

# 基于改进 YOLOv8m 的稻田害虫识别方法

谭泗桥<sup>1,2</sup>, 陈 涵<sup>1</sup>, 朱 磊<sup>1,2</sup>, 孙浩然<sup>1</sup>, 张政兵<sup>3</sup>, 尹 丽<sup>3</sup>, 黄婉婉<sup>1\*</sup>

(1. 湖南农业大学信息与智能科学技术学院, 长沙 410128; 2. 湖南省农村农业信息化工程技术研究中心, 长沙 410128;  
3. 湖南省植保植检站, 长沙 411229)

**摘 要:**为解决现有基于机器视觉的稻田害虫监测过程中存在的检测速度慢、小目标检测精度低、害虫堆积遮挡时检测精度低以及样本不平衡等问题,该研究提出了一种基于改进 YOLOv8m 模型的稻田害虫识别方法 FieldSentinel-YOLOv8。该方法首先简化了 YOLOv8m 模型,并用双检测头代替三检测头,以减少小目标细粒度信息的丢失,降低模型的复杂度;其次将卷积注意力模块 CBAM (convolutional block attention module) 添加到 YOLOv8m,使模型抑制背景等一般特征信息,更加关注害虫区域,从而提高被遮挡害虫的识别准确率;最后使用 Focal-CIoU Loss 来替换 Ciou Loss (complete intersection over union),以减少样本类别不平衡对模型精度的影响。FieldSentinel-YOLOv8 模型的平均精度均值 (mean average precisoin) mAP<sub>0.5</sub> 为 73.64%,精确率为 65.43%,召回率为 75.17%,检测帧率为 199.88 帧/s。与原模型 YOLOv8m 相比,FieldSentinel-YOLOv8 的模型参数量从 25.86 M (million) 降到 10.34 M,所需浮点运算数从 79.10 G (1 G=10<sup>9</sup>) 降到 62.80 G,召回率、平均精度均值和检测帧率分别提升 7.05、2.72 个百分点和 52.73 帧/s。该研究还采用 Pest24 数据集作为源域,自建数据集作为目标域的迁移学习方法训练 FieldSentinel-YOLOv8 模型,得到高精度 FieldSentinelTransfer-YOLOv8 模型,进一步提升模型检测性能,使用迁移学习方法后, mAP<sub>0.5</sub> 再次提升 3.36 个百分点,达到 77.00%,精确率为 69.90%,召回率为 77.73%。在自建数据集上进行模型对比试验,结果表明,FieldSentinel-YOLOv8 模型具有较高的识别准确率及较强的鲁棒性,该模型的轻量化方法可为农业害虫的精准且快速识别提供技术参考。高精度 FieldSentinelTransfer-YOLOv8 模型精度的大幅提升,也表明迁移学习在农业害虫检测上提供了技术支持。

**关键词:** 虫害; 深度学习; 图像识别; YOLOv8m; 卷积注意力模块; Focal-CIoU; 迁移学习

doi: 10.11975/j.issn.1002-6819.202406125

中图分类号: S24

文献标志码: A

文章编号: 1002-6819(2025)-02-0185-11

谭泗桥, 陈涵, 朱磊, 等. 基于改进 YOLOv8m 的稻田害虫识别方法[J]. 农业工程学报, 2025, 41(2): 185-195. doi: 10.11975/j.issn.1002-6819.202406125 <http://www.tcsae.org>  
TAN Siqiao, CHEN Han, ZHU Lei, et al. Identifying pests in rice fields using improved YOLOv8m[J]. Transactions of the Chinese Society of Agricultural Engineering (Transactions of the CSAE), 2025, 41(2): 185-195. (in Chinese with English abstract) doi: 10.11975/j.issn.1002-6819.202406125 <http://www.tcsae.org>

## 0 引 言

害虫会导致大量农作物死亡,造成种植者收入和经济效益的大幅下降<sup>[1]</sup>。农药可以解决绝大部分害虫,但是如果种植者在农作物上大量使用农药,这对农作物和自然环境都有害,并且一种农药不能杀死所有种类的害虫,种植者只能对症下药,根据害虫的特性,喷洒针对此种害虫的农药,如胃毒或触杀剂农药可以用来防治咀嚼式口器害虫,具有内吸作用的农药可以用来防治刺吸式口器害虫<sup>[2]</sup>,所以检测害虫的种类对根除害虫、农作物经济增长、农作物绿色健康和保护自然环境具有重要意义。快速并且准确的检测害虫可以加快害虫的清除速度,现在田间害虫的识别大多由种植者基于昆虫的外部形态<sup>[3-4]</sup>进行肉眼观察,由于肉眼观察费时、费力、繁琐、

成本高、具备主观性,所以人工检测是一种相当缓慢和低效的方式。

随着人工智能技术的发展,目标检测变成计算机视觉中的一项基本任务,旨在对图像或视频中感兴趣的目标进行定位和分类<sup>[5]</sup>。近年来,目标检测任务已取得了显著的进展,并成为许多现实世界应用的关键组成部分,如农业机器人、安全监控和医疗诊断<sup>[6]</sup>。甘雨等<sup>[7]</sup>利用轻量化的 EfficientNet 来实现对害虫的快速识别,准确率达到了 69.5%,相比原模型准确率提升了 4.01 个百分点。王金星等<sup>[8]</sup>用 ResNeXt 作为主干网络去改进 MaskR-CNN 网络,并引入 Boundary 损失函数构建害虫检测模型,试验结果表明,相比于原始 MaskR-CNN 模型,该模型提高 4.21 个百分点。REZA 等<sup>[9]</sup>结合迁移学习和数据增强策略训练 Inception-v3 实现害虫种类识别。ANWAR 等<sup>[10]</sup>提出了 VGG16、VGG19、ResNet50 融合的集成害虫检测模型,实现了约 82.5% 的准确率。肖德琴等<sup>[11]</sup>提出一种改进的 FasterR-CNN 算法,采用 ResNet101 作为特征提取网络时,平均精度均值为 87.14%。虽然这些模型取得了出色的准确率结果,但是 FasterR-CNN、MaskR-CNN 等网络检测速度较慢,不能满足实时检测需求。而

收稿日期: 2024-06-19 修订日期: 2024-11-29

基金项目: 国家自然科学基金青年基金项目 (62403194, 62202163); 湖南省自然科学基金青年基金项目 (2022JJ40190)

作者简介: 谭泗桥, 博士, 教授, 硕士生导师, 研究方向为机器学习及智能表型技术。Email: [tsq@hunau.edu.cn](mailto:tsq@hunau.edu.cn)

\*通信作者: 黄婉婉, 博士, 讲师, 研究方向为图像处理与图像融合。Email: [hnnnydx1993@163.com](mailto:hnnnydx1993@163.com)

YOLO<sup>[12-19]</sup> (you only look once) 系列模型作为主流的目标检测网络之一, 检测速度更快、可扩展性更高, 因此更适合实际应用, YOLO 系列模型已被众多研究人员应用于农业害虫检测领域。赵辉等<sup>[20]</sup>在 YOLOv7 模型中把跨阶段局部瓶颈模块 CSP Bottleneck (cross stage partial bottle neck mudile) 与基于移位窗口 Transformer (swin transformer) 自注意力机制相结合, 让模型可以更好地获取密集害虫目标位置信息, 比原模型的精度高出 3 个百分点。DAI 等<sup>[21]</sup>在 YOLOv5m 网络中引入 SWinTR (swin transformer) 和一个新的 Transformer 机制, 并在主干中加入 ResSPP (在空间金字塔池化中加入残差结构, 命名为 ResSPP), 此外, 还加入权重特征融合, 该方法取得了 95.7% 的精确率。ZHANG 等<sup>[22]</sup>在 YOLOv8 网络中, 在 DenseBlock 的基础上对 C2F 模块进行改进, 选取 Mish 激活函数将 CBS (channel-wise batch normalization with sigmoid) 模块替换为 CBM 模块 (channel-wise batch normalization with mish), 让模型可以更好地学习数据的非线性特征, 有效解决算法训练过程中的梯度消失问题, 在 IP102 数据集上 mAP<sub>0.5</sub>、精确率和召回率上相比原模型分别提升了 2、1.3 和 3.7 个百分点, mAP<sub>0.5</sub> 达到 60.8%。HU 等<sup>[23]</sup>在 YOLOv5s 的基础上, 增加一个检测头以扩大检测尺度范围, 用双向特征金字塔网络 BiFPN (bi-directional feature pyramid network) 代替路径聚合网络 PANet (path aggregation network) 以提高特征融合效果, 并引入 Swin Transformer 自注意力机制, 所提出的模型的平均精度均值达到 79.8%。刘忠等<sup>[24]</sup>将 YOLOv8 颈部网络中的卷积模块替换为 GSCConv (group-shuffle convolution), 将 C2f 模块替换为 VoV-GSCSP (variety of view group shuffle cross stage partial network), 使模型在保持较高识别准确性的基础上降低计算复杂度, 并加入了高效多尺度注意力机制 EMA (efficient multi-scale attention), 提升模型对复杂环境中荷叶病虫害的特征提取能力, mAP<sub>0.5</sub> 比原 YOLOv8 模型提高了 1.6 个百分点。郭嘉璇等<sup>[25]</sup>提出了融入全局响应归一化 GRN (global response normalization) 注意力机制的 YOLOv5 农作物害虫识别模型, 并利用形状交并比 SIoU (shape intersection over union) 损失函数提高模型收敛速度和识别精度, 平均精度均值达到了 72.3%。WEI 等<sup>[26]</sup>用 FasterNet 模块替换了 YOLOv9 原来的主干结构, 从输入数据中提取特征, 并在整个网络的下游添加了反向残留移动块 iRMB (inverted residual mobile block) 注意机制模块, 改进后的 YOLOv9 型号命名为 YOLO-Fi, 该模型平均精度均值达到 86.2%。HU 等<sup>[27]</sup>在 YOLOv8s 的基础上进行优化, 引入基于内容的特征采样策略以获取更丰富的目标特征信息, 并采用分布移位卷积技术, 在提高检测精度的同时成功减小了模型的规模, 在 Pest24 数据集上平均精度均值达到了 71%。这些方法在害虫的检测中取得了一定的成功, 为实现害虫的精准检测提供了有力支撑。

