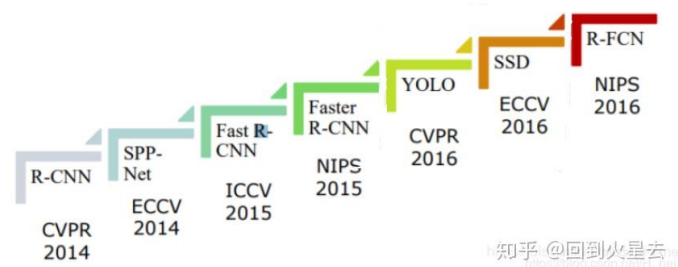


作业三

题目：理解和解释 RCNN、Fast-RCNN、Faster-RCNN



目标检测的模型发展图

R-CNN (Region CNN)

R-CNN 的全称是 Region-CNN (区域卷积神经网络)，是第一个成功将深度学习应用到目标检测上的算法。R-CNN 基于卷积神经网络(CNN)，线性回归，和支持向量机(SVM)等算法，实现目标检测技术。



流程是：

1. 输入图像
2. 利用 selective search 对图像生成 1K~2K 的候选区域 (region proposal)，这个量比传统的算法要少得多。具体一点，选出 region proposal 的方法是运行图像分割算法，对于分割算法跑出来的块，把它作为可能的 region proposal 输出。
3. 提取特征：将 region proposal resize 为统一大小，送进去掉了 softmax 的 CNN，对每个候 region proposal 提取特征
4. 对区域进行分类：对从 CNN output 出来的特征向量送进每一类的 SVM 分类，如果我有十个类别，那么每个 region proposal 要跑 10 个 SVM，得到类别。这里为什么要用 SVM 而不是 softmax，有一种说法是为了解决样本不均衡的问题，另外是早期神经网络还不如现在这样发达，当时 SVM 还是比较领先的分类器。
5. 修正：对 CNN output 的特征向量 (这个特征向量和第 4 步中拿去喂给 SVM 的是一个向量) 做回归 (左上角右下角的四个坐标)，修正 region proposal 的位置。

有几点问题：

1. 可以明显地感受到它的计算量是非常大的，毕竟要对每个候选区域都进行特征计算。
2. 冗余计算太多了，毕竟候选区域高度重叠。
3. 同时又不是端到端的训练，还麻烦。

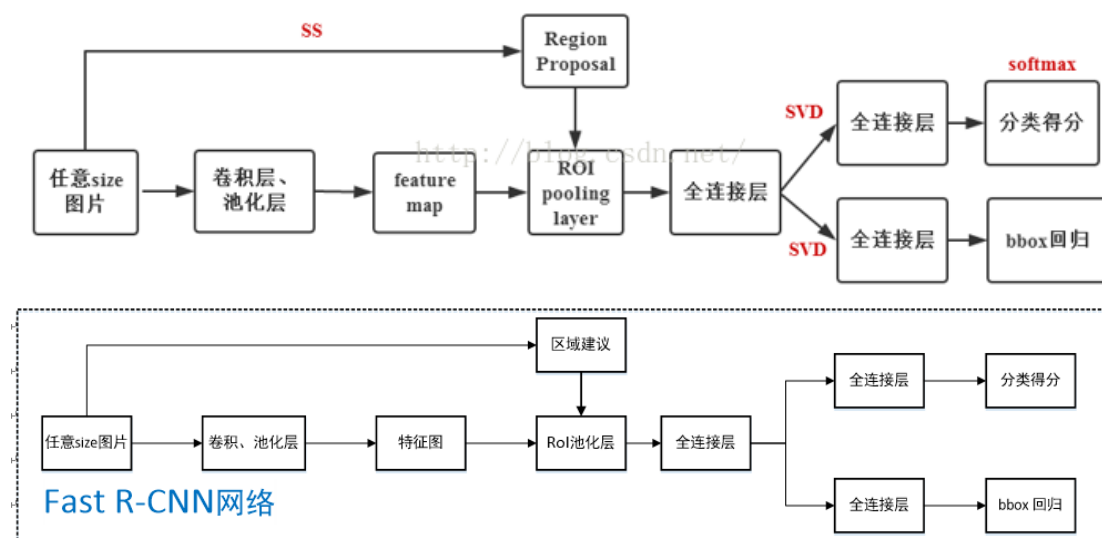
4. 内存占用：需要储存多个 SVM 分类器和 bounding box 回归器
5. 对输入图片的大小有硬性要求

