DOI: 10.13382/j.jemi.B2104724

# 基于改进 YOLO 及 NMS 的水果目标检测*

徐印赟[1,2] 江 明[1,2] 李云飞[1,2] 吴云飞[1,2] 卢桂馥[1,3]

(1. 安徽工程大学高端装备先进感知与智能控制教育部重点实验室 芜湖 241000;2. 安徽工程大学电气工程学院 芜湖 241000;3. 安徽工程大学计算机与信息学院 芜湖 241000)

**摘 要**:为使水果采摘机器人在复杂情况下如树叶遮挡、果实目标尺度变化大等情况能准确地检测出水果,提出一种 YOLO(you only look once)改进模型与 NMS(non-maximum suppression)改进算法的目标检测方法。首先,对传统 YOLO 深度卷积神经网络架构进行改进,设计一种更细化的 SPP5(spatial pyramid pooling)特征融合网络模块,强化特征图多重感受野信息的融合,并基于此模块提出一种 YOLOv4-SPP2-5 模型,在标准 YOLOv4 网络中跨层添加并改进 SPP 层,重新分布池化核大小,增强感受野范围,从而降低目标误检率;其次,提出一种 Greedy-Confluence 的 NMS 改进算法,通过对高度接近的检测框直接抑制和对重叠检测框综合考虑距离交并比 DIOU(distance-intersection over union)和加权接近度 WP(weighted proximity)的方法,均衡 NMS 的计算消耗并减少检测框的错误抑制,从而提高遮挡、重叠物体的检测精度;最后,分别对改进方法进行性能测试,验证方法的可行性,随后制作水果检测数据集并进行格式转换和标签标注,然后采用数据增强技术对训练数据进行扩充,并使用 K-means++聚类方法获取先验锚定框,在计算机上进行了水果检测实验。结果表明,基于改进 YOLO 网络及改进 NMS 的水果检测方法在准确率方面有显著的提高,平均精度均值(mean average precision,MAP) 在 YOLOv4 上达到了 96.65%,较原网络提升 1.70%,并且实时性也得到了保证,在测试设备上达到了 39.26 帧/s。

**关键词**:水果目标检测;YOLO 网络;SPP 模块;NMS;信息熵

**中图分类号**:TP242 TP391.4 TH89 **文献标识码**:A **国家标准学科分类代码**:510.4050

# Fruit target detection based on improved YOLO and NMS

Xu Yinyun[1,2] Jiang Ming[1,2] Li Yunfei[1,2] Wu Yunfei[1,2] Lu Guifu[1,3]

(1. Key Laboratory of Advanced Perception and Intelligent Control of High-End Equipment, Ministry of Education, Anhui Polytechnic University, Wuhu 241000, China; 2. School of Electrical Engineering, Anhui Polytechnic University, Wuhu 241000, China; 3. School of Computer and Information, Anhui Polytechnic University, Wuhu 241000, China)

**Abstract**: To enable fruit picking robots that accurately detect target in complex conditions such as leaf covering and variances of fruit sizes, etc., improved YOLO(you only look once) model and NMS(non-maximum suppression) algorithm are proposed. First, the traditional YOLO deep convolutional neural network architecture is upgraded. A more fine-grained SPP5(spatial pyramid pooling) feature fusion network module is generated to enhance the integration of multiple sensory field information in feature maps, based on which a YOLOv4-SPP2-5 model is proposed. The SPP layer is added and improved in the standard YOLOv4 network across layers and redistributed the pooling kernel size and to enlarge perceptual field range, thus decreasing the false detection rate. Moreover, an improved Greedy-Confluence NMS algorithm is proposed. Through direct suppression of high-proximity detection boxes and comprehensive consideration of Distance-Intersection over Union (DIOU) and weighted proximity (WP) for overlapping detection boxes, the computational consumption of NMS was balanced and the error suppression of detection boxes was reduced, so as to improve the detection accuracy of occlusion and overlapping objects. Finally, performance tests are conducted to verify the feasibility of the method, followed by format converting and annotation labelling with fruit training datasets. The training datasets are expanded via data augmentation techniques and the K-means ++ clustering approach is utilized to obtain a priori anchor frames, and the fruit detection

收稿日期:2021-09-11 Received Date:2021-09-11

*基金项目:国家自然科学基金(61976005)项目资助

experiments are carried out on a computer. The results demonstrate that the improved YOLO network and NMS algorithm-based approach significantly increase the accuracy rate of fruit detection. The mean average precision (MAP) reaches 96.65% at YOLOv4, which is 1.70% higher than the previous network. Real-time performance is also guaranteed, hitting 39.26 frames per second on the test device.
**Keywords**: fruit target detection; YOLO network; SPP block; NMS; information entropy

## 0 引　言

目标检测问题的定义是确定目标在给定图像中的位置(目标定位)以及每个目标所属的类别(目标分类)[1]。当前,目标检测技术已经在自动化生产、医疗、农业中得到重要应用[2],如红外目标检测[3]、缺陷检测[4]等。目标检测的主流方法采用深度学习的方法。基于深度学习的目标检测方法主要分为两类。一类是基于候选区域的双阶段目标检测算法,典型方法包括 R-CNN[5]、SPP-Net[6]、Fast R-CNN[7]、R-FCN[8]、Mask R-CNN[9]、Cascade R-CNN[10]等;另一类为基于回归的单阶段目标检测算法,主要包括 SSD[11]、YOLO[12]、RetinaNet[13]、EfficientDet[14]等。由于双阶段算法需要生成大量候选区域,因此相对消耗时间长,单阶段算法在实际应用中更受欢迎[15]。

