

# 基于 PBM-YOLOv8 的水稻病虫害检测

刘鹏, 张天翼, 冉鑫, 史佳霖, 毕誉轩, 王彩霞

(长春理工大学电子信息工程学院, 长春 130022)

**摘要:** 为提高水稻病虫害检测精度, 解决病虫害种类繁多、尺度不一、生长环境复杂导致的误检漏检问题, 便于模型在边缘设备进行部署, 提出一种基于改进 YOLOv8 的水稻病虫害检测方法 PBM-YOLOv8。首先使用部分卷积 (partial convolution, PConv) 设计 PCBLOCK 结构, 替换 YOLOv8 特征提取模块中的瓶颈 (bottleneck) 结构, 以减少模型参数量, 提升检测速度; 其次为了减少非相邻层语义信息特征融合时的稀释, 在颈部网络增加平衡特征融合层, 重构特征金字塔 (feature pyramid networks, FPN) 为平衡特征金字塔 (balanced feature pyramid, BFP), 对融合的特征层进行特征再提取, 并引入嵌入高斯非局部注意力 (embedded Gaussian non-local attention, EGNA) 消除多层融合导致的混叠效应, 最大程度减小特征丢失; 最后将损失函数更换为 MPDIUO, 改善因样本差异性大而导致的检测框失真, 同时降低模型训练的计算负担。试验结果表明, 改进模型 PBM-YOLOv8 在水稻病虫害数据集上取得了更为优异的试验效果, 相较于原始 YOLOv8n 基线模型精确度及平均准确率均值分别提高了 1.3 和 1.1 个百分点。将 PBM-YOLOv8 部署在 RK3588 上经多线程优化后检测速度可达到 71.4 帧/s, 满足实际应用的需求, 可实现对水稻病虫害的实时精准检测。

**关键词:** 图像识别; 深度学习; 目标检测; 水稻病虫害; YOLOv8; BFP; 多线程优化

doi: 10.11975/j.issn.1002-6819.202405194

中图分类号: TP391.4

文献标志码: A

文章编号: 1002-6819(2024)-20-0147-10

刘鹏, 张天翼, 冉鑫, 等. 基于 PBM-YOLOv8 的水稻病虫害检测[J]. 农业工程学报, 2024, 40(20): 147-156. doi: 10.11975/j.issn.1002-6819.202405194 <http://www.tcsae.org>

LIU Peng, ZHANG Tianyi, RAN Xin, et al. Detecting rice disease using PBM-YOLOv8[J]. Transactions of the Chinese Society of Agricultural Engineering (Transactions of the CSAE), 2024, 40(20): 147-156. (in Chinese with English abstract) doi: 10.11975/j.issn.1002-6819.202405194 <http://www.tcsae.org>

## 0 引言

中国是世界上稻作历史最悠久、水稻遗传资源最丰富的国家之一。2023 年全国水稻播种面积 2 894 万  $\text{hm}^2$ , 总产量 20 660 万 t, 约占世界稻作总面积的 1/4, 产量约为世界稻谷总产量的 37%, 全国粮食总产量的 30%。作为中国主要粮食作物之一, 水稻的生长安全对农户利益至关重要, 同时也是维护国家粮食安全的关键因素<sup>[1]</sup>。水稻产量威胁主要来源于害虫侵害和病原体感染, 为保证其产量和质量, 就需要在种植过程中及早对病虫害进行精准识别和检测。

传统的水稻病虫害识别主要依赖于经验丰富的农民在田间进行人眼识别, 这种方式需要消耗大量人力资源, 并且容易受主观影响, 准确率较低。为了提高识别精度和效率, 早期的自动识别方法主要依赖于数字图像处理和传统的机器学习技术。然而这些方法大多基于人工构建的模型, 由于其拓展性较差, 无法适应中国多样的种植地域地貌, 因此难以在实际应用中得到推广。

