基于SSD的检测

目标检测的基本概念

在基本的目标检测任务中，有多种进行检测的实现方式，一种传统的方式是使用滑窗，使用**不同大小、不同长宽比**的候选框在整幅图像上进行穷尽式的滑窗，然后提取窗口内的特征后送入分类器进行识别，判断其中是否包含需要检测的目标，但这一方法往往要求较高的计算能力，且其需要处理的图像往往较多，因此效率较低但准确率较高，被用于`RCNN`和`Fast RCNN`等模型中。

为了提升检测的速度，可以使用Anchor的机制进行模型的学习。

1.边界框

通常使用边界框（bounding box）来描述目标位置。边界框是一个矩形框，可以由矩形左上角的x和y轴坐标与右下角的x和y轴坐标确定。

目标检测算法通常会在输入图像中采样大量的区域，然后判断这些区域中是否包含我们感兴趣的目标，并调整区域边缘从而更准确地预测目标的真实边界框

2.锚点

一言以蔽之，锚框**就是在图像上预设好的不同大小，不同长宽比的参照框。**

实际应用中，以图像的每个像素为中心生成不同形状的锚框，可以通过预设长宽比等方式降低生成的锚框的数目。

3.IoU

需要某一种机制用于判断锚框和真实边界框之间的相似度，在这里使用了$Jaccard$系数：

$$J(A,B)=\frac{|A\cap B|}{|A\cup B|}$$

当衡量两个边界框的相似度时，我们通常将$Jaccard$系数称为交并比(intersection over union)

4.训练

在训练集中，我们将每个锚框视为一个训练样本。为了训练目标检测模型，我们需要为每个锚框标注两类标签：一是锚框所含目标的类别，简称类别；二是真实边界框相对锚框的偏移量，简称偏移量（offset）。在目标检测时，我们首先生成多个锚框，然后为每个锚框预测类别以及偏移量，接着根据预测的偏移量调整锚框位置从而得到预测边界框，最后筛选需要输出的预测边界框。

因此，在这一过程中，涉及到了**如何为锚框分配与其相似的真实边界框**这一问题。

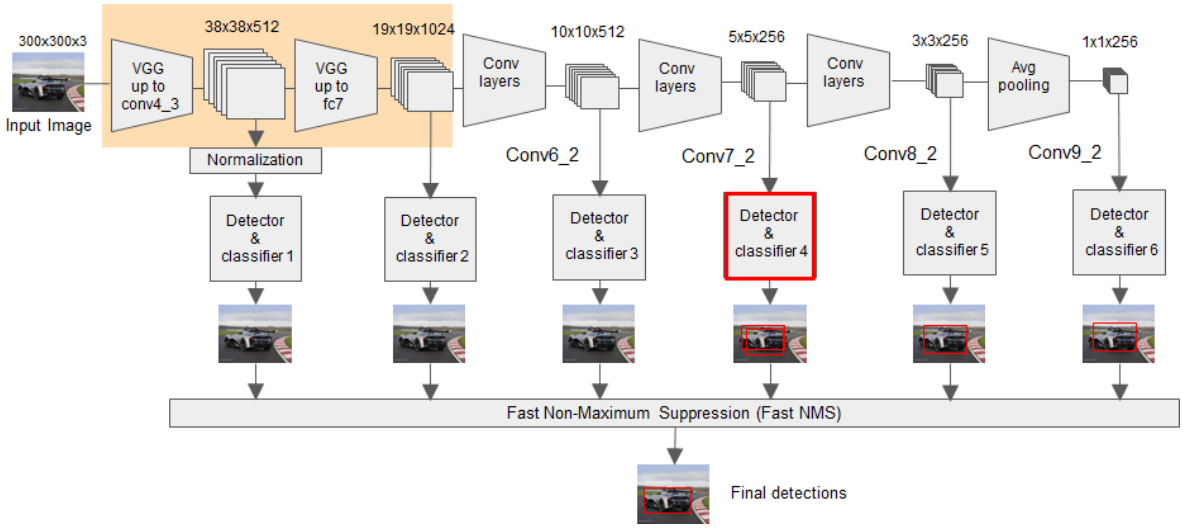
假设图像中的锚框为$A_1,A_2,\dots A_n$，真实边界框分别为$B_1,B_2,\dots B_m$,且满足$n>m$，这样就可以得到一个$n\times m$的矩阵，其中的X_{ij}代表着第i个锚框与第j个真实边界框的IoU值。

