附件4

**基于YOLOv2的左右手识别优化研究与实现**

专业班级：暴印行

指导教师：靳冰 讲师

**摘 要：**目前全球最先进的一些物体检测方法都需要设置有目的性的分类器来执行检测，而YOLOv2（You Only Look Once）是一种全新的实时物体检测方法,将边界框和类别概率关联到起来，设计成一个简单的回归问题。它使用多种分辨率数据输入的训练方法，在精度和速度之间做了一个平衡，从而使其性能达到最优。要实现基于YOLOv2的左右手识别，首先需要组建一个网络模型，然后采集并标注大量应用场景数据，进行模型训练及优化，得到权重文件，最后运用这个训练好的权重文件把图片中的左右手识别出来。基于YOLOv2的左右手识别速度和精度都相当高。基础模型速度可以达到45帧每秒，同时Mean AP (平均精度)达到78.6%。与目前的最先进的方法相比，虽然YOLOv2的位置检测误差较大，对小目标检测效果不理想，但是对于背景它有更低的误检，同时，YOLOv2能够轻松学习到更泛化的特征。由于在触控行业，触控要做到实时响应，并且应用场景中摄像头捕获的图片中手的像素宽高均不小于78px，不算是小目标，所以YOLOv2完全可以胜任这项工作。

**关键词：**YOLO；深度学习；左右手

**1 绪论**

1.1 课题的背景、目的和意义

目前全世界的触控屏按照工作原理及传输信息的介质，可将其分成四类：电阻式、电容感应式、红外线式和表面声波式。而把计算机视觉和深度学习用于触控产品上目前还是全球首例。

最近现有的深度学习方法，如R-CNN运用了区域检测方法，首先在图片中生成大量的可能存在的边界框，然后以这些边界框为基础设计分类器。分类后，再剔除误检测的边界框，消除有重复的检测，每种类别在小范围内只留一个，并根据场景中的其它目标从新定位框的位置。这些非常复杂的计算难以优化而且很慢，因为每个独立的组件都必须单独训练。再比如SSD在神经网络最后一层加入一个带有滑动式窗口算法的全链接层，它对最后的特征图每个位置都进行了评估，方式是非常经典的，但是将会付出沉重的时间复杂度以及空间复杂度代价。

YOLO相比较以前的方法就显得更加精小，它将目标检测从新设计为单个回归问题，直接将图片划分为7\*7的网格，评估以网格中心为中心的两个边界框，也就是总共98个边界框待处理评估。YOLO只需要在图片上运行一遍神经网络，就能从图片像素直接得出边界框坐标与类的概率。

1.2 课题目标

触控产品需要做到实时响应触控点，这对深度模型的效率要求很高，触控产品还需要做到精准的触控点位，这对深度模型的精度要求也很高，需要一个综合性能很强的神经网络框架才能实现。

左右手识别要做到实时检测，并且Mean AP要高于75%，还要让误检率尽可能的少。

1.3 主要内容

解决左右手识别问题主要要做两件事：

1.研究YOLO深度学习框架工作基本思路及其使用方法

2.设计网络模型与训练优化。编写网络模型配置文件，将YOLOv2原有的平方差损失函数替换成交叉熵损失函数，然后根据训练结果调整超参数，使网络模型达到最优状态，保存权重文件，有网络模型配置文件与权重文件就可以用于左右手识别了。

1.4 论文的组织结构

章节1是绪论，主要讲课题的背景、目的、意义和论文组织结构。

章节2是理论基础，主要讲YOLO 深度学习框架工作基本思路。

章节3是算法描述，主要讲网络模型中两个重要的算法简述。

章节4是设计实现，主要讲深度学习算法的详细算法实现以及设计流程。

章节5是优化，主要讲交叉熵的导入。

章节6是结果与结论，主要讲左右手识别的效果及性能提升。

**2 理论基础**

YOLO很精巧。用YOLO处理图片非常简单而且直接。YOLO将输入图片大小调整为448px\*448px，在输入图片上运行单个卷积网络，以及用边界框的置信度与网络阈值对比，留下置信度最高的边界框。另外YOLO卷积网络可以同时预测所有格子的多个边界框与类别的概率。