随着近年来硬件计算能力的显著提高, 许多研究人员采用图像分类的深度学习方法对水稻病害进行识别。

黄双萍等<sup>[2]</sup>引用使用 GoogLeNet 模型对自制的水稻穗瘟病数据集进行训练, 该模型的缺点是在不同环境下检测精度差别较大, 泛化能力较弱。范春全等<sup>[3]</sup>利用基于 ResNet50 的迁移学习模型在样本较少的情况下完成识别, 并建立二分类模型进行数据过滤, 以防止非水稻病害被识别为水稻病害的不合理情况, 然而对相似的细小特征提取结果不显著, 不利于病害特征识别。FENG 等<sup>[4]</sup>使用卷积神经网络对水稻的病虫害特征进行提取, 然后将支持向量机方法应用于病虫害分类和预测, 其精度高于传统的 BP 神经网络模型, 不过受限于其数据集样本单一, 鲁棒性较差。为满足病虫害识别和定位的双重需求<sup>[5]</sup>, 研究者将二阶段目标检测算法 Fast R-CNN 等<sup>[6]</sup>及一阶段目标检测算法 SSD<sup>[7]</sup>、YOLO 系列<sup>[8-11]</sup>等应用在水稻病虫害检测领域, 取得了丰硕成果<sup>[12]</sup>。RAHAMAN<sup>[13]</sup>等针对大模型不适合移动设备部署的问题, 提出了一种两阶段轻量级卷积神经网络对水稻病虫害进行检测, 但该模型面向的是孟加拉国常见的 4 种水稻病虫害, 对中国种类繁多的病害类别并不能够精准地识别。余颖等<sup>[14]</sup>提出一种基于 SSD 网络模型的水稻害虫识别方法, 该算法用表征能力更强的特征金字塔代替 SSD 原有的多尺度特征图, 使得模型对小目标的识别率更高、收敛性更好, 提高了水稻害虫的识别率与检测速度。吴子炜等<sup>[15]</sup>使用多个深度可分离卷积提出一种轻量级 YOLOv5 算法, 改进

收稿日期: 2024-05-28 修订日期: 2024-08-29

基金项目: 吉林省科技发展计划项目 (20210201021GX)

作者简介: 刘鹏, 博士, 副研究员, 博士生导师, 研究方向为计算机视觉、边缘计算和图像处理。Email: liup@cust.edu.cn

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 个百分点。上述检测方法虽取得一定成效,但在水稻的实际种植场景中,植株上往往患有多种病虫害,不同病虫害间形态差异较大,当前模型难以解决病虫害种类多、尺度不一、生长环境复杂而导致的误检漏检问题。

为解决上述问题,本文在 YOLOv8<sup>[16]</sup> 的基础上结合部分卷积<sup>[17]</sup> 和嵌入高斯非局部注意力<sup>[18]</sup>,改进特征提取网络并重构特征金字塔为平衡特征金字塔,提出改进模型 PBM-YOLOv8,使得模型在复杂的田间环境下准确检测病虫害,在保证检测效果的同时控制模型的计算复杂度,便于模型在 RK3588 边缘设备上部署;在实际部署于嵌入式平台时,从应用工程的角度通过多线程的方法优化与加速模型的检测速度,进一步保证模型的实时性。本研究致力于探索一种准确快速检测水稻病虫害的模型,及时针对病害采取措施,避免农药的滥用,为智慧农业提供技术支持。

## 1 试验数据材料与方法

### 1.1 试验数据采集

水稻病虫害表型数据采集自吉林省长春市郊区,此地位于松辽平原中部松花江流域,采集时间为 2023 年 7—9 月,正值水稻灌浆期、乳熟期和蜡熟期,此时植株分蘖良好,为病虫害的识别提供了有利条件<sup>[19]</sup>。共获得患病水稻图片 1 854 张,保存为 jpg 格式。其次通过 Scrapy 爬虫框架爬取高质量水稻病虫害图片 326 张,筛选后共获得图片 2 118 张,部分图片如图 1 所示。其中虫害包括稻螟蛉虫、稻螟蛉虫卵、大螟、稻绿蝽四类,病害包括稻瘟病和纹枯病两类。采集的病害类别及种类占比信息见表 1。