水果检测作为目标检测技术的一种,可以应用于水果采摘、品质检测、分类、成熟度鉴定和病虫害检测等。当前,由于算力资源、遮挡情况、光照条件、目标尺度相差较大等复杂场景因素影响,目标检测任务很难实现完美应用[16-17]。近年来,这一领域的研究也取得了一些进展。文献[18]提出了使用苹果不同时期的图像并进行数据增强以及利用 DenseNet 模块改进 YOLOv3 模型,用于检测果园不同生长期的苹果;文献[19]提出一种利用 SE_ResGNet34 结构替换 DarkNet 结构的改进 YOLOv3 模型用于自然环境下柠檬检测;文献[20]提出一种利用 MobileNet 结构替换 DarkNet 结构的改进 YOLOv3 模型用于果园火龙果检测;文献[21]提出一种利用 EfficientNet 结构替换 CSPDarknet53 结构的改进 YOLOv4 模型和树叶插图数据增强方法用于复杂情况下苹果检测;文献[22]提出一种包含 DenseNet 结构和用圆形边界框(C-Bbox)代替传统的矩形边界框(R-Bbox)的 YOLOv3 改进方法进行番茄检测与定位。

特征金字塔(feature pyramid networks, FPN)[23]是解决目标检测多尺度变化的常用方法,近年来,很多研究人员对此方法提出了很多其他的派生改进方法,文献[24]提出了一种尺度归一化方法(SNIP),通过选择性的训练每个图像尺度上合适大小的目标,增强了特征金字塔的多尺度检测能力;文献[25]提出了一种利用神经结构搜索(NAS)的技术来搜索相对较好的结构(NAS-FPN);文献[26]提出了一种自适应空间特征融合(ASFF)的金字塔特征融合策略,并在 YOLOv3 上获得了不错的结果;文献[27]提出了一种在 FPN 的基础上增加一路自下而上的增强层(PANet),提升特征金字塔架构的能力。此外,在其他方面,文献[28]提出了一种 YOLOv3-SPP3 的模型结构,提升多重感受野的信息融合能力。

目标检测在运行预测任务过后,会得到很多的检测框,即使是同一个物体也会产生很多检测框,最终在大量检测框中选出最优的检测框,通过非极大值抑制(NMS)的方法来完成。非极大值抑制通过递归地计算检测框之间的重叠程度(intersection over union, IOU)来判断是否将检测框抑制。但是在遇到目标被遮挡或重叠的情况时,如水果检测任务中树叶遮挡和果实簇,此时算法会错误的将遮挡物体抑制而造成漏检。文献[29]通过将所有框的得分衰减成与此框重叠程度的连续函数,来减少检测框的误抑制,文献[30]考虑两个检测框之间的中心点位置信息,采用 DIOU 的计算方法替换传统 IOU 来减少误抑制,文献[31]提出一种采用归一化曼哈顿距离替换 IOU 计算,然后根据曼哈顿距离聚类梯度删选检测框的方法。

水果检测以上问题的本质是当前目标检测算法的问题。要想解决以上问题,首先需要提升当前目标检测算法的多尺度检测能力以及对重叠、遮挡物体的检测性能。本文首先对通用的目标检测算法进行改进,提出了一种 YOLOv4-SPP2-5 的 YOLO 改进模型,旨在增强 YOLOv4[32]网络的多尺度检测能力;同时为了解决目标重叠、遮挡时 NMS 过程检测框错误抑制造成漏检的问题,提出了一种 Greedy-Confluence 的检测框抑制方法,并首先在公开数据集上验证,同时与原算法比较,证明本文的方法能够提升当前目标检测算法的多尺度检测精度,降低误抑制并均衡测试时间。然后将本文的改进方法在水果检测任务上根据任务特点作细节改进,制作水果检测数据集并进行数据增强和锚定框重定,并通过实际实验验证改进后的方法性能。

## 1 改进 YOLO 模型

### 1.1 传统 YOLO 模型

YOLOv3[33]使用 DarkNet53 作为模型的基本结构,主体结构由数个残差单元组成,YOLOv3 在多个尺度上进行了目标的位置估计和类别检测,通过将低分辨率的特征图上采样并进行跨尺度拼接,借鉴 FPN[23]的融合方

式,形成了3个尺度的特征图检测模型,当输入图片设为416×416大小时,3个检测层的特征图大小分别为13×13,26×26,52×52。YOLOv4是基于YOLOv3的改进模型。YOLOv4使用CSPDarkNet53替换DarkNet53作为模型的基本结构,并在3个检测层前引入PANet结构,进一步加强模型的多尺度检测能力。

### 1.2　改进的YOLO模型

随着卷积神经网络的网络层次加深,深层特征图的图像信息高度抽象,造成小目标检测的精度较差。SPP[6]模块结构可以实现多尺度局部特征和全局特征的融合,丰富特征图的表达能力,从而提高小目标的检测精度。SPP模块结构如图1所示。

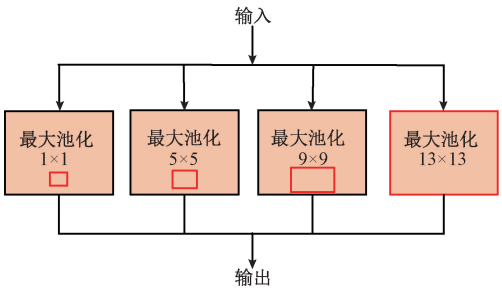



图1　SPP模块结构

Fig. 1　SPP module structure

在水果检测任务中,小目标较多且目标尺度变化大。为了解决以上问题,本文对SPP模块和YOLO模型网络结构进行重新设计,提出一种更加细化的SPP5模块,如图2所示,将池化核大小细化为1×1,4×4,7×7,10×10,13×13,增强感受野范围。并基于此模块设计一种YOLOv4-SPP2-5模型,如图3所示,将YOLOv4模型的第1个SPP模块替换为细化的SPP5模块,并在跨层连接处增加第2个SPP模块,增加特征图多尺度感受野信息的融合。如图3所示。为了便于比较,将YOLOv3也进行相同的改进,并调整第2个SPP模块的池化核大小为1×1,18×18,35×35,52×52。

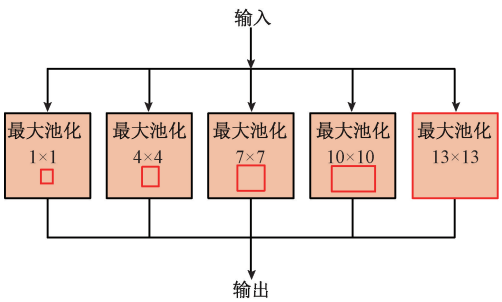



图2　SPP5模块结构

Fig. 2　SPP5 module structure

信息论可知:信息熵可以作为信息的量化度量的参数[34]。假设$H(p)$表示神经网络添加SPP模块后的输出特征图的信息熵,用来度量神经网络经过SPP模块后输出的图像特征信息量的期望。根据香农信息量定义[35]:

