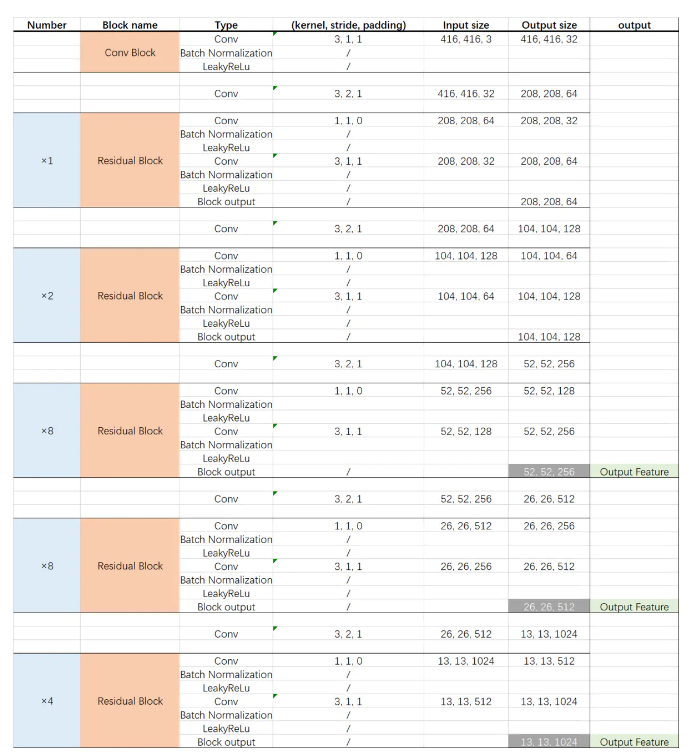
**实验二 目标检测**

1. **实验要求**

使用 YOLO\_v3 模型，在PASCAL VOC训练数据集上训练，并在测试集上进行测试。

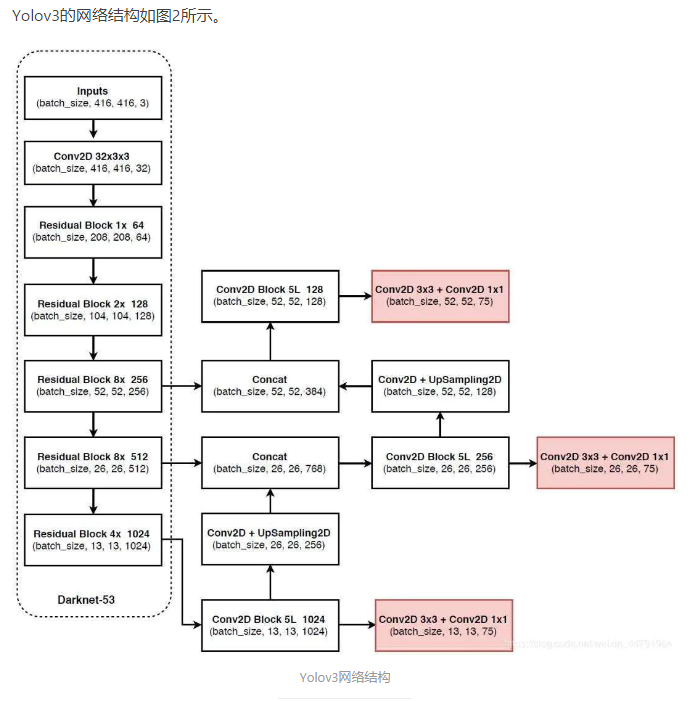
1. **实验原理**
2. YOLO原理简介

首先要明确地是YOLO系列模型是一步的目标检测模型，其主体就是一个卷积神经网络，该网络对图像进行处理后可同时得到若干关于目标的包围盒以及盒中目标的类别。YOLOv3采用Darknet作为其卷积神经网络部分，图1.1给出了Darknet53的构建细节：



