doi:10.6041/j.issn.1000-1298.2020.08.002

基于轻量化 YOLOv3 卷积神经网络的苹果检测方法

武　星[1]　齐泽宇[1]　王龙军[1]　杨俊杰[1]　夏　雪[2]

(1. 南京航空航天大学机电学院, 南京 210016; 2. 中国农业科学院农业信息研究所, 北京 100081)

摘要: 为使苹果采摘机器人在复杂果树背景下能快速准确地检测出苹果,提出一种轻量化 YOLO(You only look once)卷积神经网络(Light-YOLOv3)模型与苹果检测方法。首先,对传统 YOLOv3 深度卷积神经网络架构进行改进,设计一种同构残差块串联的特征提取网络结构,简化目标检测的特征图尺度,采用深度可分离卷积替换普通卷积,提出一种融合均方误差损失和交叉熵损失的多目标损失函数;其次,开发爬虫程序,从互联网上获取训练数据并进行标注,采用数据增强技术对训练数据进行扩充,并对数据进行归一化,针对 Light-YOLOv3 网络训练,提出一种基于随机梯度下降(Stochastic gradient descent,SGD)和自适应矩估计(Adaptive moment estimation,Adam)的多阶段学习优化技术;最后,分别在计算机工作站和嵌入式开发板上进行了复杂果树背景下的苹果检测实验。结果表明,基于轻量化 YOLOv3 网络的苹果检测方法在检测速度和准确率方面均有显著的提高,在工作站和嵌入式开发板上的检测速度分别为 116.96、7.59 f/s,F1 值为 94.57%,平均精度均值(Mean average precision,mAP)为 94.69%。

关键词: 苹果; 采摘机器人; 目标检测; YOLOv3 网络; 深度可分离卷积; 网络训练

中图分类号: TP391.4　　**文献标识码**: A　　**文章编号**: 1000-1298(2020)08-0017-09　　**OSID**:

Apple Detection Method Based on Light-YOLOv3 Convolutional Neural Network

WU Xing[1]　QI Zeyu[1]　WANG Longjun[1]　YANG Junjie[1]　XIA Xue[2]

(1. *College of Mechanical and Electrical Engineering*, *Nanjing University of Aeronautics and Astronautics*, *Nanjing* 210016, *China*
2. *Agricultural Information Institute*, *Chinese Academy of Agricultural Sciences*, *Beijing* 100081, *China*)

Abstract: An apple detection method (Light-YOLOv3) based on lightweight YOLO (You only look once) convolutional neural network was proposed for apple picking robots to detect apples quickly and accurately in the complex background of fruit trees. Firstly, in order to improve the traditional YOLOv3 deep convolutional neural network architecture, a feature extraction network structure containing cascaded homogeneous residual blocks was designed, and the dimensionality of the feature map for object detection was simplified. In this architecture, the conventional convolution was replaced by the depth wise separable convolution, and a multi-objective loss function was defined in terms of the mean square error loss and the cross entropy loss. Secondly, the training data was obtained from the Internet by means of a crawler program, and then labelled. The data augmentation technique was used to expand the training data and normalize it. Thirdly, a multi-stage learning optimization approach based on stochastic gradient descent (SGD) and adaptive moment estimation (Adam) was proposed to train Light-YOLOv3 network. Finally, an apple detection experiment in the complex background of fruit trees was performed on a computer workstation and an embedded processor, respectively. The experimental results showed that the apple detection method based on Light-YOLOv3 network improved the detection speed and accuracy significantly, i.e., the detection speed on the computer workstation and the embedded processor can reach 116.96 f/s, 7.59 f/s, F1 value can reach 94.57%, and the mean average precision (mAP) can reach 94.69%.

Key words: apple; picking robot; object detection; YOLOv3 network; depth wise separable convolution; network training

收稿日期: 2019-12-03　修回日期: 2020-03-22

基金项目: 国家自然科学基金项目(61973154)、国防基础科研计划项目(JCKY2018605C004)、中央高校基本科研业务费专项资金项目(NS2019033)和南京航空航天大学研究生创新基地(实验室)开放基金项目(kfjj20190516)

作者简介: 武星(1982—),男,副教授,主要从事移动机器人智能感知与控制研究,E-mail: wustar5353@nuaa.edu.cn

0 引言

目前,我国农业生产不断向规模化、集约化、精准化方向发展,对具有智能化、自动化的农业智能装备的需求也快速增加[1]。苹果是我国产量最大的水果,由于果园环境复杂,目前仍依靠人工采摘[2-3]。因此,在农业劳动力紧缺、采摘成本不断增加的情况下,以苹果采摘机器人代替人工采摘具有重要的现实意义和广阔的应用前景[4]。

快速准确的苹果检测是苹果采摘机器人自动采摘的关键。复杂果树背景下的苹果快速准确检测受诸多因素影响:果实之间相互重叠遮挡,树叶也会对果实产生遮挡;现场的光照环境非常复杂;苹果采摘机器人本身的计算资源有限,复杂算法运行效率不高。这些为苹果快速准确检测带来诸多困难。

苹果检测本质上属于目标检测。目前,目标检测方法主要分为传统方法和深度学习方法。传统方法受光照变化和复杂背景的影响较大,在此方面已有一些针对不同应用场景的改进方法[5-8]。文献[9]研究了基于区域增长和颜色特征的图像分割方法,提出一种基于支持向量机的苹果识别分类算法。文献[10]在不同光线和阴影的影响下,采用 $R-G$ 色差分量和改进型 Otsu 法进行图像分割,求取目标质心,并使用同面积的圆形进行拟合,较好地实现了对成熟苹果的识别。文献[11]通过颜色和纹理检测可能属于苹果的像素,由这些像素组成种子区域,然后与理想的特征模型进行比较,从而判断该区域是否包含苹果。

