

邓泓,王祖明,尹乘乐,等. 基于 KSGM-YOLO 的轻量级小目标实蝇检测模型[J]. 江苏农业科学,2025,53(5):213-223.

doi:10.15889/j.issn.1002-1302.2025.05.025

# 基于 KSGM-YOLO 的轻量级小目标实蝇检测模型

邓泓<sup>1</sup>,王祖明<sup>1</sup>,尹乘乐<sup>2</sup>,李越千<sup>1</sup>,黄伟继<sup>1</sup>,桂露<sup>1</sup>,周帅<sup>1</sup>,彭莹琮<sup>1</sup>

(1. 江西农业大学软件学院,江西南昌 330000; 2. 德布勒森大学,匈牙利德布勒森 4032)

**摘要:**实蝇害虫会对果蔬作物造成减产,进而影响农民的果蔬经济效益。由于实蝇类间相似性高且采集的样本多为小目标,导致传统的害虫识别模型不准确。为此,提出一种基于 YOLO v7-tiny 的轻量级小目标检测模型 KSGM-YOLO。该模型首先设计了针对实蝇数据集的锚框聚类算法,生成了更有效的锚框;其次在 Backbone 中引入 SimAM 注意力机制,以增强模型对实蝇语义信息的特征提取能力;同时原模型的 Neck 层中设计了 GSCBL 与 GSELAN,以降低模型的参数量与计算量;最后采用 MPDIoU 损失函数计算定位损失,优化模型对小目标实蝇标注框的计算。此外本研究创建了实蝇数据集 *Drosophila-Four* 并在此数据集上进行了多项试验。试验结果显示,KSGM-YOLO 相较于原模型,在精度方面提高了 2.3 百分点且参数量和计算量分别下降了 6.3% 和 8.3%。这表明本研究提出的模型实现了更为准确的小目标实蝇检测需求,同时也取得了一定的轻量级优化效果。综上所述,本研究结果为果蔬作物提供了一种更准确的针对小目标害虫的轻量级检测方法,并为在边缘设备部署提供了可行性,能帮助农业工作者及时发现实蝇害虫,提高果蔬作物的产量和质量。

**关键词:**实蝇;YOLO v7-tiny;KSGM-YOLO;小目标检测;轻量级

**中图分类号:**S126;TP391.41 **文献标志码:**A **文章编号:**1002-1302(2025)05-0213-10

实蝇是危害果蔬安全生产的主要原因之一。果蔬作物在世界农业生产板块中占据重要份额<sup>[1]</sup>,大量果蔬产品因实蝇的危害而严重影响其经济效益<sup>[2]</sup>。我国果蔬种植面广,实蝇寄主广泛,虫害扩散十分迅速,在没有防控的情况下,能轻易造成损失。因此及时准确地检测果蔬的虫害类型具有重大的科研、经济和社会价值,能快速采取针对性防治措施,降低虫害对果蔬的生产损失,又可以避免盲目使用农药造成生态环境污染,研究有效的果蔬害虫防治方法具有重大现实意义。

传统实蝇监测主要依赖人工经验,这种方法劳动强度大,且容易出现误判,导致成本较高。随着机器学习技术被应用在农业虫害检测中,从一定程度上解决了上述问题。多层感知器(MLP)<sup>[3]</sup>、支持向量机(SVM)<sup>[4-5]</sup>、BP 神经网络<sup>[6]</sup>通过对颜色、纹

理、大小等特征的筛选,能够完成对农业害虫的检测任务。但是采用机器学习需要高质量图像且需要手动进行特征提取,在现实的果园中实蝇背景复杂且处于移动状态,这将显著影响实蝇的检测精度及检测速度。

由于使用机器学习方法进行农业害虫检测存在一定局限性,越来越多的研究人员使用深度学习技术来进行农业上的虫害识别工作。Cheng 等利用 AlexNet 与残差学习对复杂农田背景下的害虫图像进行识别,与传统 BP 神经网络相比,该方法识别性能更强<sup>[7]</sup>。Yang 等利用特征金字塔将低层次的特征与高层次的特征连接起来,并改进了轻量级的 SqueezeNet 模型,使网络模型适用于园林昆虫的研究,在针对小目标昆虫时仍然存在一定局限<sup>[8]</sup>。肖德琴等基于 Faster R-CNN 模型提出了一种端到端的 MPF R-CNN,通过加深主干网络层数并结合 FPN 对黄板诱捕到的害虫进行检测,但是由于其模型复杂,运行时间较长,导致其在实际场景部署时存在一定挑战<sup>[9]</sup>。

与上述传统深度学习检测方法相比,YOLO 系列检测方法具有更快的推理速度和重要的工程应用价值<sup>[10-14]</sup>。Amrani 等通过集成自适应特征融合模块进一步改进了 YOLO v3 的原始特征金字塔网

收稿日期:2023-12-27

基金项目:国家自然科学基金(编号:62262028);江西省教育厅科学技术研究项目(GJJ210438、GJJ210434、GJJ2200423)。

作者简介:邓泓(1977—),男,江西南昌人,硕士,副教授,硕士生导师,主要从事农业信息化、计算机视觉研究。E-mail: jxaudh@jxau.edu.cn。

通信作者:彭莹琮,硕士,教授,硕士生导师,主要从事农业信息化、图像处理研究。E-mail: jneyq@jxau.edu.cn。

络,实现了对农作物害虫的实时检测<sup>[15]</sup>。Song 等将 DenseNet 引入到 YOLO v4 的主干网络中,增强模型的特征提取能力,在自制的害虫数据集上取得了优良的检测效果<sup>[16]</sup>。Wu 等通过对 YOLO v5 的主干网络添加卷积层及在 SPPF 结构添加注意力机制实现了对红棕象甲的实时检测,能够有效地运用在野外环境<sup>[17]</sup>。Wang 等提出的 YOLO v7 是目前图像识别与目标检测领域较为先进的深度学习算法<sup>[18]</sup>,郑果等对 YOLO v7 模型做改进,使用 MobileNet v3 作为主干网络对常见的 6 类水稻害虫做检测任务,实现了高精度的水稻虫害检测<sup>[19]</sup>。因 YOLO v7 的参数量与计算量较大,需要更高端的设备进行实地部署,作者同时提出了适用于部署在边缘 GPU 的 YOLO v7 - tiny<sup>[18]</sup>。

虽然基于卷积神经网络的深度学习方法在农作物害虫的检测识别方面具有一定的优势,YOLO v7 - tiny 也能胜任常见的目标检测场景,但是针对小目标实蝇检测仍然存在一定挑战:(1)与传统的检测任务相比,实蝇体积小,标注框分布与检测器预设的锚框差距较大,在训练时影响定位效果,给小目标检测造成了障碍。(2)实蝇类间差距较小且检测环境的背景复杂,Backbone 提取特征能力有限,难以关注到小目标语义信息。(3)在特征融合结构中的 ELAN 模块<sup>[18]</sup>虽然能增强网络的学习能