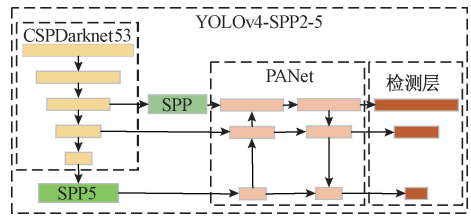



图3　改进模型网络结构

Fig. 3　Network structure diagram of improved model

$$h(x_0) = -\log p(x_0) = \log \frac{1}{p(x_0)} \tag{1}$$

其中,$h(x_0)$表示随机事件$X = x_0$的信息量,$p(x)$为随机事件$X = x_0$的概率分布函数$p(x) = \Pr(X = x)$,$x \in X$。

则信息熵为:

$$H(p) = E[I(x_i)] = -\sum_{i=1}^{n} p(x_i)\log p(x_i) = \sum_{i=1}^{n} p(x_i)\log \frac{1}{p(x_i)}, (i = 1,2,\cdots n) \tag{2}$$

当输入图片为416×416时,第1个SPP5模块前特征图的大小为13×13,由此,为了方便计算,假设$\boldsymbol{A}$为SPP模块前的特征图的二值像素矩阵,假定每一个像素点的特征信息均不相同,即特征图上每个像素的值均不相等,设特征图的像素值为:

$$\boldsymbol{A} = \begin{bmatrix} a_1 & a_2 & \cdots & a_{13} \\ a_{14} & a_{15} & \cdots & a_{26} \\ \vdots & \vdots & \ddots & \vdots \\ a_{157} & a_{158} & \cdots & a_{169} \end{bmatrix} \tag{3}$$

其中,为了简化计算,假设$a_1 < a_2 < \cdots < a_{169}$,在YOLO中池化操作采取padding操作即填充0以实现输入输出的特征图维度相同,因此可以得出:用4×4的池化核进行最大值池化操作后,输出特征图的像素值形如:

$$\boldsymbol{A}_1 = \begin{bmatrix} a_{43} & a_{44} & \cdots & \overbrace{a_{52}}^{4} & \cdots & a_{52} \\ a_{56} & a_{57} & \cdots & a_{65} & \cdots & a_{65} \\ \vdots & \vdots & \ddots & \vdots & \cdots & \vdots \\ a_{160} & a_{160} & \cdots & a_{160} & \cdots & a_{160} \\ a_{160} & a_{160} & \cdots & a_{160} & \cdots & a_{160} \\ a_{160} & a_{160} & \cdots & a_{160} & \cdots & a_{160} \\ a_{160} & a_{160} & \cdots & a_{160} & \cdots & a_{160} \end{bmatrix} \tag{4}$$

则很容易得出特征图经过池化操作后图像特征信息

熵为：

$$H(p,k) = (S-k)^2 \times \left(-\frac{1}{S^2}\log\frac{1}{S^2}\right) + 2 \times (S-k) \times \left(-\frac{k}{S^2}\log\frac{k}{S^2}\right) + \left(-\frac{k^2}{S^2}\log\frac{k^2}{S^2}\right) \tag{5}$$

其中，$S$ 为输入特征图大小，$k$ 为池化核大小，则原 SPP 模块之后的图像特征信息熵为：

$$H(p) = H(p,k=1) + H(p,k=5) + H(p,k=9) + H(p,k=13) \tag{6}$$

而 SPP5 模块之后的图像特征信息熵为：

$$H(p)_1 = H(p,k=1) + H(p,k=4) + H(p,k=7) + H(p,k=10) + H(p,k=13) \tag{7}$$

经过计算，可知 $H(p)_1 > H(p)$，神经网络经过 SPP5 模块后产生的融合特征图的信息熵比 SPP 模块后的要大，即经过 SPP5 模块后神经网络所蕴含的图像特征信息更加丰富，所包含的信息更加复杂，在图像上表现为细节特征更多。应当指出，当一张图像所包含的信息越复杂，其二值矩阵越接近以上推理所假定的理想情况，SPP5 模块的效果理论上应当更明显。

同理，如图 2 所示，假设第 2 个 SPP 模块后产生的特征图的信息熵为 $H(p)_2$，神经网络前向直连传递层分支产生的特征图的信息熵为 $H(p)_3$，由信息熵 $H(p) \geqslant 0$，故在跨层连接处，有：

$$H(p)_2 + H(p)_3 \geqslant H(p)_3 \tag{8}$$

因此增加 SPP 模块，显然增加了特征图信息的融合。

## 2　改进非极大值抑制算法

### 2.1　传统的 Greedy-NMS

在水果检测过程中，运行一次检测任务的结果如图 4 所示，其中包含大量的冗余检测框，传统的目标检测任务对大量冗余检测框进行去除操作是通过 NMS 算法来完成的。标准的 Greedy-NMS 算法对所得检测框进行按类别划分，并对每个类别的检测框按从高到低的结果排序得到一个降序的列表，然后每个列表递归的选取得分最高的检测框并删除那些与此检测框重叠度即 $iou$ 大于阈值 $N_t$ 的检测框来达到抑制效果。$iou = \dfrac{insection}{union}$ 为两个检测框的交集与并集之比。

传统 Greedy-NMS 可以定义为：

$$S_i = \begin{cases} S_i, iou(M,b_i) < N_t \\ 0, iou(M,b_i) > N_t \end{cases} \tag{9}$$