近年来,以图像特征自学习为优势的深度学习方法开始运用于苹果检测。文献[12]针对 4 种水果提出了一种改进的 SSD(Single shot multibox detector)水果检测模型,采用 ResNet-101 模型替换 SSD 模型中的 VGG16 输入模型。文献[13]借鉴 DenseNet 网络的思想对 YOLOv3 网络进行改进,对遮挡、重叠情况及不同生长阶段的苹果进行有效检测。文献[14]改进设计了基于 ResNet-44 的 R-FCN(Region-based fully convolutional networks)网络,能够识别疏果前的苹果目标。虽然上述方法都具有一定的鲁棒性和泛化性,但是实验平台均为高性能计算机,缺乏考虑采摘机器人有限计算资源的影响。

YOLO[15](You only look once)网络是一种通用的 one-stage 目标检测算法,通过单个卷积神经网络处理图像可直接计算出多种目标的分类结果与位置坐标。然而,苹果采摘机器人只需检测单种目标(苹果),该网络结构过于庞大,计算复杂度高,检测效率不足,无法满足采摘实时应用。因此,本文在 YOLOv3 网络的基础上提出一种轻量化卷积神经网络改进设计方法,采用深度可分离卷积[16]替换普通卷积,定义一种融合均方误差损失和交叉熵损失的多目标损失函数,在网络训练时采用一种基于随机梯度下降(Stochastic gradient descent,SGD)和自适应矩估计(Adaptive moment estimation,Adam)的多阶段学习优化技术,并在工作站和嵌入式开发板上分别进行实验,对改进后的模型和方法进行检测速度和准确率的验证。

1 轻量化卷积神经网络设计

1.1 传统 YOLOv3 网络结构

YOLOv3 网络的整体结构如图 1 所示,使用 Darknet-53 网络来提取特征,Darknet-53 网络的主体结构如图 1 蓝色虚线框内所示,该网络借鉴了残差网络[17]的结构,在一些网络层之间加了跳跃连接,由此构建了残差单元,使得网络结构可以更深。网络结构中的基本单元如图 1 红色虚线框内所示,由卷积层、批归一化层(Batch normalization,BN)[18]和 Leaky ReLU 激活函数组成;由 2 个这样的基本单元和跳跃连接组成图 1 绿色实线框内所示的残差单元,再分别由 1、2、8、8、4 个残差单元组成 5 个残差块,最终组成网络的主体结构。

为了改善网络对小尺寸物体检测性能差的问题,YOLOv3 使用 13×13、26×26、52×52 这 3 个不同尺度的特征图来进行目标检测,并将低分辨率的特征图上采样后与高分辨率的特征图进行拼接,如图 1 蓝色实线框内所示。

1.2 轻量化网络结构改进

1.2.1 网络结构简化

YOLOv3 使用的基础网络为 Darknet-53。然而,在苹果检测这一个单类别检测任务中,并不需要如此复杂的网络。并且,神经网络本身具有较高的冗余性,因此可以对网络结构进行简化以减少计算量。本文仿照 Darknet-53 网络的结构设计了一个新的基础网络,该网络由 5 个相同结构的残差块串联组成,每个残差块包含 2 个残差单元和一个步长为 2 的卷积,减少了网络的层数和计算量。

由于本文设计的网络应用于苹果采摘机器人上,实现苹果的识别定位,所以可以不对采摘机器人工作空间之外的苹果进行检测。本文使用的苹果采摘机器人的工作空间是半径为 0.85 m 的球体。在网络输入为 416 像素×416 像素的条件下,52×52 特征图中单个网格的尺寸为 8 像素×8 像素。该尺