综上所述, 现在基于深度神经网络技术的各类害虫

识别方法已取得诸多进展。但是, 对小目标害虫的检测仍然是一项具有挑战性的任务, 原因有 3 点: 首先, 害虫会快速移动导致难以检测, 这对实时检测提出了挑战, 模型需要更加轻量化来提高检测速度; 其次, 害虫体积较小、数量较多、分布不均且会存在害虫相互遮挡的情况, 从而影响模型的准确率; 最后, 样本类别不平衡会导致识别难度加大, 从而影响检测的准确性。因此, 为了解决这 3 点问题, 本研究提出一种基于 YOLOv8m 改进的害虫检测模型 FieldSentinel-YOLOv8, 以期能够更好地解决害虫识别过程中识别难度大、速度慢等问题。

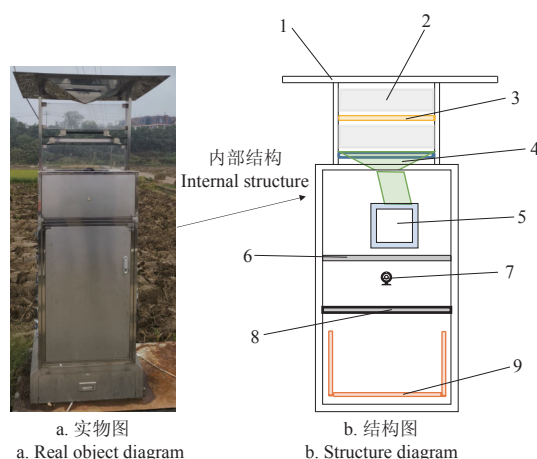
## 1 材料与方法

### 1.1 数据集

#### 1.1.1 数据集介绍

大多数试验研究使用的害虫数据集未考虑实际情况, 众多研究人员所使用的害虫数据集一般为 3 种类型, 第一种: 在实验室的环境下拍摄, 未考虑复杂的田间环境; 第二种: 此类数据集类似 IP102 数据集<sup>[28]</sup>, 虽然考虑了复杂的田间环境, 但是拍摄距离却离害虫特别近, 这样就算将摄像头放入田间, 只要害虫不在摄像头附近的农作物上, 那么检测准确率就会有所降低, 虽然使用这种数据集的模型在验证集和测试集上的精度较高, 但是难以实施在真实田间环境中; 第三种: 此类数据集类似 Pest24 数据集<sup>[29]</sup>, 由虫情监测装备先诱捕再拍摄, 可以在不用考虑复杂的田间环境下, 体现真实的田间害虫种类, 实施条件只需在田间放入虫情监测装备。

本研究使用的数据集类似于 Pest24。数据来源于湖南省植保植检站, 数据采集场地为邵阳县五峰镇的病虫害监测基地, 数据集由虫情监测装备采集而得, 虫情监测装备摆放位置为稻田附近。虫情监测装备大致结构如图 1 所示。



1. 遮雨棚 2. 撞击版 3. 诱虫灯 4. 接虫漏斗 5. 远红外杀虫仓 6. 电控翻转底座 7. 高清摄像头 8. 接虫板 9. 收集单元  
1. Rain shelter 2. Impact plate 3. Insect trap lamp 4. Insect trap funnel 5. Far infrared insecticidal warehouse 6. Electrically actuated flip-up base 7. High-definition camera 8. Insect collection plate 9. Collection unit

图 1 虫情监测装备

Fig.1 Insect monitoring equipment

虫情检测装备作业时, 紫外线诱虫灯发出令害虫敏感的光线致使害虫飞扑, 撞击玻璃屏落到下端漏斗顺着



打开的落虫通道滑进杀虫仓，杀虫仓利用远红外加热处理害虫，活虫落入后 3~5 min 内会有不少于 95% 的害虫被杀死，活下来的害虫也将会变得行动缓慢或无法行动，之后害虫会通过电控翻转底板到达接虫板，高清摄像头对害虫进行拍摄。只需将虫情监测装备放在稻田农作物附近，采集出来的数据集不用考虑田间复杂情况，也能反应农作物附近出现的害虫，虫情监测装备可以像田间哨兵（field sentinel）一样去监控害虫。

数据集共有 834 张，共标注了 12 类虫子（螟蛾、鳞翅目、隐翅虫、灯蛾、二化螟、水龟甲、其他昆虫、鞘翅目、步甲、蜡、稻螟蛉、蝼蛄），数据集样本如图 2 所示，数据集中的害虫体型大多都比较小，比如：螟蛾、隐翅虫等大多只有几毫米，并且有些数据中的害虫较为密集，图片的分辨率为 4 024 像素×3 036 像素，数据集的时间跨度为 2019 年 8 月 30 日—2020 年 1 月 7 日，数据集以 8:1:1 的比例划分训练集、验证集和测试集。

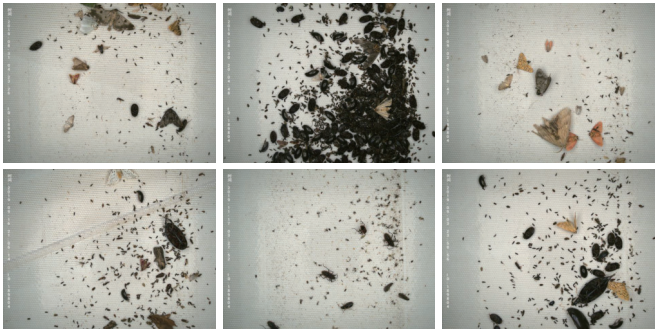


图 2 害虫样本  
Fig.2 Samples of pests

1.1.2 数据集构建

数据集使用 Labelling 工具进行图像标注工作，生成.txt 格式标签文件。数据增强是计算机视觉中使用的一种重要技术，用于增加训练数据的规模和多样性，可提高检测模型的泛化性能<sup>[30]</sup>。本研究使用了 3 种数据增强技术来增强数据集：垂直翻转、随机调整大小和随机旋转，更多数据增强的细节可以参考 SU 等<sup>[31]</sup>的方法。

图 3 显示了数据增强前后的图像。

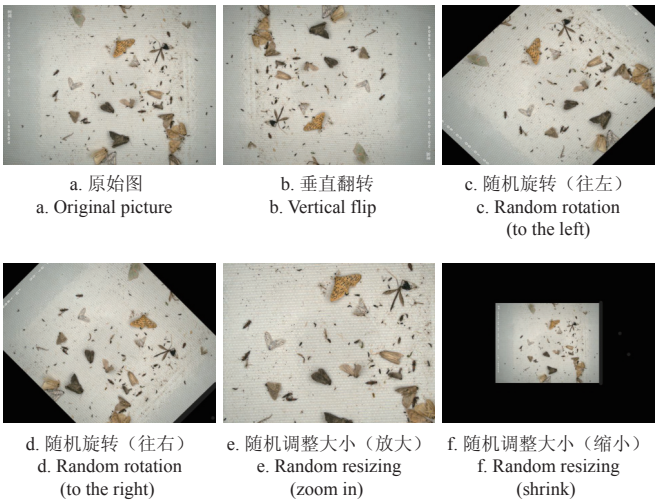


图 3 数据增强前后的图像  
Fig.3 Images before and after data augmentation

对于数据集中的每一张图像，都进行了一次增强，数据增强后共有 1 668 张图片（其中训练集 1 334 张、验证集 167 张、测试集 167 张）。通过应用这些数据增强技术，本研究能够训练出一个更稳定和准确的模型来检测害虫。数据增强后每种害虫的标签数量如表 1 所示，可以发现二化螟、稻螟蛉和蝼蛄的标签数量太少，导致验证集和测试集中并未出现这 3 类害虫，所以本研究选择螟蛾、鳞翅目、隐翅虫、灯蛾、水龟甲、其他昆虫、鞘翅目、步甲和蜡这 9 类害虫做识别任务。

| 表 1 每种害虫的标签数量<br>Table 1 Number of labels for each pests |                                    |                                      |                                |
|--|------------------------------------|--------------------------------------|--------------------------------|
| 检测目标<br>Objects for detection                            | 训练集标签数量<br>Number of training sets | 验证集标签数量<br>Number of validation sets | 测试集标签数量<br>Number of test sets |
| 螟蛾 <i>Pyrallidae</i>                                     | 328                                | 30                                   | 46                             |
| 鳞翅目 <i>Lepidoptera</i>                                   | 1 869                              | 236                                  | 263                            |
| 隐翅虫 <i>Staphylinidae</i>                                 | 2 345                              | 269                                  | 272                            |
| 灯蛾 <i>Arctiidae</i>                                      | 414                                | 40                                   | 54                             |
| 二化螟 <i>Chilo suppressalis</i>                            | 4                                  | 0                                    | 0                              |
| 水龟甲 <i>Hydrophilidae latreille</i>                       | 2 009                              | 316                                  | 227                            |
| 其他昆虫 <i>Other insects</i>                                | 514                                | 64                                   | 62                             |
| 鞘翅目 <i>Coleoptera</i>                                    | 798                                | 126                                  | 135                            |
| 步甲 <i>Carabidae</i>                                      | 961                                | 143                                  | 82                             |
| 蜡 <i>Pentatomidae</i>                                    | 338                                | 45                                   | 33                             |
| 稻螟蛉 <i>Naranga aenescens</i>                             | 13                                 | 0                                    | 0                              |
| 蝼蛄 <i>Gryllotalpidae orientalis</i>                      | 2                                  | 0                                    | 0                              |