其中，$iou$ 表示列表中最高得分所对应的检测框 $M$ 和列表中其他检测框 $b_i$ 之间的交并比。

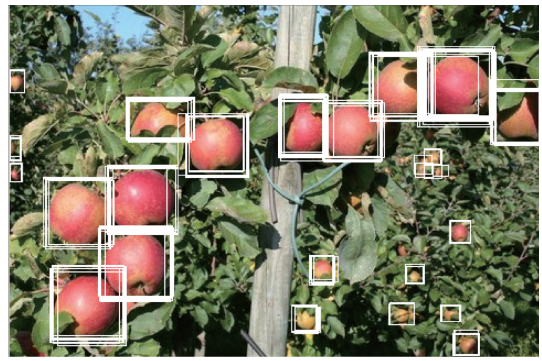


图 4　目标检测任务结果（未进行 NMS）

Fig. 4　Object detection task results (no NMS)

### 2.2　DIOU-NMS

DIOU-NMS[30] 在 NMS 过程中考虑了两框中心点位置距离的信息，即采用 DIOU 的计算方式替换了传统的 IOU，DIOU-NMS 可以定义为：

$$S_i = \begin{cases} S_i, IOU - R_{diou}(M,b_i) < N_t \\ 0, IOU - R_{diou}(M,b_i) > N_t \end{cases} \tag{10}$$

其中，$R_{diou} = \dfrac{\rho^2(b,b^{gt})}{c^2}$，$b,b^{gt}$ 表示两个检测框 $M,b_i$ 的中心点，$\rho$ 表示欧氏距离，$c$ 表示包含两个检测框的最小封闭框的对角线长度。

### 2.3　Confluence

Confluence[31] 在进行检测框抑制时，不根据检测框之间的 IOU 或 DIOU 去删除冗余的框，使用曼哈顿距离，在一个簇中选取和其他框都是距离最近的那个框，然后去除与此框加权曼哈顿距离低于一个阈值的框。

曼哈顿距离为所有点的水平和垂直距离的和，两点 $u=(x_1,y_1)$ 和 $v=(x_2,y_2)$ 之间的曼哈顿距离表示为：

$$MH_{(u,v)} = |(x_1 - x_2)| + |(y_1 - y_2)| \tag{11}$$

如图 5 所示，两个框之间的接近程度可以表示为左上角点和右下角点的曼哈顿距离的和：

$$P = P_{(u,v,m,n)} = MH_{(u,v)} + MH_{(m,n)} \tag{12}$$

即为：

$$P = |(x_1 - p_1)| + |(x_2 - p_2)| + |(y_1 - q_1)| + |(y_2 - q_2)| \tag{13}$$

$P$ 越小表示接近程度越高，在目标检测中就表示两个检测框越可能表示同一个物体。但由于每个框的尺寸不一，为了使任意一对检测框的曼哈顿距离能够与其他检测框进行比较，因此 Confluence 对检测框进行归一化，如图 5 所示的坐标：

$$X = \{x_1, x_2, p_1, p_2\}$$
$$Y = \{y_1, y_2, q_1, q_2\} \tag{14}$$

其归一化公式为：

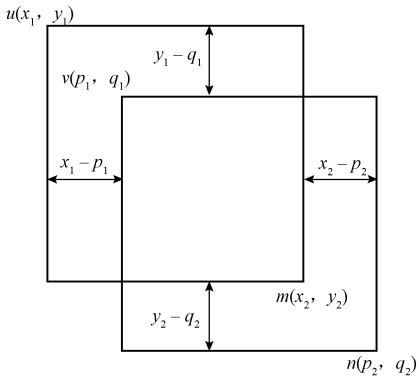



图5　曼哈顿距离表示图

Fig. 5　Manhattan distance representation diagram

$$norm(x_i, y_i) = \left(\frac{x_i - \min(X)}{\max(X) - \min(X)}, \frac{y_i - \min(Y)}{\max(Y) - \min(Y)}\right) \tag{15}$$

将所有的坐标归一化到0~1之后,两个相交的检测框之间的接近度会小于2,Confluence将两个框接近度$P$值小于2的归为同一个簇,然后在簇内获取最优的簇内检测框。

Confluence在进行检测框递归抑制时同时考虑物体的置信度得分$c$和$P$值,以一个加权的接近度代入计算,即:

$$WP = WP_{(u,v,m,n)} = \frac{P_{(u,v,m,n)}}{c} \tag{16}$$

Confluence可以定义为:

$$S_i = \begin{cases} 0, & P < 2 \&\& WP < \varepsilon \\ S_i, & P > 2 \,||\, WP > \varepsilon \end{cases} \tag{17}$$

其中,$P$表示两个框之间的归一化曼哈顿距离,$WP$表示加权归一化曼哈顿距离,$\varepsilon$表示接近度阈值。

### 2.4　改进的NMS:Greedy-Confluence

在原YOLOv4框架中,使用传统Greedy-NMS或者DIOU-NMS,传统Greedy-NMS由于贪婪的采用IOU作为判断是否删除检测框的标准,在遇到物体重合或者遮挡情况时,容易出现误抑制。而DIOU-NMS耗时较长,并且其性能在重叠、遮挡情况下依旧需要提高。而在水果检测任务中,水果受到遮挡和重叠的情况较多,为了解决这一问题,本文基于Confluence的思想,力求方便地插入到YOLO网络中,提出一种Greedy-Confluence策略,借鉴Greedy-NMS的贪婪思维,对所有检测框按类别分类并为每个类别按得分将检测框进行排序分别得到一个降序的列表,然后每个列表递归地选取得分最高的检测框并删除那些与此检测框曼哈顿距离$P$小于阈值$\varepsilon_1$的检测框,再将剩下的检测框计算$WP$和DIOU,再递归地将$WP$小于阈值$\varepsilon_2$且DIOU小于阈值$N_t$的检测框进行删除。

根据文献[31]可知,同一物体的冗余检测框十分密集,因此冗余检测框之间曼哈顿距离也相对很小。利用这一特性,本文直接剔除高度密集的检测框,减少计算时间并且保证性能优良。水果检测NMS过程的难点在于遮挡目标和重合目标的检测,本文将剩下的少量检测框综合考虑WP和DIOU,结合两者的优点,从而有效减少错误的抑制。本文算法沿用Greedy-NMS的贪婪思维,因此可以方便地插入到YOLO框架中而不增加实现负担。本文的Greedy-Confluence可以定义为:

$$S_i = \begin{cases} 0, & P < \varepsilon_1 \\ 0, & P \geqslant \varepsilon_1 \,\&\&\, WP < \varepsilon_2 \,\&\&\, D_{diou} \leqslant N_t \\ S_i, & P \geqslant \varepsilon_1 \,\&\&\, (WP \geqslant \varepsilon_2 \,||\, D_{diou} \leqslant N_t) \end{cases} \tag{18}$$