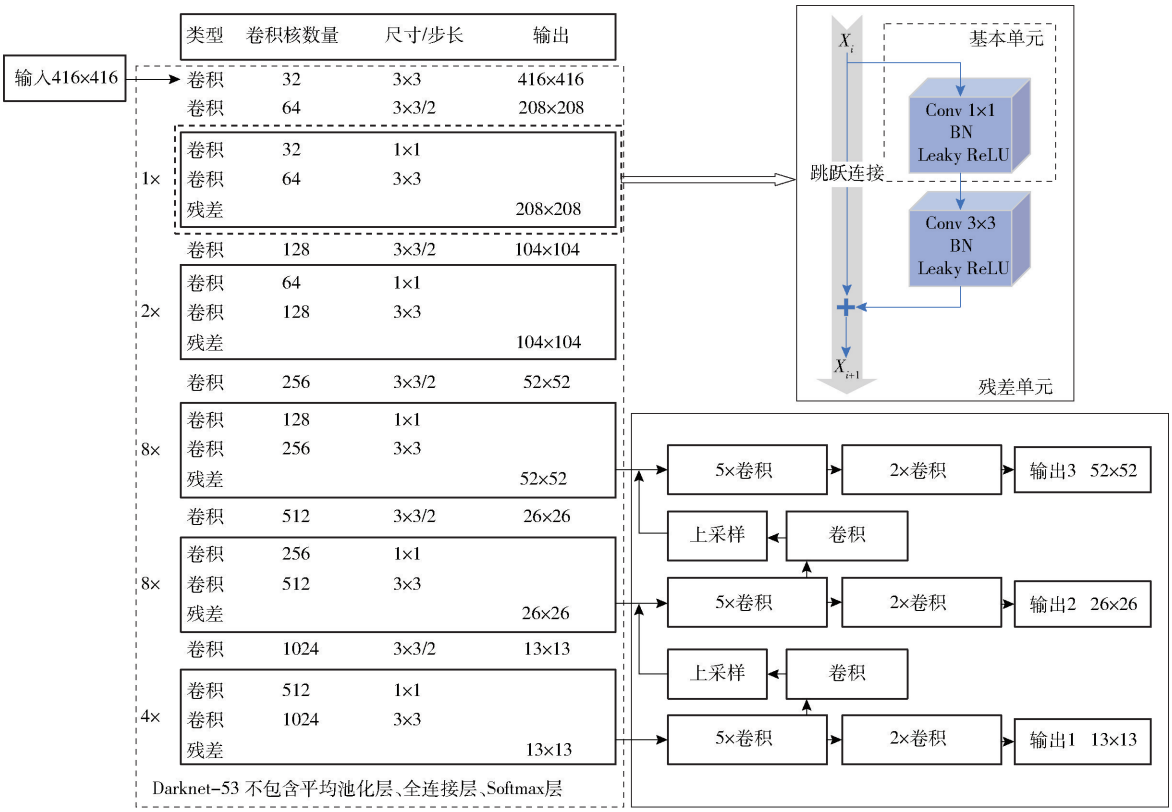


图 1　YOLOv3 网络结构

Fig. 1　YOLOv3 network structure

寸转换到本文使用的 1 280 像素 ×780 像素图像分辨率下，单个网格的尺寸约为 24.6 像素 ×24.6 像素。根据相机内参的标定结果，在机器人工作空间内，该网格所对应的最大物体的二维投影尺寸约为 22.5 mm ×22.4 mm，远小于正常苹果的二维投影尺寸。因此无需在该尺度上进行苹果检测，只需要使用 13 ×13 和 26 ×26 这 2 个尺度的特征图进行苹果检测，从而删除生成 52 ×52 尺寸特征图的相关网络结构，减少计算量。

1.2.2　使用深度可分离卷积

使用深度可分离卷积替换原网络中所有 3 ×3 的普通卷积。深度可分离卷积的整个工作流程如图 2 所示，包括深度卷积和逐点卷积 2 部分，输入样本的尺寸为 H(行数) × W(列数) × N(通道数)。首先，深度卷积对输入向量的每个通道使用一个单独的 $k \times k \times 1$ 的卷积核进行卷积运算，经过 N 个深度卷积运算得到 N 个尺寸为 $H \times W$ 的向量。其次，逐点卷积完成对上一环节输出向量的通道数调整，使用 $1 \times 1 \times N$ 的卷积核进行卷积运算将其压缩为 $H \times W \times 1$ 的向量，经过 M 个逐点卷积运算得到 M 个尺寸为 $H \times W$ 的向量。最终组成 $H \times W \times M$ 的输出结果。

深度可分离卷积相对于常规卷积的优点是可以显著减少参数量。例如，对于一个 N 通道的输入向量，若想用 $k \times k$ 的常规卷积核去得到一个 M 通道的输出向量，需要的参数量为

$$a = k^2 NM \tag{1}$$

如果使用深度可分离卷积完成同样的工作，需

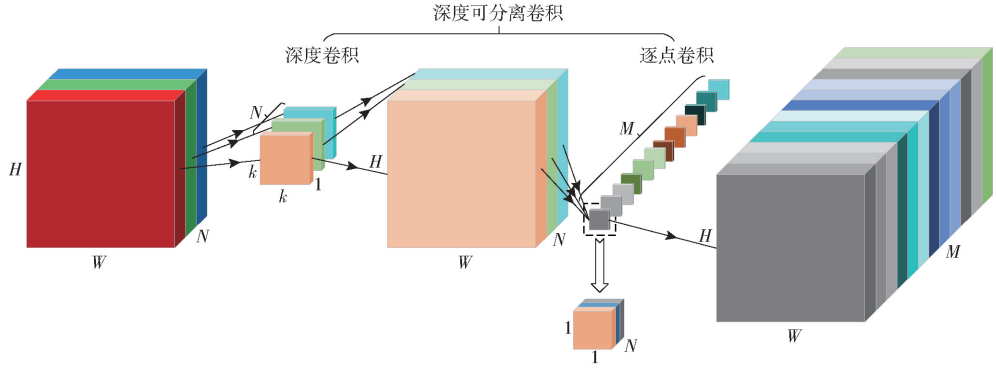


图 2　深度可分离卷积原理图

Fig. 2　Depthwise separable convolution principle

要的参数量为

$$b = k^2N + NM \tag{2}$$

两种情况下参数量的比值为

$$\frac{b}{a} = \frac{k^2N + NM}{k^2NM} = \frac{1}{M} + \frac{1}{k^2} \tag{3}$$

由此可见,当输出向量的通道数很大时,2 种卷积参数量的比值近似与卷积核尺寸的平方成反比。当卷积核尺寸为 3 ×3 时,使用深度可分离卷积最多可以将参数量减少到原来的 1/9。

1.3 改进结果

改进后的 Light - YOLOv3 网络结构如图 3a 所示。网络输入是 416 像素 ×416 像素 ×3 通道,输出是 13 ×13 和 26 ×26 两种尺寸的特征图。残差块结构如图 3b 所示,包含 1 个步长为 2 的卷积单元和 2 个残差单元。残差单元结构如图 3c 所示。文献[19]实验结果表明:当预激活结构采用 BN 层 + 激活层 + 卷积层的排列顺序时,包含该预激活结构的残差单元神经网络具有更好的收敛性、精度和泛化能力。因此,本文设计的网络中也采用 BN 层、激活层、卷积层的排列顺序。卷积单元具体结构如图 3d 所示,由 3 部分组成,每个部分均包含 1 个 BN 层、Leaky ReLU 激活函数和卷积层,卷积核的尺寸分别是 1 ×1、3 ×3 和 1 ×1,同样采用 BN 层 + 激活层 + 卷积层的排列顺序,当卷积单元用于下采样时,第 1 个卷积的步长为 2。