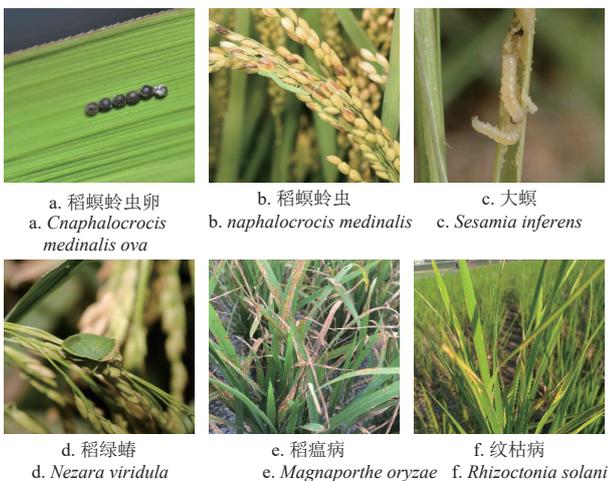


图 1 部分水稻病害图像数据

Fig.1 Part of rice disease image data

### 1.2 数据集构建

现实场景中,不同环境例如强光直射、田间杂草以及叶片间的相互遮挡等都会对病害特征的提取产生影响。

为提高样本鲁棒性,采用旋转、翻转、高斯模糊、随机裁剪、随机填充的方法对原始图片进行扩充,提升网络的泛化能力。为确保模型输出结果一致,减少运算中的变量,加速训练和推理过程,以病害为中心,使用 MATLAB 视觉库对图像进行裁剪和归一化处理,将图像分辨率统一为 640 像素×640 像素。图像使用 LabelImg 标注病害的位置和类别,标签保存为 txt 格式,并对标注结果进行校正,减少错误标注产生的噪声。

表 1 水稻病虫害数据信息

Table 1 Data information of rice pest

类别 Classes	图片数量 Number of images	病害特征 Disease characteristics
稻螟蛉虫卵 <i>Cnaphalocrocis medinalis</i> <i>ova</i>	249	呈半圆形或卵形,初产时淡黄色,孵化前变紫色
稻螟蛉虫 <i>Cnaphalocrocis medinalis</i>	197	呈淡黄色或粉红色,身体细长,有黑色头部,尾部有一对尖角
大螟 <i>Sesamia inferens</i>	435	呈淡绿色,身体细长,体侧有白色纵纹
稻绿蝽 <i>Nezara viridula</i>	313	长椭圆形,青绿色,腹下色较淡
稻瘟病 <i>Magnaporthe oryzae</i>	476	纺锤形病斑,中间灰色边缘棕色,两头有褐色延长线
纹枯病 <i>Rhizoctonia solani</i>	448	黄色或褐色的细长条纹,纹路通常沿着叶子的纵向生长

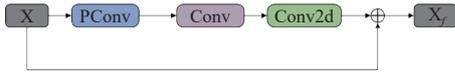
通过对数据进行扩充与增强<sup>[20]</sup>,图像由 2 118 张扩充至 5 506 张。利用 Python 脚本,将图像按照 8:1:1 的比例随机划分训练集、验证集和测试集,构建数据集。

## 2 PBM-YOLOv8

本文针对水稻病虫害种类多、尺度不一、生长环境复杂导致的准确率低及边缘设备检测实时需求对 YOLOv8<sup>[21]</sup> 进行改进。首先使用 PConv 设计 PCBlock 结构,替换 YOLOv8 特征提取模块 C2f 中的 Bottleneck,以减少模型参数量,提升检测速度。其次为了减少非相邻层的语义信息在特征融合时的稀释,在 Neck 部分增加平衡特征融合层<sup>[22]</sup>,重构特征金字塔为平衡特征金字塔,对融合的特征层进行特征再提取,并引入嵌入高斯非局部注意力消除多层融合导致的混叠效应,得到增强后的特征层,最大程度减小特征丢失。最后将损失函数更换为 MPDIU<sup>[23]</sup>,改善因样本差异性大而导致的检测框失真,同时降低模型训练的计算负担。

### 2.1 部分卷积特征提取块

为设计低延迟高吞吐量且快速高效的目标检测模型,本文引入可同时减少冗余计算和内存访问,进行有效提取空间特征的 PConv,改善浮点运算 (floating point operations, FLOPs) 效率较低的问题。并基于 PConv 设计一种轻量化的特征提取结构 PCBlock,用于减少因特征图通道数增多,多通道特征图相似而造成的特征图冗余,确保模型部署后的实时性。其结构如图 2 所示。

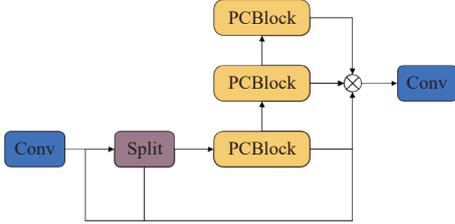


注：X 为接收特征图，PCConv 为部分卷积操作，Conv 为卷积操作，Conv2 d 为 1×1 卷积，X<sub>f</sub> 为输出特征图。  
 Note: X is the receiving feature map, PCConv is a partial convolution operation, Conv is a convolution operation, Conv2 d is a 1×1 convolution, X<sub>f</sub> is the output feature map.