Fast R-CNN (Fast Region CNN)

Fast R-CNN 组合了 classification 和 regression, 做成 single Network, 实现了端到端的训练, 实际上它相对 RCNN 最大的改进是抛弃了多个 SVM 分类器和 bounding box 回归器的做法, 一起输出 bbox 和 label, 很大程度上提升了原始 RCNN 的速度。

它的流程是：

1. 将任意 size 的图片输入 CNN, 得到特征图。在 RCNN 中, 先生成 region proposals 再做卷积, 相当于做了多次卷积, 浪费时间。
2. 对原始图片使用 selective search 算法得到约 2k region proposals (相当于 RCNN 的第一步)
3. 在特征图中找到每一个 region proposals 对应的特征框。在 ROI 池化层中将每个特征框池化到统一大小
4. 统一大小的特征框经过全连接层得到固定大小的特征向量, 分别进行 softmax 分类 (使用 softmax 代替了 RCNN 里面的多个 SVM 分类器) 和 bbox 回归



这里出现了一个新的概念, ROI Polling

ROI Pooling (Region of Interest)

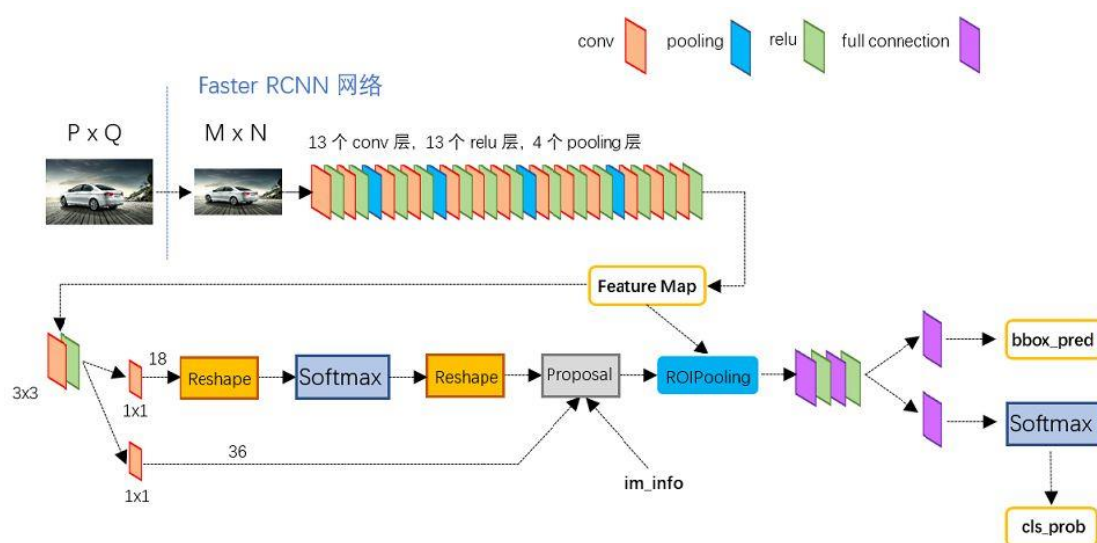
它的输入是特征图, 输出则是大小固定的 $\text{channel} \times H \times W$ 的 vector。ROI Pooling 是将一个个大小不同的 region proposals, 映射成大小固定的 $(W \times H)$ 的矩形框。它的作用是**根据 region proposals 的位置坐标在特征图中将相应区域池化为固定尺寸的特征图**, 以便进行后续的分类和输出回归框操作。它可以加速处理速度。这个 ROI Pooling 在下面的 Faster RCNN 也会出现, 它的作用是类似的, 就是将 region proposals 池化成同样大小的 vector, 便于传入后续分类网络。