首先，我们找出矩阵X中最大元素，将该元素的行索引与列索引分别记为i_1,j_1。这就意味着此时A_{i1}与B_{j1}具有最高的相似度，因此可以将它们进行配对，然后将X矩阵中的i_1行上与j_1列上的所有元素丢弃，重复以上步骤，直到为每一个真实框各分配了一个锚边界框，且它们之间满足一一对应的关系。

接下来，应该遍历剩余的$n-m$个锚框：给定其中的锚框A_i，根据矩阵X的第i行找到与A_i交并比最大的真实边界框B_j，且只有当该交并比大于预先设定的阈值时，才为锚框A_i分配真实边界框B_j。在这一基础上，完成了对锚框的基本初始化操作。

SSD的基本思路

SSD的基本模型如下：



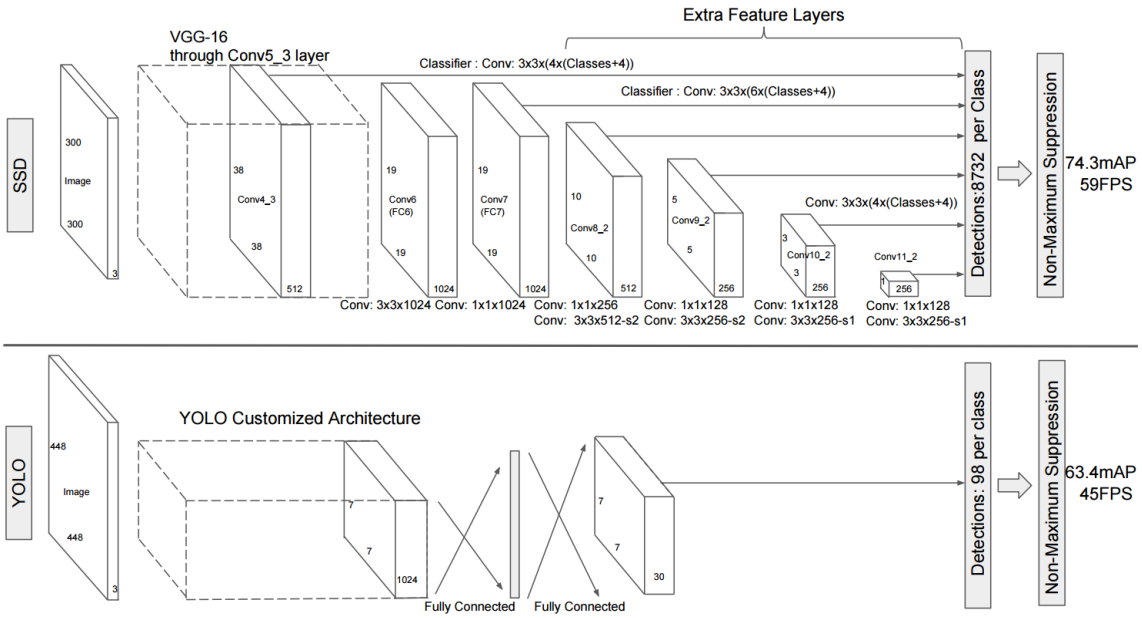
SSD的基本思想可以归结于以下几点：

- 同$YOLO$相似，将分类问题转化为回归问题，基于此，其使用的损失函数将会包含分类误差项与边界框误差项两部分
- SSD使用了one-stage的方法，并不会进行重采样，而是一次性完成classification以及bounding box regression
- SSD使用了金字塔架构，即使用多种feature map并在它们身上同时进行预测操作，这样做的好处是比较大的特征图来用来检测相对较小的目标，而小的特征图负责检测大目标