YOLOv2的检测系统如图2-1所示。

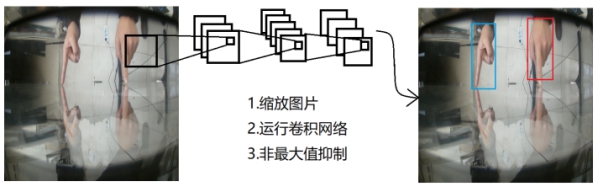


图2-1 YOLOv2的检测系统

**3 算法描述**

3.1 神经网络框架相关的概念

3.1.1激活函数

激活函数有很多，但常用的有四个：sigmoid、linear 、relu、leaky relu[1]。大体功能都很相似，都是把数据一个映射到一个小的区间，但有微小差别:sigmoid把数据映射到[0,1]；tahn把数据映射到[-1,1]；relu负数全部映射到0，但损失一些特征；leaky relu负数乘以一个小于1的正数值。

3.1.2损失函数

损失函数（loss function）是用来评估网络模型的预测值与真实值的差异程度，然后通过反向传播用这个误差更新权重[2]。它的输出通常是一个不小于零的实数,损失函数越小，模型的稳定性就越好。平均精度越高，模型的性能就越好。损失函数是反向传播的核心部分。用于模型训练的损失函数多种多样，比如，Leaky ReLU, ReLU, MSE, CE。

3.1.3 池化层

池化层通常放在卷积层的后面，一般卷积层输出的特征图比较冗余，池化层的目标是消减卷积层输出的信息，从而节省一些内存开销，避免权值的冗余，更具体一点就是，池化层可以把卷积层的输出作为它的输入并且输出一个更加紧凑的特征向量传递给下层网络。

3.1.4 全链接层

全链接层（Fully Connected Layers，FC）在整个卷积的神经网络中饰演着“分类器”的角色。如果说池化层和卷积层等层结构是将输入数据映射到零散特征图的话，全链接层就起到将前边卷积层学到的零散的特征映射到联系紧凑的特征图的作用[3]。然后根据这个紧凑的特征图就可以把目标分类。

3.1.5 Dropout

Dropout是为防止模型过拟合的而生的一种手段[4]。它的基本思想是，随机屏蔽某块区域的特征图，使其暂时不参与本轮的运算过程。这样做，模型可以更加的鲁棒，因为这样做，模型才不会过度的以来某些局部特征使最终的训练方向跑偏。到了下一轮的训练中，这块区域还是有可能参与运算的，因为这个过程是随机的，它只是阻止本轮的这块区域参与运算。

3.2 使用方法

3.2.1 输入输出

图片测试的输入输出如图3-1所示。

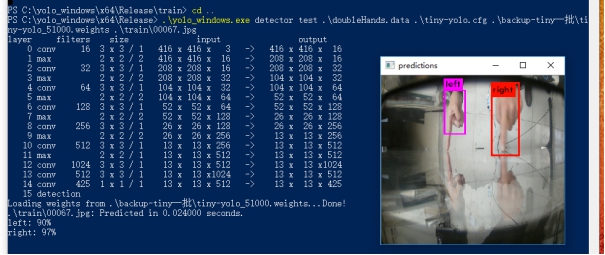


图3-1 图片测试

输入分别是：

detector：YOLOv2的目标检测方法，YOLO是detect；

test:图片测试方法；

.\doubleHands.data:配置class name文件位置；

.\tiny-yolo.cfg:配置模型架构及超参数；

.\backup-tiny一批\tiny-youlo\_51000.weights:模型权重；

.\00067.jpg:要测试的图片。

输出分别是：

left:90%：左手的置信度为90%；

right:97%：右手的置信度为97%；

图片是可视化输出。

3.2.2 流程

网络在检测的时候，只需要输入图片（输入视频的时候，网络会把视频拆分成图片），图片经过卷积层得到零散的特征图，再经过池化层提炼重要特征，前两步过程通常要循环多次，卷积层层数越多，提炼的特征就越丰富，最后经过全链接层把这些零散的特征图紧密联系起来，输出一个tensor作为检测结果，将tensor内容解析出来可视化表示，这时候就可以得到一张或者一帧带有标框及类别的图像了。

