

# 目录

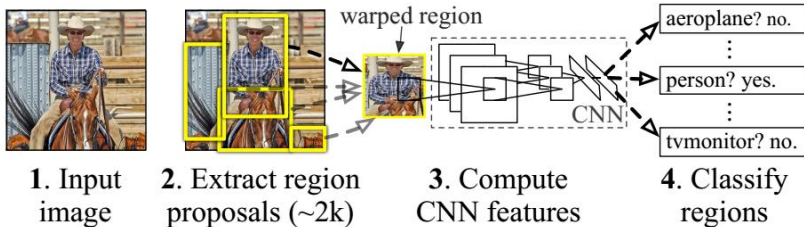
1 R-CNN 2013

2 Fast R-CNN 2015

# 数据流程图

2013 Rich feature hierarchies for accurate object detection and semantic segmentation

## R-CNN: *Regions with CNN features*



输入：图像及 selective search 算法提出的 2000 个 proposal box。

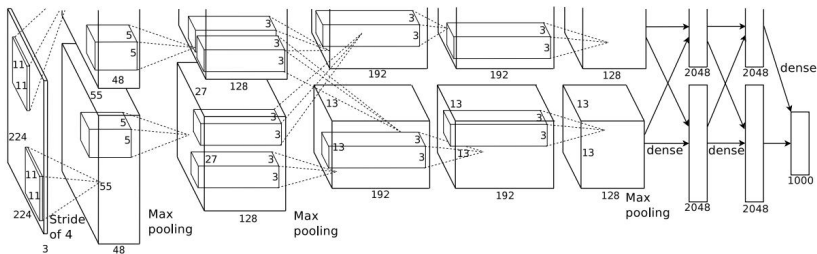
输出：proposal box 包含的目标类别，及修正后的位置坐标。

## proposal box 的形状变换



A 是原图的 proposal box, B 是 proposal box 的最小外接正方形, 短边使用原图填充, 然后放大到  $227 \times 227$ , C 与 B 的相同, 只是填充短边时使用图像均值, D 是直接 warp, 即直接进行放大。每列图像, 上一行是直接对 proposal box 处理, 即 context padding = 0, 下一行的 context padding = 16。实验表明使用 D 方法中的 warp with context padding = 16 时, 效果最好, 提高 3-5 mAP%

# 特征提取网络



采用了和 Alexnet 一样的结构，5 层卷积，3 层全连接的分类网络。在做特征提取时，只计算到第二层全连接输出 4096 维特征向量。

## proposal box 的分类

这里使用 category-specific linear SVM 对上一步中 proposal box 中提取的 4096 维特征向量进行分类。

测试时，一张图像提出 2000 个 proposal box，对应的特征向量写成一个矩阵就是  $2000 \times 4096$  维，乘以 SVM 系数矩阵  $4096 \times N$  ( $N$  指目标类别数目) 得到每一个 proposal box 的分数。

## class-specific bounding box regressor

proposal box 的参数表示:  $P^i = (P_x^i, P_y^i, P_w^i, P_h^i)$

ground truth box 的参数表示:  $G = (G_x^i, G_y^i, G_w^i, G_h^i)$

回归的目标:

$$t_x = (G_x - P_x) / P_w$$

$$t_y = (G_y - P_y) / P_h$$

$$t_w = \log(G_w / P_w) = \log(G_w) - \log(P_w)$$

$$t_h = \log(G_h / P_h) = \log(G_h) - \log(P_h)$$


回归使用的是线性模型, 损失函数是均方误差加上  $L_2$  正则项:

$$loss = \sum_i^N (t_\star^i - \hat{w}_\star^T \phi_5(P^i))^2 + \lambda \|\hat{w}_\star\|^2$$

其中,  $\phi_5(P^i)$  表示 CNN 的 pool 5 层在 proposal box 内提取的特征,  $w_\star$  表示回归要学习的参数。

# IOU

Intersection Over Union: 两个框重叠度的计算

$$\text{IoU} = \frac{\text{Area of Overlap}}{\text{Area of Union}}$$


<http://blog.csdn.net/lanqianhui>

# 训练

- CNN 的训练

先在 ILSVRC2012 图像分类数据集上进行预训练，之后在目标检测数据集 VOC 与 ILSVRC203 上进行微调，微调时把 proposal box 中所有与 ground-truth box 的 IOU 大于 0.5 的框看做正样本，其余的 proposal box 都是负样本。训练时，batch-size 为 128，其中 32 个为正样本，96 个背景样本。