1.2 轻量化害虫识别模型—FieldSentinel-YOLOv8

1.2.1 YOLOv8 网络模型

YOLO 算法通过实时性、一步到位、多尺度特征融合、预测框设计和多任务学习等创新点，实现了高效准确的目标检测，成为目标检测领域的重要算法之一。YOLOv8 是 YOLO 的第 8 个版本，它是在 YOLOv5 的基础上进行了改进。

1) 主干网络和颈部网络。YOLOv8 使用了与 YOLOv5 类似的主干，只是在 CSPLayer 上做了一些更改，现在称为 C2f（cross-stage partial bottleneck with two convolutions）模块。C2f 模块将高层特征与上下文信息相结合，以提高检测精度，C2f 模块针对不同比例模型调整了通道数量，从而显著提高了模型性能。此外，颈部网络弥补了骨干网络输出的特征表示和头部网络预测之间的差距。

2) 检测头。与 YOLOv5 相比，检测头 Detect 发生了较大的变化，取而代之的是目前流行的解耦头结构。

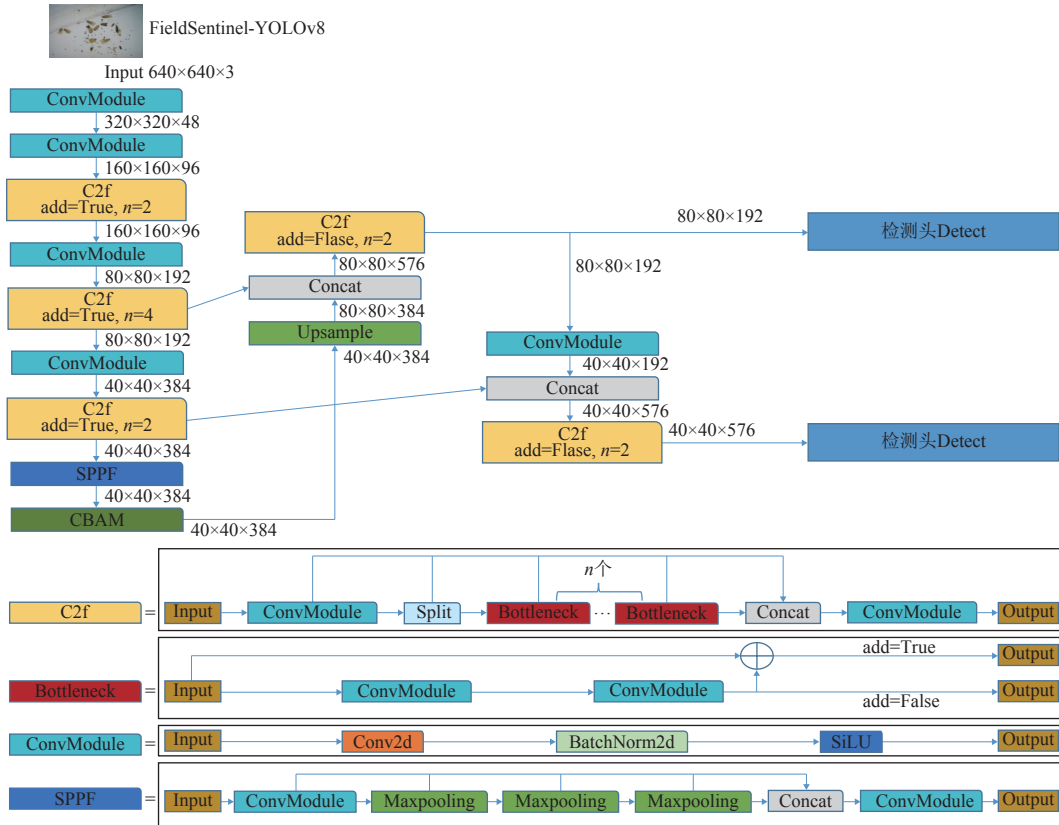
3) 损失函数。YOLOv8 的损失函数计算有 2 个分支：分类损失和回归损失，分类损失采用二元交叉熵损失，BCE（binary cross entropy）损失函数，而回归损失是由分布焦点损失 DFL（distribution focal loss）和 CIoU 结合使用，这样预测框就可以更加的符合真实框<sup>[32]</sup>。

1.2.2 FieldSentinel-YOLOv8 介绍

改进一：简化模型。网络深度的增加，通常导致特

征图的尺寸减小, 这种下采样过程势必造成空间信息的丢失, 使得小目标在高层特征图中可能只剩下很少的像素, 难以捕捉细节, 从而使得小目标的一些细粒度的特征信息丢失<sup>[33]</sup>。因此, 较深的网络层不适应于检测害虫目标。为了解决网络层数增加带来的问题, 本研究选择删除 YOLOv8m 主干网络中的最后一个 ConvModule 层

和 C2f 层, 保留快速空间金字塔池化 SPPF (spatial pyramid pooling - fast) 层, 删除颈部网络中最后一个 ConvModule 层之后的所有卷积, 再将 YOLOv8m 的三检测头改为双检测头。通过以上这些操作将 YOLOv8m 简化, 这些操作统一命名为 SMO (simplify model operations), 具体的简化方式如图 4 所示。



注: CBAM 为卷积注意力模块; Concat 为连接操作; Split 为切片操作; Conv2d 为卷积操作; BatchNorm2d 为批归一化操作; SiLU 为激活函数; Maxpooling 为最大池化操作。  
Note: CBAM is convolutional block attention module; Concat is concatenate operation; Split is a slicing operation; Conv2d is a convolution operation; BatchNorm2d is a batch normalization operation; SiLU is the activation function; Maxpooling is the maximum pooling operation.

图 4 FieldSentinel-YOLOv8 模型  
Fig.4 FieldSentinel-YOLOv8 model

改进二: 卷积注意力模块 CBAM。为动态地强调重要的特征并抑制不重要的特征来增强模型的代表能力, 本研究在 YOLOv8m 中加入了卷积注意力模块 CBAM<sup>[34]</sup>, 卷积注意力模块 CBAM 的结构如图 5 所示, CBAM 通过两个阶段的注意力机制来实现这一点:

1) 通道注意力机制: 首先, CBAM 通过全局平均池化和全局最大池化来捕获不同通道的特征统计信息。然后, 通过这两个池化层的输出, 使用全连接层和 ReLU 激活函数来生成通道权重。这些权重能够突出那些对识别任务更有信息量的通道, 同时抑制那些包含较少有用信息的通道。

2) 空间注意力机制: 在通道注意力的基础上, CBAM 进一步通过空间注意力机制来聚焦于图像中的关键区域。这一步通过在通道注意力输出的特征图上应用空间卷积层来实现, 以突出图像中对目标检测更为重要的局部区域。

通过这种双注意力机制, CBAM 能够使 YOLOv8m 模型更加关注于图像中害虫的特征, 尤其是在害虫被背景或其他物体遮挡时。这种关注机制有助于模型更好地

区分害虫与背景, 从而提高对被遮挡害虫的识别准确率。CBAM 整体流程如下: 首先, 特征图  $F$  经过通道注意力模块  $M_c$  得到  $M_c(F)$ ,  $M_c(F)$  的计算如式 (1), 特征图  $F$  与  $M_c(F)$  相乘得到特征图  $F'$ ; 其次, 特征图  $F'$  经过空间注意力模块  $M_s$  得到  $M_s(F')$ ,  $M_s(F')$  的计算如式 (2), 特征图  $F'$  与  $M_s(F')$  相乘得到特征图  $F''$ ; 最后, 特征图  $F$  与特征图  $F''$  相加得到输出  $F^*$ 。

$$M_c(F) = \sigma(MLP(Avgpool(F)) + MLP(Maxpool(F))) \quad (1)$$

$$M_s(F') = \sigma(f^{7 \times 7}([Avgpool(F'); Maxpool(F')])) \quad (2)$$

式中  $\sigma$  为 Sigmoid 激活函数;  $MLP$  为多层感知机,  $Avgpool$  为平均池化操作;  $Maxpool$  为最大池化操作;  $f^{7 \times 7}$  为滤波器大小为  $7 \times 7$  的卷积操作;  $[\cdot]$  为特征图的连接操作。

改进三: Focal-CIoU 损失函数。在进行目标检测任务时, 数据集中某类目标数量较少时会导致此类目标

别难度加大，为了解决这一问题，LIN 等<sup>[35]</sup>提出了 Focal Loss，用来解决正负样本类别不平衡时识别难度较大的问题，Focal Loss 通过动态调整损失函数中易分类样本的权重，减少易分类样本的损失贡献，增加难分类样本的损失贡献，从而使得模型更加关注那些难以分类的样本。本研究也将其中 Focal Loss 和 CIoU 相结合，引入 Focal-CIoU 来替换掉 CIoU，Focal-CIoU 定位损失有助于降低样本不平衡对检测结果的影响，提升模型的整体性能， $L_{Focal-CIoU}$  计算式为：

$$L_{Focal-CIoU} = IoU^\gamma L_{CIoU} \quad (3)$$

$$IoU = \frac{|A \cap B|}{|A \cup B|} \quad (4)$$

$$L_{CIoU} = 1 - IoU + \frac{(x - x_{gt})^2 + (y - y_{gt})^2}{(W_g^2 + H_g^2)} + \alpha v \quad (5)$$