其中,$D_{diou} = IOU - R_{diou}(M, b_i)$,$P$表示两个框之间的曼哈顿距离,$WP$表示加权曼哈顿距离,$\varepsilon_1$,$\varepsilon_2$表示接近度阈值,$N_t$表示DIOU阈值,$\varepsilon_1 < 2$。

## 3　实验与分析

### 3.1　实验环境

本文使用Darknet框架构建网络,程序在Ubuntu16.04系统下运行,训练模型工作站处理器为Intel i5-10600KF @ 4.10 GHz,内存16 GB和显存11 GB的NVIDIA GeForce GTX 1080Ti,CUDA版本为10.0,CUDNN版本为7.6.5,OPENCV版本为3.2.0,Python版本为3.8。测试设备处理器为Intel i7-9750 H@2.60 GHz,内存16 GB和显存6GB的NVIDIA GeForce GTX 1660Ti,其他环境配置与工作站一致。

### 3.2　评估指标

为了验证本文改进方法的可用性,首先对模型进行评估,统计平均精度均值(MAP)以及运行验证集所花费的时间(Time)等指标。对于二分类问题,可以根据样本的真实类别和模型预测类别的组合将样本划分为4种类型:预测为正的正样本(true positive,TP);预测为负的正样本(false negative,FN);预测为正的负样本(false positive,FP);预测为负的负样本(true negative,TN)。

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,$P$越大表示预测结果中正确的结果样本占比越高,误检越低。计算公式为:

$$P = \frac{T_P}{T_P + F_P} \times 100\% \tag{19}$$

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,R值越高表示预测结果中正样本被正确检测出来的越多,漏检越低。计算公式为:

$$R = \frac{T_P}{T_P + F_N} \times 100\% \tag{20}$$

MAP 表示所有类别平均精度 AP 值的平均值。MAP 值越高,目标检测模型各个类别的平均检测效果越好。其中,AP 计算公式为:

$$AP = \int_0^1 p(r)\,\mathrm{d}r \tag{21}$$

MAP 计算公式为:

$$MAP = \frac{1}{n}\sum_{i=1}^{n} AP_i \tag{22}$$

### 3.3　改进 YOLO 模型的验证

为了验证所设计的模型的可用性及其性能,本文对设计的模型进行性能比较,本文首先选择在 PASCAL VOC 数据集上进行训练,迭代 40 000 次,避免使用个人数据集带来的偶然性。MAP 计算采用 VOC 数据集中的验证集进行验证,输入图片统一为 736×736。并且 NMS 都采用默认的 Greedy-NMS。所得结果如表 1 所示。改进模型及原模型的 PR 曲线如图 6 所示。

**表 1　改进模型及原模型性能对比**

**Table 1　Performance comparison of improved model and original model**

| 方法 | 精度 MAP | 时间 Time/s |
|---|---|---|
| YOLOv3 | 74.48 | **339.73** |
| YOLOv3-SPP | 77.30 | 344.40 |
| YOLOv3-SPP3 | 77.90 | 373.11 |
| YOLOv3-SPP2-5 | **79.17** | 460.97 |
| YOLOv4 | **82.70** | **303.00** |
| YOLOv4-SPP2-5 | **83.13** | **306.00** |

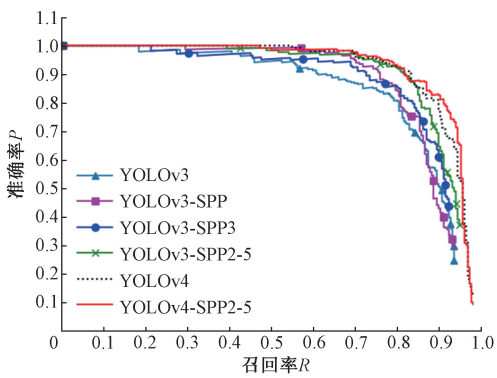



图 6　改进模型及原模型的 PR 曲线

Fig. 6　PR curves of the improved model and the original model

改进后 YOLOv4-SPP2-5 模型相比较 YOLOv4 提高 0.43%,在测试集上只增加了 3 s。总体来说,YOLOv4-SPP2-5 模型比 YOLOv4 模型性能要好,速度也能保证。

实验结果与式(6)、(7)和(8)的信息熵计算结果吻合,当特征图包含的信息足够复杂,神经网络经过 SPP5 模块以及添加第 2 个 SPP 模块后产生的融合特征图的信息熵比 SPP 模块和未加第 2 个 SPP 模块后的特征图信息熵要大,即改进模型的神经网络所蕴含的信息更加丰富,在图像上表现为包含的物体细节特征信息更全面,使得神经网络的多尺度检测能力得到提升,进而提高了模型的检测精度。可以得出结论:当一张图像所包含的信息越复杂,改进模型的效果理论上应当更明显,即改进模型对复杂环境的检测表现应当较原模型更好。推测若将输入图片分辨率增大,此时如果参与计算的特征图信息变得更丰富,那么最终模型的效果可能会有一定提升。因此改进模型应当适合水果检测的复杂情况。

### 3.4　改进 NMS 算法的验证

由于 COCO 数据集具有较多的小目标和遮挡、重叠目标,因此为了验证本算法对重叠、遮挡目标的可行性,本文首先采用更权威的 COCO 数据集作为 NMS 改进的验证数据集,在 COCO test-dev 2017 官方测试服务器上对本文算法进行测试,并详细对比各项平均精确度(average precision, AP)指标,以及在测试设备上运行测试集所花的时间(Time),并基于此数据获取用于检测任务时算法参数的预数值。输入图片统一设置为 608×608,为了反映原模型添加 Greedy-Confluence 的结果,不修改 YOLO 模型网络的结构,模型的网络结构和权重文件均采用 YOLO 原版公开文件。将 DIOU-NMS 的 beta 参数设置为默认最优 0.6[31],NMS 和 DIOU-NMS 的阈值 $N_t$ 设置为 0.45,Greedy-confluence 的 $\varepsilon_1$,$\varepsilon_2$ 分别设置为 0.4 和 1.1。