图 2 PCBlock 结构图  
 Fig.2 PCBlock structure diagram

在上述结构中，PCConv 具有高效快速的优点，对于接收的特征图  $X \in R^{c \times h \times w}$ ，它只需在输入通道的 1/4 部分应用  $c$  个滤波器  $w \in R^{k \times k}$  进行空间特征提取，保持其余 3/4 通道不变。当内存访问是连续或规则的时候，PCConv 会将第一个或最后一个连续的通道  $c_p$  作为完整的特征图来进行计算。在特征图经过 3×3 的 PCConv 处理后，使用一个 Conv 将输出特征图的通道数扩充 2 倍，并从中提取有效的信息，兼顾所有通道，在轻量化的前提下保证网络的精度。接下来 1×1 的卷积将通道数还原成扩张前，确保特征图的两路分支具有相同的大小和维度，将未计算部分与已计算部分两路分支合并，进一步避免信息丢失，得到输出特征图  $X_f$ 。

为减小模型的计算量，本文将 YOLOv8 特征提取模块中原有的 Bottleneck 结构替换为 PCBlock 结构，改进后的 C2f-PCBlock 模块如图 3 所示。



注：Split 为分割操作，PCBlock 为合成模块  
 Note: Split is a segmentation operation, PCBlock is synthetic modules.

图 3 C2f-PCBlock 模块  
 Fig.3 C2f-PCBlock Module

C2f-PCBlock 特征提取模块输入  $h \times w \times c_{in}$  通道数的特征图，经过 Conv 处理得到的输出通道数为  $h \times w \times c_{out}$ ，将通道数为  $h \times w \times 0.5c_{out}$  的特征图存入列表  $i$  中。Conv 输出的全部特征图分割成 2 份通道数为  $h \times w \times 0.5c_{out}$  的特征图用于实现跳跃连接，之后串联  $n$  个 PCBlock 结构，第一个 PCBlock 对特征进行压缩，后续的 PCBlock 将特征高维映射并与输入融合。每个 PCBlock 结构的输出作为下一个 PCBlock 结构的输入，同时也需将输出存入列表  $i$ 。全部 PCBlock 结构处理完成后，将列表  $i$  中的特征图按通道数拼接，得到一个  $h \times w \times 0.5(n+2)c_{out}$  维特征图，由第二个 Conv 将特征图通道数压缩为  $h \times w \times c_{out}$  后输出。

PCConv 的 FLOPs 和内存访问可表述为下式：

$$q \times w \times k^2 \times c_p^2 \quad (1)$$

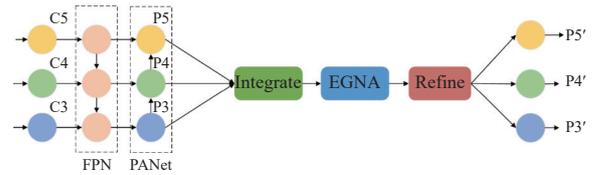
由此可得出 PCConv 的计算量是常规卷积的 1/16，1×1 卷积相较 ConvModule 的计算量则可以忽略不计。由

此得出本文设计的 C2f-PCBlock 计算开销可近似为原 Bottleneck 结构的 1/16。YOLOv8 检测模型中存在着大量的 Bottleneck 结构，所以本文的改进策略能够极大的减少计算量，便于后续模型部署并提升检测速度。

## 2.2 平衡特征金字塔

根据 PCConv 改进的特征提取网络，可以得到包含目标信息的特征层，此时如果只使用最后一层特征图进行目标的检测，则很难解决实际场景中病虫害目标尺度多变的问题，致使检测效果不佳。为此 YOLOv8 采用 FPN 和 PANet<sup>[24]</sup> 对不同特征层进行自上而下和自下而上的两次融合。

由于 FPN 关注的是相邻层的信息，以顺序的方式会使得融合的特征更多地关注相邻分辨率而较少关注其他分辨率，非相邻层的特征在融合过程中遭到稀释，不利于检测任务中对于浅层特征的需要。为减少融合过程中的特征丢失，本文引入了平衡特征融合层，增加不同维度特征之间的依赖，经嵌入高斯非局部注意力优化后得到增强后的特征层，重构后特征金字塔，即平衡特征金字塔如图 4 所示。



注：Cx, Px, Px' 表示不同尺度特征图，Integrate 表示特征融合，EGNA 表示嵌入高斯非局部注意力，Refine 表示特征再提取。  
 Note: Cx, Px, Px' represents feature maps of different scales, Integrate represents feature fusion, EGNA represents embedded Gaussian non-local attention, and Refine represents feature reextraction.