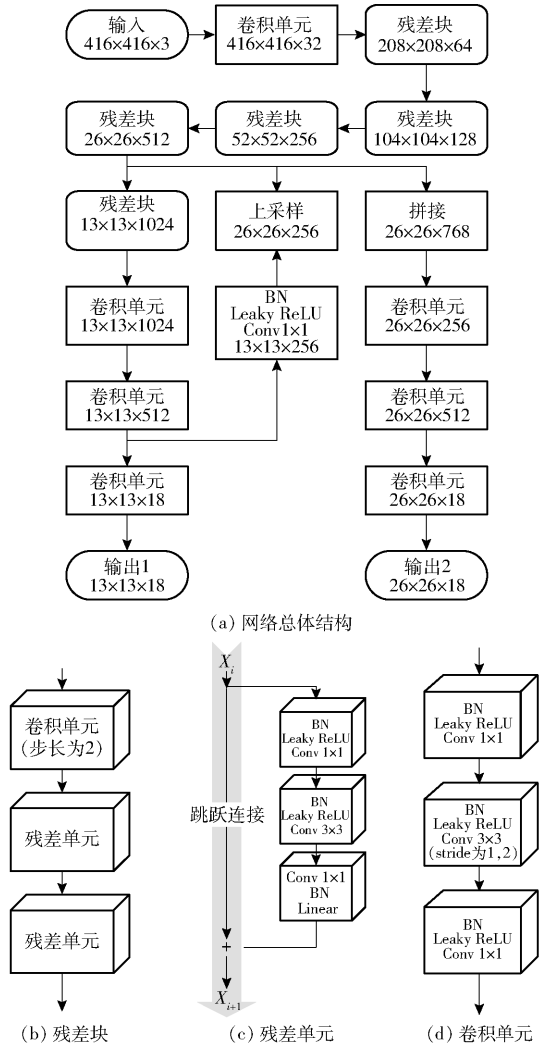


图 3 Light - YOLOv3 网络结构

Fig. 3 Light - YOLOv3 network structure

1.4 损失函数

损失函数 L 由 4 部分组成,分别是边界框定位损失值 L_{xy}、边界框尺寸损失 L_{wh}、置信度损失 L_{conf} 和类别损失 L_{cls},即

$$L = L_{xy} + L_{wh} + L_{\text{conf}} + L_{\text{cls}} \tag{4}$$

其中边界框定位损失 L_{xy} 使用均方误差损失函数表示为

$$L_{xy} = \sum_{i=0}^{S^2}\sum_{j=0}^{B} I_{ij}^{\text{obj}}[(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2] \tag{5}$$

式中 S^2——输入图像被划分网格数

B——单个网格预测边界框数,取值为 3

I_{ij}^{obj}——当第 i 个网格预测的第 j 个边界框检测到某个目标时,取值为 1,否则为 0

(x_i, y_i)——预测边界框中心点坐标

$(\hat{x}_i, \hat{y}_i)$——实际边界框中心点坐标

边界框尺寸损失 L_{wh} 使用均方误差损失函数表示为

$$L_{wh} = \sum_{i=0}^{S^2}\sum_{j=0}^{B} I_{ij}^{\text{obj}}[(w_i - \hat{w}_i)^2 + (h_i - \hat{h}_i)^2] \tag{6}$$

式中 w_i、h_i——预测边界框宽度、高度

$\hat{w}_i$、$\hat{h}_i$——实际边界框宽度、高度

置信度损失 L_{conf} 使用交叉熵损失函数表示为

$$L_{\text{conf}} = \lambda_{\text{obj}}\sum_{i=0}^{S^2}\sum_{j=0}^{B} I_{ij}^{\text{obj}}[-\hat{C}_i\ln C_i - (1-\hat{C}_i)\ln(1-C_i)] + \lambda_{\text{nobj}}\sum_{i=0}^{S^2}\sum_{j=0}^{B} I_{ij}^{\text{nobj}}[-\hat{C}_i\ln C_i - (1-\hat{C}_i)\ln(1-C_i)] \tag{7}$$

式中 λ_{obj}——权重系数,取值为 1

λ_{nobj}——权重系数,取值为 100,这样使得不包含目标的边界框产生更大的损失值,表明此时的模型误差较大

I_{ij}^{nobj}——当第 i 个网格预测的第 j 个边界框检测到某个目标时,取值为 0,否则为 1

C_i、$\hat{C}_i$——预测目标、实际目标的置信度

类别损失 L_{cls} 使用交叉熵损失函数表示为

$$L_{\text{cls}} = \sum_{i=0}^{S^2}\sum_{j=0}^{B}\sum_{c\in\text{class}} I_{ij}^{\text{obj}}[-\hat{p}_i(c)\ln(p_i(c)) - (1-\hat{p}_i(c))\ln(1-p_i(c))] \tag{8}$$

式中 c——检测到的目标所属类别

$\hat{p}_i(c)$——第 i 个网格检测到某个目标时,该

目标属于类别 c 的实际概率

$p_i(c)$——第 i 个网格检测到某个目标时，该目标属于类别 c 的预测概率

式(4)是单个尺度上的损失函数。本文改进的网络在两个尺度上进行检测，所以最终的损失函数值 L_{sum} 是在 2 个检测尺度上分别计算损失函数值相加得到的，即

$$L_{sum} = L_{13\times13} + L_{26\times26} \tag{9}$$

式中　$L_{13\times13}$——针对 13×13 特征图计算的损失函数值

$L_{26\times26}$——针对 26×26 特征图计算的损失函数值

2　数据集制作

目前网络上有大量苹果相关的图像，本文使用 Python 语言开发了图像爬虫对这些图像进行批量下载，减小了数据采集的成本，提高了数据采集的效率。

图像的主要来源为百度、Bing 和谷歌等网站，按照关键字“苹果”和“苹果树”分别保存一定数量的图像，然后进行人工筛选并去除重复图像，最终获得 815 幅图像，其中只包含单个苹果的图像 262 幅，包含复杂光照下的苹果图像 141 幅，存在树叶遮挡的苹果图像 150 幅，多个且不重叠苹果图像 69 幅，多个且存在重叠苹果图像 415 幅，未成熟苹果图像 47 幅。然后使用标注工具按 PASCAL VOC 数据集的标注格式对图像进行标注，生成 XML 类型的标注文件。

训练深度神经网络需要大量的数据，过小的数据集会导致神经网络过拟合，本文对采集到的数据集使用数据增强技术，增加数据集中的数据量，由此可以提高模型的泛化能力，提升模型的鲁棒性。