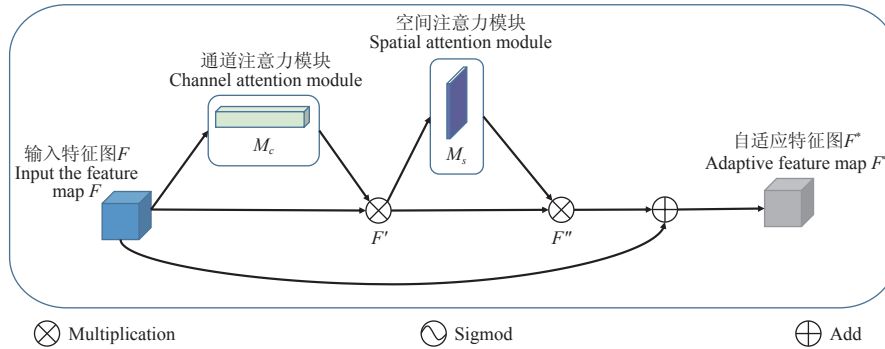
$$v = \frac{4}{\pi^2} \left( \tan^{-1} \frac{w}{h} - \tan^{-1} \frac{w_{gt}}{h_{gt}} \right)^2 \quad (6)$$

$$\alpha = \frac{v}{(1 - IoU) + v} \quad (7)$$

式中  $L_{Focal-CIoU}$  为 Focal-CIoU 损失函数， $L_{CIoU}$  为 CIoU 损失函数， $IoU$  为交并比； $v$  用来度量预测框和目标框之间

长宽比的一致性； $\alpha$  是一个平衡参数，用来动态调整  $v$  在损失函数中的影响程度； $\gamma$  为控制异常值抑制程度的参数； $A \cap B$  为预测框与真实框交集的面积； $A \cup B$  为预测框与真实框并集的面积； $x$  和  $y$  分别为预测框中心点的横坐标、纵坐标； $w$  和  $h$  分别为预测框的宽、高； $x_{gt}$  和  $y_{gt}$  分别为真实框中心点的横坐标、纵坐标； $w_{gt}$  和  $h_{gt}$  分别为真实框的宽、高； $W_g$  和  $H_g$  分别为预测框和真实框的最小外接矩形的宽和高。

进一步提升：迁移学习。深度学习需要大量数据驱动，所以一个模型从零开始训练是十分耗时且成本较高的方法<sup>[36]</sup>。深度学习中常见的一个训练技巧为迁移学习，该方法将在源域上训练完成的模型权重参数移植到需要进行新目标学习的模型中，不仅可以提高模型的收敛速度，还能在一定程度上减少模型过拟合现象的发生<sup>[37]</sup>。本研究使用迁移学习去进一步提升 FieldSentinel-YOLOv8 模型的精度，流程如图 6 所示，Pest24 数据集与自制害虫数据集的背景与害虫部分有很多相似的特征，所以将 Pest24 数据集作为源域，训练 FieldSentinel-YOLOv8 模型，让模型充分学习害虫的通用特征，再将自制的害虫数据集作为目标域，接着训练 FieldSentinel-YOLOv8 模型，使模型具有田间害虫识别的能力，最终得到具有更高精度的 FieldSentinelTransfer-YOLOv8 模型。



注：⊗ 为矩阵相乘；⊕ 为矩阵相加； $M_c$  为通道注意力模块；特征图  $F'$  为特征图  $F$  与通道注意力特征图  $M_c(F)$  相乘的结果； $M_s$  为空间注意力模块；特征图  $F''$  为特征图  $F'$  与空间注意力特征图  $M_s(F')$  相乘的结果。

Note: ⊗ is matrix multiplication; ⊕ is matrix addition;  $M_c$  is channel attention module; Feature map  $F'$  is the result of multiplying the original feature map  $F$  with the channel attention feature map  $M_c(F)$ ;  $M_s$  is spatial attention module; Feature map  $F''$  is the result of multiplying the original feature map  $F'$  with the spatial attention feature map  $M_s(F')$ .

图 5 卷积注意力模块

Fig.5 Convolutional block attention module

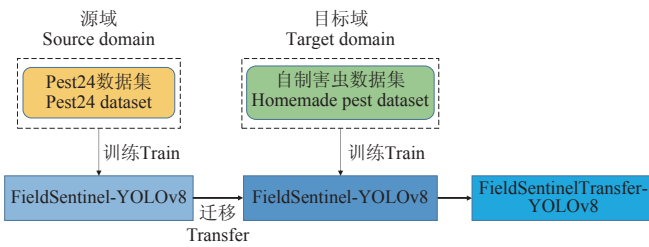


图 6 迁移学习

Fig.6 Transfer learning

### 1.3 试验设计

#### 1.3.1 试验平台

本试验在 64 位操作系统 Ubuntu 22.04.2 LTS 下进行，服务器配置：GPU 为 RTX4090、内存为 128GB、AMD

EPYC 7302 CPU，在 Python-3.8、PyTorch-2.0.1、CUDA-11.8 搭建的深度学习环境中完成。

训练过程中训练批次 batch 为 16，一共训练 150 轮，学习率为 0.01，权重衰减系数为 0.0005，学习率动量为 0.937。

#### 1.3.2 模型评价指标

模型从数据集的 12 类害虫中选出标签数量较多的 9 类害虫进行识别，并对识别结果进行分析，本研究采用精确率 (precision,  $P$ )、召回率 (recall,  $R$ )、平均精度均值 (mean average precision, mAP)、所需浮点运算数 (floating-point operations per second, FLOPs)、模型参数量 (params) 和帧率 FPS (frames per seconds) 评价模型性能。

mAP 是指先计算每一类害虫的平均精度，然后再综



合全部类别的平均精度,一般指  $mAP_{0.5}$ ,即 IoU 阈值为 0.5 的平均精度均值。 $P$ 、 $R$ 、 $AP$ 、 $mAP$  具体计算式如下:

$$P = \frac{TP}{TP + FP} \quad (8)$$

$$R = \frac{TP}{TP + FN} \quad (9)$$

$$AP = \int_0^1 P(R) dR \quad (10)$$

$$mAP = \frac{\sum_{i=1}^N AP_i}{N} \quad (11)$$

式中  $TP$  为真阳性样本,表示正确识别害虫的数量,即预测框与标注框类别相同且交并比大于 0.5;  $FP$  为假阳性样本,表示错误识别为害虫的数量;  $FN$  为假阴性样本,表示未识别的害虫数量;  $AP$  为每种害虫的平均精度;  $N$  为检测类别数,为 9。

## 2 结果与分析

### 2.1 CBAM 与其他注意力机制的对比试验与分析

为验证加入 CBAM 注意力机制的有效性,本研究分析 CBAM 注意力与坐标注意力 (coordinate attention, CA) [38]、高效通道注意力 (efficient channel attention, ECA) [39]、全局注意力 (global attention mechanism, GAM) [40] 和激励注意力 (squeeze and excitation networks, SE) [41] 的性能差异。对比结果如表 2 所示。

表 2 不同注意力机制的结果对比

Table 2 Comparison of the results of different attention mechanisms

| 模型<br>Model  | $P/\%$ | $R/\%$ | $mAP_{0.5}/\%$ | 浮点运算数<br>Floating-point<br>operations per second/G | 参数量<br>Params<br>/M |
|--------------|--------|--------|----------------|--|---------------------|
| YOLOv8m      | 68.52  | 68.12  | 70.92          | 79.10  | 25.86               |
| YOLOv8m+CA   | 64.65  | 69.98  | 69.12          | 79.80  | 26.12               |
| YOLOv8m+ECA  | 67.20  | 70.95  | 71.95          | 79.80  | 26.08               |
| YOLOv8m+GAM  | 68.60  | 65.85  | 67.93          | 86.40  | 34.38               |
| YOLOv8m+SE   | 64.55  | 71.86  | 71.11          | 79.80  | 26.13               |
| YOLOv8m+CBAM | 69.10  | 67.33  | 72.32          | 80.10  | 26.42               |

注:  $P$  为精确率,  $R$  为召回率,  $mAP_{0.5}$  为平均精度均值 (IoU=0.5), IoU 为平均交并比, CA 为坐标注意力, ECA 为高效通道注意力, GAM 为全局注意力, SE 为激励注意力。

Note:  $P$  is precision,  $R$  is recall,  $mAP_{0.5}$  is mean average precision (IoU=0.5), and IoU is intersection over union, CA is coordinate attention, ECA is efficient channel attention, GAM is global attention mechanism, SE is squeeze and excitation networks.

在 YOLOv8m 中分别加入每个注意力机制,没有任何的其他改动。从表 2 的注意力机制试验结果可以看出, CA 和 GAM 注意力机制都会让平均精度均值  $mAP_{0.5}$  有所下降, ECA、SE 和 CBAM 注意力机制加入到 YOLOv8m 中,会让  $mAP_{0.5}$  的值都有所提升,但 CBAM 注意力机制的  $mAP_{0.5}$  是最高的。

### 2.2 Focal-CIoU 与其他损失函数的对比试验与分析

为验证 Focal-CIoU 损失函数的有效性,本研究分析 Focal-CIoU 与广义交并比损失函数 (generalized intersection over union loss, GIoU) [42]、距离交并比损失函数 (distance intersection over union loss, DIoU) [43]、高效交并比损失函数 (efficient intersection over union loss, EIoU) [44] 和形状交并比损失函数 (shape intersection over union loss, SIoU) [45] 的性能差异。在 YOLOv8m 中分别修改每一种损失函数,没有任何其他改动。从表 3 的损失函数试验结果可以看出,GIoU 会使 YOLOv8m 原模型的精确率  $P$  和平均精度均值  $mAP_{0.5}$  均有所下降, EIoU 会使 YOLOv8m 原模型召回率  $R$  和平均精度均值  $mAP_{0.5}$  均有所下降,虽然 DIoU、SIoU 和 Focal-CIoU 会让 YOLOv8m 的精确率有所下降,但是  $mAP_{0.5}$  的值都有所提升,而 YOLOv8m+Focal-CIoU 的  $mAP_{0.5}$  是最高的。

表 3 不同损失函数的结果对比

Table 3 Comparison of results of different loss functions

| 模型<br>Model | 损失函数<br>Loss function | $P/\%$ | $R/\%$ | $mAP_{0.5}/\%$ |
|-------------|-----------------------|--------|--------|----------------|
| YOLOv8m     | CIoU                  | 68.52  | 68.12  | 70.92          |
|             | GIoU                  | 64.61  | 69.71  | 70.51          |
|             | DIoU                  | 65.44  | 70.77  | 70.99          |
|             | EIoU                  | 69.40  | 64.21  | 69.44          |
|             | SIoU                  | 64.18  | 72.78  | 71.40          |
|             | Focal-CIoU            | 66.51  | 69.81  | 71.47          |

注: CIoU 为 YOLOv8m 原模型所使用的损失函数。GIoU 为广义交并比损失函数, DIoU 为距离交并比损失函数, EIoU 为高效交并比损失函数, SIoU 为形状交并比损失函数, Focal-CIoU 为焦点完全交并比损失函数。

Note: CIoU is the loss function used in the original YOLOv8m model, GIoU is generalized intersection over union loss, DIoU is distance intersection over union loss, EIoU is efficient intersection over union loss, SIoU is shape intersection over union loss, Focal-CIoU is focal complete intersection over union loss.

