

边框回归(Bounding Box Regression)

详解

原创 置顶 南有乔木NTU 2017-08-31 22:38:05 127492 版权

★ 收藏 650

分类专栏: 目标检测 文章标签: 目标检测算法

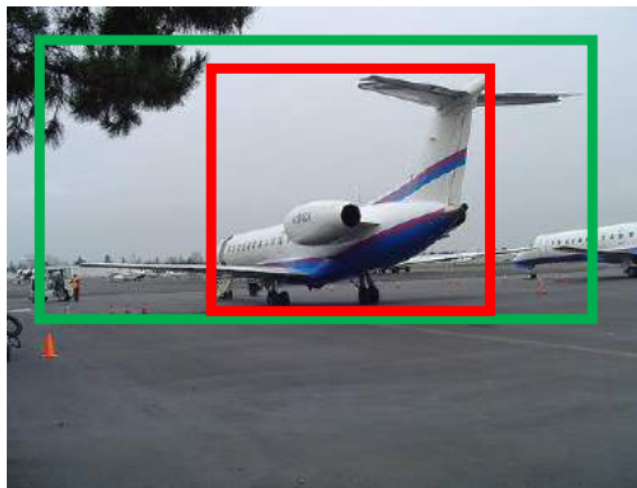
Bounding-Box regression

最近一直看检测有关的Paper, 从rcnn, fast rcnn, faster rcnn, yolo, r-fcn, ssd, 到今年cvpr最新的yolo9000。这些paper中损失函数都包含了边框回归, 除了rcnn详细介绍了, 其他的paper都是一笔带过, 或者直接引用rcnn就把损失函数写出来了。前三条网上解释比较多, 后面的两条我看了很多paper, 才得出这些结论。

- 为什么要边框回归?
- 什么是边框回归?
- 边框回归怎么做的?
- 边框回归为什么宽高, 坐标会设计这种形式?
- 为什么边框回归只能微调, 在离Ground Truth近的时候才能生效?

为什么要边框回归?

这里引用王斌师兄的理解, 如下图所示:



<http://img.csdn.net/zijin0802034>

对于上图, 绿色的框表示Ground Truth, 红色的框为Selective Search提取的Region Proposal。那么即使红色的框被分类器识别为飞机, 但是由于红色的框定位不准($IoU < 0.5$), 那么这张图相当于没有正确的检测出飞机。如果我们能对红色的框进行微调, 使得经过微调后的窗口跟Ground Truth 更接近, 这样岂不是定位会更准确。确实, Bounding-box regression 就是用来微调这个窗口的。

边框回归是什么?

继续借用师兄的理解: 对于窗口一般使用四维向量 (x, y, w, h) 来表示, 分别表示窗口的中心点坐标和宽高。对于

点赞Mark关注该博主, 随时了解TA

回归窗口 \hat{G} 。


$$(\hat{G}_x, \hat{G}_y, \hat{G}_w, \hat{G}_h) \approx (G_x, G_y, G_w, G_h)$$

边框回归怎么做的？

那么经过何种变换才能从图 2 中的窗口 P 变为窗口 \hat{G} 呢？比较简单的思路就是：平移+尺度放缩

1. 先做平移 $(\Delta x, \Delta y)$, $\Delta x = P_w d_x(P)$, $\Delta y = P_h d_y(P)$ 这是R-CNN论文的:

$$\hat{G}_x = P_w d_x(P) + P_x, (1)$$

$$\hat{G}_y = P_h d_y(P) + P_y, (2)$$

2. 然后再做尺度缩放(S_w, S_h),

$S_w = \exp(d_w(P)), S_h = \exp(d_h(P))$, 对应论文中:

$$\hat{G}_w = P_w \exp(d_w(P)), (3)$$

$$\hat{G}_h = P_h \exp(d_h(P)), (4)$$

观察(1)-(4)我们发现，边框回归学习就是

$d_x(P), d_y(P), d_w(P), d_h(P)$ 这四个变换。下一步就是设计算法那得到这四个映射。

线性回归就是给定输入的特征向量 X , 学习一组参数 W , 使得经过线性回归后的值跟真实值 Y (Ground Truth) 非常接近。即 $Y \approx WX$ 。那么 Bounding-box 中我们的输入以及输出分别是什么呢?