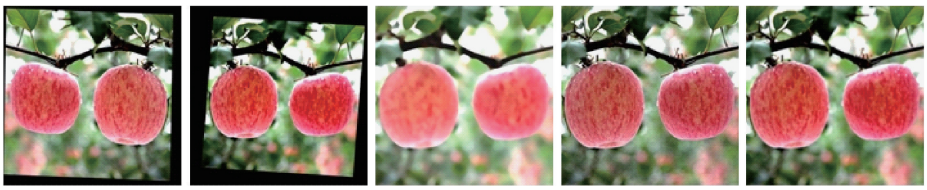
使用翻转、缩放、旋转、裁剪、平移、添加噪声和色彩抖动(包括对图像亮度、饱和度和对比度进行调整)7 种方法的随机组合对采集到的图像进行数据增强，原始图像和增强后图像分别如图 4 和图 5 所示，并对每幅图像对应的标注文件进行同步变换。

图 4　原始图像

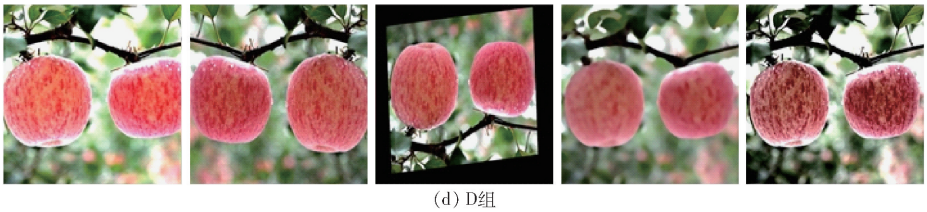
Fig. 4　Original image

(a) A组

(b) B组



(c) C组



(d) D组

图 5　数据增强后的图像

Fig. 5　Images after data augmentation

每幅图像生成 20 幅数据增强图像,最终有 2 幅图像数据增强失败,得到有效图像 17 113 幅。按照 8∶1∶1 的比例划分训练集 13 690 幅、验证集 1 712 幅与测试集 1 711 幅。

为了加快训练时网络的收敛速度以及防止梯度爆炸,对标注数据进行归一化处理,处理流程如图 6 所示。

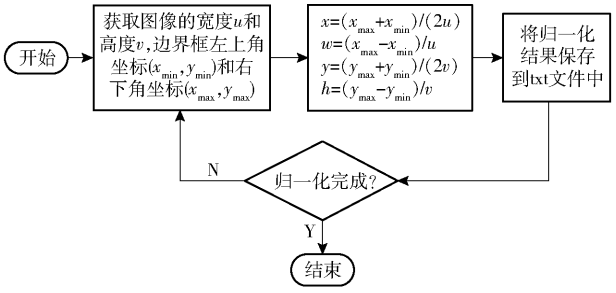


图 6 归一化流程图

Fig. 6 Normalized flowchart

将结果保存到 txt 格式的文件中,数据格式为 class_id x y w h,其中 class_id 表示类别编号,在本文使用的训练集中 class_id 为 0 表示类别为苹果;x、y、w 和 h 分别表示归一化后的目标物体边界框的中心点坐标、宽度和高度。

3 训练与实验

3.1 训练

使用 Pytorch 框架搭建网络,并在工作站上进行训练。工作站的配置为 i7 -9800X@3.80 GHz×16、内存 32 GB 和显存 11 GB 的 GeForce RTX 2080 Ti,使用的系统为 Ubuntu16.04,安装了 CUDA 和 cuDNN 库,Python 版本为 3.6,Pytorch 版本为 1.3。

训练前还需要在数据集上使用 K-means 算法进行聚类,得到预设的 6 个模板框尺寸,在本文制作的数据集上得到的结果为(77,77)、(131,127)、(194,189)、(261,254)、(395,395)、(693,667)。

训练时实际的批尺寸为 4,每迭代 2 次更新一次权值,等效的批尺寸为 8,总共训练 50 代,每代保存一次模型权重文件,每代迭代 3 423 次,共迭代 171 125 次,耗时 7.5 h。

训练时使用 SGD 和 Adam 2 种优化算法。经过实验发现,单独使用 SGD 优化算法时,学习率偏大会导致算法难以收敛,学习率偏小又会导致算法收敛很慢;而单独使用 Adam 优化算法时,最后训练的结果会比单独使用 SGD 时差[20],但优点是具有自适应学习率,算法收敛快。所以本文将训练过程分为 2 个阶段。第 1 阶段为前 30 代,使用的优化算法为 Adam,使用默认的初始化参数;第 2 阶段为后 20 代,使用的优化算法为 SGD,参考经验值并结合多次实验的结果,设置初始学习率为 0.000 1,动量参数为 0.9,每经过一代学习率减小为原来的 0.9。训练时第 2 阶段的学习率变化曲线如图 7 所示。

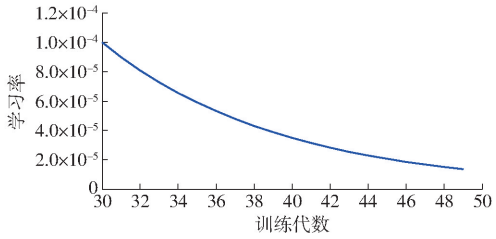


图 7 第 2 阶段学习率变化曲线

Fig. 7 Learning rate curve at the second stage

训练过程中每代完成后在验证集上对模型进行评估,计算 F1 值、mAP、准确率 P 和召回率 R 这 4 个指标,将这些数据保存至日志文件中,并使用 Tensorboard 软件对训练过程进行实时的监控。对于二分类问题,可以根据样本 S 的真实类别和模型预测类别的组合将样本 S 划分为 4 种类型:预测为正的正样本(True positive,TP),数量为 T_P;预测为负的正样本(False negative,FN),数量为 F_N;预测为正的负样本(False positive,FP),数量为 F_P;预测为负的负样本(True negative,TN),数量为 T_N[21]。