- SVM 的训练样本

正样本只有 ground-truth box 提出的特征，负样本是 proposal box 中与 ground-truth box 的 IOU 小于 0.3 的。

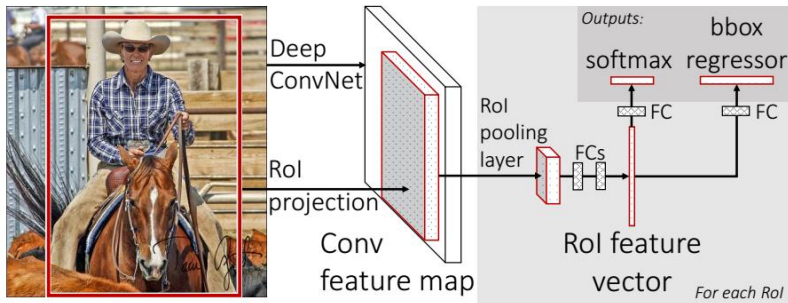
- bounding-box regressor 的训练样本

对于每一个 proposal box，计算它与所有的 ground-truth box 的 IOU，然后选出 IOU 最大的那一个，如果此 IOU 值大于 0.6，那么这个 proposal box 就配对成功，看做一个训练样本，配对不成功的 proposal box 被舍弃。



# 数据流程图

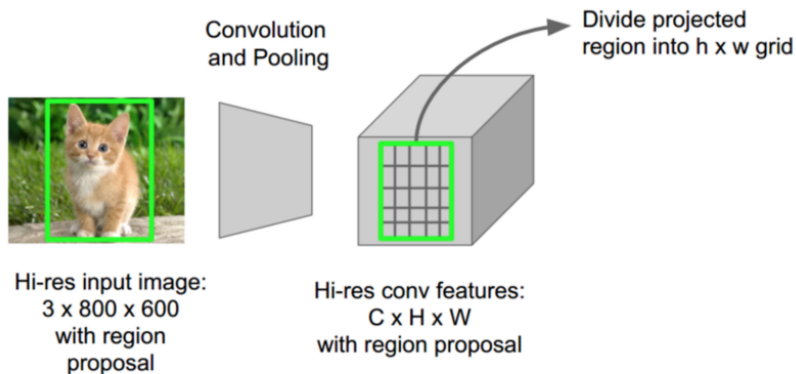
2015 Fast R-CNN



输入：图像及 selective search 算法提出的 2000 个 proposal box。

输出：proposal box 包含的目标类别，及修正后的位置坐标。

# ROI pooling



Roi pooling 使不同大小，不同长宽比的 proposal box 都能提取到固定长度的特征向量。

# Multi-task loss

分类器与坐标回归的损失函数：

$$L(p, u, t^u, v) = L_{cls}(p, u) + \lambda[u \geq 1]L_{loc}(t^u, v)$$

其中， $p$  为预测的分类概率， $u$  真实分类的 label， $t^u$  为预测的坐标偏差，因为坐标回归针对每一类都有一个回归器，所以它是  $u$  的函数， $v$  为真实的坐标偏差 label， $[u \geq 1]$  艾佛森括号，表示  $u \geq 1$  时，输出为 1，否则为 0。

# Multi-task loss

分类器与坐标回归的损失函数：

$$L(p, u, t^u, v) = L_{cls}(p, u) + \lambda[u \geq 1]L_{loc}(t^u, v)$$

分类损失：

$$L_{cls} = -\log p_u$$

坐标回归损失：

$$L_{loc}(t^u, v) = \sum_{i \in \{x, y, w, h\}} \text{smooth}_{L1}(t_i^u - v_i)$$

与 R-CNN 不同，这里使用了  $\text{Smooth}_{L1}$  损失：

$$\text{smooth}_{L1}(x) = \begin{cases} 0.5x^2 & \text{if } |x| < 1 \\ |x| - 0.5 & \text{otherwise} \end{cases} \rightarrow \text{求导} \begin{cases} x \\ \pm 1 \end{cases}$$