Input:

$RegionProposal \rightarrow P = (P_x, P_y, P_w, P_h)$, 这个是什么? 输入就是这四个数值吗? 其实真正的输入是这个窗口对应的 CNN 特征, 也就是 R-CNN 中的 Pool5 feature (特征向量)。(注: 训练阶段输入还包括 Ground Truth, 也就是下边提到的 $t_* = (t_x^*, t_y^*, t_w^*, t_h^*)$)

点赞Mark关注该博主,随时了解TA

Output:

需要进行的平移变换和尺度缩放 $d_x(P), d_y(P), d_w(P), d_h(P)$, 或者说 $\Delta x, \Delta y, S_w, S_h$ 。我们的最终输出不应该是 Ground Truth 吗? 是的, 但是有了这四个变换我们就可以直接得到 Ground Truth, 这里还有个问题, 根据(1)~(4)我们可以知道, P 经过 $d_x(P), d_y(P), d_w(P), d_h(P)$ 得到的并不是真实值 G, 而是预测值 \hat{G} 。的确, 这四个值应该是经过 Ground Truth 和 Proposal 计算得到的真正需要的平移量 (t_x, t_y) 和尺度缩放 (t_w, t_h) 。这也就是 R-CNN 中的(6)~(9):

$$t_x = (G_x - P_x) / P_w, (6)$$

$$t_y = (G_y - P_y) / P_h, (7)$$

$$t_w = \log(G_w / P_w), (8)$$

$$t_h = \log(G_h / P_h), (9)$$

那么目标函数可以表示为 $d_*(P) = w_*^T \Phi_5(P)$, $\Phi_5(P)$ 是输入 Proposal 的特征向量, w_* 是要学习的参数 (*表示 x,y,w,h, 也就是每一个变换对应一个目标函数), $d_*(P)$ 是得到的预测值。我们要让预测值跟真实值 $t_* = (t_x, t_y, t_w, t_h)$ 差距最小, 得到损失函数为:

$$Loss = \sum_i^N (t_*^i - \hat{w}_*^T \phi_5(P^i))^2$$

函数优化目标为:

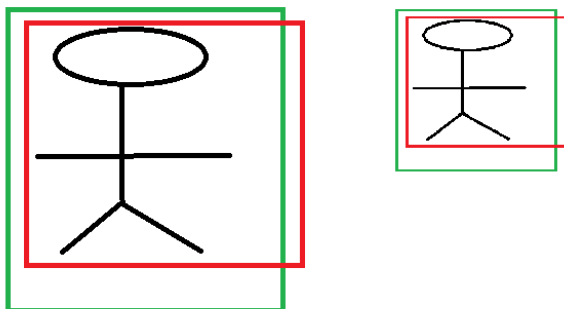
$$W_* = \operatorname{argmin}_{w_*} \sum_i^N (t_*^i - \hat{w}_*^T \phi_5(P^i))^2 + \lambda ||\hat{w}_*||^2$$

利用梯度下降法或者最小二乘法就可以得到 w_* 。

为什么宽高尺度会设计这种形式?

这边我重点解释一下为什么设计的 t_x, t_y 为什么除以宽高, 为什么 t_w, t_h 会有 log 形式!!!

