

## RCNN -> Fast-RCNN -> Faster-RCNN

这里不得不先提的就是为什么会有RCNN这一系列的检测算法，以及为什么它们会被称为深度对象检测的开山之作，我们知道，在CNN火起来之前，对象检测这一问题基本是遵循着“设计手工特征(Hand-crafted feature)+分类器”的思路，而且由于存在着区域搜索的步骤，所以可以认为是计算机用一个小的矩形窗口不断在图像上滑动、缩放，然后用分类器预测当前滑动窗口所在区域是否存在一个感兴趣的对象，自从CNN在CV领域流行起来以后，很多人都开始想，既然CNN的特征比传统手工特征好这么多，那么为什么不用深度神经网络做检测呢？

RCNN算法的核心思想就是对每个区域通过CNN提取特征，然后接上一个分类器预测这个区域包含一个感兴趣对象的置信度，也就是说，转换成了一个图像分类问题（类似imagenet），后面接的这个分类器可以是独立训练的svm也可以是简单的softmax分类。在RCNN论文里，作者还提到两个保证检测速度的关键点：1.所有类别的分类器共享相同的特征输入；2.与传统特征相比，深度特征维度一般比较低，比如VGG16里的4096维。

但是很可惜，即使使用了selective search等预处理步骤来提取潜在的bounding box作为输入，但是RCNN仍会有严重的速度瓶颈，原因也很明显，就是计算机对所有region进行特征提取时会有重复计算，Fast-RCNN正是为了解决这个问题诞生的，作者提出了一个可以看做单层sppnet的网络层，叫做ROI Pooling，这个网络层可以把不同大小的输入映射到一个固定尺度的特征向量，而我们知道，conv、pooling、relu等操作都不需要固定size的输入，因此，在原始图片上执行这些操作后，虽然输入图片size不同导致得到的feature map尺寸也不同，不能直接接到一个全连接层进行分类，但是可以加入这个神奇的ROI Pooling层，对每个region都提取一个固定维度的特征表示，再通过正常的softmax进行类型识别。另外，之前RCNN的处理流程是先提proposal，然后CNN提取特征，之后用SVM分类器，最后再做bbox regression，而在Fast-RCNN中，作者巧妙的把bbox regression放进了神经网络内部，与region分类和并成为了一个

multi-task模型，实际实验也证明，这两个任务能够共享卷积特征，并相互促进。Fast-RCNN很重要的一个贡献是成功的让人们看到了Region Proposal+CNN这一框架实时检测的希望，原来多类检测真的可以在保证准确率的同时提升处理速度，也为后来的Faster-RCNN做下了铺垫。

Fast-RCNN之后的问题已经非常清晰，就是我们能不能把region proposal部分也放到GPU上？rbg大神给的答案当然又是yes，于是有了Faster-RCNN，出现了一个end-to-end的CNN对象检测模型。作者提出，网络中的各个卷积层特征其实可以用来预测类别相关的region proposal，不需要事先执行诸如selective search之类的算法，但是如果简单的在前面增加一个专门提proposal的网络又显得不够elegant，所以最终把region proposal提取和Fast-RCNN部分融合进了一个网络模型，虽然训练阶段仍然要分多步，但是检测阶段非常方便快捷，准确率也与原来的Fast-RCNN相差不多，从此，再也不用担心region proposal提取耗时比实际对象检测还多这种尴尬场景了。