以下是SSD的一些基本操作步骤：

SSD网络架构

SSD的基本架构如下：



SSD采用$VGG16$作为基础模型，然后在$VGG16$的基础上新增了卷积层来获得更多的特征图以用于检测。这些卷积层可以用来检测不同大小的特征图并预测它们的类别以及偏移量。

可以发现SSD使用了多尺度的特征图进行检测，抛弃了$YOLO$中的全连接层而是使用了$Conv7$，$Conv8_2$，$Conv9_2$，$Conv10_2$，$Conv11_2$，以及$Conv4_3$作为检测所用的特征图。共提取了6个特征图，之后，在每一个特征图上会设置一系列尺度和大小不同的初始框，这些初始框都会反向映射到原图的某一个位置，其生成的规律可以总结为：随着网络层数加深（特征图的变小），初始框的尺度线性增加。其具体的尺度公式如下:

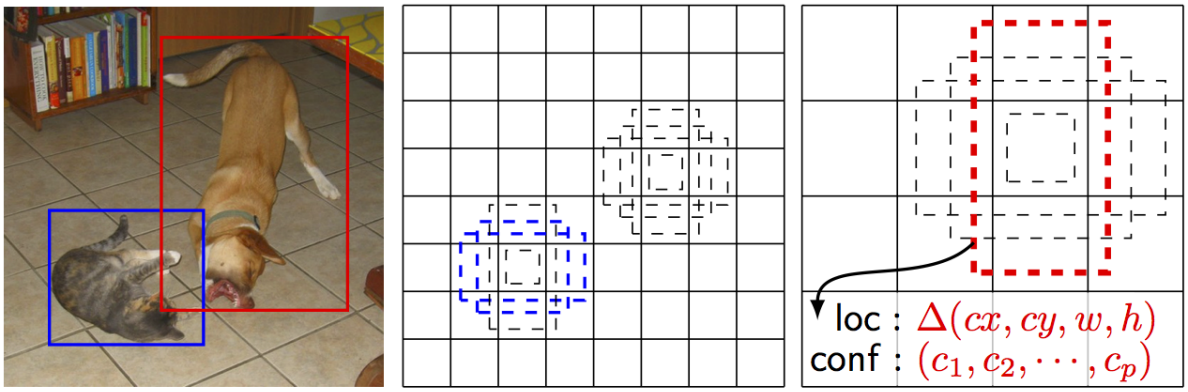
$$s_k = s_{min} + \frac{s_{max} - s_{min}}{m-1}(k-1), k \in [1, m]$$

SSD的基本是输入进行了全部物体样本标注的原始图。

在最后，SSD对于每个预测框，首先根据类别置信度确定其类别（置信度最大者）与置信度值，并过滤掉属于背景的预测框。然后根据置信度阈值（如0.5）过滤掉阈值较低的预测框。之后进行解码操作获得真实的位置参数，最终在经过一个non-maximum suppression层后输出结果，以此过滤掉那些重叠度较大的预测框。。

先验框的生成

每个单元设置尺度或者长宽比不同的先验框，预测的边界框（bounding boxes，default boxes）是以这些先验框为基准的，在一定程度上减少训练难度。一般情况下，每个单元会设置多个先验框，其尺度和长宽比存在差异，而可以看出，下面的实例中每个点包含了四个先验框。



(a) Image with GT boxes (b) 8×8 feature map (c) 4×4 feature map

在SSD中，分别使用预设的长宽比用于在每一个中心点生成`default box`:

$$a_r \in \left\{1, 2, 3, \frac{1}{2}, \frac{1}{3}\right\}$$
$$width: \quad w_k^a = s_k\sqrt{a_r}$$
$$height: \quad h_k^a = \frac{s_k}{\sqrt{a_r}}$$