首先CNN具有尺度不变性, 以图3为例:



<http://blog.csdn.net/zi jin0802034>

x,y 坐标除以宽高

点赞Mark关注该博主, 随时了解TA

点赞384

评论108

分享

收藏650

打赏

举报

关

上图的两个人具有不同的尺度，因为他都是人，我们得到的特征相同。
假设我们得到的特征为 ϕ_1, ϕ_2 ，那么一个完好的特征应该具备 $\phi_1 = \phi_2$ 。
ok，如果我们直接学习坐标差值，以x坐标为例， x_i, p_i 分别代表第i个框的x坐标，学习到的映射为 f ， $f(\phi_1) = x_1 - p_1$ ，同理
 $f(\phi_2) = x_2 - p_2$ 。从上图显而易见， $x_1 - p_1 \neq x_2 - p_2$ 。也就是说同一个x对应多个y，这明显不满足函数的定义。边框回归学习的是回归函数，然而你的目标却不满足函数定义，当然学习不到什么。

宽高坐标Log形式

我们想要得到一个放缩的尺度，也就是说这里限制尺度必须大于0。我们学习的 t_w, t_h 怎么保证满足大于0呢？直观的想法就是EXP函数，如公式(3), (4)所示，那么反过来推导就是Log函数的来源了。

为什么IoU较大，认为是线性变换？

当输入的 Proposal 与 Ground Truth 相差较小时(RCNN 设置的是IoU>0.6)，可以认为这种变换是一种线性变换，那么我们就可以用线性回归来建模对窗口进行微调，否则会导致训练的回归模型不 work（当Proposal跟 GT 离得较远，就是复杂的非线性问题了，此时用线性回归建模显然不合理）。这里我来解释：

Log函数明显不满足线性函数，但是为什么当Proposal 和Ground Truth 相差较小的时候，就可以认为是一种线性变换呢？大家还记得这个公式不？参看高数1。

$$\lim_{x \rightarrow 0} \log(1+x) = x$$

现在回过来看公式(8)：

$$t_w = \log(G_w/P_w) = \log\left(\frac{G_w + P_w - P_w}{P_w}\right) = \log\left(1 + \frac{G_w - P_w}{P_w}\right)$$

当且仅当 $G_w - P_w = 0$ 的时候，才会是线性函数，也就是宽度和高度必须近似相等。

对于IoU大于指定值这块，我并不认同作者的说法。我个人理解，只保证Region Proposal和Ground Truth的宽高相差不多就能满足回归条件。x,y位置到没有太多限制，这点我们从YOLOv2可以看出，原始的边框回归其实x, y的位置相对来说对很大的。这也是YOLOv2的改进地方。详情请参考我的博客YOLOv2。

总结

里面很多都是参考师兄在caffe社区的[回答](#)，本来不想重复打字的，但是美观的强迫症，让我手动把latex公式巴拉巴拉敲完，当然也为了让大家看起来顺眼。后面还有一些公式那块资料很少，是我在阅读paper+个人总结，不对的地方还请大家留言多多指正。

RCNN:[Bounding-Box \(BB\) regression](#) Bixiwen_liu的博客 1万+
本博客主要介绍RCNN中的[Bounding-box](#)的回归问题，这个是RCNN定准确定位...

【目标检测】[边框回归\(Bounding Box Reg...](#) zhw864680355的博客 3367
转载来源: <https://blog.csdn.net/zijin0802034/article/details/77685438> [Bounding...](#)



优质评论可以帮助作者获得更高权重

评论



戏水献东柳：给作者一个赞，看完确实对box regression有了更深的认识。对于“边框回归为什么宽高，坐标会设计这种形式？”我的理解：第一，(6, 7)两式对平移量除以宽和高的处理是为了做尺度归一化，因为大的box可能绝对偏移量会比小的box大，除以宽和高消除这种影响；第二，(8, 9)两式比值取对数是线性回归中的常见操作（参考对数线性回归），目的也是在一定程度上将数值的绝对变化转换为相对的变化。 1年前 回复 ●●

27

点赞Mark关注该博主，随时了解TA

TWSF 回复 龚大龙：我的理解是R

点赞384

评论108

分享

收藏650

打赏

举报

关