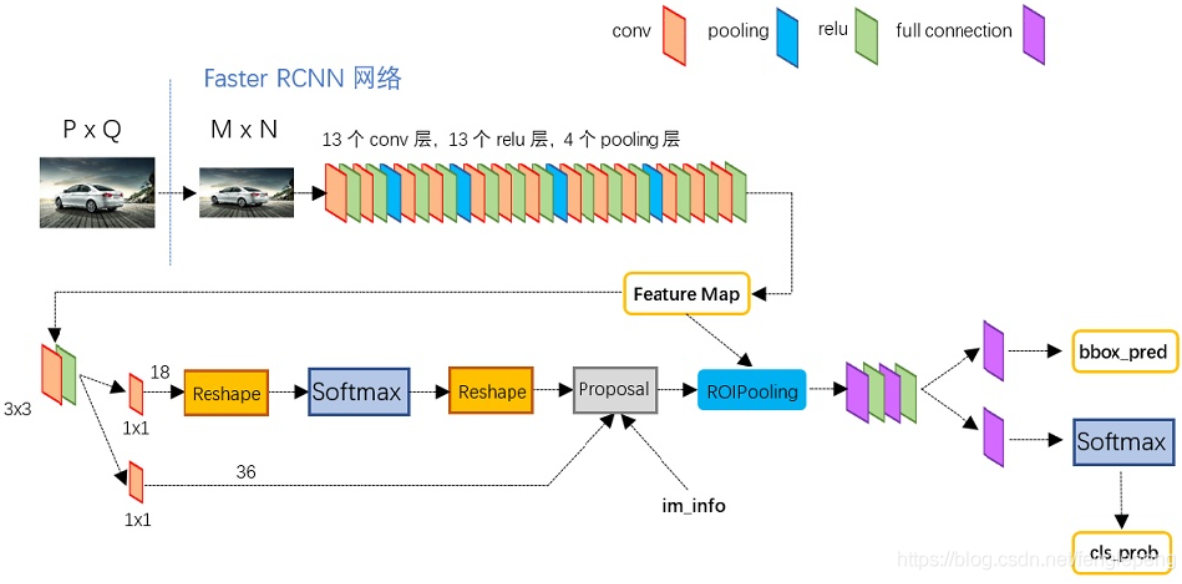
# RPN : 区域候选网络



1 conv feature map 到 intermediate layer 的卷积过程

RPN网络前面是一个提特征的网络，比如VGG,Res等，传给RPN网络的是一个特征图，其实也就是一个 tensor，比如用ZF网络（论文里面用的）

输入特征图：13 ∗ 13 ∗ 256

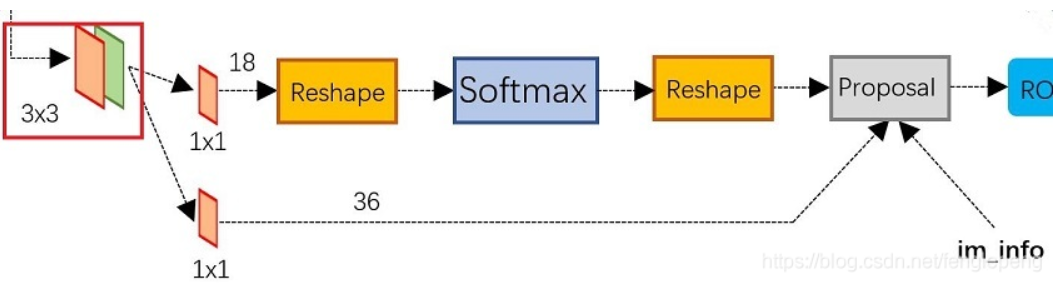
拿到模型的特征，RPN网络首先加了一个 3 ∗ 3 ∗ 256 ∗ 256 步长为 1 的卷积层（可能是为了扩大感受野）

这样就会得到 11 ∗ 11 ∗ 256的输出，前面 11 ∗ 11图形矩阵，其中每一个点在原图中都是一个很大的区域，256表示这个区域的特征。

在这个区域内可能有目标，为了能更能逼近目标，我们需要3种尺度，和3种形状，就是图中的那9种物体框。假设原图中有一个物体，那我们通过在原图上平移框，就总能找到一个颜色框能正好把物体框在里面，而且尺度啊形状最接近。

那么如何平移的呢？在原图上你一个像素的平移没意义啊，难道要重复提取特征？所以平移必须在特征图上平移，因为特征图最后总能映射回原图。11 ∗ 11的特征图区域，在原图中就表示 11 ∗ 11 个大黑框，每个大黑框里面又有9个小颜色框，这样就会产生 11 ∗ 11 ∗ 9 个不同位置，不同尺度，不同形状的物体框，基本足够框出所有物体了。

