FCOS: Fully Convolutional One-Stage Object Detection

1. 介绍

1.1. 传统锚框方式

在以往的目标检测中，预定义锚框一直被认为是成功的关键。尽管像是Faster R-CNN，SSD，YOLOv2,v3获得了巨大成功，但是预定义锚框依旧有很大缺陷。

- 预测的性能对锚框的尺寸，长宽比，数量很敏感。因此对超参数的微调要求很高。
- 预定义锚框限制了泛化能力，遇到形状变化较大的物体会遇到困难，特别是小物体。因此每次进行新检测任务需要重新设计。
- 为了提高召回率，设置较为密集的锚框导致负样本过多，正负样本不平衡导致训练低效。
- 锚框会导致复杂计算：IoU。

1.2. 借鉴FCN的检测探索

借鉴FCN这样的像素级实例分割，以**每像素预测方式**或许可以解决无锚框的目标检测问题。并且取得比锚框更好的性能。

在以往已经有开展此项研究的模型，比如DenseBox。这些基于FCN框架的模型预测一个4D向量（描述bounding box的位置相对偏移量）以及一个类别。

为了处理不同尺寸的bounding box，DenseBox只能在图像金字塔上面进行检测，这与FCN的一次性计算所有卷积的观念相违背。

更重要的是，这些模型只能应用于相对特殊的检测领域，比如场景文本检测或人脸识别。因为当出现目标物体重叠的模棱两可情况，像素不知回归到哪个bounding box。

1.3. 我们的研究

我们的研究特点：

- 使用FPN可以很大程度上消除目标物体重叠的模棱两可情况，并取得跟传统锚框一样的效果。
- 发明的center-ness分支（只有一层）通过预测一个像素点在其对应的bounding box中心的偏差来降低权重的方式，可以抑制低质量的bounding box，并将结果合并到NMS（Non-Maximum Suppression）中。

我们的研究优势：

- 检测跟其他CV任务统一，如语义分割。
- 没有锚框，减少了大量调参，简化了训练。
- 避免了计算IoU，使得训练加快，内存占用变少。
- FCOS可作为two-stage中的RPNs，并取得比锚框更好的性能，或称为下一个评定的标准。
- 因为检测器的方法兼容性，所以可以拓展到别的研究中，比如实例分割。

2. 相关工作

2.1. Anchor-based 检测器

基于锚框的检测器继承了传统滑动窗口和基于proposals的检测器思想，如Faster R-CNN。利用了CNN的特征图，避免了重复的运算，大大加快了检测过程。

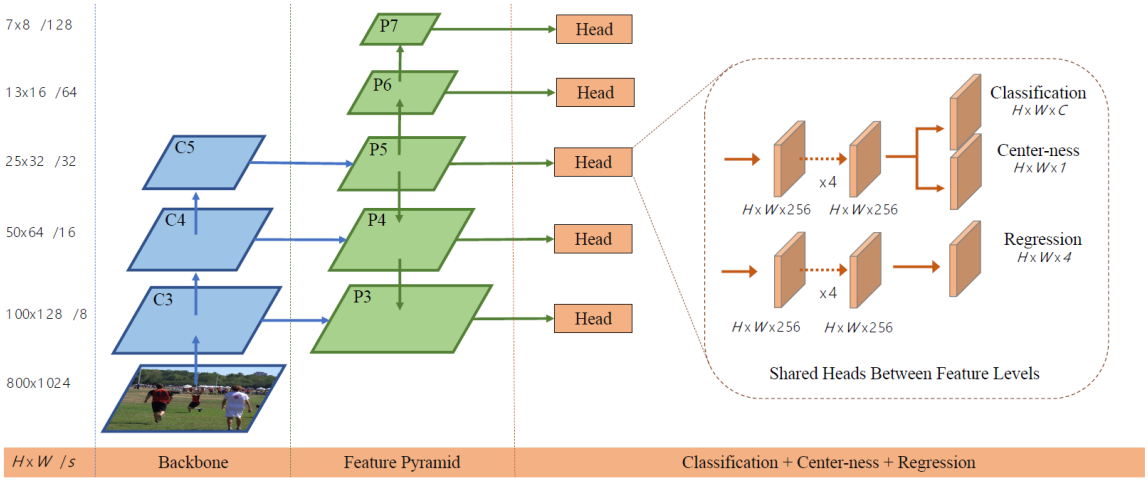
2.2. Anchor-free 检测器

代表为YOLOv1,2。

3. 我们的方法

- 以per-pixel预测的方式重新表述物体检测。
- multi-level 预测提高召回率。
- 解决由重叠的bounding box导致的模棱两可的状况。
- "center-ness"分支。有助于一直低质量的bounding box，并且大幅度提高整体性能。