流程如图3-2所示。

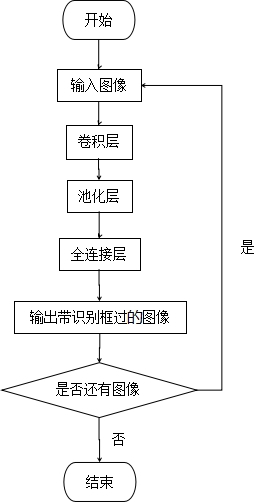


图3-2 YOLO测试流程图

**4 设计实现**

YOLO的网络架构的思想来自于GoogLeNet图片分类模型。可实现实时检测和端到端训练，同时还能保持较高的平均精度[5]。YOLO有二十四个卷积层，后接两个全链接层。与GoogLeNet运用的初始模块不同的是，YOLO简单地运用了1\*1的缩放层，然后是3\*3的卷积层,完整的网络如图4-1所示：

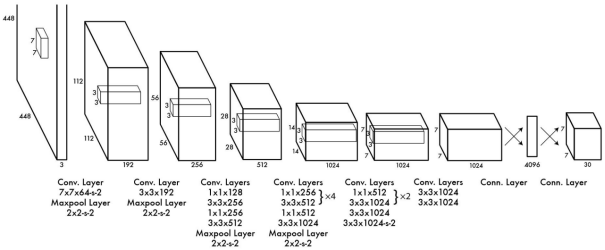


图4-1 YOLO网络结构

**5 优化**

将原架构中的均方差损失函数换成交叉熵损失函数，交叉熵(Cross-Entropy，CE)最大的优点是训练时可以避免函数的梯度消失，主要用于度量两个概率分布之间的相似性[6]。为了解决参数更新效率下降这一问题，代码中，用交叉熵的损失函数替代了网络框架中的平方差的损失函数。如果预测值与实际值的误差越大，那么在反向传播的过程中权重调整的幅度就要越大，从而能够使训练加速收敛。

当分类器为二分类时:

C:\Users\byhang\AppData\Local\Temp\ksohtml\wpsB3E2.tmp.jpg

当分类器为非二分器时：

C:\Users\byhang\AppData\Local\Temp\ksohtml\wpsB3E3.tmp.jpg

其偏导函数是;

C:\Users\byhang\AppData\Local\Temp\ksohtml\wpsB3E4.tmp.jpg

C:\Users\byhang\AppData\Local\Temp\ksohtml\wpsB3E5.tmp.jpg

**6 结果与结论**

优化后的YOLOv2表现很好，在 自己的测试集上Mean AP可以跑到78.6%。其中Batch norm：批正则化；Hight resolution classifier：高分辨率分类器；Anchor boxes：锚点框；Dimension priors：维度聚类；Location prediction：本地特征预测；Passthrough：直通层；Multi-scale：多图像尺寸训练；Cross-Entropy:交叉熵；‘Y’是指加入了该项优化。YOLO在没有优化的情况下，平均精度是63.4%，YOLOv2相对于YOLO增加了批正则化、高分辨率分类、全链接层换成卷积层、用新网络消减计算量、维度聚类、本地特征预测、直通层和多图像尺寸训练，综合起来平均精度提升了13.4个点，有了一个质的飞跃，但是YOLOv2用的还是效率平平的均方差损失函数，还有一定的优化空间，而在YOLOv2的基础上又加上交叉熵损失函数，预测值与实际值的误差越大，那么在反向传播的过程中权重调整的幅度就要越大，从而能够使训练加速收敛，有效地使模型训练加速收敛，节省了训练时间，平均精度又提升了1.8个点。如表6-1所示。