2 intermediate layer 的256维向量后面对应两条分支



cls layer 分支是目标和背景的二分类（classification），因为k等于9，所以通过 1 × 1 × 256 × 18的卷积核得到 2 × 9 = 18个分数，分别是目标和背景的评分。

reg layer分支。如果候选框是目标区域，就去判断该目标区域的候选框位置在哪，这个时候另一条分支就过 1 × 1 × 256 × 36的卷积得到 4 × 9个值，每个框包含4个值（x，y，w，h），就是9个候选区域对应的框应该偏移的具体位置，Δxcenter,Δycenter,Δwidth,Δheight。如果候选框不是目标区域，就直接将该候选框去除掉，不再进行后续位置信息的判断操作。这里预测的值都是通过模型不断训练得到的。

**3 proposals layer**

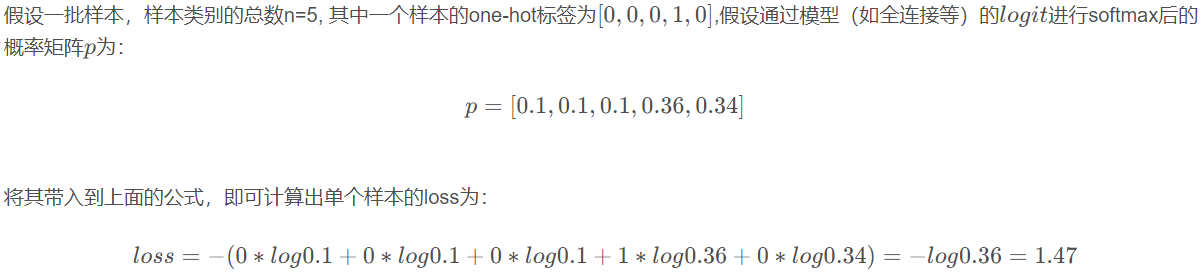
proposals layer 3个输入：一个是分类器结果 foreground softmax scores，一个是 anchor 回归 regression：[ dx( A ) , dy( A ) , dw( A ) , dh( A ) ] ，还有一个是img-info，包含图像缩放的信息。proposal layer 步骤：

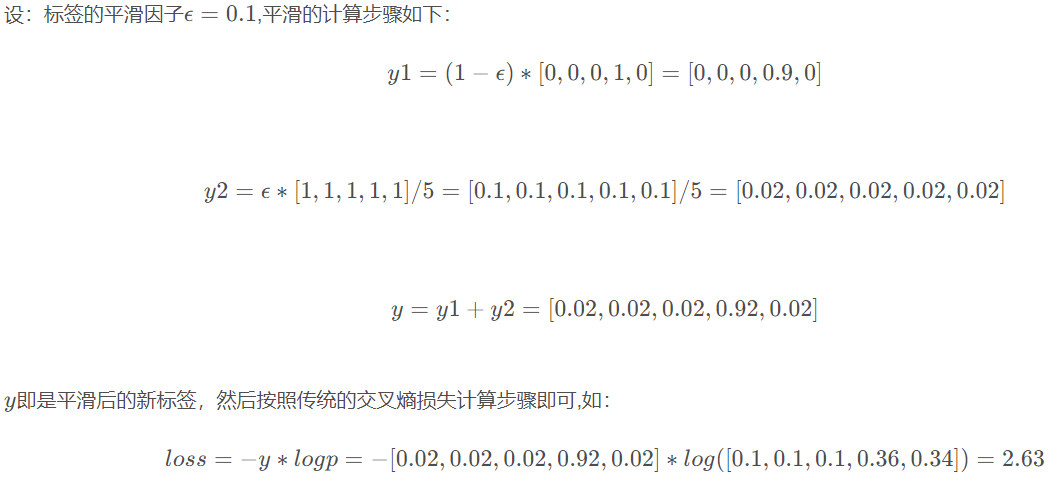
1. 生成 anchors（anchors 的坐标是相对于原图的坐标），然后利用[ dx( A ) , dy( A ) , dw( A ) , dh( A ) ]  对所有的 anchors 做 bbox regression 回归（这里的 anchors 生成和训练时完全一致）
2. 按照输入的 foreground softmax scores 由大到小排序 anchors，提取前pre\_nms\_topN(e.g. 6000) 个anchors，即提取修正位置后的 foreground anchors。
3. 判断fg anchors是否大范围超过边界，剔除严重超出边界fg anchors，剔除非常小（width<threshold or height<threshold）的 foreground anchors
4. 进行 nms，按照 nms 后的 foreground softmax scores 由大到小排序fg anchors，提取前 post\_nms\_topN(e.g. 300) 结果作为 proposal 输出