结构图：



3.1. 定义与组件

设$F_i \in \mathbb{R}^{H \times W \times C}$作为在第$i$层，backbone为CNN（比如ResNet50）的特征图，并且s是在此特征图之前的总步长(stride)。

一个输入图像的ground-truth box被定义为$\{B_i\}$，定义如下：

$$B_i = (x_0^i,\, y_0^i,\, x_1^i,\, y_1^i,\, c^i) \in \mathbb{R}^4 \times \{1,\, 2 \ldots C\}$$

其中，$(x_0^i,\, y_0^i,\, x_1^i,\, y_1^i)$是ground-truth bounding box的左上角和右下角坐标，c^i是ground-truth bounding box中的物体所属的类。C是类的总数量，对于MS-COCO数据集来说$C = 80$.

对于特征图F_i每个位置$(x,\, y)$来说，将它们映射回到原图位置的公式是：$(\lfloor \frac{s}{2} \rfloor + xs, (\lfloor \frac{s}{2} \rfloor + ys)$

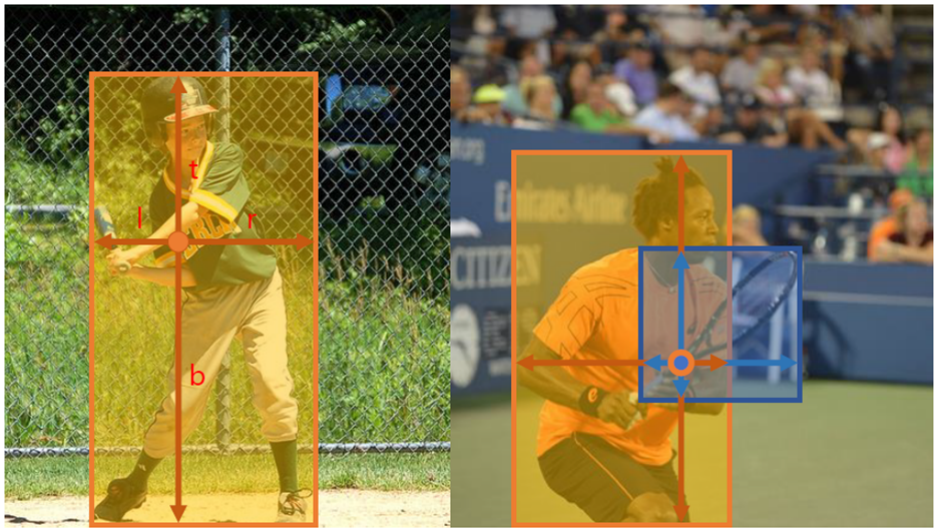
与anchor_based模型不同的是，**FCOS的训练样本是视为（多个）anchor boxes的中心位置**，并对基于这些位置的anchor box进行回归。而不是anchor_based的anchor box作为训练样本。

> **正、负、模糊样本区分：**
>
> - 正样本：特征图的位置$(x,\, y)$映射回原图中（公式如上）落入ground-truth box里面并且预测的类别正确，即位置预测类别c^*等于bounding box的类别c。
> - 负样本：除了正样本以外全是负样本。
> - 模糊样本：特征图的位置$(x,\, y)$映射回原图中（公式如上）落入**多个**ground-truth box里面。

优势：相比于anchor-based，FCOS解决了正负样本不平衡的问题。因为基于anchor-based中每个特征都锚框的方式增加了很多负样本。

如果位置$(x,\, y)$与一个bounding box B_i 相关联，则该位置的训练回归目标为left，right，top，bottom：

$$l^* = x - x_0^{(i)}, \quad t^* = y - y_0^{(i)}$$
$$r^* = x_1^{(i)} - x, \quad b^* = y_1^{(i)} - y$$



FCOS可以利用尽可能多的前景样本进行训练。

> **实现细节：**[2]
>
> 1. 使用Resnet作为backbone（r50，r101，ResNeXt）
> 2. 卷积层为卷积+GN+RuLU，Group Normalization而非Batch Normal
> 3. 使用FPN作为neck
> 4. head权重共享，head预测三个分支：
>   - Classification
>   - Center-ness
>   - Regression

3.2. 损失函数

计算Loss的整体流程大概如下：[2]

FCOS的前向传播，得到三个分支的预测值：Classification; Center-ness; Regression

1. 根据FPN的每个特征图，得到training samples

   training samples是指在所有level的FPN特征图上的坐标点
2. 对training samples采样，确定哪些是正样本和负样本
3. 对正样本计算目标center-ness
4. 最后计算每个部分的loss
   - Classification，对应loss为RetinaNet提出的FocalLoss
   - Center-ness，对应loss为BCELoss二分类交叉熵（因其范围在0到1之间）
   - Regression，对应loss为IoULoss