**图1.1** Darknet53的构建细节

图1.2给出了Darknet53的结构详情：



**图1.2** （YOLOv3）Darknet53的网络结构

根据图1.1、图1.2，我们不难发现YOLOv3具有如下特点：

1. 模型在三个尺度上分别输出特征图。根据图1.1，我们不难得知YOLOv3一共有六组卷积，其中最后三组分别对输入图像做8倍、16倍和32倍下采样，于是YOLOv3便在这三种尺度上分别输出一个特征图。值得注意的是，在YOLOv3的最后三层，特征信息的流动并不是仅仅自上而下的过程，32倍下采样的特征信息通过上采样又流回了16倍下采样层，紧接着，16倍下采样的特征信息混合着32倍下采样的特征信息再经上采样后又流回了8倍下采样层。所以说这三层特征图间的信息是共享的，只是分辨率不同而已；
2. 使用1x1卷积作为输出层。与传统的卷积神经网络不同，YOLOv3没有采用全连接，而是采用1x1卷积作为输出层，这是其采用对图片做格栅化处理以回归出包围框所需输出的形式导致的。从图2中不难看出，1x1卷积在特征图形状方面仅仅只是做了一个通道数的变换；
3. 包围框的回归方式以及模型输出的形式。YOLOv3和YOLOv1在包围框回归方面方法相同，都是把图片分成若干个小个，YOLOv1中是7x7，而YOLOv3中为13x13、26x26以及52x52三种，分别对应三种尺度的模型输出。从而，为确定出一个包围框，我们只需知道这个包围框的中心点落在哪个格子中、相对格子的位置怎样以及包围框的长和宽即可，如此就得到了输出的形式，每一个小格子应当给出至少一个输出，并且输出应当是一个5+目标类别数nC（YOLOv3中nC=70）的向量，该向量决定了一个包围框，以及其中是否存在目标、存在目标的情况下目标的类别。其每一维的含义分别是包围框中可能含有目标的概率（置信得分）、包围框中心点横坐标相对于当前格子的水平归一化偏移量、包围框中心点纵坐标相对于当前格子的垂直归一化偏移量、包围框的归一化宽度、包围框的归一化高度以及包围框中目标所属每一个类别的概率（共nC个值）。假设我们要求每个格子给出nA个输出（YOLOv1中nA=2，YOLOv3中nA=9），三种尺度下，一张图片被划分成小格子的数量分别是13x13、26x26以及52x52，于是就得到三个四维张量的输出，形状分别为13x13xnAx(5+nC)、26x26xnAx(5+nC)以及52x52xnAx(5+nC)，对于其中任意一个张量，其索引为[i, j, a]的元素表示图片中第i行第j列的小格子所给出的第a个包围盒的信息；
4. 锚盒子。锚盒子（anchor box）就是从训练集中真实框（ground truth）中统计或聚类得到的几个不同尺寸的框，避免模型在训练的时候盲目的找，有助于模型快速收敛。YOLOv3中引入了锚盒子机制，对预测的对象范围进行约束，并加入了尺寸先验经验，从而实现了多尺度学习的目的；
5. 根据（3）中的分析，在进行预测时，对于每一张输入图片，YOLOv3都将为其预测并生成13x13xnAx(5+nC)+26x26xnAx(5+nC)+52x52xnAx(5+nC)个可能的包围框，然而YOLOv3采用了非极大抑制（NMS）算法，从中找到可能性最高的若干个包围框；
6. 损失函数。YOLOv3的损失函数与YOLOv1基本一致，总的损失中包含三种损失——置信损失、坐标损失和分类损失，分别采用二进制交叉熵、MSE和交叉熵进行计算。
7. **实验结果**

为更加深入地了解YOLO的工作原理和实现细节，本次实验中我们在代码（GitHub地址为：https://github.com/muyiguangda/pytorch-yolov3）的基础上对其进行修改并完成了实验。