# 标签平滑

在进行多分类时，很多时候采用one-hot标签进行计算交叉熵损失，而单纯的交叉熵损失时，只考虑到了正确标签的位置的损失，而忽略了错误标签位置的损失。这样导致模型可能会在训练集上拟合的非常好，但由于其错误标签位置的损失没有计算，导致预测的时候，预测错误的概率比较大，也就是常说的过拟合。标签平滑可以在一定程度上防止过拟合

实例：





有上面实例可以看出，带有标签平滑的损失要比传统交叉熵损失要更大。换言之，带有标签平滑的损失要想下降到传统交叉熵损失的程度，就要学习的更好，迫使模型往正确分类的方向走。

# 行人重识别算法性能指标

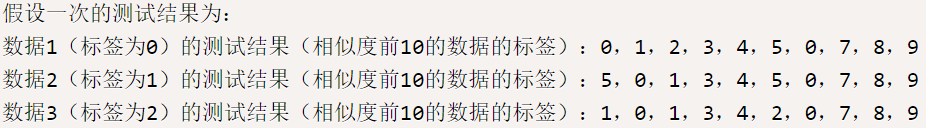
**1. AP(Average Precision平均准确度)**

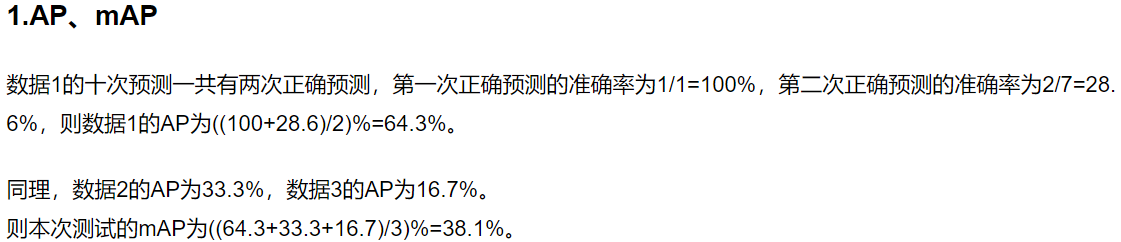
是每一次正确预测的准确率的平均数

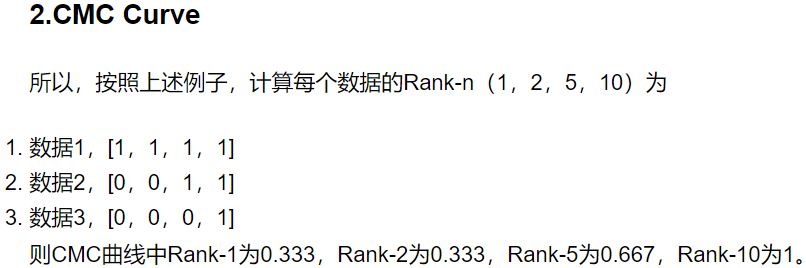
**2. mAP (mean Average Precision平均准确度均值)**

**3. CMC（Cumulative Match Characteristic累计匹配特征）**

横轴代表Rank-n，纵轴代表Accuracy。如果选取Rank-n排序列表为[1，2，5，10]的话，则CMC曲线就是一个由四个点连接而成的曲线。其中每个点的Rank-n的值都为所有测试数据的Rank-n的平均值。Rank-n代表前n个预测是否包含正确预测，取值要么为1要么为0







# 长距离依赖

广义上指句子中具有依存关系的词之间有较多词汇间隔的现象

# 非局部注意力网络（Non-local Attention Network）

主要用于扩大神经网络的感受野，传统卷积核的感受野大小等于其自身的大小，而一般的神经网络里卷积核大小设置的都很小，这导致其感受野只能感知到非常有限的局部区域，非局部注意力网络将感受野扩大到整个矩阵上，使得矩阵 中每个点都能参与运算，这样可以引入全局信息，进而将全局信息传到后面的层，因此非常适合捕获数据中的长距离依赖。

# 表征学习与度量学习

行人重识别表征学习和度量学习的区别是根据损失函数的不同而区分的。

**表征学习**的方法没有直接训练网络的时候考虑图片间的相似度，而把行人重识别任务当做分类问题或者验证问题来看待。

**度量学习**旨在通过网络学习出两张图片的相似度。在行人重识别问题上，表现为同一行人的不同图片间的相似度大于不同行人的不同图片

# 样本

**正样本**： 与[真值](https://so.csdn.net/so/search?q=%E7%9C%9F%E5%80%BC&spm=1001.2101.3001.7020)对应的目标类别来说该样本为正样本。  
**负样本**： 与真值不对应的其他所有目标类别来说该样本为负样本。  
**简单样本**： 预测时与真值标签误差较小的样本。  
**困难样本**： 预测时与真值标签误差较大的样本。