1.1 动机

从上一篇论文中可以发现 RCNN 有几个缺点：

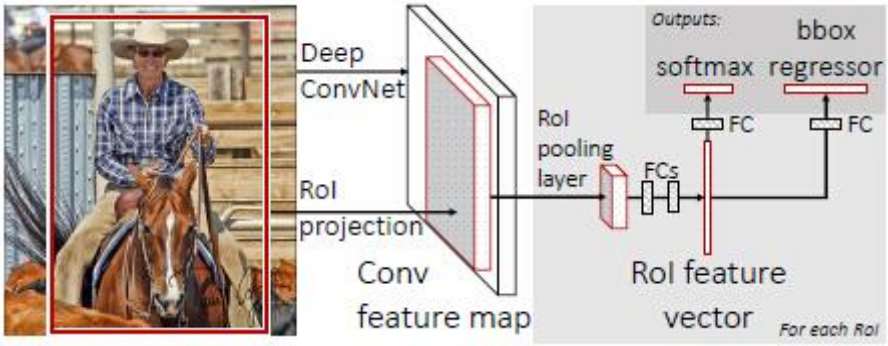
（1） 训练的时候，pipeline 是隔离的，先提候选框，然后 CNN 提取特征，之后用 SVM 分类器，最后再做 bbox regression。FRCN 实现了 end-to-end 的 joint training(提 proposal 阶段除外)。

（2） 训练时间和空间开销大。RCNN 中 ROI-centric 的运算开销大，所以 FRCN 用了 image-centric 的训练方式来通过卷积的 share 特性来降低运算开销；RCNN 提取特征给 SVM 训练时候需要中间要大量的磁盘空间存放特征，FRCN 去掉了 SVM 这一步，所有的特征都暂存在显存中，就不需要额外的磁盘空间了。

（3） 测试时间开销大。依然是因为 ROI-centric 的原因，这点 SPP-Net 已经改进，然后 FRCN 进一步通过 single scale testing 和 SVD 分解全连接来提速。

1.2 原理

Fast RCNN 的大致框架如下：




（1） selective search 在一张图片中得到约 2k 个 object proposal，即 ROI

（2） 缩放图片的 scale 得到图片金字塔，FP 得到 conv5 的特征金字塔。

（3） 对于每个 scale 的每个 ROI，求取映射关系，在 conv5 中 crop 出对应的 patch。并用一个单层的 SPP layer（这里称为 RoI pooling layer）来统一到一样的尺度（对于 AlexNet 是 6x6）。

（4） 继续经过两个全连接得到特征，这特征有分别 share 到两个新的全连接，连接上两个优化目标。第一个优化目标是分类，使用 softmax，第二个优化目标是 bbox regression，使用了一个 smooth 的 L1-loss.

FRCN 中 RoI pooling layer 的作用主要有两个，一个是将 image 中的 rol 定位到 feature map 中对应 patch，另一个是用一个单层的 SPP layer 将这个 feature map patch 下采样为大小固定的 feature 再传入全连接层。而且 FRCN 有两个 loss，对于分类 loss，是一个 N+1 路的 softmax 输出，其中的 N 是类别个数，1 是背景。对于回归 loss，是一个 4xN 路输出的 regressor，也就是说对于每个类别都会训练一个单独的 regressor 的意思，比较有意思的是，这里 regressor 的 loss 不是 L2 的，而是一个平滑的 L1。