本文的实验结果如表 2 所示,相关原算法测试结果可以参考文献[32]。可以看出,在 COCO2017(test-dev:20 288 张图片)数据集上 YOLOv4-Greedy-Confluence 的 MAP 较 YOLOv4 提升了 0.6%,较 YOLOv4-DIOU-NMS 耗时降低 433 s,且算法检测精度基本相同。YOLOv3-Greedy-Confluence 也比 YOLOv3 提升了 0.2%,较 YOLOv3-DIOU-NMS 提升了 0.1%,而 YOLOv3-SPP-Greedy-Confluence 更是比 YOLOv3-SPP-Greedy-NMS 提升了 0.7%,较 YOLOv3-SPP-DIOU-NMS 提升了 0.2% 且耗时降低 17.27 s。Greedy-confluence 提高了检测精度,并均衡了检测时间消耗,减少了检测框的错误抑制。

### 3.5　水果检测数据集制作

当前网络上有很多水果检测的数据集,本文使用百度 AI Studio 数据集库中 fruit-detection 数据集,由于此数据集量相对太小,并且数据集中的遮挡和小目标数据量很少,因此本文选取 kaggal 中的 Fruit-Recognition 数据集中的部分图片,综合考虑光线、姿势变化、阴影、目标大小、数量、遮挡等情景,对各种情况分别选取适量图片作为数据集,为了提高模型泛化能力,在百度上通过“果树”检索关键词搜索并筛选获取相应类别的具有大量果实和遮挡果实的图片。最终获得 601 幅图像,其中包括香蕉、橙子、苹果 3 类,编写程序将图片统一为 jpg 格式,

表2　改进NMS及原NMS性能对比

Table 2　Performance comparison of improved NMS and original NMS

| 方法 | AP@0.5:0.95 | AP@0.5 | AP@0.75 | AP small | AP medium | AP large | Time/s |
|---|---|---|---|---|---|---|---|
| v3 | 33.0 | **57.9** | 34.4 | 18.3 | 35.4 | 41.9 | **1 184.86** |
| v3+DIOU-NMS | 33.1 | 57.6 | 34.5 | 18.4 | 35.5 | **42.1** | 1 228.24 |
| v3+Greedy-Confluence | **33.2** | 56.8 | **35.2** | **18.5** | **35.7** | **42.1** | 1 220.95 |
| v3+SPP | 36.2 | 60.6 | 38.2 | **20.6** | 37.4 | 46.1 | 1 192.42 |
| v3+SPP+ DIOU-NMS | 36.7 | **60.7** | 38.9 | 19.6 | 38.6 | 47.8 | 1 210.95 |
| v3+SPP+Greedy-Confluence | **36.9** | 60.3 | **39.6** | 19.8 | **38.9** | **48.0** | **1 193.68** |
| v4 | 43.5 | **65.7** | 47.3 | 26.7 | 46.7 | 53.3 | 1 183 |
| v4+DIOU-NMS | **44.1** | 65.1 | **48.8** | **27.1** | **47.4** | **53.7** | 2 317 |
| v4+Greedy-Confluence | **44.1** | 65.3 | 48.6 | **27.1** | 47.3 | **53.7** | 1 884 |

然后使用LabelImg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,然后转换成YOLO框架的TXT类型标签文件。训练深度神经网络通常需要大量的数据,因此为了加强模型的泛化能力,同时考虑制作大型数据集的成本,本文对此数据集采用数据增强技术,提升模型的鲁棒性。使用平移、翻转、旋转、缩放、添加噪声、改变图像亮度、饱和度和对比度8种方法的随机组合对采集到的图像进行数据增强,同时编写程序对每幅图像对应的标注文件进行同步数据变换与生成。对每幅图像生成10幅数据增强图像,得到有效图像6 611幅。然后按照8∶2的比例划分训练集和验证集,分别得到测试集5 288张图片,验证集1 323张图片,随后在验证集中取80%用作测试集,得到测试集1 060张图片。

### 3.6　数据集锚定框聚类

由于YOLO框架是对COCO数据集设置的锚定框,与本文数据集具有较大出入,因此采用K-means++重新聚类先验锚定框,设置输入图片大小为416×416,迭代次数100次,最终得到聚类的锚定框为:(14,17),(49,43),(67,84),(99,117),(115,164),(189,148),(153,227),(225,269),(335,348)。

### 3.7　模型训练及性能对比

本文选择YOLOv4网络作为水果检测实际任务的基础网络模型,分别在原网络和改进网络上对制作的数据集进行训练,并对精度和测试所花时间等指标进行统计,取$\varepsilon_1$为0.44,测试结果如表3所示。结果表明:改进后的YOLOv4-SPP2-5模型较YOLOv4模型MAP提升1.65%,检测速度为每秒40.77帧,而改进后的YOLOv4-SPP2-5+Greedy-Confluence模型较YOLOv4模型MAP提升1.70%,检测速度为每秒39.26帧。总体来说,本文改进方法精度得到提升,速度仍有保证,满足实时性需求。图7为不同模型的PR曲线,图8为不同模型效果的一些对比实例。

表3　水果检测各种模型性能对比

Table 3　Comparison of performance of various fruit detection models

| 方法 | 精度 | | | | 时间 | 速度 | 单张时间 |
|---|---|---|---|---|---|---|---|
| | 平均精度MAP | AP(香蕉) | AP(橙子) | AP(苹果) | Time/s | $V/(\mathrm{f\cdot s^{-1}})$ | Time per image/s |
| YOLOv4 | 94.95 | 95.30 | 96.39 | 93.17 | **26.0** | **40.77** | **0.024 5** |
| YOLOv4-SPP2-5 | 96.60 | **97.44** | 97.04 | 95.32 | **26.0** | **40.77** | **0.024 5** |
| YOLOv4-SPP2-5+ DIOU-NMS | **96.65** | 96.79 | 97.50 | 95.65 | 28.0 | 37.86 | 0.026 4 |
| YOLOv4-SPP2-5+Greedy-Confluence | **96.65** | 96.70 | **97.51** | **95.72** | **27.0** | **39.26** | **0.025 4** |