力,但其参数量较大,且对实蝇的形状与尺寸的改变敏感性不佳。(4)在实际拍摄过程中存在实蝇在图像中占比小、运动模糊、遮挡、变形等问题,导致数据集中的部分标注框与物体的位置存在偏差,对默认的坐标损失计算影响显著。针对上述问题,本研究以 YOLO v7 - tiny 为基础网络进行改进,提出了实蝇检测模型 KSGM - YOLO,实现了对自然环境下小目标实蝇的实时检测;保证模型检测的准确性,同时最小化模型的参数数量和计算资源,以满足部署在果园实时检测的要求,为果园小目标实蝇害虫的检测提供智能解决方案。

## 1 材料与方法

### 1.1 数据集的构建

本研究以橘小实蝇、具条实蝇、瓜实蝇及南瓜实蝇对果蔬危害严重的 4 类实蝇作为检测对象,选用的实蝇标本来自江西农业大学农学院实验室,数据集的制作时间为 2023 年 7 月中旬。其中具条实蝇 262 张,南瓜实蝇 252 张,橘小实蝇 247 张,瓜实蝇 251 张,拍摄到的图片分辨率为 3 024 像素 × 3 024 像素,存储格式为 jpg 格式。由图 1 可见,这些图像包含了具有复杂和简单背景的场景,用于模拟果园中实蝇虫害的识别状态以及与诱捕设备的相关情境。



图1 数据集中部分实蝇图像

本研究使用 Labelimg 对采集的实蝇图像进行数据集标注。由于原数据有限,为了避免模型在训练时产生过拟合的问题,利用 Albumentations 对原始数据集进行颜色及几何变换等操作来扩充数据集<sup>[20]</sup>,增强模型的泛化性和鲁棒性,最后得到 3 480 张包含 4 类实蝇的数据集 Drosophila - Four。此外

YOLO v7 - tiny 的数据预处理部分带有 Mix Up 以及 Mosaic 的数据增强<sup>[21]</sup>,如图 2 所示,扩充后数据集能够模拟实蝇害虫多角度及不同光照下的状态,对于提升训练模型泛化能力至关重要。数据增强的引入使得模型能够更全面地学习并适应不同的场景,从而在实际应用中表现出更强的鲁棒性。



本研究采用 6 : 2 : 2 的数据集划分比例,将数据集 *Drosophila - Four* 进行随机划分,分别用于训练集、验证集和测试集,其分布见表 1。

表 1 4 类实蝇的数据量

类别	划分比例 (%)	类别	标签数量 (个)	图片数量 (张)
训练集	60	具条实蝇	2 245	631
		瓜实蝇	2 427	630
		南瓜实蝇	3 210	649
		橘小实蝇	2 105	637
验证集	20	具条实蝇	665	183
		瓜实蝇	609	160
		南瓜实蝇	819	163
		橘小实蝇	483	152
测试集	20	具条实蝇	906	234
		瓜实蝇	832	214
		南瓜实蝇	895	196
		橘小实蝇	664	199

1.2 YOLO v7 - tiny 网络

YOLO v7 是高性能检测器,准确度和速度方面较 YOLO v5 有明显的优势。YOLO v7 - tiny 是 YOLO v7 的简化版本,是为边缘 GPU 设计的网络模型,由 Input、Backbone、Neck 及 Detected Head 等 4 个部分组成,Input 部分负责对输入的图片进行图像预处理操作;Backbone 对输入的图像进行多尺度的特征提取;模型的 Neck 利用 PAN 结构对 Backbone 提取到的浅层位置信息及深层语义信息进行特征融合<sup>[22]</sup>;Detected Head 执行分类检测回归,使用 IDetect 作为 Head1 到 Head3 的 3 个检测头<sup>[23]</sup>,分别检测小型、中型和大型目标。

$$Loss = Loss_{loc} + Loss_{class} + Loss_{conf} \quad (1)$$

YOLO v7 - tiny 主要计算 3 个损失函数,如式 (1) 所示,即边界框损失、分类损失和目标置信损

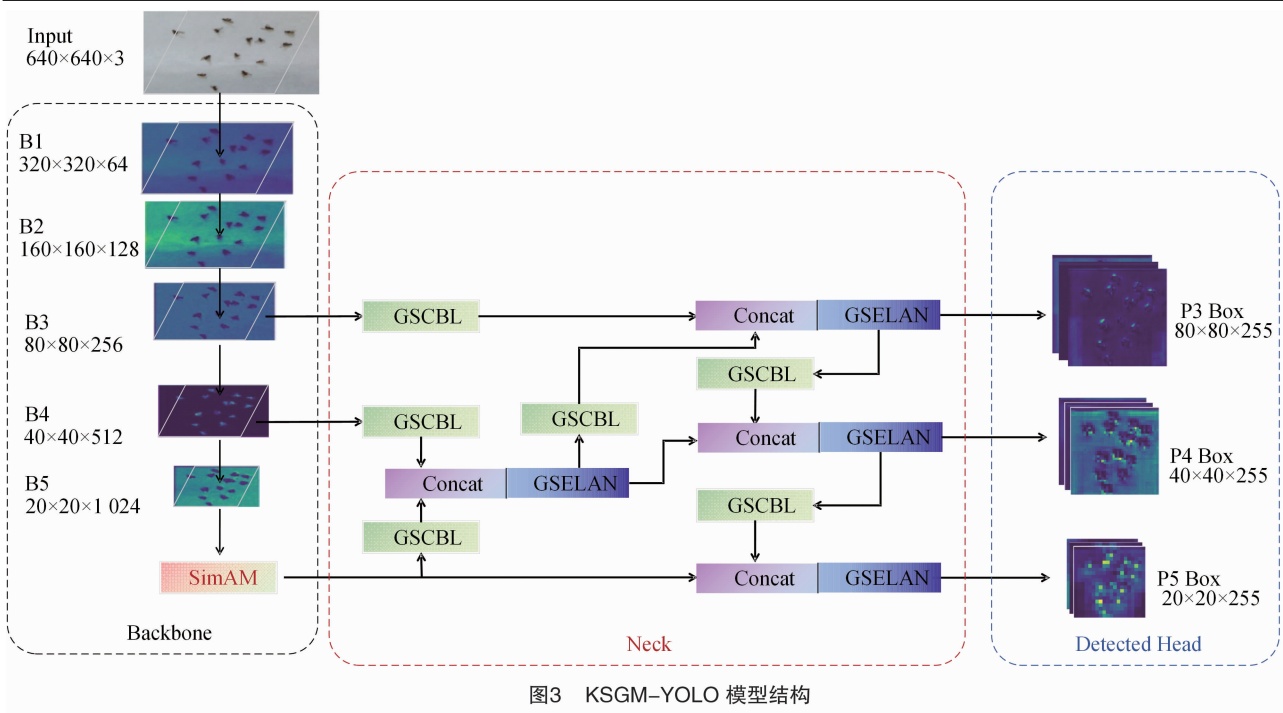
失。在目标检测阶段的后处理过程中,YOLO v7 - tiny 使用非最大抑制 (NMS) 过滤多个目标候选框并消除冗余候选框,从而确保最终每个对象只能获得 1 个检测框。

