：

  - 使用[1]中提到的粗略实例分割方法将整个图片划分为若干个区域$Image = \{R_1, R_2, ..., R_n\}$，每个区域都包含一个可能实例。
  - 对于这些实例区域进行基于相似度的层次聚类，聚为2000类，将聚类区域使用矩形框框出，作为我们的目标区域。
  - 将目标区域进行插值，使得插值后区域大小为网络标准输入大小

R-CNN提出了什么样的架构，解决了什么问题

R-CNN所确定的框架如下：



其中，Bound-Box Regression接受卷积后的输出向量作为输入，并输出相应的4个位置坐标:x, y, w, h。其中x,y表示左上角的坐标，而w,h为矩形框的宽和高。在分类任务中，采用SVM对输出向量进行分类。

在训练过程中，我们一般在目标周围进行采样，将采样区域与目标区域的IoU高于0.5的作为正样本，小于0.3的作为负样本，并用Bounding Box的Ground Truth作为标签进行训练。

Fast R-CNN提出了什么架构，在哪些方面改进了R-CNN

Fast R-CNN主要在以下四个方面对R-CNN进行了改进:

- Region Proposal算法
- Region插值过程
- 分类过程

Fast R-CNN提出了什么架构，在哪些方面改进了R-CNN

- Region Proposal算法

  R-CNN提出的结构是先采用Selective Search进行Region Proposal，但是这意味着我们对一个图片的多个被重叠的部分要进行重复计算。Fast R-CNN提出了RoI Projection的方法，先对整张图直接进行卷积，然后再用Selective Search提取输入图片的区域，将这些区域依照卷积操作直接映射到最后一层的Feature Map上，直接提取Feature Map的Patch进行分类与Bounding Regression，具体算法如下:

  - 记录原图的RoI左上与右下的坐标$(x_L, y_L), (x_R, y_R)$.
  - 假设输入尺寸是$w * h$,每经过一层$Kernel = n * m, Stride = s, Padding = p$的卷积层，做坐标变换为:

$$x_{L,R} = min([\frac{x_{L,R} + p}{stride}], \frac{w - n + 2p + s}{s})$$
$$y_{L,R} = min([\frac{y_{L,R} + p}{stride}], \frac{h - m + 2p + s}{s})$$

Fast R-CNN提出了什么架构，在哪些方面改进了R-CNN

Fast R-CNN主要在以下四个方面对R-CNN进行了改进:

- Region插值过程

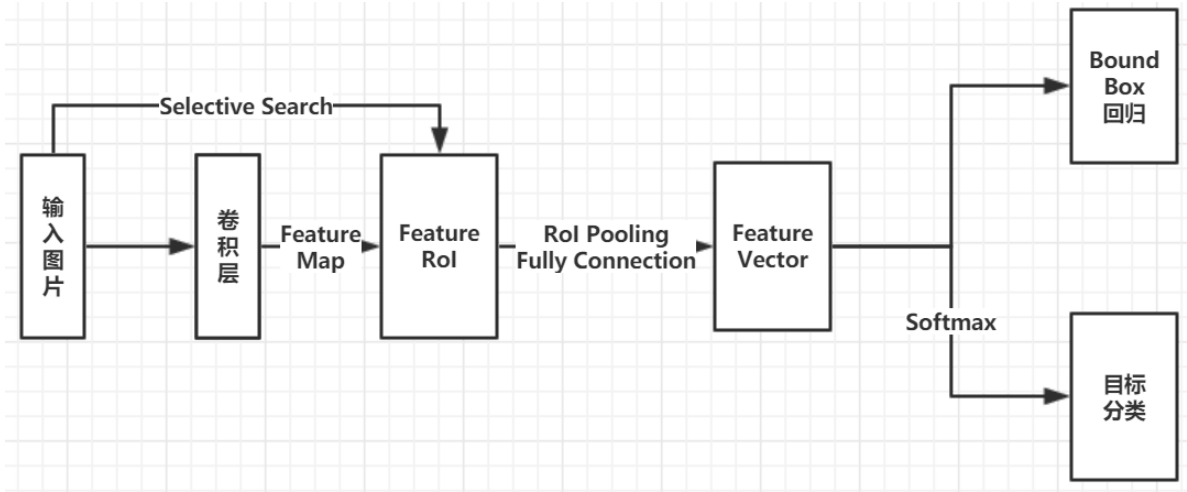
  Fast R-CNN对于原图中需要将图像改变为固定的输出尺寸时使用的插值方法提供了改进的建议，因为插值的方法并不能做反向传播。Fast R-CNN采用了ROI Pooling的做法来进行输出尺寸更改。即如果我们想要$W * H$的固定输出，但是输入是$w * h$的话，我们就采用大小为$\frac{w}{W} * \frac{h}{H}$的Maxpooling框来使它达到固定的输入。具体实现细节可以参阅Github上的代码。

