layout: post
title: "yolo系列文章之yolov3详解"
date: 2020-07-09
description: "目标检测"

tag: 目标检测

论文地址：YOLOv3: An Incremental Improvement

yolo官网：https://pjreddie.com/yolo/

主要介绍

YOLOv3在YOLO上做了一些更新，做了一些小的改动，网络性能变得更好，网络整体比YOLOV2大了一些，但是算法的精度更高，速度也没有很大的下降。在320×320 YOLOv3运行22.2ms，28.2 mAP，像SSD一样准确，但速度快三倍，在使用AP0.5评估时，它的速度是RetainNet的3.8倍，但是精度相当。

算法介绍

Bounding box prediction

沿用YOLO9000使用维度聚类的方式生成特定的anchor boxes来预测bbox，网络为每个bbox输出四个坐标，tx，ty，tw，th。如果对应的cell对于图像的左上角偏移cx，cy。cell的宽高分别为pw，ph，那么对应的预测边界框就是：

$$
\begin{aligned}
b_x &= \sigma(t_x) + c_x \\
b_y &= \sigma(t_y) + c_y \\
b_w &= p_w e^{t_w} \\
b_h &= p_h e^{t_h}
\end{aligned}
$$

在训练的时候采用平方误差和损失，如果ground truth对应的值为t*，预测的值为t1，那么梯度就是 t*-t1,t*利用上边的公式可以很容易的计算得到。

YOLOv3使用逻辑回归为每个bbox预测一个物体得分，如果一个先验框与ground truth的IOU超过某个阈值，那么这个bbox的得分就为1，如果这个先验框不是最好的，但是IOU超过某个阈值，那么就忽视这个预测值，阈值为0.5，我们的系统只为每个ground truth对象分配一个边界框，如果先验框没有匹配到ground truth。

Class Prediction

每个bbox使用多标签分类来预测其可能包含的标签类别，使用独立的逻辑分类器，不使用softmax，因为采用softmax给数据集强加了一个默认的约束，即所有的标签是完全互斥的。而在open image dataset中很多标签并不是完全互斥的，有很多重叠的标签，例如女性和人物。在训练的过程中采用二元交叉熵损失来进行类别的预测。

YOLOv3在三个不同尺度上预测框（boxes）。使用类似于特征金字塔网络的思想来提取这些尺度的特征。在基本特征提取器中，添加了几个卷积层。其中最后一个预测了3-d张量编码边界框，对象和类别预测。在COCO实验中，我们预测每个尺度的3个框，所以对于4个边界框偏移量，1个目标性预测和80个类别预测，张量为N×N×[3 *（4 + 1 + 80）]。

然后，从之前的两层中取得特征图（feature map），并将其上采样2倍。我们还从网络中的先前的层中获取特征图，并使用element-wise addition将其与我们的上采样特征进行合并。这种方法使我们能够从早期特征映射中的上采样特征和更细粒度的信息中获得更有意义的语义信息。然后，我们再添加几个卷积层来处理这个组合的特征图，并最终预测出一个相似的张量，虽然现在是两倍的大小。

我们再次执行相同的设计来预测最终尺度的方框。因此，我们对第三种尺度的预测将从所有先前的计算中获益，并从早期的网络中获得细粒度的特征。

仍然使用k-means聚类来确定我们的边界框的先验。只是选择了9个聚类（clusters）和3个尺度（scales），然后在整个尺度上均匀分割聚类。在COCO数据集上，9个聚类是：（10×13）；（16×30）；（33×23）；（30×61）；（62×45）；（59×119）；（116×90）；（156×198）；（373×326）。

Feature Extractor

特征提取器采用新的DarkNet53，是YOLOv2中的DarkNet19与残差链接混合的方法，使用连续的3x3和1x1的卷积层实现，加入了残差链接，一共有53层。

新网络比DarkNet19枪弹你很多，并且比ResNet152更加有效。Darknet-53比ResNet-101更好，速度更快1：5倍。 Darknet-53与ResNet-152具有相似的性能，速度提高2倍。

train

YOLOv3训练完整的图像，没有困难样本挖掘或者类似的其他方法。我们使用多尺度训练，大量的data augmentation，batch normalization，以及所有标准的东西。我们使用Darknet神经网络框架进行训练和测试。

一些不成功的尝试

1. **预测锚框的偏移(Anchor box x, y offset predictions)**，擦汗给你是使用正常的anchor box预测机制，使用线性激活函数来预测x，y的offset为bbox宽高的倍数，降低模型的稳定性，降低性能
2. **使用线性xy预测而不是逻辑预测（Linear x, y predictions instead of logistic）**，尝试使用线性激活来直接预测x，y offeset 而不是逻辑激活。这导致mAP下降了几个点。
3. **Focal loss**，使用focal loss。它使得mAp降低了2个点。YOLOv3对focal loss解决的问题可能已经很强大，因为它具有单独的对象预测和条件类别预测。
4. **Dual IOU thresholds and truth assignment**，Faster R-CNN在训练期间使用两个IOU阈值。如果一个预测与ground truth重叠达到0.7，将它看作是一个正样本，如果达到0.3-0.7，忽略它，如果小于0.3，这是一个负样本。尝试了类似的策略，但无法取得好成绩。

总结

- 使用金字塔网络（类似于多尺度的信息）
- 用逻辑回归替代softmax作为分类器
- Darknet-53作为backbone