由表3及图8(a)、(b)两组结果可以看出,YOLOv4-SPP2-5模型较YOLOv4模型的多尺度水果目标检测能力更强,能够检测出更多的水果目标,由图8(b)、(c)两组结果可知,加入Greedy-Confluence方法后,重叠、遮挡水果目标的检测精度得到提升。因此,改进后的模型及NMS方法在水果检测任务上得到了更好的结果。

最终实验结果符合信息熵计算结果和NMS改进算法的效果期望。加入改进的NMS后端处理算法后,如表3所示,总体检测精度上升,香蕉的检测精度较只有模型改进的方法有略微下降,原因可能是:1)数据集制作问题,包括标签标注不准确,数据集中香蕉目标的尺度单一,数据集泛化能力欠缺等;2)NMS改进算法的阈值设置可能更趋向达到总体检测精度提升至最优的合适值,香蕉和苹果、橙子的形状差异较大,NMS改进算法无法

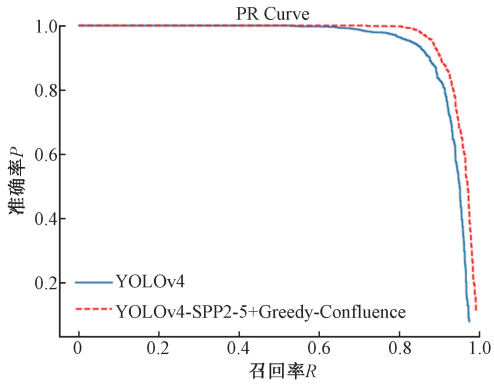
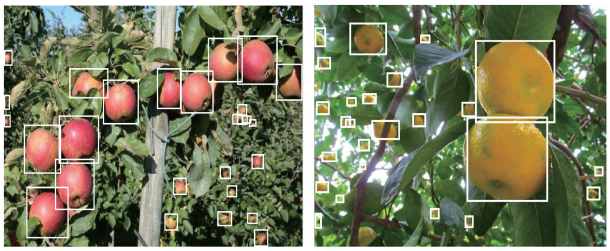



图 7　水果检测改进模型及原模型的 PR 曲线

Fig. 7　Fruit detection PR curves of the improved model and the original model



(a)YOLOv4模型检测结果
(a)Detection results of YOLOv4 model

(b)YOLOv4-SPP2-5模型检测结果
(b)Detection results of YOLOV4-SPP2-5 model

(c)YOLOv4-SPP2-5+Greedy-Confluence模型检测结果
(c)Detection results of YOLOV4-SPP2-5+Greedy-Confluence model

图 8　水果检测效果实例

Fig. 8　Examples of fruit detection effect

在每一个类别上均达到最优。本文的相关实验代码上传在:https://gitee.com/xyy0521/yolo-spp2-5-g-c.git。

## 4　总　结

为了解决水果检测任务中目标尺度变化大,重叠、遮挡情况较多以及小目标检测精度问题,本文提出一种 SPP5 模块,增强了感受野范围,并基于此模块设计一种 YOLOv4-SPP2-5 模型,提高了模型的多尺度信息融合能力,从而提高了模型的检测精度。对于 YOLO 模型的后端处理模块,本文提出一种 Greedy-Confluence 策略,综合考虑了 DIOU、Confluence 的思想,提高了算法性能,保证了 NMS 处理的精度,并均衡了检测时间消耗。

本文将改进方法在水果检测任务上进行了实验。结果表明,基于改进 YOLO 网络及改进 NMS 的水果检测方法在准确率方面有显著的提高,平均精度均值(MAP)在 YOLOv4 上得到了一定提升并且实时性也得到了保证。但对于形状差异较大的不同物体,改进方法的性能有待进一步提升,重叠遮挡严重的情况下模型的回归框定位不够准确,今后将对目标检测深度学习模型进一步研究突破,着重提高目标检测方法在复杂自然场景下的识别和定位精度问题。

## 参考文献

[1] ZHAO Z Q, ZHENG P, XU S T, et al. Object detection with deep learning: A review[J]. IEEE Trans Neural New Learn Syst, 2019, 30(11): 3212-3232.

[2] XIAO Y, TIAN Z, YU J, et al. A review of object detection based on deep learning[J]. Multimedia Tools and Applications, 2020, 79(33-34): 23729-23791.

[3] 曹红燕, 沈小林, 刘长明, 等. 改进的 YOLOv3 的红外目标检测算法[J]. 电子测量与仪器学报,2020, 34(8):188-194.
CAO H Y, SHEN X L, LIU CH M, et al. Improved infrared target detection algorithm of YOLOv3[J]. Journal of Electronic Measurement and Instrumentation, 2020,34(8):188-194.

[4] 伊欣同, 单亚峰. 基于改进 Faster R-CNN 的光伏电池内部缺陷检测[J]. 电子测量与仪器学报, 2021, 35(1): 40-47.
YI X T, SHAN Y F. Photovoltaic cell internal defect detection based on improved Faster R-CNN[J]. Journal of Electronic Measurement and Instrumentation, 2021, 35(1): 40-47.

[5] GIRSHICK R, DONAHUE J, DARRELL T, et al. Rich feature hierarchies for accurate object detection and semantic segmentation[C]. 2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2014: 580-587.

[6] HE K, ZHANG X, REN S, et al. Spatial pyramid pooling in deep convolutional networks for visual recognition[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2015, 37(9): 1904-1916.

[7] GIRSHICK R. Fast R-CNN [C]. International

Conference on Computer Vision (ICCV), 2015: 1440-1448.

[8] DAI J, LI Y, HE K, et al. R-FCN: Object detection via region- based fully convolutional networks [C]. Conference on Neural Information Processing Systems, 2016: 379-387.