1.3 改进的网络

本研究提出的高效轻量级小目标实蝇检测模型 KSGM - YOLO 的网络结构如图 3 所示。

本研究针对原有的 YOLO v7 - tiny 模型进行改进,首先设计了针对小目标实蝇的锚框生成算法,重新聚类实蝇数据集集中的标注框,构建一批适用于小目标实蝇检测的锚框;接着在 Backbone 中引入 SimAM 模块,加强模型对多类别实蝇相似特征的有效提取能力;同时利用 GSConv 轻量化 Neck 网络,减轻模型运行的参数量同时保持精度;最后利用 MPDIoU Loss 改进原始损失函数,使得收敛速度更快且有更好的定位结果。

1.3.1 实蝇数据集锚框重设算法 YOLO v7 - tiny 采用的初始锚框是从 COCO 数据集<sup>[24]</sup>中的标注框聚类而来。针对本研究使用的实蝇数据集锚框大小和 COCO 数据集锚框大小分布存在差异<sup>[24]</sup>,因此本研究设计了一种实蝇数据集锚框重设算法 (图 4)。算法提出了一种更适合小目标实蝇的度量方式对标注框进行聚类;并解决了传统锚框聚类算法对噪声值及聚类初始中心敏感的问题。锚框重设算法旨在优化聚类过程,确保生成的锚框能更有效地适应小目标实蝇检测任务,为此做了 2 点改进。(1) 在传统算法聚类过程中,当锚框与标注框高度重叠时,微小的位置变化或 IoU 微小变化可能导致不同的锚框被分配到相同的标注框或不同的聚类中心,从而导致聚类过程变得不稳定,因此本研究利用 GIoU 设计了一种度量方式<sup>[25]</sup>,用以改进聚类度量,进一步保证模型的性能和泛化能力。(2) 为了进一步提高目标框与锚点的匹配度并减少随机选



#### Algorithm 1: 实蝇数据集锚框重设算法

**Input:** 实蝇数据集的标注框集合  $B=\{x_1, x_2, x_3\}$ , 聚类数  $K$

**Output:** 最终的聚类中心集合  $Centers$

```

1 初始化一个空集合  $Centers$ ;
2 从  $B$  中随机选择一个样本作为第一个聚类中心  $Center_1$  并且将其添加到  $Centers$ ;
3 repeat
4   for  $i$  from 2 to  $K$  do
5     for  $B$  中的每一个样本  $x$  do
6       计算距离  $d = 1 - GloU(x, nearest\_center(x))$ 
7       其中  $nearest\_center(x)$  是最近的聚类中心;
8       计算选择概率  $P(x) = \frac{d(x, nearest\_center(x))^2}{\sum_{i=1}^N d(x, nearest\_center(x))^2}$ ;
9     在  $B$  中以概率  $P(x)$  随机选择下一个聚类中心  $Center_i$  并将其添加到  $Centers$ ;
10 until 选择出  $K$  个初始中心点;
11 初始化迭代次数  $iteration$  为 0;
12 repeat
13   for  $B$  中的每一个样本  $x$  do
14     计算它与  $Centers$  已有的聚类中心点的距离, 并将  $x$  分配到最近的  $Center_i$ ;
15   for 每一个簇中心  $Center_i$  do
16     利用  $Center_i$  分配到样本的平均值计算其新的聚类中心点  $NewCenter_i$ ;
17   将  $NewCenter_1, NewCenter_2, ..., NewCenter_K$  更新到  $Centers$  中;
18   更新  $iteration$ ;
19 until  $iteration$  达到最大值或者满足收敛条件;
20 return  $Centers$ ;

```

图4 锚框重设算法

取初始值对结果的潜在影响,本研究采用  $K - means++$  作为聚类算法重新设计了锚框的尺寸<sup>[26]</sup>。对实蝇数据集 *Drosophila - Four* 进行锚框重设,最终得到的锚框尺寸如表 2 所示。

1.3.2 SimAM 注意力机制 通过观察发现在 *Drosophila - Four* 数据集中,存在较多数量的小目标

实蝇,且不同类别之间的实蝇形状纹理差距较小,为了进一步提升模型对小目标实蝇语义信息的捕捉能力,使得卷积神经网络能够在小目标实蝇样本上学习鲁棒的特征表示,本研究在检测模型的 Backbone 中添加 SimAM 注意力机制。

人类的视觉系统中,注意力机制能够更快捷高



表 2 在 *Drosophila* - Four 数据集中预设锚框  
与聚类后的锚框大小对比

特征图层/下采样倍数	P3/8	P4/16	P5/32
COCO 数据集锚框尺寸	(10,13)	(30,61)	(116,90)
	(16,30)	(62,45)	(156,198)
	(33,23)	(59,119)	(373,326)
<i>Drosophila</i> - Four 数据集锚框尺寸	(9,22)	(41,63)	(149,106)
	(20,17)	(73,49)	(124,195)
	(27,25)	(102,71)	(162,132)

效的分析复杂场景信息。神经元能够根据人类视觉提取信息的丰富性表现出独特的放电模型,从而可以评估神经元的重要性。相对于 Squeeze and Excitation Net<sup>[27]</sup> 和 Convolutional Block Attention Module<sup>[28]</sup>, Yang 等基于完善的神经科学理论提出 SimAM 模块<sup>[29]</sup>,很好地解决了传统注意力机制灵活性不足及操作性复杂的问题。SimAM 的示意图如图 5 所示,从当前的神经元推理出三维的权重,然后反过来去优化这些神经元,让网络学习到更具区分性的神经元。在小目标实蝇检测中这些关键神经元通常在提取目标的显著特征方面发挥作用。其

计算过程见公式(2)~公式(5):

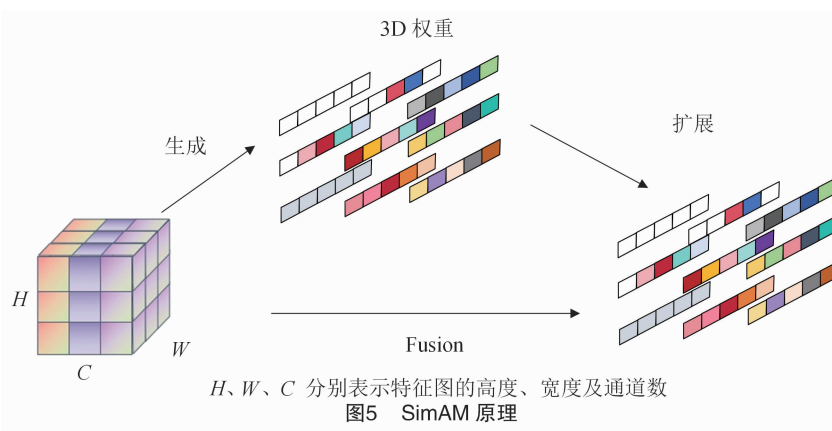
$$Y = \text{sigmoid}\left(\frac{1}{E}\right) \times X; \quad (2)$$

$$e_i^* = \frac{4(\sigma^2 + \lambda)}{(t - \mu)^2 + 2\sigma^2 + 2\lambda}; \quad (3)$$

$$\mu = \frac{1}{Q} \sum_{i=1}^Q x_i; \quad (4)$$

$$\sigma^2 = \frac{1}{Q} \sum_{i=1}^Q (x_i - \mu)^2. \quad (5)$$

其中,  $Y$  表示实蝇语义信息得到增强后的特征图;  $e_i^*$  表示每个通道上的能量函数;  $E$  表示每个神经元位置的能量值;  $Q$  表示每个特征通道的神经元总数;  $X$  表示输入实蝇的特征图;  $\sigma^2$  表示输入实蝇特征图中每个通道数上的方差;  $\lambda$  是超参数;  $t$  表示实蝇目标的神经元;  $\mu$  表示输入实蝇特征图中每个通道上的平均值;  $x_i$  表示第  $i$  个特征通道上其他的神经元。  $e_i^*$  越低说明神经元与周围的神经元的区别越大,利用  $\frac{1}{e_i^*}$  来评估每个神经元的重要性,并通过 sigmoid 激活函数做特征细化后与原特征做加权。