针对每一个box，需要将其进行标记，标记的方式基于IoU参数，每一个标记主要分为两个部分。第一部分是各个类别的置信度或者评分，值得注意的是SSD将背景也当做了一个特殊的类别，如果检测目标共有 c 个类别，SSD其实需要预测 $c+1$个置信度值，其中第一个置信度指的是不含目标或者属于背景的评分，此外，SSD采用了`hard negative mining`，就是对负样本进行抽样，以此来维持正负样本的比例。第二部分就是边界框的location，包含4个值 (cx, cy, w, h)，分别表示边界框的中心坐标以及宽高。但是真实预测值应该为相对于先验框的偏移。

Loss Function

根据上文中的先验框生成结果，可以定义其损失函数：

$$L(x,c,l,g)=\frac{1}{N}(L_{conf}(x,c)+\alpha L_{loc}(x,l,g))$$

其中$L_{conf}(x,c)$为类别损失函数，其基于交叉熵，而$L_{loc}(x,l,g))$为位置损失函数，它是`Smooth L1 loss`由被预测的框与真实框的位置参数决定，而α是两者的权重，默认为1。

小物体标记

对比Faster R-CNN等架构，针对小物体的标记对于SSD架构相对难以实现良好效果，基于此，需要采取一系列数据扩增的操作，诸如随机采集块域（Randomly sample a patch）以获取小目标训练样本；或者以较小的IoU阈值进行采样操作。

此外可以使用"zoom in"与“zoom out"操作生成更多的经过放大或者缩小尺寸的实例进行训练。

实验结果

对模型进行分析，首先需要评估多重卷积层的效果：

Prediction source layers from:						mAP use boundary boxes?		# Boxes
conv4_3	conv7	conv8_2	conv9_2	conv10_2	conv11_2	Yes	No	
✔	✔	✔	✔	✔	✔	74.3	63.4	8732
✔	✔	✔	✔	✔		**74.6**	63.1	8764
✔	✔	✔	✔			73.8	68.4	8942
✔	✔	✔				70.7	69.2	9864
✔	✔					64.2	64.4	9025
	✔					62.4	64.0	8664

采用多尺度的特征图用于检测也是至关重要的，可以极大地提升模型的效果。

此外可以验证预设长宽比对模型性能的影响：

	SSD300				
more data augmentation?		✔	✔	✔	✔
include $\{\frac{1}{2},2\}$ box?	✔		✔	✔	✔
include $\{\frac{1}{3},3\}$ box?	✔			✔	✔
use atrous?	✔	✔	✔		✔
VOC2007 `test` mAP	65.5	71.6	73.7	74.2	**74.3**

可以发现，**使用不同长宽比的先验框可以得到更好的结果**，而一系列数据增强操作也可以极大地提高整个模型的性能。

源码分析：

总结：

SSD针对主干CNN结构在不同层次特征图，以小窗口卷积的方式，预测以预设框为基准的位置变换参数以及目标类别得分，从而得到不同尺度、不同形状的结果。

SSD是一种良好的one-stage模型，它采用单一CNN结构、以一阶段的模式实现目标检测，也是一个彻底的端到端模型，在速度方面也达到了实时目标检测的水准。SSD引入了类似Faster R-CNN的锚点机制，即在特征图的所有位置都绑定多个具有不同尺度、不同长宽比的预设框。除此之外，为了实现对不同尺寸目标的识别能力的提升，SSD在不同层次特征图上进行识别操作。

但其对小尺寸的目标的检测能力仍比较差，这是因为其更倾向于将小目标检测的任务交给浅层特征来完成，但是浅层特征往往表现出更多的纹理信息，而对语义信息的表达不够充分，造成错误判断。

参考

SSD Model Summary
Another review on this topic
Original Code
Code based on Keras