准确率 P 表示预测为正的所有样本中真正为正的样本所占的比例,计算公式为

$$P=\frac{T_P}{T_P+F_P}\times 100\% \tag{10}$$

召回率 R 表示真正为正的样本中被预测为正的样本所占的比例,计算公式为

$$R=\frac{T_P}{T_P+F_N}\times 100\% \tag{11}$$

F1 值(F_1)可以综合考虑准确率和召回率,是基于准确率和召回率的调和平均,定义为

$$F_1=\frac{2PR}{P+R}\times 100\%=\frac{2T_P}{S+T_P-T_N}\times 100\% \tag{12}$$

在目标检测中每个类别都可以根据准确率 P 和召回率 R 绘制 $P-R$ 曲线,AP 值就是 $P-R$ 曲线与坐标轴之间的面积,而 mAP 就是所有类别 AP 值的平均值。

3.2 训练数据分析

训练完成后,从日志文件中读取每一次迭代的损失值并绘制曲线,结果如图 8 所示。由图 8 可以看出,前 30 000 次迭代损失值迅速减小直至稳定,后面的训练过程中损失值在小范围内振荡。

根据日志中记录的数据绘制训练过程中 F1 值、mAP、准确率和召回率的变化曲线,如图 9 所示。由

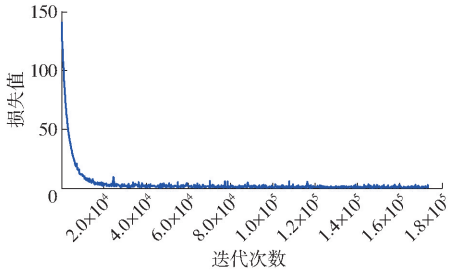


图 8　损失值变化曲线

Fig. 8　Loss value curve

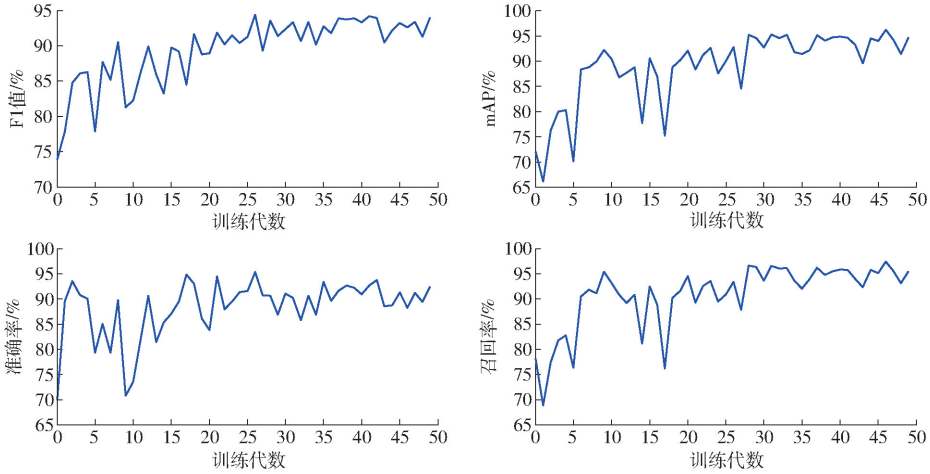


图 9　训练过程中各项指标变化曲线

Fig. 9　Curves of various indicators during training

本文主要考虑 F1 值和 mAP 这 2 个指标对训练结果进行评估，最终选取第 50 代保存的模型权重文件作为最终的训练结果，具有较高的 F1 值和 mAP，同时具有较高的准确率与召回率。

3.3　实验

3.3.1　工作站上对比实验

针对本文所提出的 YOLOv3 网络结构改进措施，分别进行单项改进措施的验证实验。将仅使用深度可分离卷积的改进网络记为网络 A，仅使用同构残差块串联的特征提取网络并简化特征图尺度的改进网络记为网络 B，使用相同的数据集与训练参数对 YOLOv3 网络、网络 A 和网络 B 进行训练，在测试集上分别进行检测，测试集中共包含 1 711 幅图像。对比每种网络的检测时间与各项性能指标，检测时间如表 1 所示，F1 值、mAP、准确率和召回率如表 2 所示。

表 1　网络检测时间对比

Tab. 1　Comparison of network detection time

网络	图像数/幅	总检测时间/ms	平均检测时间/ms	检测速度/(f·s^{-1})
YOLOv3	1 711	19 416. 17	11. 35	88. 11
网络 A	1 711	17 865. 04	10. 44	95. 79
网络 B	1 711	15 543. 83	9. 08	110. 13

图 9 可知，在前 30 代的训练过程中，各项指标变化幅度较大，但是总体趋势是增长的；在后 20 代的训练过程中，各项指标逐渐趋于稳定，在小范围内振荡。

F1 值最大是 94. 4%，对应第 27 代；mAP 最大值为 96. 2%，对应第 47 代；准确率最大值为 95. 4%，对应第 27 代；召回率最大值为 97. 4%，对应第 47 代。

表 2　网络性能指标对比

Tab. 2　Comparison of network performance indicators

%

网络	F1 值	mAP	准确率	召回率
YOLOv3	91. 56	92. 76	89. 16	94. 10
网络 A	92. 99	93. 93	91. 24	94. 80
网络 B	92. 82	94. 88	89. 94	95. 91

由表 1 和表 2 可知，相对于传统的 YOLOv3 网络，本文改进的网络 A 和网络 B 检测时间有较明显下降，准确率也稍有提高，实验数据验证了前文方法分析的正确性。在网络 A 中，使用深度可分离卷积替换普通卷积，减少了参与计算的卷积参数量，从而提高了网络的检测速度。在网络 B 中，针对单一目标检测的应用，使用同构残差块串联的特征提取网络，删除了生成 52 × 52 的特征图的相关网络结构，减少了参与计算的网络分支，从而提高了网络的检测速度。并且，由于网络结构的简化，一定程度上避免了复杂网络在训练阶段对样本数据的过拟合，提高了网络的泛化能力。因此，网络对验证集样本的检测准确率也稍有提高。

同时采用这 2 种措施改进 YOLOv3 网络的结构组成，得到结构精简的 Light - YOLOv3 网络。YOLOv3 网络与 Light - YOLOv3 网络训练得到的模型权重文件容量、参数量和理论计算量如表 3 所示，