表6-1 YOLOv2加入优化的对比

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | YOLO | YOLOv2 | | | | | | |  |
| Batch norm |  | Y | Y | Y | Y | Y | Y | Y | Y |
| Hight resolution classifier |  |  | Y | Y | Y | Y | Y | Y | Y |
| Anchor boxes |  |  |  | Y | Y |  |  |  |  |
| New network |  |  |  |  | Y | Y | Y | Y | Y |
| Dimension priors |  |  |  |  |  | Y | Y | Y | Y |
| Location prediction |  |  |  |  |  | Y | Y | Y | Y |
| Passthrough |  |  |  |  |  |  | Y | Y | Y |
| Multi-scale |  |  |  |  |  |  |  | Y | Y |
| Cross-Entropy |  |  |  |  |  |  |  |  | Y |
| Mean AP | 63.4 | 65.8 | 69.5 | 69.2 | 69.6 | 74.4 | 75.4 | 76.4 | 78.6 |

**参考文献：**

1. Joseph Redmon, Santosh Divvala, Ross Girshick, Ali Farhadi.You Only Look Once:Unified, Real-Time Object Detection[J].arXiv:1506.02640v5 ,2016,1～10
2. Joseph Redmon, Ali Farhadi.YOLO9000:Better, Faster, Stronger[J]. arXiv:1612.08242v1.2016,1～9
3. K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition[J]. arXiv preprint arXiv:1512.03385,2015. 2～5
4. K. Lenc and A. Vedaldi. R-cnn minusr[J].arXiv preprint arXiv:1506.06981, 2015. 5～6
5. J. Redmon and A. Angelova. Real-time grasp detection using convolutional neural networks[J].CoRR, abs/1412.3128, 2014.5
6. lan Goodfellow, Yoshua Bengio, Aaron Courville. DEEP LEARNING[M].北京人民邮电出版社.2017年8月.201～226
7. 周志华.机器学习[M].清华大学出版社.2016年1月.97～246
8. 李航.统计学习方法[M].清华大学出版社.2012年3月.1～53
9. J. Dong, Q. Chen, S. Yan, and A. Yuille. Towards unified object detection and semantic segmentation. In Computer Vision–ECCV 2014, 2014.7 299～314
10. K. Lenc and A. Vedaldi. R-cnn minus r.arXiv preprint arXiv:1506.06981, 2015. 5, 6
11. C. Szegedy, S. Ioffe, and V. Vanhoucke.Inception-v4, inception-resnet and the impact of residual connections on learning. CoRR, abs/1602.07261, 2016. 2
12. 谢剑斌等.视觉机器学习20讲[M].清华大学出版社.2015年6月.59～164

**Optimization and Implementation of**

**Left and Right Hand Recognition Based on YOLOv2**

**Abstract:** At present, some advanced object detection methods need to set up a purposeful classifier to perform the detection, and YOLOv2 (You Only Look Once) is a brand-new real-time object detection method that associates the bounding box and class probability, designed as a simple regression problem. It uses multiple resolution data training methods to achieve a balance between accuracy and speed, thereby do the best performance. To realize left and right hands recognition based on YOLOv2, we first need to set up a network model, then collect and label a large number of application scene data, run model training and optimization, then get a weight file, and finally use this trained weight file to identify the left and right hands in the image. The speed and accuracy of Right and left hands recognition based on YOLOv2 are quite high. The basic model speed can reach 45 frames per second, while Mean AP (Mean Average Precision) reaches 78.6%. Compared with the current state-of-the-art methods, although YOLOv2's position detection error is larger and the detection effect on small targets is not ideal, but it has lower misdetection on the background, and at the same time, YOLOv2 can easily learn more generalized features. In the touch industry, touch control needs real-time response, and the pixel width of the hand captured by the camera in the application scene is not less than 78px, which is not a small target. Therefore, YOLOv2 is very suitable for this task.

**Key words:** YOLO; deep learning; left and right hand