- 分类过程

  Fast R-CNN采用softmax替代了R-CNN的SVM进行分类，取得了很好的效果。

Fast R-CNN提出了什么架构，在哪些方面改进了R-CNN

因此，Fast R-CNN的框架可以概括如下：



在网络训练的过程中，我们先训练分类部分，再固定分类部分的卷积层参数训练Bounding Box Regression部分，再训练分类部分，依次进行。每次训练Bounding Box Regression部分的时候都固定卷积层参数。

Faster R-CNN提出了什么架构，在哪些方面改进了Fast R-CNN

Faster R-CNN主要在Region Proposal生成过程方面对Fast R-CNN进行了改进:

- Region Proposal生成过程 Faster R-CNN在Detection部分保留了Fast R-CNN的网络结构，并在结构基础上直接增加了可训练的Region Proposal Network以直接取代Selective Search的步骤，具体有以下几个步骤：
  - 选取Detection部分输出的最后一层Feature Map，对Feature Map的每一个点以该点为中心生成3*3的区域，对该区域做全连接生成$256-d$向量。
  - 将该点反RoI投影到输入图像所对应的像素位置(与前面类似，公式为$x_{inv}=x*stride+\frac{kernel-1-2p}{2}$),以该像素点为中心，按3组宽度与3组纵横比的组合生成9个锚盒(Anchor Box)并编号。
  - 对第一步得到的256-d的向量做两次预测。第一次输出$1*9$的向量，每一个向量的值代表对应的锚盒有目标物体的confidence，是一个二值预测。第二次输出$4*9$的向量，代表每一个锚盒的Bounding Box Regression预测。
  - 将有目标物体的锚盒以及相应的Bounding Box作为Region Of Interest，将这些锚盒送入Detection Net进行检测。