其中 FLOPs 表示浮点运算次数。

表 3 网络模型权重对比

Tab. 3 Comparison of network model weight

网络	网络权重文件容量/MB	参数量	理论计算量(FLOPs)
YOLOv3	234.0	6.15×10^7	3.28×10^{10}
Light-YOLOv3	87.1	2.27×10^7	1.52×10^{10}

由表 3 可知,Light-YOLOv3 网络相比于 YOLOv3 网络,网络权重文件容量减小了 62.78%,参数量减少了 63.09%,理论计算量减小了 53.66%。

在工作站平台上,使用 2 种网络对整个测试集(包含 1 711 幅图像)分别进行检测,对比 2 种网络的检测时间与各项性能指标,检测时间如表 4 示,F1 值、mAP、准确率和召回率如表 5 所示。

表 4 工作站检测时间对比

Tab. 4 Comparison of detection time on workstation

网络	图像数/幅	总检测时间/ms	平均检测时间/ms	检测速度/(f·s^{-1})
YOLOv3	1 711	19 416.17	11.35	88.11
Light-YOLOv3	1 711	14 636.91	8.55	116.96

表 5 工作站网络性能对比

Tab. 5 Comparison of network performance on workstation %

网络	F1 值	mAP	准确率	召回率
YOLOv3	91.56	92.76	89.16	94.10
Light-YOLOv3	94.57	94.69	93.64	95.51

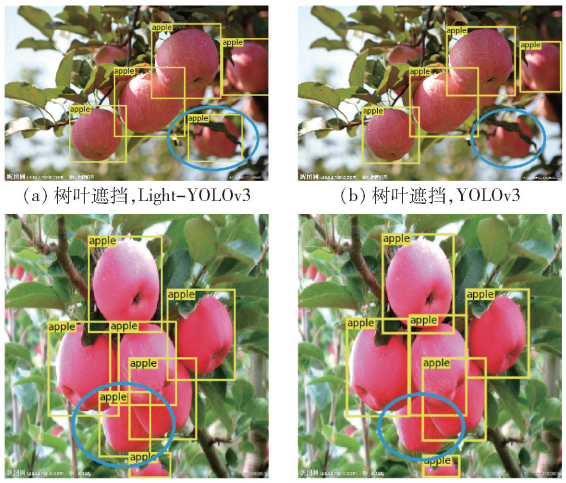
由表 4 和表 5 可知,Light-YOLOv3 网络相比较于 YOLOv3 网络,各项性能指标均有提高,同时检测速度更快。Light-YOLOv3 网络 F1 值提高了 3.01 个百分点,mAP 提高了 1.93 个百分点,准确率提高了 4.48 个百分点,召回率提高了 1.41 个百分点,单幅图像的平均检测时间减少了 24.67%,检测速度提高了 32.74%。

检测效果对比如图 10 所示。可以看到 Light-YOLOv3 网络具有更好的检测效果,能够检测图 10 中蓝色圆圈所包含的被遮挡的苹果。

3.3.2 嵌入式开发板上对比实验

最后在 Nvidia TX2 嵌入式开发板上进行实验,使用的系统为 Ubuntu 18.04,安装了 CUDA 和 cuDNN 库,Python 版本为 3.6,Pytorch 版本为 1.3,并使用 TensorRT 库实现神经网络推理加速。

在 Nvidia TX2 上,使用 2 种网络对整个测试集(包含 1 711 幅图像)分别进行检测,对比 2 种网络的检测时间,结果如表 6 所示。



(a) 树叶遮挡,Light-YOLOv3 (b) 树叶遮挡,YOLOv3

(c) 苹果间相互遮挡,Light-YOLOv3 (d) 苹果间相互遮挡,YOLOv3

图 10 网络检测效果对比

Fig. 10 Comparison of network detection effect

表 6 嵌入式开发板检测时间对比

Tab. 6 Comparison of detection time of embedded system

网络	图像数/幅	总检测时间/ms	平均检测时间/ms	检测速度/(f·s^{-1})
YOLOv3	1 711	319 207.88	186.56	5.36
Light-YOLOv3	1 711	225 286.53	131.67	7.59

由表 6 可知,Light-YOLOv3 网络在嵌入式开发板上运行时单幅图像平均检测时间为 131.67 ms,相比 YOLOv3 网络减少了 29.42%,在嵌入式开发板上检测速度可以达到 7.59 f/s,提高了 41.60%。

4 结论

(1)在分析 YOLOv3 网络结构的基础上,设计了由 5 个相同结构的残差块串联组成的特征提取网络,在 2 个尺度上进行苹果检测,并采用深度可分离卷积替换普通卷积,采用均方误差损失和交叉熵损失组成的复合损失函数,减少了参数量和降低了计算量。

(2)采集数据对网络进行训练,将训练过程分为 2 个阶段,分别使用 SGD 和 Adam 优化算法,加快了模型收敛,并获得较好的训练结果。分别在工作站和嵌入式开发板上进行实验,结果表明,改进后参数量减少了 63.09%;在工作站上单幅图像的平均检测时间为 8.55 ms,在嵌入式开发板上的平均检测时间为 131.67 ms,比改进前分别减少了 24.67% 和 29.42%;在工作站上的检测速度为 116.96 f/s,在嵌入式处理器上的检测速度为 7.59 f/s,比改进前分别提高了 32.74% 和 41.60%;在测试集上的 mAP 为 94.69%,F1 值为 94.57%,比改进前分别提高了 1.93 个百分点和 3.01 个百分点。

参 考 文 献

[1] 田娜,杨晓文,单东林,等. 我国数字农业现状与展望[J]. 中国农机化学报,2019,40(4):210 - 213.
TIAN Na, YANG Xiaowen, SHAN Donglin, et al. Status and prospect of digital agriculture in China[J]. Journal of Chinese Agricultural Mechanization, 2019, 40(4):210 - 213. (in Chinese)