Loss计算公式：

$$
\begin{aligned}
L(\{\boldsymbol{p}_{x,y}\},\{\boldsymbol{t}_{x,y}\}) = & \frac{1}{N_{\text{pos}}} \sum_{x,y} L_{\text{cls}}(\boldsymbol{p}_{x,y}, c^*_{x,y}) \\
& + \frac{\lambda}{N_{\text{pos}}} \sum_{x,y} \mathbb{1}_{\{c^*_{x,y}>0\}} L_{\text{reg}}(\boldsymbol{t}_{x,y}, \boldsymbol{t}^*_{x,y}),
\end{aligned} \tag{2}
$$

N_{pos}指所有正样本的数量。

3.3. 推断

通过网络前向传播，可以得到特征图F_i上每个位置的分类scores$p_{x,y}$和回归预测$t_{x,y}$。我们选定$p_{x,y} > 0.05$的位置作为正样本，并预测bounding box。

3.4. 网络输出

关于输出，网络的最后一层预测的是一个80D的分类标签向量p（相对于COCO数据集来说），以及一个4D的向量$t = (l, t, r, b)$为bounding box坐标。

训练的分类器是C个二元分类器，代表C个类别。

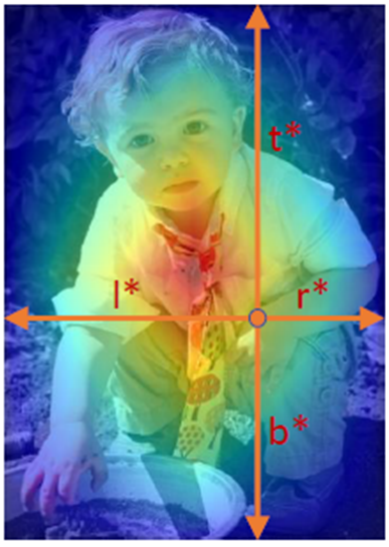
3.5. FPN多级预测

预测的bounding box由$max(l^*, t^*, r^*, b^*) > m_i$或者$max(l^*, t^*, r^*, b^*) < m_{i-1}$判断是否该在$i$层特征图上检测。

适用于重叠的gt box尺寸差距较大的，如果进行FPN分开的特征图以后依旧有重叠的ground-truth box，则位置归于较小的ground-truth box中。

head共享参数，但是由于不同特征级需要回归到不同的尺寸范围，因此head不同。

3.6. center-ness



利用FPN结构进行多级预测后，依旧跟anchor_based存在差距。这是因为远离物体中心的位置产生了很多低质量的bounding box，因为正样本中的有些位置点距离中心点较远（如上图所示）。因此引入了center-ness的概念，计算bounding box相对于ground-truth box的中心度，即从位置到该位置负责的对象中心之间的标准化距离。

center-ness的目标定义如下：

$$center^* = \sqrt{\frac{min(l^*, r^*)}{max(l^*, r^*)} \times \frac{min(t^*, b^*)}{max(t^*, b^*)}}$$

因为centerness范围是0到1，因此采用二交叉熵损失函数（BCE）来训练。由此，这些低质量bounding box可以被最后的非最大抑制（NMS）过程过滤掉，由此提高检测性能。

> 根据20年update：
>
> 代替centerness的还有一个方法为指利用gound-truth box的中心的一部分作为正样本，代价是多出一个超参数。

4. 实验结果

Method	C_5/P_5	w/ GN	nms thr.	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L	AR_1	AR_{10}	AR_{100}
RetinaNet	C_5		.50	35.9	56.0	38.2	20.0	39.8	47.4	31.0	49.4	52.5
FCOS	C_5		.50	36.3	54.8	38.7	20.5	39.8	47.8	31.5	50.6	53.5
FCOS	P_5		.50	36.4	54.9	38.8	19.7	39.7	48.8	31.4	50.6	53.4
FCOS	P_5		.60	36.5	54.5	39.2	19.8	40.0	48.9	31.3	51.2	54.5
FCOS	P_5	✓	.60	37.1	55.9	39.8	21.3	41.0	47.8	31.4	51.4	54.9
Improvements												
+ ctr. on reg.	P_5	✓	.60	37.4	56.1	40.3	21.8	41.2	48.8	31.5	51.7	55.2
+ ctr. sampling [1]	P_5	✓	.60	38.1	56.7	**41.4**	**22.6**	41.6	**50.4**	32.1	52.8	56.3
+ GIoU [1]	P_5	✓	.60	38.3	57.1	41.0	21.9	42.4	49.5	32.0	52.9	56.5
+ Normalization	P_5	✓	.60	**38.6**	**57.4**	**41.4**	22.3	**42.5**	49.8	**32.3**	**53.4**	**57.1**

5. Reference

[1] FCOS:一阶全卷积目标检测

[2] FCOS中的损失函数实现细节