[CV] Anchor-Free Target Assignment 高斯核半径

简介

有两篇经典 Anchor-Free 算法，CornerNet 和 CenterNet，Target Assignment 过程都是用了位置和包围核尺度生成自适应高斯分布，即高斯核半径是通过包围核尺度计算得到的。这里作者论文都是一带而过，然而如何计算还是有很多学问在里面。Github也有对于他们的讨论。我参考了以下资料：

> 1. *知乎：说点 Cornernet/Centernet 代码里面 GT heatmap 里面如何应用高斯散射核*
> 2. *[Github issue: How to compute the gaussian_radius?Who can tell me the formula about it?Thank you! 这里给出了另一种功能更为精细的方式*
> 3. *Gi thub issue: Bugs in `gaussian_radius` (这里发现公式用错了，但是对实验结果影响不大)*

代码

代码对应链接在这 sample/utils，计算了三个可能的半径，选取其中最小的一个。

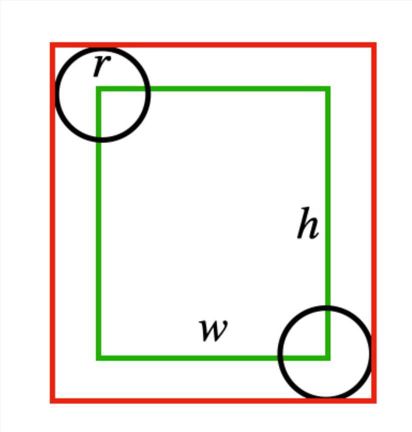
```python
def gaussian_radius(det_size, min_overlap):
    height, width = det_size

    a1 = 1
    b1 = (height + width)
    c1 = width * height * (1 - min_overlap) / (1 + min_overlap)
    sq1 = np.sqrt(b1 ** 2 - 4 * a1 * c1)
    r1 = (b1 - sq1) / (2 * a1)

    a2 = 4
    b2 = 2 * (height + width)
    c2 = (1 - min_overlap) * width * height
    sq2 = np.sqrt(b2 ** 2 - 4 * a2 * c2)
    r2 = (b2 - sq2) / (2 * a2)

    a3 = 4 * min_overlap
    b3 = -2 * min_overlap * (height + width)
```

```
    c3 = (min_overlap - 1) * width * height
    sq3 = np.sqrt(b3 ** 2 - 4 * a3 * c3)
    r3 = (b3 + sq3) / (2 * a3)
    return min(r1, r2, r3)
```

原理解释

CornerNet高斯半径确定方式如下，在 bbox top-left 和 right-down 两个位置以高斯半径绘制圆，检测结果的 top-left 和 right-down 两个结果在改半径内，并且与原始 bbox IoU大于一定阈值，就认为是有效的结果。可以明确，只需要IoU阈值和原始bbox的尺度，就可以推理出高斯核半径。

情况1



先考虑上图的情况：

$$\frac{w \cdot h}{(w+2r)(h+2r)} > IoU$$

化简可以得到：

$$4 \cdot IoU \cdot r^2 + 2 \cdot IoU \cdot (w+h) \cdot r + (IoU - 1) \cdot w \cdot h < 0$$

令：

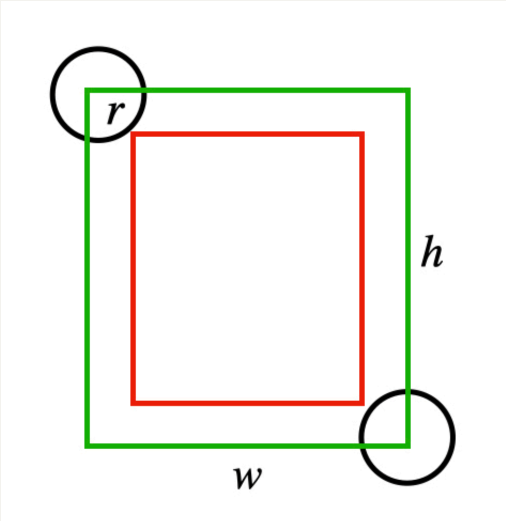
$$a = 4$$
$$b = 2 \cdot IoU \cdot (w+h)$$
$$c = (IoU - 1) \cdot w \cdot h$$

由于高斯半径需要大于0， r最大可以取到的值为

$$r = \frac{-b + \sqrt{b^2 - 4ac}}{2a}$$

这对应个代码里面的`r3`

情况2



同理，如果预测的框在原始bbox内部，可以写出第二个公式

$$\frac{(w-2r)(h-2r)}{wh} > IoU$$

$$4r^2 - 2(w+h)r + (1-IoU)wh > 0$$

令：

$$\begin{gathered} a = 4 \\ b = -2(w+h) \\ c = (1-IoU)wh \end{gathered}$$

同理选择r的最大情况，同时要满足在预测框在原始框内部的条件，有

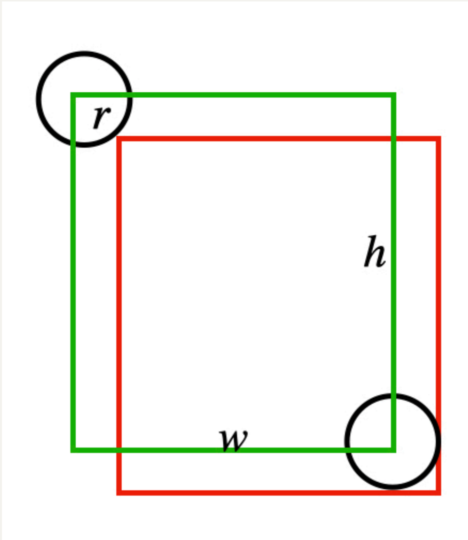
$$2r < \min(h, w)$$

最终确定

$$r = \frac{-b - \sqrt{b^2 - 4ac}}{2a}$$

对应代码里面的`r2`

情况3



如图，对应公式为

$$\frac{(w-r)(h-r)}{(w+r)(h+r)-2r^2} > IoU$$

化简得到:

$$r^2-(w+h)r+\frac{1-IoU}{1+IoU}wh > 0$$

同情况2，这里对应的是代码的`r1`