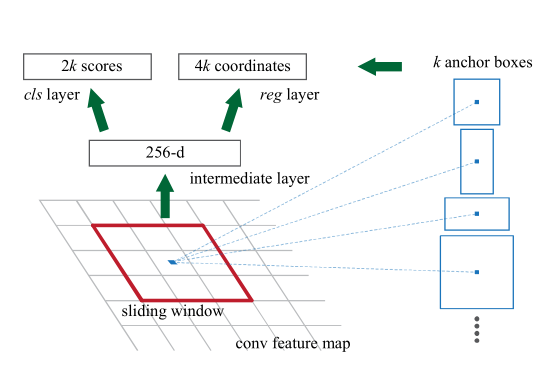
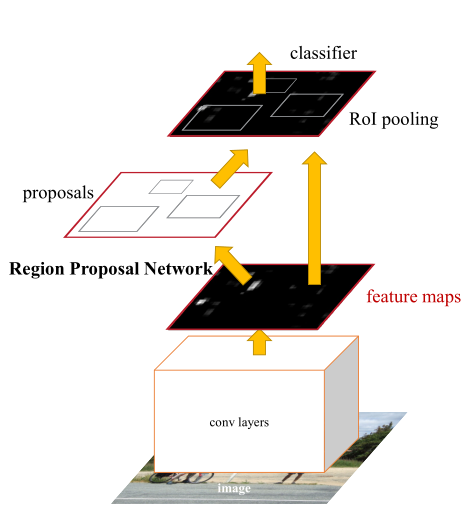
**[Faster R-CNN] NIPS 2015 | Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun**

**主要思想：**Faster RCNN与Fast RCNN非常相似，基于Fast RCNN中SS的时间占一半的缺点，本文提出了全新的**区域推荐网络RPN**替代SS，其余不变。RPN是个全卷积网络，与CNN共享权值，卷积层的特征图作为RPN的输入，输出是每一点预测的**k个边界框相对anchor的位置**(即共k个anchor)和**包含和不包含目标的分数**。边界框不是绝对位置，而是**首创anchor机制**使用相对位置表示。

**贡献/缺点：**Faster RCNN对Fast RCNN的主要改进是在region proposal的方法上用首创的RPN替代SS，**进一步加速**。RPN相当于**attention机制**，它使Faster RCNN成为**第一个完全可微分**的模型。所以，之前的Fast RCNN可以说只是一个高性能的分类器，不具备预测边界框位置的功能。

**论文细节：网络结构：**Faster R-CNN = Fast R-CNN+RPN-ss。



**RPN**

**RPN流程：**前面的CNN以AlexNet为例，conv5特征图的尺寸为13\*13\*256，这一层的特征别送入RPN，在这个特征图上用n\*n\*256的卷积核，一共用了**256个**(n=3是考虑感受野)。那么卷积核一次卷积之后的特征就是1\*1\*256（全卷积），也就是右图中的256-d，这时只是在某一点的位置，之后该特征出两个分支。**第一个分支**用4k个1\*1\*256的卷积核卷积，最后输出**4k个数**，这里的4是一个建议框的参数，即**(x,y,w,h)**；**第二个分支**用2k个1\*1\*256的卷积核卷积，最后输出**2k个数**，这里的2是分别是**有物体和没有物体的概率**。k是Anchor box的类型数，文中k=9，是**3个scale\*3个ratio**。

**Anchor机制:**以上是描述在某一点上RPN怎么操作，RPN第一个分支输出的(x,y,w,h)不是推荐框的绝对位置，而是**相对于anchor的偏置**。在某一点的时候，这个3\*3的卷积中心作为**锚心**，会自动对应3个scale、3个ratio共k=9个锚箱，该支路输出的4k个参数就是这k个锚箱抛出的k\*4个锚（k个预测框的相对位置）。使用3\*3的卷积窗口在特征图上**滑动**，就可以得到13\*13个锚心，13\*13\*9=1521个region proposal。 在通过第二个支路输出的objectness score**筛选**，保留其中分数高的一部分。

**多尺度预测的方法：**预测多尺度的目标，第一种方法是图像/特征金字塔，第二种是卷积核金字塔，本文的方法是首创的具有3个不同尺度的anchors，也可以说是**pyramids of anchors**。

**多尺度的边界框回归：**最后的边界框回归中，在处理不同尺寸的边界框时，SPP和Fast RCNN的思路是共享regressor的参数，但是输入的RoIf eature尺寸不定，本文Faster R-CNN思路是输入的feature尺寸是**固定的3\*3窗口大小**，但是k个不同尺寸的anchor尺寸对应着**k个独立的regressor**。

多任务损失函数：分类加回归。



交替训练：**分步训练**，第一次训练RPN，再第一次训练Fast-RCNN网络，然后重复一遍。

**为什么要用Anchor?**假设，已经知道图片中有两个objects，首先想到的是，训练一个网络，输出 8 个值：两对元组 xmin, ymin, xmax, ymax分别定义了每个 object 的边界框。这种方法存在基本问题，例如，当图片的尺寸和长宽比不一致时，训练模型来预测会非常复杂。另一个问题是无效预测：预测 xmin​ 和 xmax时，需要保证 xmin​<xmax​。一种更加简单的方法来预测 objects 的边界框就是采用anchor机制，学习相对于参考boxes 的偏移量。 假设参考 box：xcenter,ycenter,width,height，待预测量 Δxcenter, Δycenter, Δwidth, Δheight一般都是很小的值。