Faster RCNN (Faster Region CNN)

在 Fast RCNN 的基础上, Faster RCNN 在性能上又有了进步。Faster RCNN 将特征抽取(feature extraction), proposal 提取, bounding box regression, classification 都整合在了一个网络中, 使得综合性能有较大提高, 在检测速度方面尤为明显。对比起它哥哥 Fast-RCNN, 其实最重要的一点就是使用 RPN 来代替原来使用分割算法生成候选框的方式, 极大的提升了检测框生成速度。总的来说, Faster RCNN 对 Fast RCNN 的改进点在于获得 region proposals 的速度要快很多。

1. **提取特征:** 输入固定大小的图片, 经过卷积层提取特征图 feature maps
2. **生成 region proposals:** 然后经过 Region Proposal Networks(RPN)生成 region proposals。该层通过 softmax 判断 anchors 属于 foreground 或者 background, 再利用 bounding box 回归修正 anchors 获得精确的 proposals (候选区域)。
3. **ROI Pooling:** 该层的输入是 feature maps 和 proposals, 综合这些信息后提取 proposal feature maps
4. **Classification:** 将 Roi pooling 生成的 proposal feature maps 分别传入 softmax 分类和 bounding box regression 获得检测物体类别和检测框最终的精确位置。

具体一点的图长这样:



二、R-CNN、Fast R-CNN、Faster R-CNN三者关系

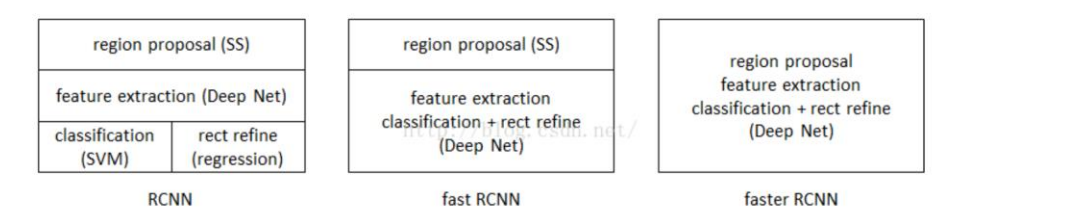


图2 三者关系

表1 三者比较

	使用方法	缺点	改进
R-CNN (Region-based Convolutional Neural Networks)	1、SS提取RP; 2、CNN提取特征; 3、SVM分类; 4、BB盒回归。	1、训练步骤繁琐（微调网络+训练SVM+训练bbox）; 2、训练、测试均速度慢; 3、训练占空间	1、从DPM HSC的34.3%直接提升到了66%（mAP）; 2、引入RP+CNN
Fast R-CNN (Fast Region-based Convolutional Neural Networks)	1、SS提取RP; 2、CNN提取特征; 3、softmax分类; 4、多任务损失函数边框回归。	1、依旧用SS提取RP(耗时2-3s，特征提取耗时0.32s); 2、无法满足实时应用，没有真正实现端到端训练测试; 3、利用了GPU，但是区域建议方法是在CPU上实现的。	1、由66.9%提升到70%; 2、每张图像耗时约为3s。
Faster R-CNN (Fast Region-based Convolutional Neural Networks)	1、RPN提取RP; 2、CNN提取特征; 3、softmax分类; 4、多任务损失函数边框回归。	1、还是无法达到实时检测目标; 2、获取region proposal，再对每个proposal分类计算量还是比较大。	1、提高了检测精度和速度; 2、真正实现端到端的目标检测框架; 3、生成建议框仅需约10ms。