[2] 刘晓洋,赵德安,贾伟宽,等. 基于超像素特征的苹果采摘机器人果实分割方法[J/OL]. 农业机械学报,2019,50(11):15 - 23.
LIU Xiaoyang, ZHAO Dean, JIA Weikuan, et al. Fruits segmentation method based on superpixel features for apple harvesting robot[J/OL]. Transactions of the Chinese Society for Agricultural Machinery, 2019, 50(11):15 - 23. http://www.j-csam.org/jcsam/ch/reader/view_abstract.aspx? flag = 1&file_no = 20191102&journal_id = jcsam. DOI:10.6041/j.issn.1000-1298.2019.11.002. (in Chinese)

[3] 王丹丹,宋怀波,何东健. 苹果采摘机器人视觉系统研究进展[J]. 农业工程学报,2017,33(10):59 - 69.
WANG Dandan, SONG Huaibo, HE Dongjian. Research advance on vision system of apple picking robot[J]. Transactions of the CSAE, 2017, 33(10): 59 - 69. (in Chinese)

[4] 刘继展. 温室采摘机器人技术研究进展分析[J/OL]. 农业机械学报,2017,48(12):1 - 18.
LIU Jizhan. Research progress analysis of robotic harvesting technologies in greenhouse[J/OL]. Transactions of the Chinese Society for Agricultural Machinery, 2017, 48(12): 1 - 18. http://www.j-csam.org/jcsam/ch/reader/view_abstract.aspx? flag = 1&file_no = 20171201&journal_id = jcsam. DOI:10.6041/j.issn.1000-1298.2017.12.001. (in Chinese)

[5] 武星,张颖,李林慧,等. 复杂光照条件下视觉导引 AGV 路径提取方法[J/OL]. 农业机械学报, 2017,48(10):15 - 24.
WU Xing, ZHANG Ying, LI Linhui, et al. Path extraction method of vision-guided AGV under complex illumination conditions [J/OL]. Transactions of the Chinese Society for Agricultural Machinery, 2017, 48(10):15 - 24. http://www.j-csam.org/jcsam/ch/reader/view_abstract.aspx? flag = 1&file_no = 20171002&journal_id = jcsam. DOI:10.6041/j.issn.1000-1298.2017.10.002. (in Chinese)

[6] LI C, SONG D, TONG R, et al. Illumination-aware faster R - CNN for robust multispectral pedestrian detection[J]. Pattern Recognition, 2019, 85: 161 - 171.

[7] WU X, SUN C, ZOU T, et al. Intelligent path recognition against image noises for vision guidance of automated guided vehicles in a complex workspace[J]. Applied Sciences, 2019, 9(19): 4108.

[8] 赵德安,刘晓洋,陈玉,等. 苹果采摘机器人夜间识别方法[J/OL]. 农业机械学报,2015,46(3):15 - 22.
ZHAO Dean, LIU Xiaoyang, CHEN Yu, et al. Image recognition at night for apple picking robot[J/OL]. Transactions of the Chinese Society for Agricultural Machinery, 2015, 46(3):15 - 22. http://www.j-csam.org/jcsam/ch/reader/view_abstract.aspx? flag = 1&file_no = 20150303&journal_id = jcsam. DOI:10.6041/j.issn.1000-1298.2015.03.003. (in Chinese)

[9] JI W, ZHAO D, CHENG F, et al. Automatic recognition vision system guided for apple harvesting robot[J]. Computers & Electrical Engineering, 2012, 38(5): 1186 - 1195.

[10] 崔淑娟,李健. 基于色差信息的成熟苹果识别[J]. 西北大学学报(自然科学版),2011,46(6):993 - 997.
CUI Shujuan, LI Jian. The identification of mature apple based on the chromatic aberration[J]. Journal of Northwest University (Natural Science Edition), 2011, 46(6):993 - 997. (in Chinese)

[11] LINKER R, COHEN O, NAOR A. Determination of the number of green apples in RGB images recorded inorchards[J]. Computers & Electronics in Agriculture, 2012, 81:45 - 57.

[12] 彭红星,黄博,邵园园,等. 自然环境下多类水果采摘目标识别的通用改进 SSD 模型[J]. 农业工程学报,2018,34(16):155 - 162.
PENG Hongxing, HUANG Bo, SHAO Yuanyuan, et al. General improved SSD model for picking object recognition of multiple fruits in natural environment[J]. Transactions of the CSAE, 2018, 34(16):155 - 162. (in Chinese)

[13] TIAN Yunong, YANG Guodong, WANG Zhe, et al. Apple detection during different growth stages in orchards using the improved YOLO - V3 model[J]. Computers and Electronics in Agriculture, 2019, 157: 417 - 426.

[14] 王丹丹,何东健. 基于 R - FCN 深度卷积神经网络的机器人疏果前苹果目标的识别[J]. 农业工程学报,2019,35(3):156 - 163.
WANG Dandan, HE Dongjian. Recognition of apple targets before fruits thinning by robot based on R - FCN deep convolution neural network[J]. Transactions of the CSAE, 2019, 35(3): 156 - 163. (in Chinese)

[15] REDMON J, FARHADI A. YOLO V3: an incremental improvement[J]. arXiv Preprint arXiv:1804.02767, 2018.

[16] HOWARD A G, ZHU M, CHEN B, et al. Mobilenets: efficient convolutional neural networks for mobile vision applications [J]. arXiv Preprint arXiv:1704.04861, 2017.

[17] HE K, ZHANG X, REN S, et al. Deep residual learning for image recognition[C]// Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016:770 - 778.

[18] IOFFE S, SZEGEDY C. Batch normalization: accelerating deep network training by reducing internal covariate shift[C]// International Conference on Machine Learning, 2015: 448 - 456.

[19] HE K, ZHANG X, REN S, et al. Identity mappings in deep residual networks[C]// European Conference on Computer Vision. Springer, Cham, 2016: 630 - 645.

[20] KESKAR N S, SOCHER R. Improving generalization performance by switching from Adam to SGD [J]. arXiv Preprint arXiv: 1712.07628, 2017.

[21] 周志华. 机器学习[M]. 北京:清华大学出版社,2016:30.