### 2.3 消融试验

为验证本研究所提出的各个改进的有效性,以原始模型 YOLOv8m 为基线模型,通过各个改进模块的不同组合方式进行消融试验,结果如表 4 所示。

表 4 消融试验结果

Table 4 Ablation experiment results

| SMO | CBAM | Focal-CIoU | 迁移学习<br>Transfer learning | $P/\%$ | $R/\%$ | $mAP_{0.5}/\%$ | 浮点运算数<br>Floating-point operations<br>per second/G | 参数量<br>Params<br>/M | 帧率<br>Frames per seconds/<br>(帧·s <sup>-1</sup> ) |
|-----|------|------------|---------------------------|--------|--------|----------------|--|---------------------|---|
| ×   | ×    | ×          | ×                         | 68.52  | 68.12  | 70.92          | 79.10  | 25.86               | 147.15  |
| √   | ×    | ×          | ×                         | 66.04  | 70.17  | 72.20          | 62.60  | 10.19               | 208.41  |
| √   | √    | ×          | ×                         | 71.60  | 69.01  | 72.82          | 62.80  | 10.34               | 196.08  |
| √   | ×    | √          | ×                         | 70.79  | 69.08  | 72.80          | 62.60  | 10.19               | 207.99  |
| √   | √    | √          | ×                         | 65.43  | 75.17  | 73.64          | 62.80  | 10.34               | 199.88  |
| √   | √    | √          | √                         | 69.90  | 77.73  | 77.00          | 62.80  | 10.34               | 199.88  |

注: √ 表示使用该改进, × 表示不使用该改进, SMO 为简化模型操作。

Note: √ represents using this improvement, × represents not using this improvement, SMO is simplified model operation.

根据表 4 消融试验结果分析可知,通过 SMO 操作改进 YOLOv8m 原始模型的网络后,模型所需的浮点运算数和参数量明显降低,并且召回率和平均精度均值均

有所提高,分析原因在于随着网络层数的增加,小目标的一些细粒度的特征信息容易丢失,而 SMO 操作可以有效的避免这种情况发生。从消融试验中还可以看出,

CBAM 注意力机制不仅具有较小的浮点计算数和参数量，而且在 SMO 操作的基础上，添加 CBAM 注意力机制还能将模型精确率和 mAP<sub>0.5</sub> 再次提高。在 SMO 操作的基础上，添加 Focal-CIoU 损失函数后，相比基线模型精确率、召回率、mAP<sub>0.5</sub> 分别提高了 2.27、0.96 和 1.88 个百分点，有效提升了模型的边界框回归性能。同时进行 SMO 操作、添加 CBAM 注意力、引入 Focal-CIoU 损失函数之后，相比较于原始 YOLOv8m 基线网络模型，召回率、mAP<sub>0.5</sub> 和帧数分别提高了 7.05、2.72 个百分点和 52.73 帧/s，并且模型的浮点计算数和参数量分别下降了 16.30 G、15.52 M，表明本研究所提出改进算法的有效性。

**2.4 FieldSentinel-YOLOv8 与其他模型性能对比与分析**

为验证本研究提出的算法比当前热门目标检测模型的优越性，本研究将 FieldSentinel-YOLOv8 与 Faster R-CNN，YOLOv5m、YOLOv6m、YOLOv7、YOLOv8m、YOLOv9c 算法在相同条件下进行对比试验，试验结果如表 5 所示。

表 5 对比试验结果  
Table 5 Comparative experiment results

| 模型<br>Model  | P/%   | R/%   | mAP <sub>0.5</sub> /<br>% | 浮点运算数<br>Floating-point<br>operations per<br>second /G | 参数量<br>Params/M | 帧率<br>Frames per<br>seconds<br>(帧·s <sup>-1</sup> ) |
|--------------|-------|-------|---------------------------|--|-----------------|---|
| Faster R-CNN | 48.76 | 71.47 | 62.19                     | 206.60   | 41.40           | 71.37   |
| RT-DETR      | 64.77 | 64.42 | 62.39                     | 108.00   | 32.83           | 99.94   |
| YOLOv5m      | 61.35 | 72.68 | 69.12                     | 64.40  | 23.91           | 108.70  |
| YOLOv6m      | 60.06 | 69.39 | 68.16                     | 161.60   | 52.00           | 119.47  |
| YOLOv7       | 60.20 | 70.10 | 67.10                     | 105.30   | 37.25           | 138.89  |
| YOLOv8m      | 68.52 | 68.12 | 70.92                     | 79.10  | 25.86           | 147.15  |
| YOLOv9c      | 66.70 | 68.85 | 71.86                     | 103.70   | 25.54           | 90.70   |
| FS-YOLOv8    | 65.43 | 75.17 | 73.64                     | 62.80  | 10.34           | 199.88  |
| FST-YOLOv8   | 69.90 | 77.73 | 77.00                     | 62.80  | 10.34           | 199.88  |

注：FS-YOLOv8 是 FieldSentinel-YOLOv8 的缩写，FST-YOLOv8 是经过迁移学习后的高精度 FieldSentinelTransfer-YOLOv8。下同。  
Note: FS-YOLOv8 is an abbreviation of FieldSentinel-YOLOv8, and FST-YOLOv8 is a high-precision FieldSentinel-YOLOv8 after transfer learning. The same below.

根据表 5 的对比试验可以得出，Faster R-CNN 所需的浮点运算数和参数量较多，并且每秒处理的图片数量较少，所以 Faster R-CNN 不适合本数据集的轻量化实时检测要求。RT-DETR 的 mAP<sub>0.5</sub> 比 YOLOv8m 低了 8.53

个百分点，所以 RT-DETR 并不适合小目标害虫检测。YOLOv8m 相比 YOLOv5m、YOLOv6m 和 YOLOv7 模型，速度分别快了 38.45、27.68 和 8.26 帧率/s，不同程度地降低了计算量，mAP<sub>0.5</sub> 分别提高了 1.80、2.76 和 3.82 个百分点，精确率得到不同程度地提高。YOLOv9c 的平均精度均值比 YOLOv8m 高了 0.94 个百分点，但是每秒处理的图片数量却少了接近 57 张。通过以上分析可见，YOLOv8m 基线网络模型优于其他未改进模型，而本研究提出的 FieldSentinel-YOLOv8 和 FieldSentinelTransfer-YOLOv8 对比 YOLOv8m 在多个指标方面均展现出优越性，FieldSentinelTransfer-YOLOv8 模型的 mAP<sub>0.5</sub> 值更是比 YOLOv8m 基线模型高出了 6.08 个百分点。FieldSentinel-YOLOv8 模型经过迁移学习之后，得到高精度 FieldSentinelTransfer-YOLOv8，精确率、召回率和 mAP<sub>0.5</sub> 比 FieldSentinel-YOLOv8 高了 4.47、2.56 和 3.36 个百分点，表明本研究所使用的迁移学习方法的有效性。

本研究将各种模型对 9 种害虫的平均精度进行对比，如表 6 所示。FieldSentinel-YOLOv8 模型相比原基线模型 YOLOv8m，9 种害虫中有 6 种害虫的平均精度更高，FieldSentinel-YOLOv8 模型对螟蛾识别率的提升尤为明显，相比 YOLOv8m，平均精度提升了 17.63 个百分点，其余模型对螟蛾识别的平均精度较低的原因有两点：螟蛾的体型较小，一般在 5~9 mm 之间；螟蛾样本标签数量较少。表明改进后的模型能够减少小目标细粒度信息的丢失，使模型学习到了更多的细节特征，从而提高了模型对小目标的识别能力，也表明改进后的模型有效解决了数据集样本不平衡问题。高精度 FieldSentinelTransfer-YOLOv8 模型对于检测除了隐翅虫和蝽之外的害虫，平均精度均能达到最高，表明在 2 份数据集（Pest24 与自建数据集）具有类似特征时，使用迁移学习可以大幅提高模型精度。