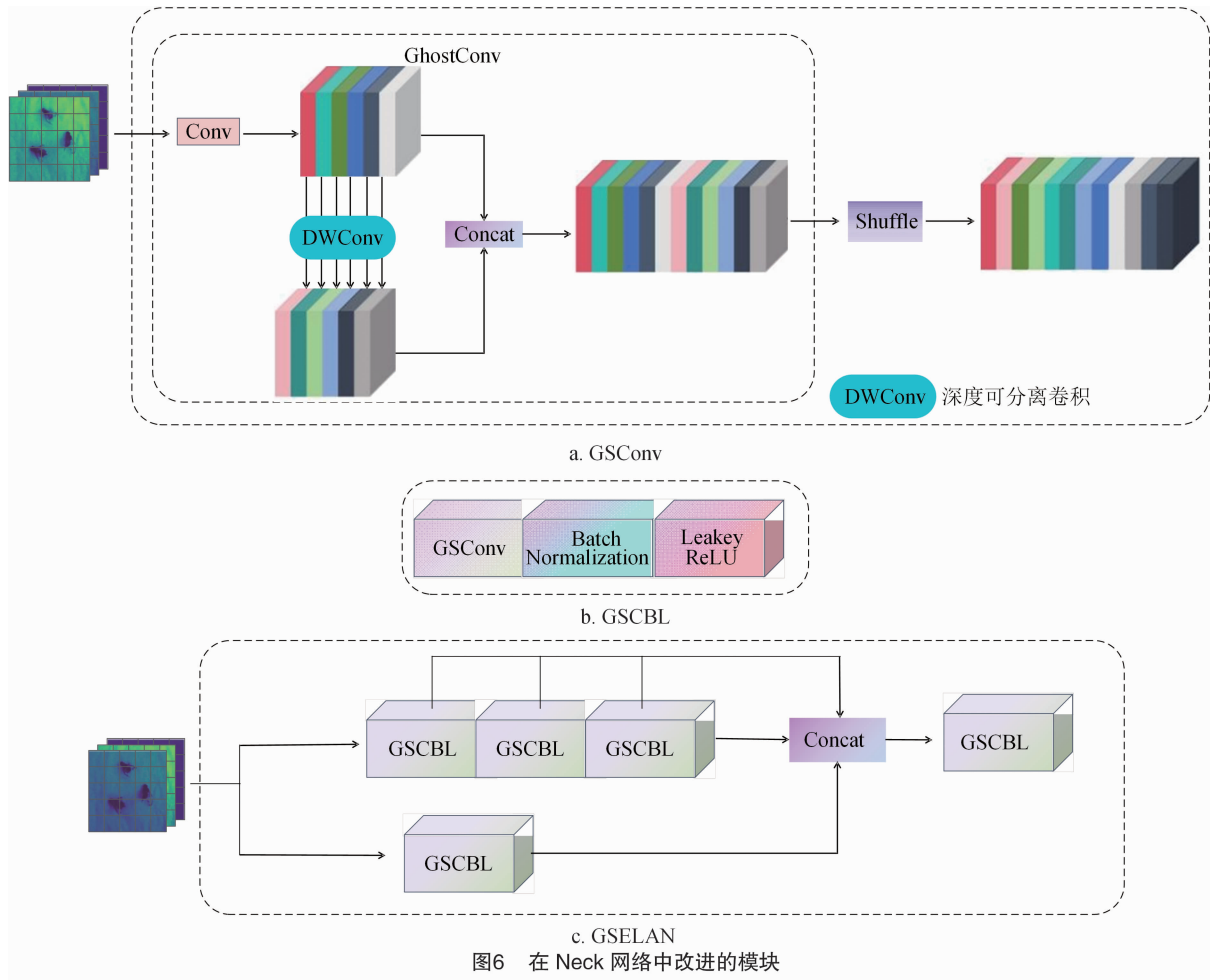
Backbone 对输入的小目标实蝇图像进行特征提取操作,随着网络层数的增加,感受野增大的同时分辨率减小,在语义上更具有全局和上下文信息,但是小目标实蝇的类间差距不明显,如果将 32 倍下采样后的特征图直接输入到颈部网络做特征融合难以关注到有效的实蝇语义信息,为此本研究在 Backbone 的最后 1 层 ELAN 模块后引入 SimAM 注意力机制,并将超参数设置为 0.000 1, SimAM 注意力通过优化能量函数来获取每个神经元的 3D 权重,会减少周围其他特征对检测目标识别的干扰,能够更好地关注到实蝇语义信息,凸显其关键特征,同时其无参数数量的特性使得其计算开销与其他

注意力机制相比更少。

1.3.3 改进的 Neck 模块 YOLO v7 - tiny 模型采用大量标准卷积以提取图像的特征信息,然而,标准卷积操作可能导致较大的时间消耗,使用过多的标准卷积对图像进行特征提取时,会造成参数数量的累积、特征的冗余,网络层数越深,影响越大。轻量级网络如 Xception<sup>[30]</sup> 和 MobileNet<sup>[31]</sup> 采用了深度可分离卷积,以有效解决标准卷积的时间消耗问题。然而,深度可分离卷积可能导致通道信息的丢失,从而降低模型的精度。为解决这一问题, Li 等提出的 GSConv 采用了标准卷积、深度可分离卷积以及 Shuffle 相结合的方式<sup>[32]</sup>,其计算成本仅为标准卷积

的 60% ~ 70%, 同时保持了与标准卷积相媲美的效果。GSConv 的结构如图 6 - a 所示: 输入进来的特征图的通道数为  $C_i$ , 输出的特征图通道数为  $C_o$ , 首先特征经过一个标准卷积通道数变为  $\frac{C_o}{2}$ , 经过深度可分离卷积时通道数不变, 将 2 个卷积后的特征图进行拼接 (concat), 最后对融合的特征图进行 Shuffle, 加强提取到的语义信息, 提升图像特征的表达能力。