[9] HE K, GKIOXARI G, DOLLAR P, et al. Mask R- CNN[C]. 2017 IEEE International Conference on Computer Vision (ICCV), 2017: 2980-2988.

[10] CAI Z, VASCONCELOS N. Cascade R-CNN: Delving into high quality object detection[C]. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018: 6154-6162.

[11] LIU W, ANGUELOV D, ERHAN D, et al. SSD: Single shot multibox detector [C]. European Conference on Computer Vision (ECCV), 2016: 21-37.

[12] REDMON J, DIVVALA S, GIRSHICK R, et al. You only look once: Unified, real-time object detection[C]. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016: 779-788.

[13] LIN T Y, GOYAL P, GIRSHICK R, et al. Focal loss for dense object detection[C]. International Conference on Computer Vision (ICCV), 2017: 2999-3007.

[14] TAN M, PANG R, LE Q V. Efficient-Det: Scalable and efficient object detection[C]. 2020 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2020.

[15] 夏浩宇,索双富,王洋,等. 基于 Keypoint RCNN 改进模型的物体抓取检测算法[J]. 仪器仪表学报, 2021, 42(4):236-246.
XIA H Y, SUO SH F, WANG Y, et al. Object grasp detection algorithm based on improved Keypoint RCNN model [J]. Chinese Journal of Scientific Instrument, 2021,42(4):236-246.

[16] WAN S, GOUDOS S. Faster R-CNN for multi-class fruit detection using a robotic vision system [J]. Computer Networks, 2020, 168: 107036.

[17] TONG K, WU Y, ZHOU F. Recent advances in small object detection based on deep learning: A review[J]. Image and Vision Computing, 2020, 97: 103910.

[18] TIAN Y, YANG G, WANG Z, 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.

[19] LI G, HUANG X, AI J, et al. Lemon-YOLO: An efficient object detection method for lemons in the natural environment[J]. IET Image Processing, 2021: 1-12.

[20] LI X, QIN Y, WANG F, et al. Pitaya detection in orchards using the mobile net-YOLO model[C]. 2020 39th Chinese Control Conference (CCC), 2020: 6274-6278.

[21] WU L, MA J, ZHAO Y, et al. Apple detection in complex scene using the improved YOLOv4 model[J]. Agronomy, 2021, 11(3): 476.

[22] LIU G, NOUAZE J C, TOUKO MBOUEMBE P L, et al. YOLO-Tomato: A robust algorithm for tomato detection based on YOLOv3[J]. Sensors, 2020, 20(7): 2145.

[23] LIN T Y, DOLLAR P, GIRSHICK R, et al. Feature pyramid networks for object detection[C]. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017: 2117-2125.

[24] SINGH B, DAVIS L S. An analysis of scale invariance in object detection snip[C]. 2018 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018: 3578-3587.

[25] GHIASI G, LIN T YI, LE Q V, et al. Nas-fpn: Learning scalable feature pyramid architecture for object detection [C]. 2018 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2019: 7036-7045.

[26] LIU S, HUANG D, WANG Y. Learning spatial fusion for single-shot object detection [J]. ArXiv Preprint, 2019,arXiv:1911.09516.

[27] LIU S, QI L, QIN H, et al. Path aggregation network for instance segmentation [C]. 2018 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018: 8759-8768.

[28] ZHANG P, ZHONG Y, LI X. SlimYOLOv3: Narrower, faster and better for real-time UAV applications [C]. 2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW), 2019: 37-45.

[29] BODLA N, SINGH B, CHELLAPPA R, et al. Soft-nms-improving object detection with one line of code[C]. 2017 IEEE International Conference on Computer Vision (ICCV), 2017: 5562-5569.

[30] ZHENG Z, WANG P, LIU W, et al. Distance-IoU loss: Faster and better learning for bounding box regression[C]. AAAI Conference on Artificial Intelligence (AAAI), 2020, 34(7):12993-13000.

[31] SHEPLEY A, FALZON G, KWAN P. Confluence: A robust Non-IoU alternative to non-maxima suppression in object detection [J]. ArXiv Preprint, 2020, arXiv: 2012.00257.

[32] BOCHKOVSKIY A, WANG C Y, LIAO H Y M. YOLOV4: Optimal speed and accuracy of object detection[J]. ArXiv Preprint,2020, arXiv: 2004.10934.

[33] REDMON J, FARHADI A. YOLOV3: An incremental improvement [J]. ArXiv Preprint, 2018, arXiv: 1804. 02767.

[34] 李国杰. 基于熵的人工神经网络系统理论初探[J]. 计算机学报, 1990,13(5):321-330.
LI G J. Entropy based system-theoretical aspect of artificial neural networks [J]. Chinese Journal of Computers, 1990,13(5):321-330.

[35] SHANNON C E. A mathematical theory of communication [J]. ACM SIGMOBILE Mobile Computing and Communications Review, 2001, 5(1): 3-55.

## 作者简介

**徐印赟**,现为安徽工程大学硕士研究生,主要研究方向为目标检测与深度学习。E-mail: 1911590204@ qq. com

**Xu Yinyun** is a M. Sc. candidate at Anhui Polytechnic University. His main research interests include object detection and deep learning.

**江明**(通信作者),1993 年于上海工业大学(现上海大学)获得硕士学位,现为安徽工程大学教授、硕士生导师,主要研究方向为机器人智能控制系统和先进检测技术。E-mail: kjjm@ ahpu. edu. cn

**Jiang Ming** (Corresponding author) received his M. Sc. degree from Shanghai Technology University (now Shanghai University) in 1993. Now he is a professor and M. Sc. supervisor at Anhui Polytechnic University. His main research interests include robotic intelligent control system and advanced detection technology.

**李云飞**,2021 年于澳门大学获得电机及电脑工程博士学位,主要研究方向包括定位鲁棒算法、安全定位算法和统计信号处理。

**Li Yunfei** received the Ph. D. degree in electrical and computer engineering from University of Macau in 2021. His main research interests include localization robust algorithm, secure localization algorithm, and statistical signal processing.