为说明不同虫害间的识别情况，本研究展示了 YOLOv8m 和 FieldSentinel-YOLOv8 的混淆矩阵对比，如图 7 所示。

表 6 不同种类害虫识别的平均精度对比  
Table 6 Comparison of average accuracy rate for different kinds of pests identification

| 模型<br>Model  | 平均精度 AP <sub>0.5</sub> |                           |                             |                        |                                       |                              |                          |                        |                          | % |
|--------------|------------------------|---------------------------|-----------------------------|------------------------|---------------------------------------|------------------------------|--------------------------|------------------------|--------------------------|---|
|              | 螟蛾<br><i>Pyralidae</i> | 鳞翅目<br><i>Lepidoptera</i> | 隐翅虫<br><i>Staphylinidae</i> | 灯蛾<br><i>Arctiidae</i> | 水龟甲<br><i>Hydrophilidae latreille</i> | 其他昆虫<br><i>Other insects</i> | 鞘翅目<br><i>Coleoptera</i> | 步甲<br><i>Carabidae</i> | 蝽<br><i>Pentatomidae</i> |   |
| Faster R-CNN | 57.18                  | 77.49                     | 40.54                       | 79.45                  | 84.10                                 | 49.91                        | 25.20                    | 65.06                  | 80.80                    |   |
| RT-DETR      | 28.30                  | 71.97                     | 85.78                       | 48.67                  | 86.59                                 | 37.29                        | 41.11                    | 70.67                  | 91.12                    |   |
| YOLOv5m      | 38.13                  | 83.91                     | 78.40                       | 75.16                  | 90.96                                 | 43.69                        | 35.89                    | 80.15                  | 95.76                    |   |
| YOLOv6 m     | 45.10                  | 80.35                     | 78.31                       | 76.87                  | 87.70                                 | 43.19                        | 37.01                    | 74.31                  | 90.61                    |   |
| YOLOv7       | 30.30                  | 80.20                     | 82.20                       | 81.20                  | 89.40                                 | 38.10                        | 38.10                    | 73.60                  | 91.00                    |   |
| YOLOv8 m     | 40.82                  | 84.20                     | 81.33                       | 74.19                  | 91.81                                 | 52.33                        | 39.24                    | 79.58                  | 94.48                    |   |
| YOLOv9c      | 50.70                  | 85.20                     | 81.45                       | 73.29                  | 90.94                                 | 49.89                        | 39.13                    | 82.99                  | 93.15                    |   |
| FS-YOLOv8    | 58.45                  | 83.87                     | 82.11                       | 77.50                  | 92.57                                 | 52.21                        | 39.03                    | 81.86                  | 95.19                    |   |
| FST-YOLOv8   | 69.50                  | 87.76                     | 82.69                       | 87.89                  | 92.49                                 | 51.72                        | 39.13                    | 86.42                  | 95.43                    |   |

相比 YOLOv8m，FieldSentinel-YOLOv8 对大多数的害虫识别的表现更好，尤其是 FieldSentinel-YOLOv8 识别螟蛾的准确率达到了 40%，而 YOLOv8m 只有 20%。混淆矩阵对比结果表明，相比 YOLOv8m，FieldSentinel-

YOLOv8 对于小目标害虫、样本数量较少的害虫的识别精度更高，表明 FieldSentinel-YOLOv8 可以更加精准的识别出体型较小的害虫，并且能够减少样本不平衡带来的影响。根据对比试验结果，选用其中表现较好的 YOLOv8m

和 YOLOv9c 与本研究的 FieldSentinel-YOLOv8 模型、经过迁移学习后的高精度 FieldSentinelTransfer-YOLOv8 模型进行可视化对比。图 8 展示了每轮的  $mAP_{0.5}$  变化情况，由此可得在训练轮次相等的情况下，FieldSentinel-

YOLOv8 的  $mAP_{0.5}$  值一般会比 YOLOv8m 和 YOLOv9c 更高，而经过迁移学习后的高精度 FieldSentinelTransfer-YOLOv8 不仅可以迅速达到精度峰值，并且在相同训练轮次下， $mAP_{0.5}$  值都为最高。

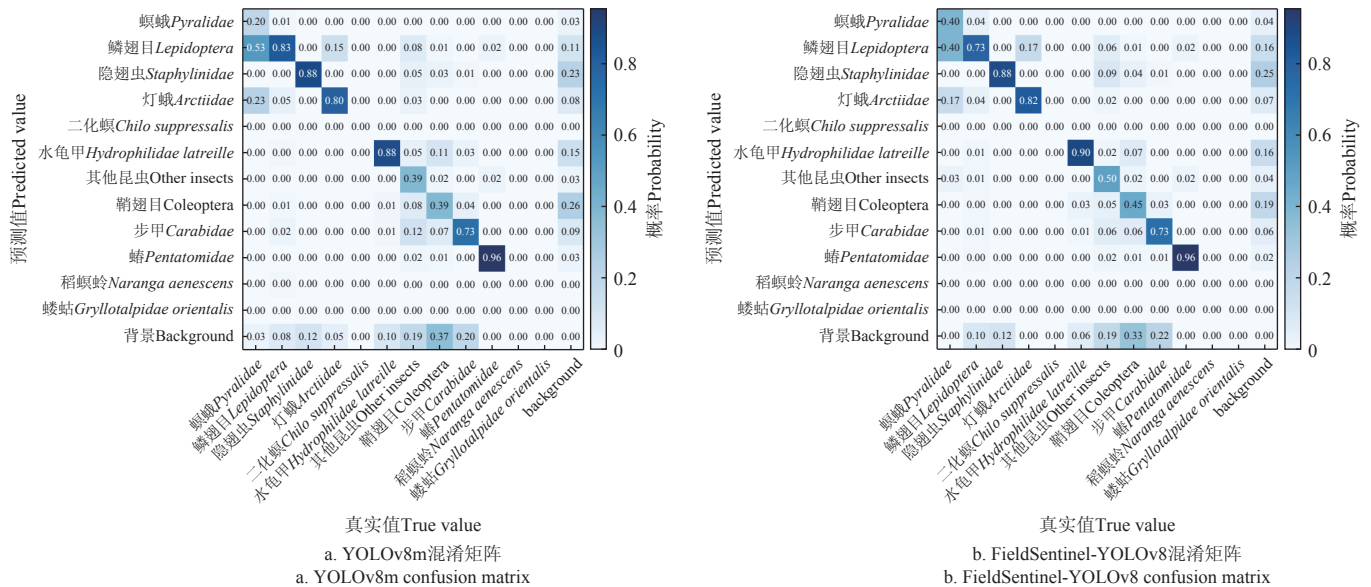


图 7 YOLOv8m 和 FieldSentinel-YOLOv8 的混淆矩阵  
Fig.7 Confusion matrix of YOLOv8m and FieldSentinel-YOLOv8

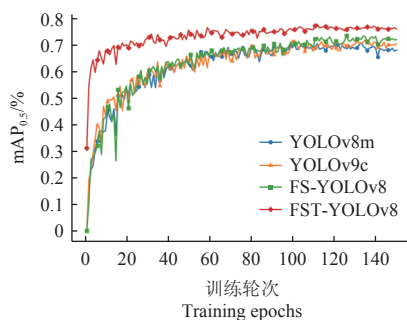


图 8 模型每一轮的平均精度均值对比  
Fig.8 The mean average precision comparison for each model iteration

此外，本研究还分析了训练过程中边界框损失和类别损失的变化。如图 9 所示，FieldSentinel-YOLOv8 与 YOLOv8m、YOLOv9c 相比，边界框损失的下降速度更快，类别损失相似。这表明 FieldSentinel-YOLOv8 优于 YOLOv8m 和 YOLOv9c。而经过迁移学习后的高精度 FieldSentinelTransfer-YOLOv8 的边界框损失和类别损失下降速度最快。

本研究分别从小目标、正常目标和密集目标的 3 种情况下的害虫图像进行检测效果展示，其结果如图 10 所示。

根据小目标的识别结果对比可以看出，在害虫较小的情况下，FieldSentinel-YOLOv8 可以识别出更多的正确小目标，证明该算法可以改善体型较小的害虫难以识别的问题。根据正常目标的识别结果对比可以看出，在多数情况下 FieldSentinel-YOLOv8 具有更高的识别精度

和更高效的边界框回归性能。根据密集目标识别结果对比可以看出，FieldSentinel-YOLOv8 检测到了图像中心被遮挡的害虫，表明 FieldSentinel-YOLOv8 能更关注害虫区域。3 种情况下 FieldSentinel-YOLOv8 的检测能力均优于 YOLOv8m 和 YOLOv9c。FieldSentinelTransfer-YOLOv8 在多数情况下有着更高效的边界框回归性能，是对比模型中检测能力最优的。

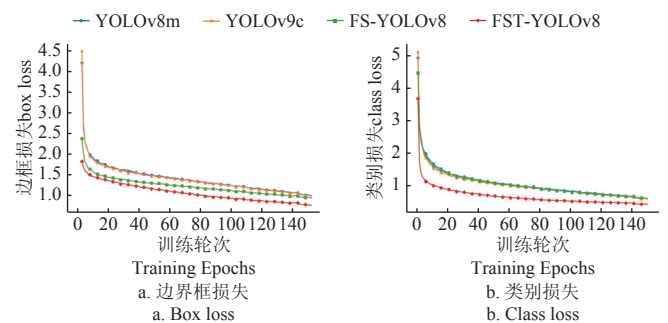


图 9 模型边界框损失和类别损失下降曲线  
Fig.9 Model box loss and class loss decline curve

## 2.5 模型特征可视化

为更加直观地观察 FieldSentinel-YOLOv8 模型对于害虫识别能力提升，本研究采用 Grad-CAM<sup>[46]</sup> (gradient-weighted class activation mapping) 技术绘制热力图，可以较为直观地看到模型对于害虫特征的学习情况。Grad-CAM 利用训练权重进行反向传播，对梯度矩阵执行空间维度上的全局平均池化，并对特征层的各个通道进行加权激活处理，从而生成可视化热力图，在热力图中可以通过每个区域的亮度深浅揭示图像中对模型预测结果影响较大的特征区域。热力图可视化对比如图 11 所