本研究为了降低模型的参数量和计算量, 对 Neck 网络进行了改进, 设计了 2 个模块 GSCBL 和 GSELAN: 首先如图 6 - b 所示, 利用 GSConv 替换 CBL 模块中的普通卷积, 使用 GSCBL 进行 Neck 网络中的下采样操作, 最大程度保证语义信息的完整; 同时对 ELAN 模块进行改进, 如图 6 - c 所示将原来的 ELAN 模块中的 CBL 模块更新为 GSCBL。利用 GSCBL 和 GSELAN 模块改进 Neck 结构, 既减轻了模型的复杂度, 又能保持模型的精度。



1.3.4 改进的损失函数 在 YOLO v7 - tiny 损失函数计算公式(1)中, 目标置信损失和分类损失采用的是 BCEWithLogitsLoss, 边框损失则是利用 CIOU Loss 进行计算<sup>[33]</sup>。CIOU Loss 的定义如公式(8)所示:

$$\alpha = \frac{v}{1 - IoU + v}; \quad (6)$$

$$v = \frac{4}{\pi^2} \left[ \arctan\left(\frac{w^{gt}}{h^{gt}}\right) - \arctan\left(\frac{w}{h}\right) \right]^2; \quad (7)$$

$$L_{CIOU} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2} + \alpha v. \quad (8)$$

在图 7 - a 中, 红色框表示预测框, 而黄色框表示标注框。在公式(6)至公式(8)中,  $IoU$  指标注框与预测框的交并比;  $w^{gt}$ 、 $h^{gt}$  分别表示标注框的宽与高;  $w$ 、 $h$  分别表示预测框的宽、高;  $b$  代表预测框,  $b^{gt}$  代表标注框。此外,  $c$  表示能够同时包含预测框和标注框的对角线距离;  $\alpha$  表示平衡参数;  $\rho$  表示预测框和标注框中心点的欧氏距离;  $v$  用来衡量长宽比是否一致。由式(7)可以得知当预测框与标注框的

长宽比值相同时,  $v$  的值为 0, 此时 CIoU 的损失函数的有效性会受到影响, 从而对不同尺度物体的敏感性差异很大, 尤其对小目标定位不友好。而实蝇图像中存在大量小目标, 采用此损失函数极易出现漏检情况。

因此, 本研究引入 MPDIoU Loss, MPDIoU 是基

于最小点距离的边界框相似性比较设计的一种度量方式<sup>[34]</sup>, 该方法考虑到了 CIoU 中长宽比偏差的问题, MPDIoU 损失对不同尺度的物体不敏感, 更适合测量小物体预测框和真实框之间的相似性, 使得模型能聚焦于低质量的锚框, 提高模型的定位能力, 并且简化了计算过程。

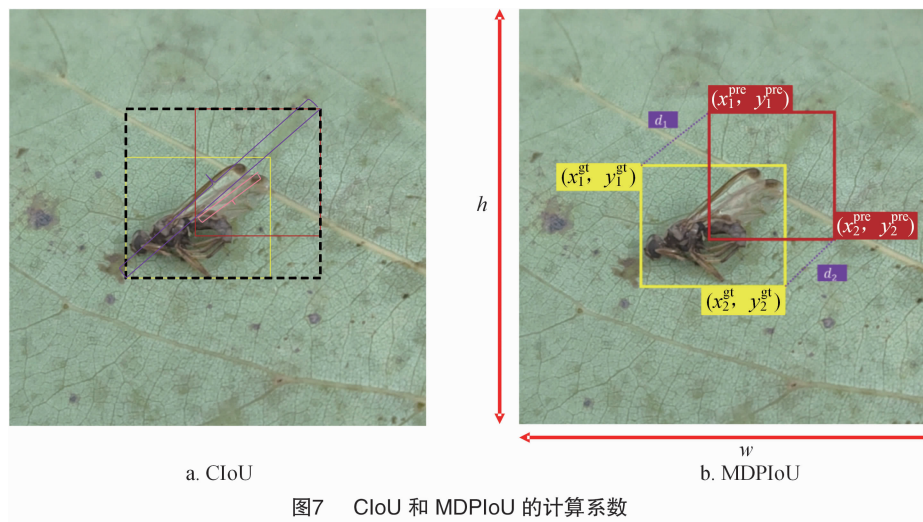


图7 CIoU 和 MPDIoU 的计算系数

MPDIoU 相关计算公式如下:

$$L_{\text{MPDIoU}} = 1 - \text{MPDIoU}; \quad (9)$$

$$\text{MPDIoU} = \text{IoU} - \frac{d_1^2}{h^2 + w^2} - \frac{d_2^2}{h^2 + w^2}; \quad (10)$$

$$d_1^2 = (x_1^{\text{pre}} - x_1^{\text{gt}})^2 + (y_1^{\text{pre}} - y_1^{\text{gt}})^2; \quad (11)$$

$$d_2^2 = (x_2^{\text{pre}} - x_2^{\text{gt}})^2 + (y_2^{\text{pre}} - y_2^{\text{gt}})^2. \quad (12)$$

图 7-b 展示了 MPDIoU 的相关指标。其中  $d_1$  表示标注框与预测框两者左上角的距离差值;  $d_2$  表示两者右下角的距离差值;  $(x_1^{\text{pre}}, y_1^{\text{pre}})$  是预测框的左上角点坐标,  $(x_2^{\text{pre}}, y_2^{\text{pre}})$  是预测框右下角点坐标;  $(x_1^{\text{gt}}, y_1^{\text{gt}})$  是标注框左上角点坐标,  $(x_2^{\text{gt}}, y_2^{\text{gt}})$  是标注框右下角点坐标。  $h$ 、 $w$  分别为输入图像的长度和宽度。 MPDIoU 简化了 2 个边界框之间的相似性比较, 同时可以适应重叠或非重叠的边界框回归, 能更好地关注到小目标实蝇。

## 2 结果与分析

### 2.1 试验环境与配置

本研究所有试验的运行时间在 2023 年 8 月至 10 月, 试验地点在江西农业大学, 运行平台使用 Ubuntu 18. 04. 5 系统, 深度学习的试验环境为 Pytorch 2. 0. 1, Cuda 11. 6。所有的训练和测试试验都是在一张 NVIDIA RTX 2080Ti GPU 和 Intel® Xeon® Silver 4116 CPU @ 2. 10 GHz 的服务器上进

行。所有模型都采用 SGD 优化器进行训练, 相关参数统一, 详见表 3。

表 3 试验参数配置

参数	值
每轮训练周期	300
批量大小	8
图像输入尺寸	640 × 640
学习率	0. 01
动量	0. 937
权重衰减	0. 000 5

### 2.2 模型评估指标

为定量分析 KSGM-YOLO 模型对实蝇的检测效果, 本研究采用检测精度与速率作为模型的衡量指标, 其中指标包含 Precision ( $P$ )、Recall ( $R$ )、Mean Average Precision ( $mAP$ )、Frames Per Second (FPS), 本研究以 IoU 为 0. 5 时取  $mAP$ 。对于目标检测任务  $mAP$  是衡量模型检测精度的最佳指标, 各指标的计算公式如下:

$$P = \frac{TP}{TP + FP}; \quad (13)$$

$$R = \frac{TP}{TP + FN}; \quad (14)$$

$$AP = \int_0^1 P(R) dR; \quad (15)$$

$$mAP = \frac{1}{N} \sum_{i=0}^N AP_i \times 100\% ; \quad (16)$$

$$FPS = \frac{FrameNum}{ElapsedTime} . \quad (17)$$

其中,  $TP$  为预测正确的正样本数量;  $FN$  为预测错误的负样本数;  $FP$  为预测错误的正样本数;  $N$  表示实蝇的类别数;  $FrameNum$  表示检测图片总数;  $ElapsedTime$  表示检测所花费的总时间。