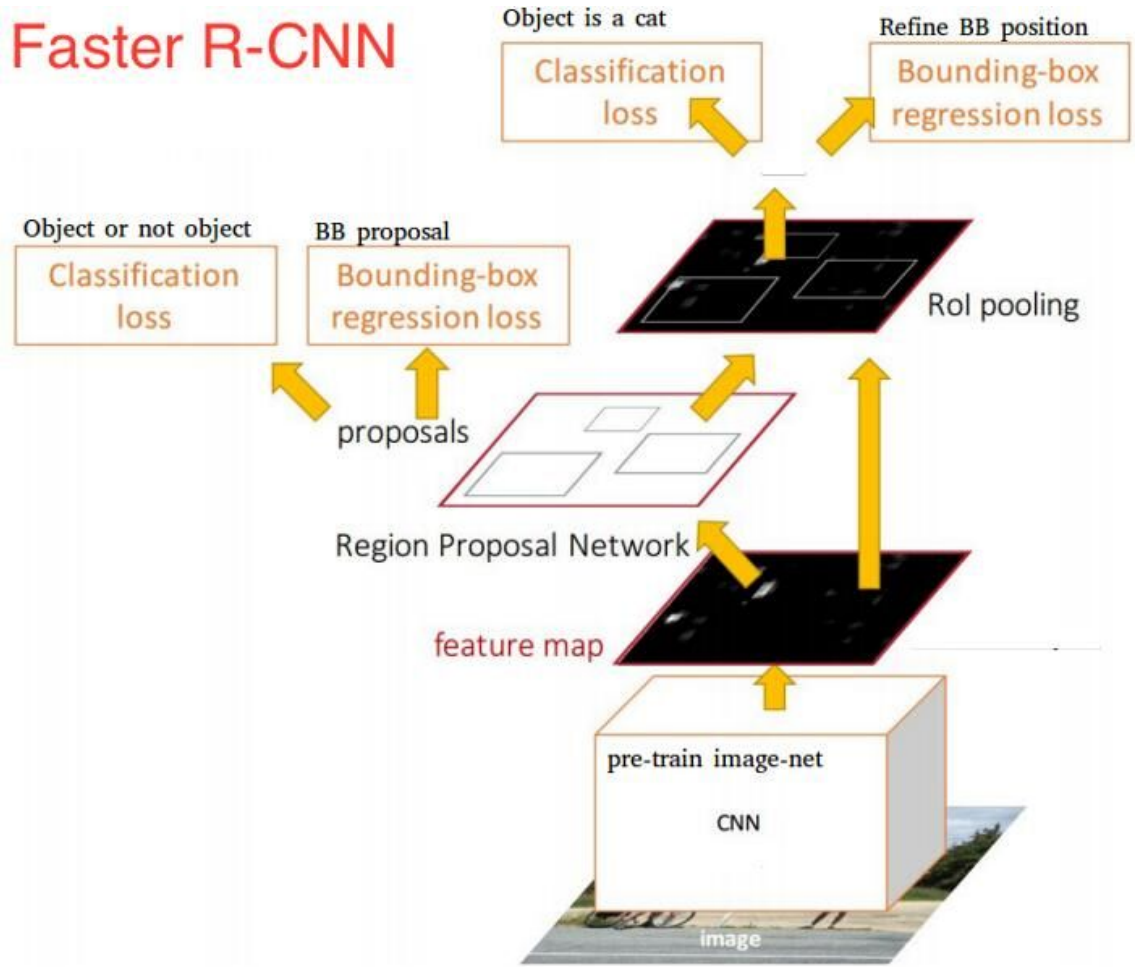
Faster R-CNN提出了什么架构，在哪些方面改进了Fast R-CNN

Region Proposal Net与Detector Net是两个共用卷积层参数的网络，因此训练是关键问题。文章给出了以下有效的训练策略：

- 首先，将所有参数初始化训练Region Proposal Net。训练中标定锚盒与Ground Truth的IoU¿0.7的为前景，标定0.1¡IoU¡0.3的为背景，按照Fast RCNN的训练步骤依次训练分类与Bounding Box Regression部分。
- 其次，利用Region Proposal Net所提供的RoI训练Detection部分网络，按照Fast RCNN的训练步骤依次训练分类与Bounding Box Regression部分。
- 最后，固定Detection部分网络的卷积层参数，再训练Region Proposal Net。
- 依次进行训练直到收敛。接下来每次训练Region Proposal Net的时候都固定卷积层参数。

Fast R-CNN提出了什么架构，在哪些方面改进了R-CNN

因此，Faster R-CNN的框架可以概括如下：




Pedro F. Felzenszwalb and Daniel P. Huttenlocher.
Efficient graph-based image segmentation.
International Journal of Computer Vision, 59(2):167–181, 2004.


Ross Girshick.
Fast r-cnn.
In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015.


Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik.
Rich feature hierarchies for accurate object detection and semantic segmentation.

In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587, 2014.


Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun.
Faster r-cnn: Towards real-time object detection with region proposal networks.
In *Advances in neural information processing systems*, pages 91–99, 2015.