示。与 YOLOv8m 和 YOLOv9m 模型相比, FieldSentinel-YOLOv8 模型的害虫目标处颜色更亮, 响应程度更高,

对于正确目标的感知加强, 使得模型更为准确地关注害虫区域的特征。

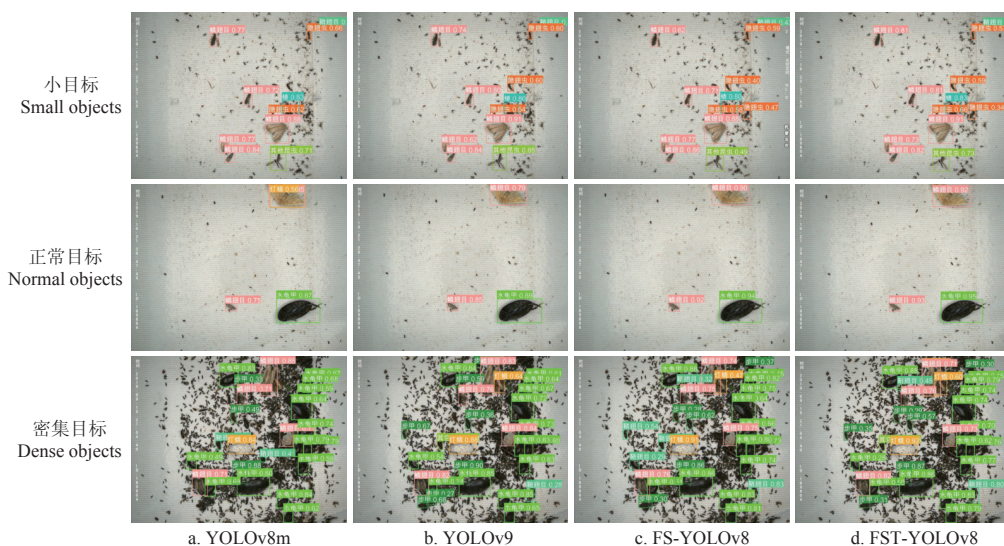


图 10 模型检测效果对比

Fig.10 Comparison of model detection effects

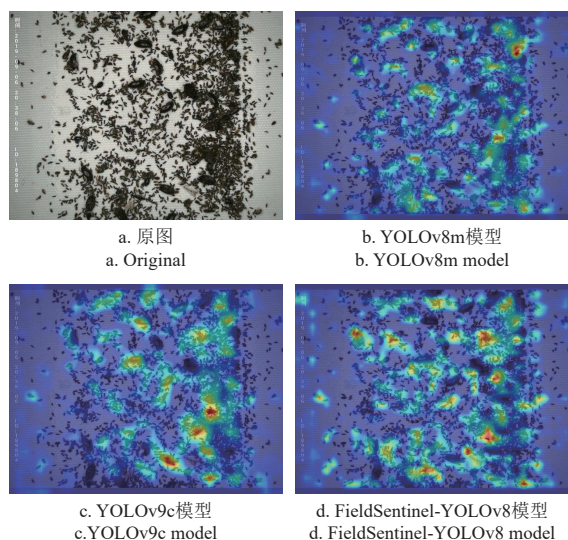


图 11 模型热力图可视化

Fig.11 Model heat map visualization

### 3 结论

本研究提出一种基于 YOLOv8m 改进的轻量化害虫识别方法 FieldSentinel-YOLOv8, 并采用迁移学习方法, 进一步提升 FieldSentinel-YOLOv8 模型的性能。通过结果对比分析可以得出以下结论:

1) 相同的试验条件下, FieldSentinel-YOLOv8 通过与 Faster R-CNN、YOLOv5m、YOLOv6m、YOLOv7、YOLOv8m 和 YOLOv9c 作对比, FieldSentinel-YOLOv8 在自建害虫数据集上取得更好的试验效果, 相比于原始的 YOLOv8m 模型, 平均精度均值  $mAP_{0.5}$  和帧数分别提高了 2.72 个百分点和 52.73 帧/s, 模型的浮点计算数和参数量分别下降了 16.30 G、15.52 M。FieldSentinel-YOLOv8 除精确率外的对比指标均优于其他对比算法, 为农业害虫快速检测提供了方法支持。本研究在使用迁

移学习后, 高精度 FieldSentinelTransfer-YOLOv8 模型  $mAP_{0.5}$  再次提升 3.36 个百分点, 达到 77.00%

2) 为验证 FieldSentinel-YOLOv8 网络模型的检测效果, 本研究设置了小目标、正常目标和密集目标 3 种情况下的可视化对比分析, 设置了模型特征可视化对比分析。结果表明, FieldSentinel-YOLOv8 网络模型对于害虫的检测效果比原始的 YOLOv8m 和 YOLOv9c 更好, 不仅能够识别出更小的害虫, 还能更准确的识别出害虫。

但本研究提出的模型对部分害虫的识别效果仍有进步空间, 考虑到或许是图像数据集较少的情况下要识别的种类较多等原因, 后续可以扩展数据集, 让模型可以更好的拟合, 让模型的精度再次提升; 今后也将完善本研究所提出的方法的实际应用能力, 助力农业病虫害智能化、数字化防控。

### [参 考 文 献]

- [1] LOWDER S K, SKOET J, SINGH S. What do we really know about the number and distribution of farms and family farms in the world? Background paper for The State of Food and Agriculture[EB/OL]. (2014-04-08)[2024-6-18]. <https://ageconsearch.umn.edu/record/288983>.
- [2] 郑钧, 陈天毅. 农作物病虫害的综合防治措施[J]. 现代农业科技, 2011(22): 202-204, 207.
- [3] ZHENG Jun, CHEN Tianyi. The integrated control measures for crop diseases and pests[J]. Journal of Modern Agricultural Science and Technology, 2011(22): 202-204,207. (in Chinese with English abstract)
- [4] DING W, TAYLOR G. Automatic moth detection from trap images for pest management[J]. Computers and Electronics in Agriculture, 2016, 123: 17-28.
- [5] SUN Y, LIU X, YUAN M, et al. Automatic in-trap pest detection using deep learning for pheromone-based Dendroctonus valens monitoring[J]. Biosystems Engineering, 2018, 176: 140-150.
- [6] ZOU Z, CHEN K, SHI Z, et al. Object detection in 20 years: A survey[J]. Proceedings of the IEEE, 2023, 111(3): 257-276.

