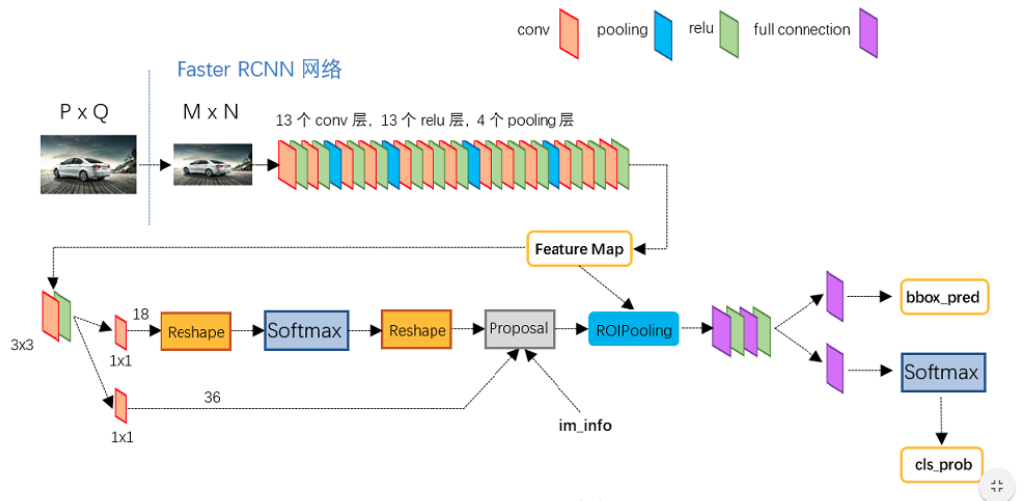


1.首先通过Conv +relu+pooling提取特征，提取的特征用于后面的RPN层和全连接层

2.RPN：用于生成region proposal ，用过softmax判断anchors是属于前景层还是背景

3.ROI Pooling:该层将特征图以及region proposals进行处理，生成proposal feature maps送入后面的全连接层进行类别判断

4.Classification:将proposal feature maps计算的proposal类别，同时再次bounding box regression获得的更精确的检测框



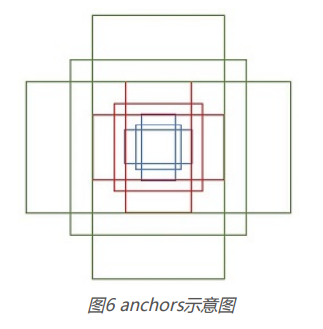
对于任意一副PxQ的图像，先缩放到MxN，然后送入卷积层，包括13个卷积层，13个relu和4个pooling层，RPN首先经过一个3x3卷积，在生成foreground anchors和bounding box regression偏移量，然后计算出proposals，ROI poling将proposal和feature map中提取proposal feature送入全连接层分类。

1.卷积层：所有的卷积层都是：keenel\_szie = 3,pad =1,所有的pooling都是：Kernel\_size =2,stride=2

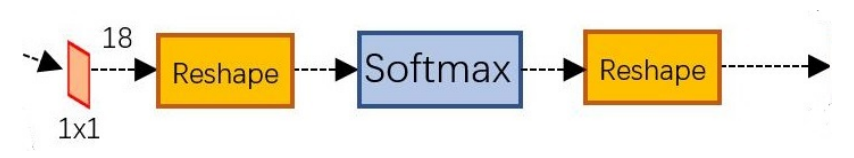
2.RPN(这是Faster Rcnn较快的地方，丢弃了传统的滑窗、ss方法)：

RPN分为两条线，上面一条通过softmax分类将anchors分为foreground和background，下面一条用于计算bounding box regression偏移量获取proposal ,同时去除太小或超出边界的proposals。

Anchors:Faster Rcnn将得到特征图每个点都分配9个长宽比例不同的检测框。



将一股MxN大小的矩阵送入Faster Rcnn后，到RPN变为（M/16）x(N/16),在softmax之前对特征图进行了1x1卷积

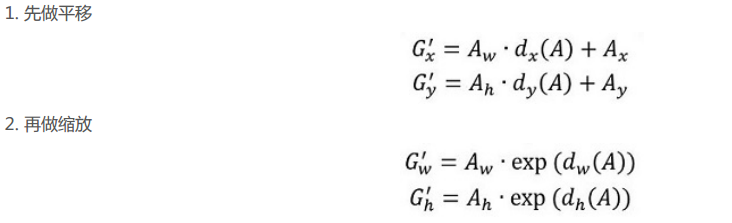


1x1卷积个数为18，因为每个点都要分配9个anchors，每个框都有可能是foreground 和background。2\*9 =18

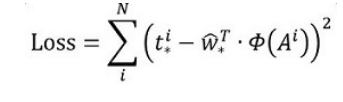
Bounding box regression原理

窗口使用（X,Y,W,H）表示，分别代表窗口的中心点坐标以及宽高，目的要是找到一种映射关系，使得输入的原始anchor经过映射得到一个与真实窗口更近的回归窗口G’

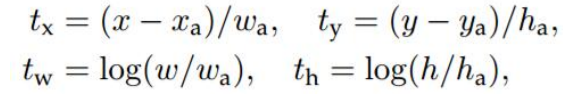
方法：



需要学习的是dx(A), dy(A), dw(A), dh(A)四个变换



W是要学习的参数，φ（A）是对应anchor对应feature map组成的特征向量，\*表示x,y,w,h,也就是每一个变换对应的目标函数，



在bounding box regression时也要进行1x1卷积，核数是36（每个点分配9个anchors,每个anchors有4个回归值）

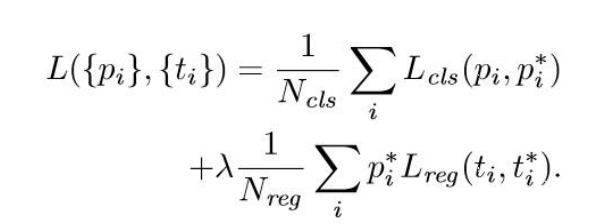
RPN就是：生成anchors->softmax分类器提取前景层anchors->bbox reg回归->Proposal Layer生成proposals

ROI pooling:

有两个输入：1.原始的特征图 2.RPN输出的proposal boxes

因为得到的RPN的框大小不一，所以需要使用ROI Pooling解决这个问题，得到的proposal是对应与MxN尺度的，所以要将其缩放到特征图尺度，之后将proposal水平竖直都分为7份，然后进行max pooling处理，这样出入的大小就都是一样的了。然后送入后续网络，通过全连接和softmax对proposal进行分类，再次对proposals进行bbox reg得到更精确的框。

损失函数：



i表示anchors index，pi表示foreground softmax predict概率，pi\*表示GT，当第i个anchor与GT间的IOU>0.7，认为是anchor是foreground，pi\*=1,反之，IOU<0.3时认为是background，pi\* =0，0.3<IOU<0.7的acnhor不参加训练。T表示predict bounding box，t\*表示GT box。由于Ncls 与Nreg差距过大，所以使用lambda来平衡他们。