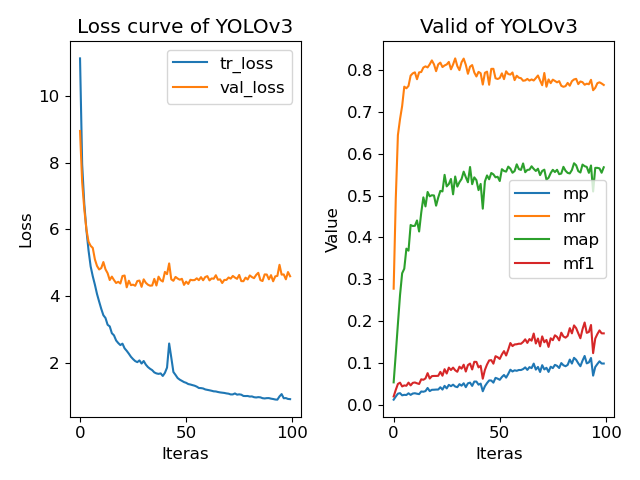
实验所用数据集为VOC 2012，其中共有2913张图片可用于进行图像分割，实验中按8:1:1的比例划分训练集、验证集和测试集，并对数据集做了预处理，在原数据集目录下新建了labels目录，用于保存处理后的标注信息，标注信息格式为“类别 包围框中心点横坐标x 包围框中心点纵坐标y 包围框宽度w 包围框高度h”，其中x、y、w、h都使用源图片的尺寸做了归一化处理。

由于VOC 2012的类别数与原YOLOv3中所采用的类别数不同，故实验中没有采用官方训练好的权重参数进行预训练后的微调。

**表1** 模型在各个类别上的AP值

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| person | bird | cat | cow | dog | horse | sheep | aeroplane | bicycle | boat |
| 0.6654 | 0.4803 | 0.5897 | 0.4260 | 0.5537 | 0.3388 | 0.497 | 0.7025 | 0.355 | 0.2391 |
| bus | car | motorbike | train | bottle | chair | dining table | potted plant | sofa | tv/  monitor |
| 0.6878 | 0.4777 | 0.454 | 0.5831 | 0.3265 | 0.2911 | 0.3742 | 0.3908 | 0.4089 | 0.3658 |

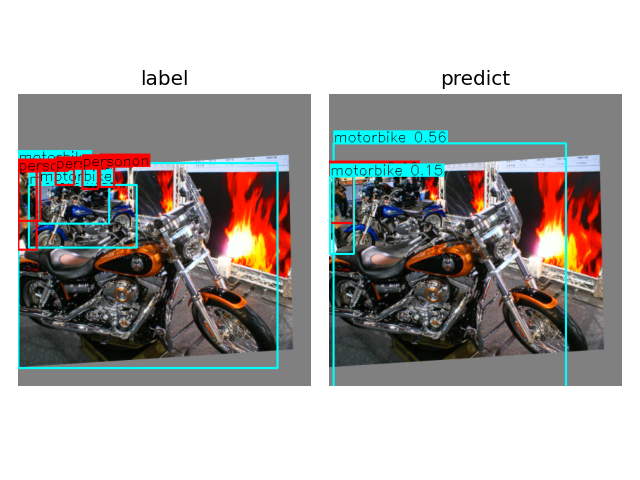
模型采用SGD方法训练，学习率为0.001，冲量为0.9，权重衰减率为0.0005，最大训练次数为100轮，批大小为32。非极大抑制的IoU阈值和非极大抑制阈值均为0.1，置信得分阈值在训练时设置为0.001以取得较高的mAP，而测试时设置为0.1从而得到有限个置信度较高的预测结果。图2.1（左）给出了模型训练过程中在训练集和验证集上的损失变化曲线，图2.1（右）则显示了训练过程中在验证集上的mp、mr、mAP、mF1这四个指标的变化情况（其中NMS的置信度阈值为0.001）。图2.2至图2.6分别给出了训练好的模型在五组测试数据上的目标检测效果（其中NMS的置信度阈值为0.1）。表1给出了模型在各个类别上的AP值：