- [6] WENG Z, MENG F, LIU S, et al. Cattle face recognition based on a Two-Branch convolutional neural network[J]. *Computers and Electronics in Agriculture*, 2022, 196: 106871.
- [7] 甘雨, 郭庆文, 王春桃, 等. 基于改进 EfficientNet 模型的作物害虫识别[J]. *农业工程学报*, 2022, 38(1): 203-211.  
GAN Yu, GUO Qingwen, WANG Chuntao, et al. Recognizing crop pests using an improved EfficientNet model[J]. *Transactions of the Chinese Society of Agricultural Engineering (Transactions of the CSAE)*, 2022, 38(1): 203-211. (in Chinese with English abstract)
- [8] 王金星, 马博, 王震, 等. 基于改进 Mask R-CNN 的苹果园害虫识别方法[J]. *农业机械学报*, 2023, 54(6): 253-263, 360.  
WANG Jinxing, MA Bo, WANG Zhen, et al. Pest identification method in apple orchard based on improved Mask R-CNN[J]. *Transactions of the Chinese Society for Agricultural Machinery*, 2023, 54(6): 253-263, 360. (in Chinese with English abstract)
- [9] REZA M T, MEHEDI N, TASNEEM N A, et al. Identification of crop consuming insect pest from visual imagery using transfer learning and data augmentation on deep neural network[C]//2019 22nd International Conference on Computer and Information Technology (ICCIT). Piscataway, NJ, USA: IEEE, 2019: 1-6.
- [10] ANWAR Z, MASOOD S. Exploring deep ensemble model for insect and pest detection from images[J]. *Procedia Computer Science*, 2023, 218: 2328-2337.
- [11] 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.  
XIAO Deqin, HUANG Yigui, ZHANG Yuanqin, et al. Pest detection algorithm of yellow plate in field based on improved faster R-CNN[J]. *Transactions of the Chinese Society for Agricultural Machinery*, 2021, 52(6): 242-251. (in Chinese with English abstract)
- [12] 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 (CVPR), Piscataway, NJ, USA: IEEE, 2016: 779-788.
- [13] REDMON J, FARHADI A. Yolov3: An incremental improvement[EB/OL]. (2018-04-08)[2024-6-18]. <https://arxiv.org/abs/1804.02767>.
- [14] BOCHKOVSKIY A, WANG C Y, LIAO H Y M. Yolov4: Optimal speed and accuracy of object detection[EB/OL]. (2020-04-23)[2024-6-18]. <https://arxiv.org/abs/2004.10934>.
- [15] JOCHER G. "YOLOv5 by ultralytics"[EB/OL]. (2020-06-19)[2024-6-18]. <https://github.com/ultralytics/yolov5>.
- [16] LI C, LI L, JIANG H, et al. YOLOv6: A single-stage object detection framework for industrial applications[EB/OL]. (2022-09-07)[2024-6-18]. <https://arxiv.org/abs/2209.02976>.
- [17] WANG C Y, BOCHKOVSKIY A, LIAO H Y M. YOLOv7: 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 BC, Canada: IEEE, 2023: 7464-7475.
- [18] JOCHER G. "YOLOv8 by ultralytics"[EB/OL]. (2023-01-10)[2024-6-18]. <https://github.com/ultralytics/ultralytics>.
- [19] WANG C Y, YE H I H, LIAO H Y M. YOLOv9: Learning what you want to learn using programmable gradient information[EB/OL]. (2024-02-21)[2024-6-18]. <https://arxiv.org/abs/2402.13616>.
- [20] 赵辉, 黄鏢, 王红君, 等. 基于改进 YOLO v7 的农田复杂环境下害虫识别算法研究[J]. *农业机械学报*, 2023, 54(10): 246-254.  
ZHAO Hui, HUANG Biao, WANG Hongjun, et al. Pest identification method in complex farmland environment based on improved YOLOv7[J]. *Transactions of the Chinese Society for Agricultural Machinery*, 2023, 54(10): 246-254. (in Chinese with English abstract)
- [21] DAI M, DORJOY M M H, MIAO H, et al. A new pest detection method based on improved YOLOv5m[J]. *Insects*, 2023, 14(1): 54.
- [22] ZHANG L, DING G, LI C, et al. DCF-Yolov8: An improved algorithm for aggregating low-level features to detect agricultural pests and diseases[J]. *Agronomy*, 2023, 13(8): 2012.
- [23] HU Y, DENG X, LAN Y, et al. Detection of rice pests based on self-attention mechanism and multi-scale feature fusion[J]. *Insects*, 2023, 14(3): 280.
- [24] 刘忠, 卢安舸, 崔浩, 等. 基于改进 YOLOv8 的轻量化荷叶病虫害检测模型[J]. *农业工程学报*, 2024, 40(19): 168-176.  
LIU Zhong, LU Ange, CUI Hao, et al. A lightweight lotus leaf diseases and pests detection model using improved YOLOv8[J]. *Transactions of the Chinese Society of Agricultural Engineering (Transactions of the CSAE)*, 2024, 40(19): 168-176. (in Chinese with English abstract)
- [25] 郭嘉璇, 王蓉芳, 南江华, 等. 融入全局相应归一化注意力机制的 YOLOv5 农作物害虫识别模型[J]. *农业工程学报*, 2024, 40(8): 159-170.  
GUO Jiaxuan, WANG Rongfang, NAN Jianghua, et al. YOLOv5 model integrated with GRN attention mechanism for insect pest recognition[J]. *Transactions of the Chinese Society of Agricultural Engineering (Transactions of the CSAE)*, 2024, 40(8): 159-170. (in Chinese with English abstract)
- [26] WEI P, YAN X, YAN W, et al. Precise extraction of targeted apple tree canopy with YOLO-Fi model for advanced UAV spraying plans[J]. *Computers and Electronics in Agriculture*, 2024, 226: 109425.
- [27] HU Y, WANG Q, WANG C, et al. MACNet: A more accurate and convenient pest detection network[J]. *Electronics*, 2024, 13(6): 1068.
- [28] WU X, ZHAN C, LAI Y K, et al. Ip102: A large-scale benchmark dataset for insect pest recognition[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Piscataway, NJ, USA: IEEE, 2019: 8787-8796.
- [29] WANG Q J, ZHANG S Y, DONG S F, et al. Pest24: A large-scale very small object data set of agricultural pests for multi-target detection[J]. *Computers and Electronics in Agriculture*, 2020, 175: 105585.
- [30] ZHANG J, CHEN L, OUYANG B, et al. Pointcutmix: Regularization strategy for point cloud classification[J]. *Neurocomputing*, 2022, 505: 58-67.
- [31] SU D, KONG H, QIAO Y, et al. Data augmentation for deep learning based semantic segmentation and crop-weed classification in agricultural robotics[J]. *Computers and Electronics in Agriculture*, 2021, 190: 106418.
- [32] LOU H, DUAN X, GUO J, et al. DC-YOLOv8: Small-size object detection algorithm based on camera sensor[J]. *Electronics*, 2023, 12(10): 2323.
- [33] 刘树东, 刘业辉, 孙叶美, 等. 基于倒置残差注意力的无人机航拍图像小目标检测[J]. *北京航空航天大学学报*, 2023, 49(3): 514-524.  
LIU Shudong, LIU Yehui, SUN Yemei, et al. Small object detection in UAV aerial images based on inverted residual attention[J]. *Journal of Beijing University of Aeronautics and Astronautics*, 2023, 49(3): 514-524. (in Chinese with English abstract)
- [34] 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, Switzerland: Springer



- International Publishing, 2018: 3-19.
- [35] LIN T Y, GOYAL P, GIRSHICK R, et al. Focal loss for dense object detection[C]//Proceedings of the IEEE international Conference on Computer Vision (ICCV), Piscataway, NJ, USA: IEEE, 2017: 2980-2988.
- [36] 汪斌斌, 杨贵军, 杨浩, 等. 基于 YOLO\_X 和迁移学习的无人机影像玉米雄穗检测[J]. *农业工程学报*, 2022, 38(15): 53-62. WANG Binbin, YANG Guijun, YANG Hao, et al. UAV images for detecting maize tassel based on YOLO\_X and transfer learning[J]. *Transactions of the Chinese Society of Agricultural Engineering(Transactions of the CSAE)*, 2022, 38(15): 53-62. (in Chinese with English abstract)
- [37] SI H, WANG Y, ZHAO W, et al. Apple surface defect detection method based on weight comparison transfer learning with MobileNetV3[J]. *Agriculture*, 2023, 13(4): 824.
- [38] HOU Q, ZHOU D, FENG J. Coordinate attention for efficient mobile network design[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA: IEEE, 2021: 13713-13722.
- [39] WANG Q, WU B, ZHU P, et al. ECA-Net: Efficient channel attention for deep convolutional neural networks[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA: IEEE, 2020: 11534-11542.
- [40] LIU Y, SHAO Z, HOFFMANN N. Global attention mechanism: Retain information to enhance channel-spatial interactions[EB/OL]. (2021-12-10)[2024-6-18]. <https://arxiv.org/abs/2112.05561>.
- [41] HU J, SHEN L, SUN G. Squeeze-and-excitation networks[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA: IEEE, 2018: 7132-7141.
- [42] 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 (CVPR), Long Beach, CA, USA: IEEE, 2019: 658-666.
- [43] ZHENG Z, 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, NY, USA, AAAI Press, 2020, 34(7): 12993-13000.
- [44] ZHANG Y F, REN W, ZHANG Z, et al. Focal and efficient IOU loss for accurate bounding box regression[J]. *Neurocomputing*, 2022, 506: 146-157.
- [45] GEVORGYAN Z. Siou loss: More powerful learning for bounding box regression[EB/OL]. (2022-05-25)[2024-6-18]. <https://arxiv.org/abs/2205.12740>.
- [46] 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 (ICCV), Piscataway, NJ, USA: IEEE, 2017: 618-626.

## Identifying pests in rice fields using improved YOLOv8m

TAN Siqiao<sup>1,2</sup>, CHEN Han<sup>1</sup>, ZHU Lei<sup>1,2</sup>, SUN Haoran<sup>1</sup>, ZHANG Zhengbing<sup>3</sup>, YIN Li<sup>3</sup>, HUANG Wanwan<sup>1\*</sup>

(1. College of Information and Intelligence, Hunan Agricultural University, Changsha 410128, China; 2. Hunan Engineering Technology Research Center of Agricultural & Rural Informatization, Changsha 410128, China; 3. Hunan Provincial Plant Protection and Inspection Station, Changsha 411229, China)

**Abstract:** Object detection can often hold the promising potential to replace the human visual recognition in smart agriculture. However, the small target pests are still challenging to detect for three reasons. Firstly, the pests can move rapidly, which is difficult to for real-time detection. Secondly, the accuracy of the model can depend mainly on the small size, the large number of groups, the uneven distribution, and the occluded pests of each other. Finally, the imbalanced sample can make the recognition more difficult, leading to the low accuracy of detection. It is necessary to efficiently and precisely identify the numerous, unevenly distributed, complex shapes, as well as small and densely packed pests. In this study, a pest recognition (called FieldSentinel-YOLOv8) was proposed using the improved YOLOv8m model. The FieldSentinel-YOLOv8 was improved as follows. Firstly, three detection heads were replaced by two ones, in order to simplify the original YOLOv8 model. The SMO (simplify model operations) was enhanced the fine-grained features for the small targets. The floating-point operations and computational burden were also reduced to streamline the YOLOv8 model; Secondly, the convolutional block attention module (CBAM) was integrated into the YOLOv8. Thus the general feature (such as background) was suppressed to focus more on the pest regions. Thus, the accuracy of the improved model was enhanced to identify the occluded pests. Lastly, the Focal-CIoU Loss was employed to replace the Ciou Loss. The class imbalance was reduced to further improve the accuracy of detection. Many current models depend heavily on the overly idealized datasets, leading to compromise in their accuracy in actual field conditions. Pest monitoring equipment was deployed directly in the field. The datasets were collected under natural environments, indicating the accurate reflection of the real-world presence of pests. The dataset was then taken by the pest monitoring equipment. A series of comparative experiments were conducted to evaluate the performance of the FieldSentinel-YOLOv8 under identical conditions using a self-constructed dataset and various object detections. FieldSentinel-YOLOv8 algorithm also demonstrated the superior performance across most metrics. Compared with the original model, the improved model was reduced the number of parameters by 15.52 M, whereas, there was the an increase in the processing speed to 52.73 frames per second. Moreover, the mAP<sub>0.5</sub> and recall rate of the improved model were enhanced by 2.72 and 7.05 percentage points, respectively. Furthermore, transfer learning was employed to train the FieldSentinel-YOLOv8 model, taking the Pest24 dataset as the source domain and a self-built dataset as the target domain. The trained model was then named the FieldSentinelTransfer-YOLOv8. The improved model was achieved in the better performance of detection after transfer learning. The mAP<sub>0.5</sub> increased by 3.36 percentage points, reaching 77.00%, with the accuracy and recall rates of 69.90% and 77.73%, respectively. Therefore, the FieldSentinel-YOLOv8 can provide the valuable technical references for the accurate and rapid identification of agricultural pests. The high-precision FieldSentinelTransferYOLOv8 model after transfer learning can also offer the technical support for the detection of agricultural pests.

**Keywords:** pest infestation; deep learning; image identification; YOLOv8m; CBAM; Focal-CIoU; transfer learning