图 4 平衡特征金字塔  
 Fig.4 Balanced feature pyramid

PANet 输出特征层记为  $P_i$ ，输出特征层个数为  $L$ ，最低和最高层的索引分别记为  $l_{min}$  和  $l_{max}$ ，分辨率从高到低依次为  $P_3$ 、 $P_4$ 、 $P_5$ 。为了融合多层特征并保持语义层次，本文分别使用池化和双线性差值分别处理  $P_5$  和  $P_3$ ，使多级特征调整至与  $P_4$  相同的分辨率。通过简单的平均获得平衡的语义特征，数学表示如式 (2) 所示：

$$P' = \frac{1}{L} \sum_{i=l_{min}}^{l_{max}} P_i \quad (2)$$

然而多个上采样特征图融合会造成特征的不连续，叠加后会使得原特征重叠、混乱，对于某些像素，多个维度特征叠加会包含重复和无用的特征。因此在特征融合后，使用卷积再对特征进行一次提取，可降低混叠效应的影响。为了使融合后的特征更具识别力，本文使用 EGNA 为特征层添加非局部依赖，优化平衡语义特征。

Non-local 的核心思想在于非局部算子 (non-local operator)，它的作用是计算当前位置与其他所有位置的相似性，得到其他所有位置对当前位置的贡献度，然后通过乘法与加法，将其他位置的值与权重相乘累加到当

前位置, 达到参考所有位置, 传递远距离信息的目的。相较于卷积操作进行局部卷积, 非局部算子可以卷积全局, 通过分配不同的权重调整卷积后的特征图, 其泛化公式可由式 (3) 表示:

$$y_i = \frac{1}{C(x)} \sum_{v_j} f(x_i, x_j) g(x_j) \quad (3)$$

式中  $y_i$  为经过非局部运算后位置  $i$  的特征图,  $x_i$  为位置  $i$  的特征图,  $f(x_i, x_j)$  用于计算位置  $i$  与  $j$  的相似的程度,  $C(x)$  为归一化函数, 保证变换前后整体信息相同。  $g(x_j)$  可简单考虑为  $x$  的线性变换函数 (式 (4)):

$$g(x_j) = W_g x_j \quad (4)$$

式中  $W_g$  是需要训练的权重矩阵参数, 本文为  $1 \times 1$  卷积。

根据非局部均值和双边滤波, 可使用嵌入高斯 (embedded Gaussian) 函数作为相似函数  $f(x_i, x_j)$ , 在嵌入空间中计算高斯距离, 即

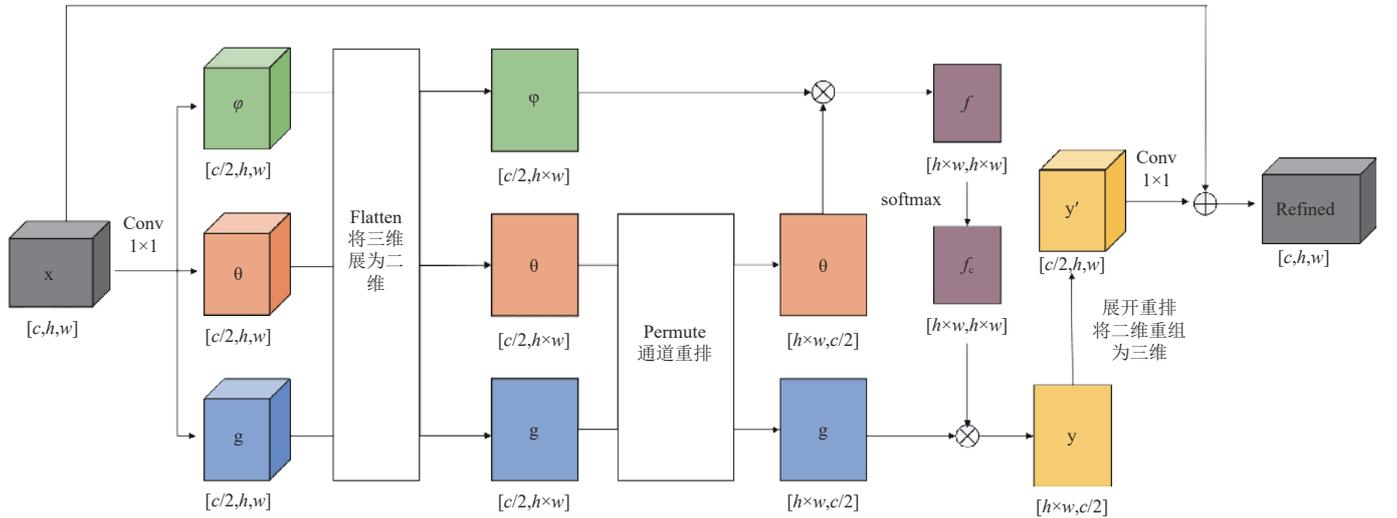
$$f(x_i, x_j) = e^{\theta(x_i)^T \varphi(x_j)} \quad (5)$$