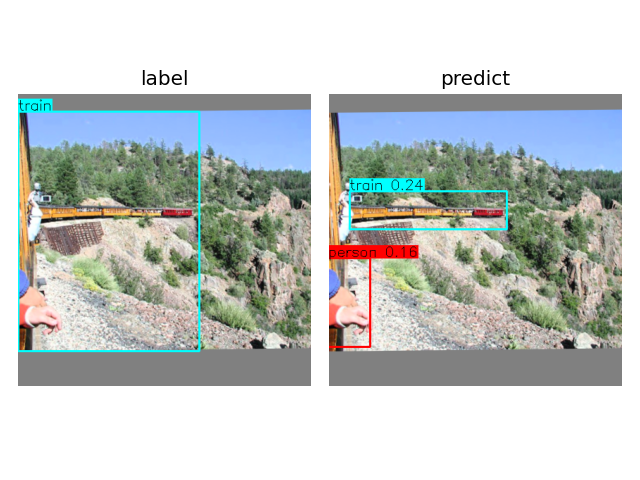
**图2.1** YOLOv3训练情况



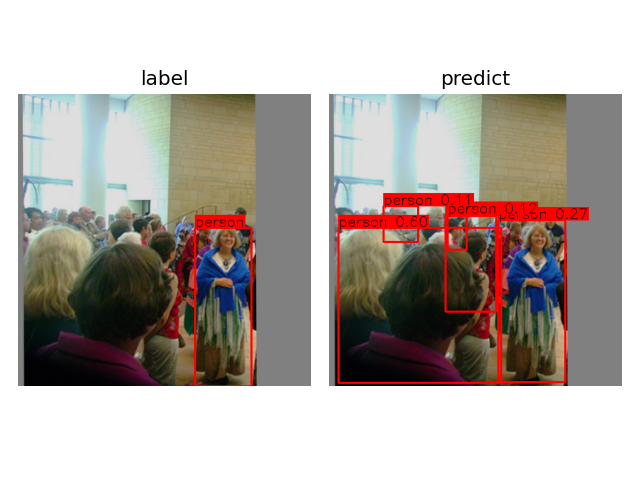
**图2.2** 模型测试效果（一）



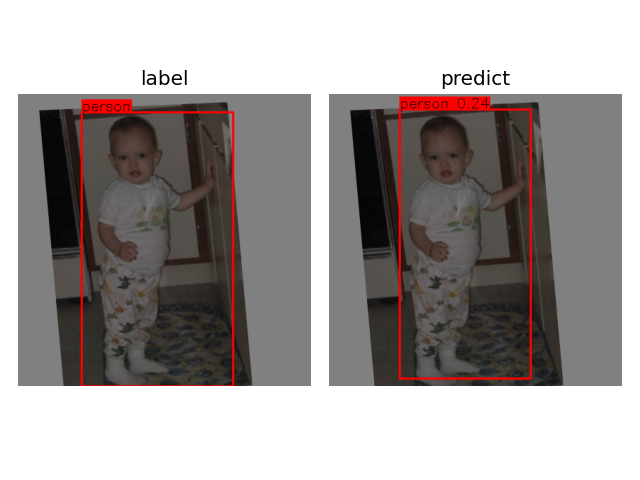
**图2.3** 模型测试效果（二）



**图2.4** 模型测试效果（三）



**图2.5** 模型测试效果（四）



**图2.6** 模型测试效果（五）

从验证结果可见，模型的识别结果总体说来还是不错的，以第三、第四组测试数据为例，模型给出的火车的包围框甚至比标注给出的还要精确，且还能够找到标注中都没有标出的人体目标，更值得一提的是模型根据目标的一部分部分（比如人的手）就可以确定目标的类别，这说明通过大量的数据，模型确实学习到了有用特征。但一个令人沮丧的问题是，即使模型准确地框出了物体的位置，但是给出的置信得分普遍偏低，如果以0.5作为判断标准的话，那么模型几乎不能做出有效的识别，这个问题以我个人目前所掌握的目标检测的知识还无法做出准确的解释。