### 2.3 对比损失函数的收敛情况

在相同的网络模型和试验环境下,本研究验证了 YOLO v7 - tiny 损失函数的收敛性。图 8 展示了 2 种损失函数随着迭代次数的变化曲线。其中, 2 条曲线分别代表了在边框损失使用 MPDIoU 和 CIoU 时的边框损失情况。

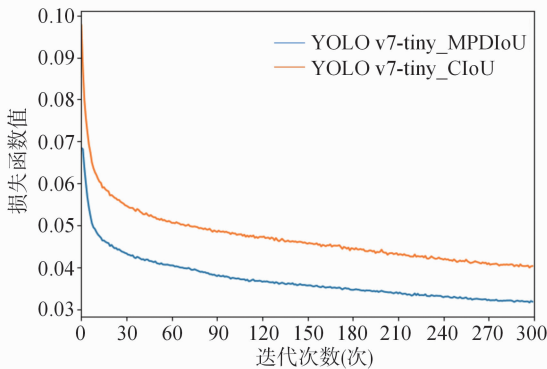


图8 使用 MPDIoU Loss 与 CIoUloss 的 train box 损失对比

从图 8 中可以观察到,随着迭代次数的增加,2 种边框损失最终都达到了收敛状态。然而,相较于 CIoU, MPDIoU 的损失值更小,最终能稳定在 0.033,

表现出更高的稳定性。因此,选择 MPDIoU 作为本研究的边框损失函数,能使得模型对小目标实蝇的检测性能得到提升。

### 2.4 消融试验

为了验证每一个提出的改进模块对网络整体性能的提升,在实蝇数据集上进行消融试验,本消融试验分为 8 组,“√”表示引入此改进方法,各组试验的结果如表 4 所示。对消融试验结果进行分析,第 1 组为原始 YOLO v7 - tiny 检测模型,第 2 组为对实蝇数据集重聚类先验框,  $mAP$  上升了 0.8 百分点,说明锚框重设能够增加对不同尺度下实蝇目标的定位能力;第 3 组为在 Neck 中使用了 GSConv 进行改进,虽然  $mAP$  略微下降,但是换来了参数量的下降;第 4 组在 Backbone 中引入了 SimAM 注意力机制,  $mAP$  上升了 1.3 百分点,说明 SimAM 能够提升模型捕捉小目标实蝇语义信息的能力;第 5 组使用了 MPDIoU 损失函数,使得模型能够更好地对待检测实蝇进行匹配,  $mAP$  上升了 1.0 百分点。第 6 组在第 2 组的基础上在网络的 Neck 使用了 GSConv,对检测模型做了轻量化的改进;第 7 组试验在第 6 组的基础上添加了 SimAM 注意力机制,  $mAP$  上升了 1.7 百分点;第 8 组是本研究所提的最终模型,总体看来其  $mAP$  较原始的 YOLO v7 - tiny 上升了 2.3 百分点,达到了 71.2%,且参数量下降了 6%,且 FPS 达到 31,能够满足在边缘 GPU 对小目标实蝇害虫实时检测的要求。

表 4 消融试验结果

组别	NEW Anchor	New Neck	SimAM	MPDIoU	$mAP$ (%)	FPS (帧/s)	参数量 (M)
1					68.9	34	6.02
2	√				69.7	34	6.02
3		√			68.2	33	5.64
4			√		70.2	32	6.02
5				√	69.9	34	6.02
6	√	√			68.8	33	5.64
7	√	√	√		70.6	31	5.64
KSGM - YOLO	√	√	√	√	71.2	31	5.64

### 2.5 对比试验

为了更全面地验证本研究提出的模型的有效性,本研究进行了与其他网络的对比试验,采用相同的训练配置。在这些试验中将本研究提出的最终模型与其他模型进行详细比较,并使用 GFLOPS

指标来评估网络的性能。GFLOPS 是计算复杂度的关键指标,直接影响模型的执行时间。模型的 GFLOPS 值越低,计算效率越高,尤其适用于在资源有限的设备上实时应用。具体的比较结果见表 5。



表 5 对比试验结果

模型	<i>mAP</i> (%)	FPS (帧/s)	参数量 (M)	GFLOPS
KSGM - YOLO	71.2	31	5.64	12.10
YOLO v3 - tiny	68.3	29	8.85	13.30
YOLO v5s	68.7	33	7.07	16.50
Faster - RCNN	66.1	27	41.14	206.68
YOLO v7 - tiny	68.9	34	6.02	13.20

相较于主流的轻量级网络 YOLO v3 - tiny 和 YOLO v5s, KSGM - YOLO 在参数量方面分别下降了 36.3% 和 20.2%, 计算量下降了 9.0% 和 26.5%, *mAP* 提升了 2.9 个百分点和 2.5 个百分点, FPS 相较于 YOLO v3 - tiny 和 Faster - RCNN 有细微的提升。与原始网络 YOLO v7 - tiny 对比, 本研究模型在参数量与计算量上分别下降了 6.3% 和 8.3%, 保证了改进网络的轻量级程度。同时, *mAP* 上升了 2.3 百分点, 这进一步提升了检测精度。尽管轻量化改进导致了 FPS 有一定程度的损失, 但仍然保持在 30 以上, 不影响实时检测任务。

这些详细的对比结果充分展示了本研究提出

的 KSGM - YOLO 模型在各方面的优越性能, 为我在实际应用中的可行性提供了有力支持。

## 2.6 模型效果可视化

为了更直观地体现 KSGM - YOLO 的有效性, 本研究采用 Grad - CAM 来比较初始模型和改进后模型的检测结果<sup>[35]</sup>。

图 9 中的红色部分表示模型较为关注的区域。相对于简单的背景, 第 2 行叶片背景下的实蝇更容易受到背景干扰, KSGM - YOLO 能够更好地将计算资源集中在目标区域, 并抑制非目标区域占用算力, 使其更加关注实蝇所在的区域。图 9 直观地验证了改进模型的有效性。通过图 9 的观察, 可以清晰地看到红色部分在实蝇周围形成了一个明显的高关注度区域。这表明 KSGM - YOLO 在处理复杂背景下的目标检测任务时, 能够更有效地聚焦于目标区域, 从而提高检测精度和减少对背景的误判。这一直观的展示进一步强化了本研究所提模型在应对复杂环境下目标检测中的优越性能。这也为读者提供了对模型内部运行机制的直观理解, 增强了文中方法的可解释性。

