1.Due to this complexity, current approaches (e.g., [9, 11, 19, 25]) train models in multi-stage pipelines that are slow and inelegant

一直以来，都是多个阶段，完成检测，Fast R-CNN，是一种端到端的检测算法，同时输出边框和类别。2014年R-CNN横空出世，首次将卷积神经网络带入目标检测领域。受SPPnet启发，rbg在15年发表Fast R-CNN，它的构思精巧，流程更为紧凑，大幅提高目标检测速度。

2.Fast R-CNN解决了哪些问题

**（1）测试速度慢**

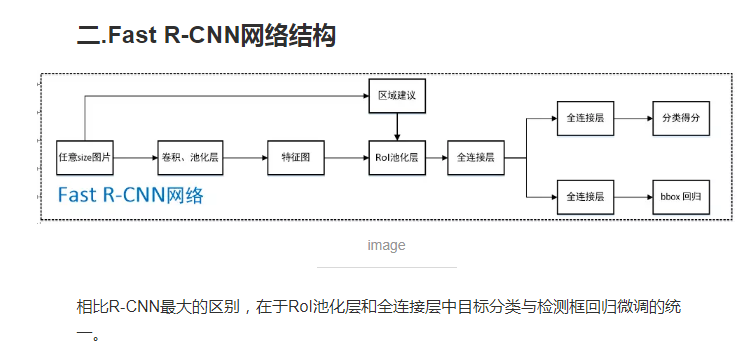
R-CNN中用CNN对每一个候选区域反复提取特征，而一张图片的2000个候选区域之间有大量重叠部分，这一设定造成特征提取操作浪费大量计算。

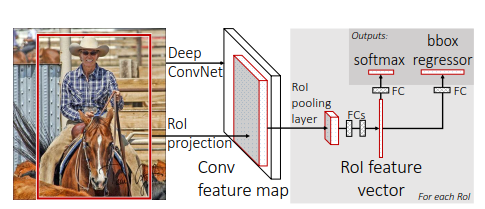
Fast R-CNN将整个图像归一化后直接送入CNN网络，卷积层不进行候选区的特征提取，而是在最后一个池化层加入候选区域坐标信息，进行特征提取的计算。

**（2）训练速度慢（同上）**

**（3）训练所需空间大**

R-CNN中目标分类与候选框的回归是独立的两个操作，并且需要大量特征作为训练样本。Fast R-CNN将目标分类与候选框回归统一到CNN网络中来，不需要额外存储特征。





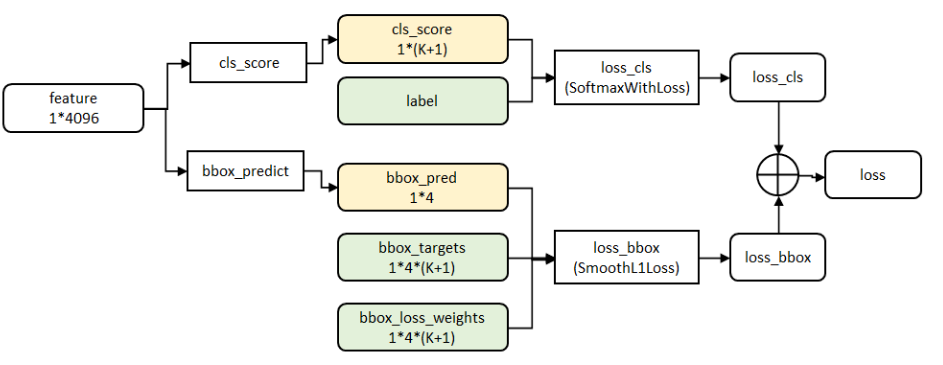
**3.RoI池化层**

RoI池化层可以说是SPP（spatial pyramid pooling）的简化版。RoI池化层去掉了SPP的多尺度池化，直接用MxN的网格，将每个候选区域均匀分成M×N块，对每个块进行max pooling。从而将特征图上大小不一的候选区域转变为大小统一的特征向量，送入下一层。

**4.特征提取方式**

Fast R-CNN在特征提取上可以说很大程度借鉴了SPPnet，首先将图片用选择搜索算法（selective search）得到2000个候选区域（region proposals）的坐标信息。另一方面，直接将图片归一化到CNN需要的格式，整张图片送入CNN（本文选择的网络是VGG），将第五层的普通池化层替换为RoI池化层，图片然后经过5层卷积操作后，得到一张特征图（feature maps），开始得到的坐标信息通过一定的映射关系转换为对应特征图的坐标，截取对应的候选区域，经过RoI层后提取到固定长度的特征向量，送入全连接层。

**5.联合候选框回归与目标分类的全连接层**



6.其它亮点

**SVD全连接层加速网络**

图像分类任务中，用于卷积层计算的时间比用于全连接层计算的时间多，而在目标检测任务中，selective search算法提取的建议框比较多【约2k】，几乎有一半的前向计算时间被花费于全连接层，就Fast R-CNN而言，RoI池化层后的全连接层需要进行约2k次【每个建议框都要计算】，因此在Fast R-CNN中可以采用SVD分解加速全连接层计算,具体实现如下：

① 物体分类和窗口回归都是通过全连接层实现的，假设全连接层输入数据为x，输出数据为y，全连接层参数为W，尺寸为u×v，那么该层全连接计算为:

*y=Wx*(计算复杂度为u×v)

② 若将W进行SVD分解，并用前t个特征值近似代替，即:

*W=U∑VT≈U(u,1:t)⋅∑(1:t,1:t)⋅V(v,1:t)T*

那么原来的前向传播分解成两步:

*y=Wx=U⋅(∑⋅VT)⋅x=U⋅z*

计算复杂度为u×t+v×t，若t<min(u,v)，则这种分解会大大减少计算量；

在实现时，相当于把一个全连接层拆分为两个全连接层，第一个全连接层不含偏置，第二个全连接层含偏置；实验表明，SVD分解全连接层能使mAP只下降0.3%的情况下提升30%的速度，同时该方法也不必再执行额外的微调操作。