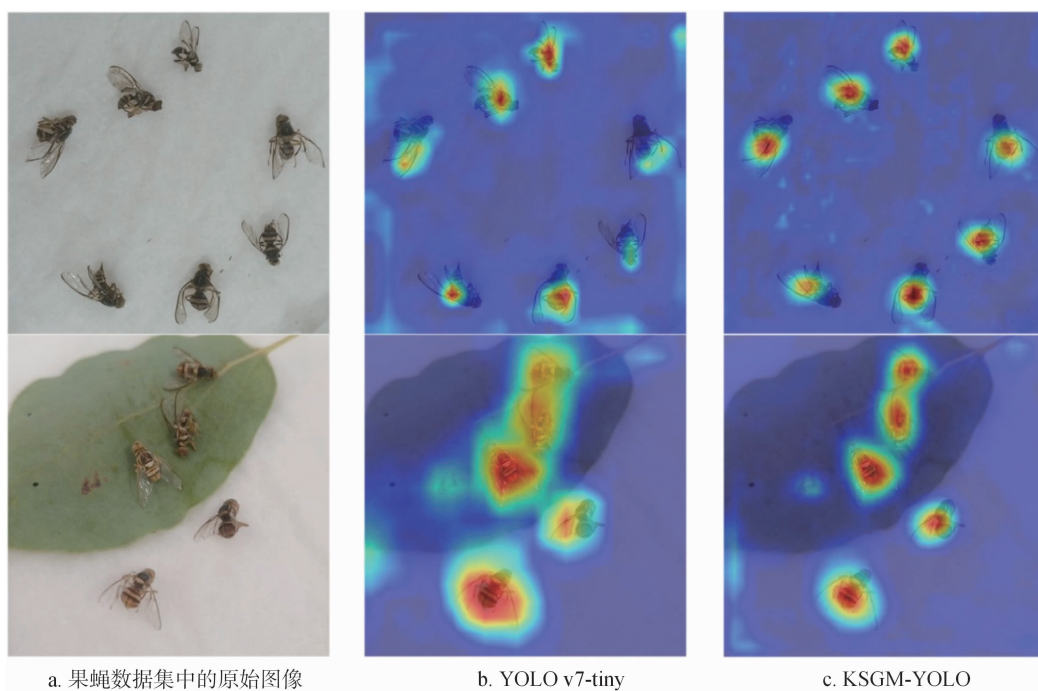


图 9 用 Grad-CAM 进行可视化

## 3 结论与讨论

本研究提出的 KSGM - YOLO 模型基于 YOLO v7 - tiny, 旨在实现对实蝇害虫的实时检测。首先, 本研究介绍了实蝇数据集 *Drosophila - Four* 和

YOLO v7 - tiny, 然后分别介绍了 KSGM - YOLO 的创新性部分: 使用实蝇数据集锚框重设算法对实蝇数据集的标注框进行聚类, 在 Backbone 中引入 SimAM 注意力机制, 通过 GSConv 重新设计颈部网络的下采样模块, 以及采用 MPDIoU 损失函数改进

模型原有损失函数。为验证模型改进的有效性,研究使用自制的 *Drosophila* - Four 数据集进行了消融试验和对比试验。

KSGM - YOLO 在试验中取得了令人满意的结果,达到了 71.2% 的 *mAP* 和 31 FPS 的运行速度。相较于原始的 YOLO v7 - tiny,提出的模型实现了 8.3% 的 GFLOPS 降低和 6.3% 参数数量的减少。这表明 KSGM - YOLO 在精度和速度上均取得了显著的提升,且通过轻量级优化进一步降低了模型的计算负担。

#### 参考文献:

- [1] 贺梅英,王 维. 中国蔬菜出口贸易潜力及贸易效率研究[J]. 北方园艺,2023(21):144 - 150.
- [2] Deng H, Cai X, Yin C L, et al. UDA - FlyRecog: unsupervised domain adaptation for *Drosophila* cross - domain recognition model [J]. Journal of Stored Products Research,2023,104:102192.
- [3] Boniecki P, Koszela K, Piekarska - Boniecka H, et al. Neural identification of selected apple pests[J]. Computers and Electronics in Agriculture,2015,110:9 - 16.
- [4] 陈西亮,张佳华,艾天成. 基于支持向量机法提取江汉平原三湖农场棉蚜危害程度的空间分布[J]. 江苏农业科学,2016,44(9):157 - 162.
- [5] Liu T, Chen W, Wu W, et al. Detection of aphids in wheat fields using a computer vision technique [J]. Biosystems Engineering, 2016,141:82 - 93.
- [6] 彭莹琼,廖牧鑫,张永红,等. 基于 BP 神经网络模型的果实蝇自动分类系统[J]. 江西农业大学学报,2016,38(6):1205 - 1210.
- [7] Cheng X, Zhang Y H, Chen Y Q, et al. Pest identification via deep residual learning in complex background [J]. Computers and Electronics in Agriculture,2017,141:351 - 356.
- [8] Yang Z K, Yang X T, Li M, et al. Automated garden - insect recognition using improved lightweight convolution network [J]. Information Processing in Agriculture,2023,10(2):256 - 266.
- [9] 肖德琴,黄一桂,张远琴,等. 基于改进 Faster R - CNN 的田间黄板害虫检测算法[J]. 农业机械学报,2021,52(6):242 - 251.
- [10] Redmon J, Divvala S, Girshick R, et al. You only look once: unified, real - time object detection [C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas, NV, USA: IEEE,2016:779 - 788.
- [11] Redmon J, Farhadi A. YOLO9000: better, faster, stronger [C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Honolulu, HI, USA: IEEE,2017:7263 - 7271.
- [12] Farhadi A, Redmon J. YOLO v3: an incremental improvement [C]//Computer Vision and Pattern Recognition. Berlin/Heidelberg, Germany: Springer,2018,1804:1 - 6.
- [13] Bochkovskiy A, Wang C Y, Liao H Y M. YOLO v4: Optimal speed and accuracy of object detection [EB/OL]. (2020 - 04 - 23) [2023 - 09 - 18]. <http://arXiv.org/abs/2004.10934>.
- [14] 曾晏林,贺壹婷,蔺 瑶,等. 基于 BCE - YOLO v5 的苹果叶部病害检测方法[J]. 江苏农业科学,2023,51(15):155 - 163.
- [15] Amrani A, Sohel F, Diepeveen D, et al. Insect detection from imagery using YOLO v3 - based adaptive feature fusion convolution network [J]. Crop and Pasture Science,2023,74(6):615 - 627.
- [16] Song L M, Liu M Y, Liu S H, et al. Pest species identification algorithm based on improved YOLO v4 network [J]. Signal, Image and Video Processing,2023,17(6):3127 - 3134.
- [17] Wu S, Wang J P, Liu L, et al. Enhanced YOLO v5 object detection algorithm for accurate detection of adult *Rhynchophorus ferrugineus* [J]. Insects,2023,14(8):698.
- [18] Wang C Y, Bochkovskiy A, Liao H Y M. YOLO v7: Trainable bag - of - freebies sets new state - of - the - art for real - time object detectors [C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Vancouver, Canada: IEEE,2023:7464 - 7475.
- [19] 郑 果,姜玉松,沈永林. 基于改进 YOLO v7 的水稻害虫识别方法[J]. 华中农业大学学报,2023,42(3):143 - 151.
- [20] Buslaev A, Iglovikov V I, Khvedchenya E, et al. Albumentations: fast and flexible image augmentations [J]. Information,2020,11(2):125.
- [21] Zhang H Y, Cisse M, Dauphin Y N, et al. Mixup: beyond empirical risk minimization [EB/OL]. (2018 - 04 - 27) [2023 - 09 - 18]. <http://arXiv.org/abs/1710.09412V2>.
- [22] Liu S, Qi L, Qin H F, et al. Path aggregation network for instance segmentation [C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Salt Lake City, UT, USA: IEEE,2018:8759 - 8768.
- [23] Wang C Y, Yeh I H, Liao H Y M. You only learn one representation: Unified network for multiple tasks [EB/OL]. (2021 - 05 - 10) [2023 - 09 - 17]. <https://arXiv.org/abs/2015.04206V1>.
- [24] Lin T Y, Maire M, Belongie S, et al. Microsoft COCO: common objects in context [C]//Computer Vision - ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6 - 12, 2014, Proceedings, Part V 13. Springer International Publishing, 2014: 740 - 755.
- [25] Rezatofighi H, Tsoi N, Gwak J Y, et al. Generalized intersection over union: A metric and a loss for bounding box regression [C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Long Beach, CA, USA: IEEE,2019:658 - 666.
- [26] Arthur D, Vassilvitskii S. k - means ++: the advantages of careful seeding [C]//Soda. New Orleans. 2007,7:1027 - 1035.
- [27] Hu J, Shen L, Sun G. Squeeze - and - excitation networks [C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Salt Lake City, UT, USA: IEEE,2018:7132 - 7141.
- [28] Woo S, Park J, Lee J Y, et al. CBAM: convolutional block attention module [C]//Proceedings of the European Conference on Computer Vision (ECCV). Cham: Springer International Publishing, 2018: 3 - 19.
- [29] Yang L, Zhang R Y, Li L, et al. Simam: a simple, parameter - free attention module for convolutional neural networks [C]//International Conference on Machine Learning, PMLR. New York,

何泽坤,黄操军,许善祥,等. 基于改进 YOLO v5 的水稻害虫识别方法[J]. 江苏农业科学,2025,53(5):223-230.

doi:10.15889/j.issn.1002-1302.2025.05.026

# 基于改进 YOLO v5 的水稻害虫识别方法

何泽坤,黄操军,许善祥,袁鑫宇,高学文,赵清

(黑龙江八一农垦大学信息与电气工程学院,黑龙江大庆 163319)

**摘要:**在水稻生长过程中,及时识别害虫种类在虫情监测环节至关重要。由于害虫种类繁多,人工识别效率低且主观性强,传统的水稻害虫识别模型易受环境因素影响,因此提高水稻害虫识别的准确性是亟待解决的问题。以水稻试验田中获取的害虫图像为对象,提出了一种基于 MSRCF 算法的图像增强和改进 YOLO v5 网络的水稻害虫识别模型。在图像预处理部分,采用 MSRCF 算法和双边滤波算法来增强因光照不均而受到影响的害虫图像。在改进 YOLO v5 网络部分,在骨干网络中引入了 SE 注意力机制,增强模型对害虫特征的理解能力,从而提升模型对各个通道害虫特征的辨别能力;引入  $\alpha$ -IoU 损失函数优化检测边界框,提高模型精度;并采用迁移学习策略训练模型,提升模型的学习效率和鲁棒性。通过消融试验可知,改进后的模型在水稻害虫图像识别上的精确率为 95.17%,召回率为 86.34%,平均精度均值为 91.30%,较原 YOLO v5 模型分别提高了 6.09、5.04、7.10 百分点。与其他模型相比,改进后的模型平均精度均值较 Faster R-CNN、原 YOLO v5 模型、YOLO v7 分别提高了 21.45、7.10、3.06 百分点。改进算法明显优于其他算法,能更好地适应水稻害虫的特殊形态和农田复杂环境,对水稻害虫的防控具有重要意义。

**关键词:**农田复杂环境;YOLO v5 模型;害虫种类识别;害虫防治;深度学习

**中图分类号:**S126;TP391.41 **文献标志码:**A **文章编号:**1002-1302(2025)05-0223-08

农业发展对国家稳定至关重要,也是人类生存和发展的基础<sup>[1]</sup>。科技进步显著提升了农产品产量,但病虫害成为影响农产品的关键因素。如何利用科学技术进行病虫害防治,是目前农业发展亟需解决的问题<sup>[2]</sup>。

随着人工智能技术的发展,图像检测技术也被

引入农业领域中。已有研究包括唐小煜等提出一种改进 YOLO v5 模型用于对小目标害虫识别定位,平均检测精度达 92%<sup>[3]</sup>。李滨等提出了一种轻量化 YOLO v5 模型,使模型体积降至 7.38 MB,且使模型的精度提升<sup>[4]</sup>。段新涛等提出了一种 YOLO v4-Corn 夏玉米主要害虫检测方法,模型平均检测精度可达 95.89%<sup>[5]</sup>。吴子炜等提出了一种轻量模块以及相应改进 YOLO v5 模型的方法,改进后的模型参数量仅为原模型的一半,对 14 类害虫的平均识别精度相较原模型提高 3.2 百分点<sup>[6]</sup>。彭红星等提出了一种兼顾检测精度和效率的害虫检测模型 YOLOF-PD,改进后的 YOLOF-PD 模型在 FieldPest5 测试集

收稿日期:2024-04-18

基金项目:黑龙江省现代农业产业技术系统创新推广体系项目。

作者简介:何泽坤(2000—),男,安徽铜陵人,硕士研究生,主要从事农业装备智能化研究。E-mail:1036869635@qq.com。

通信作者:黄操军,博士,教授,博士生导师,主要从事农业装备智能化研究。E-mail:huangcaojun@163.com。

2021;11863-11874.

[30]Chollet F. Xception: deep learning with depthwise separable convolutions[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Honolulu, HI, USA: IEEE, 2017: 1251-1258.

[31]Howard A G, Zhu M, Chen B, et al. Mobilenets: Efficient convolutional neural networks for mobile vision applications[EB/OL]. (2017-04-17)[2023-09-18]. <https://arxiv.org/abs/1704.04861>.

[32]Li H, Li J, Wei H, et al. Slim-neck by GSConv: A better design paradigm of detector architectures for autonomous vehicles[EB/OL]. (2022-08-17)[2023-09-18]. <https://arxiv.org/>

abs/2206.02424v2.

[33]Zheng Z H, Wang P, Liu W, et al. Distance-IoU loss: faster and better learning for bounding box regression[C]//Proceedings of the AAAI Conference on Artificial Intelligence. New York, 2020: 12993-13000.

[34]Siliang M, Yong X. Mpdious: a loss for efficient and accurate bounding box regression[EB/OL]. (2023-07-14)[2023-09-18]. <https://arxiv.org/abs/2307.07662>.

[35]Selvaraju R R, Cogswell M, Das A, et al. Grad-cam: visual explanations from deep networks via gradient-based localization[C]//Proceedings of the IEEE International Conference on Computer Vision. Venice, Italy: IEEE, 2017: 618-626.