由于在深度网络中点积更容易实现, 使用  $\theta(x_i)^T \varphi(x_j)$  作为点积相似度。其中  $\theta(x_i) = W_\theta x_i$ ,  $\varphi(x_j) = W_\varphi x_j$ , 此时归一化函数  $C(x) = \sum_{v_j} f(x_i, x_j)$ 。

由此得出, embedded Gaussian non-local 可表示为

$$y_i = \text{softmax}(\theta(x_i)^T \varphi(x_j)) g(x_j) = \frac{1}{\sum_{v_j} e^{\theta(x_i)^T \varphi(x_j)}} e^{\theta(x_i)^T \varphi(x_j)} W_g x_j \quad (6)$$

Embedded Gaussian 的使用可在嵌入空间中创建一个抽象的图像表达, 以便汇聚更多的信息。同时利用卷积, 可有效的消除因跨尺度融合特征而引起的混叠效应。嵌入高斯非局部注意力示意图如图 5 所示。经嵌入高斯非局部注意力处理后得到增强后的特征层, 完成平衡特征金字塔的设计。将特征层还原至原输入尺寸, 输入检测头, 完成水稻病虫害的检测任务。



注:  $h$  为输入特征图高,  $w$  为输入特征图宽,  $c$  为输入特征图长,  $\phi$ 、 $\theta$ 、 $g$  是经  $1 \times 1$  卷积后通道缩减为  $c/2$  的特征图,  $f$  为相似度计算矩阵,  $\text{softmax}$  为归一化函数,  $y$  为矩阵  $f_c$  与  $g$  进行矩阵相乘的结果。

Note:  $h$  represents the height of the input feature map,  $w$  represents the width of the input feature map,  $c$  represents the length of the original input feature map,  $\phi$ ,  $\theta$ ,  $g$  represents the feature reduced to  $c/2$  after  $1 \times 1$  convolution,  $f$  represents the similarity calculation matrix,  $\text{softmax}$  represents the normalization function,  $y$  represents the result of matrix multiplication  $f_c$  and  $g$ .

图 5 嵌入高斯非局部注意力

Fig.5 Embedded Gaussian non-local attention

### 2.3 MPDIUO

YOLOv8 采用的损失函数是常见的 CIUO Loss<sup>[25]</sup>, CIUO Loss 的计算公式为

$$L_{\text{CIUO}} = 1 - \text{IOU} + \frac{d^2}{C^2} + \alpha v \quad (7)$$

式中  $d$  为预测框和真实框中心点的距离,  $C$  为最小外接矩形的对角线距离,  $v$  是修正因子, 包含目标框纵横比。

由于 CIUO Loss 在纵横比描述的相对值上较为模糊, 可能存在不合理的优化, 在实际检测场景中受样本差异性影响较大, 无法针对多样性的检测目标进行自适应调整, 影响模型的收敛速度。因此, 本文引入一种基于最小点距离的损失函数 MPDIUO, 直接最小化预测边界框与真实边界框的左上角点和右下角点距离, 计算公式可表示为

$$L_{\text{MPDIUO}} = \frac{A \cap B}{A \cup B} - \frac{d_1^2}{w^2 + h^2} - \frac{d_2^2}{w^2 + h^2} \quad (8)$$

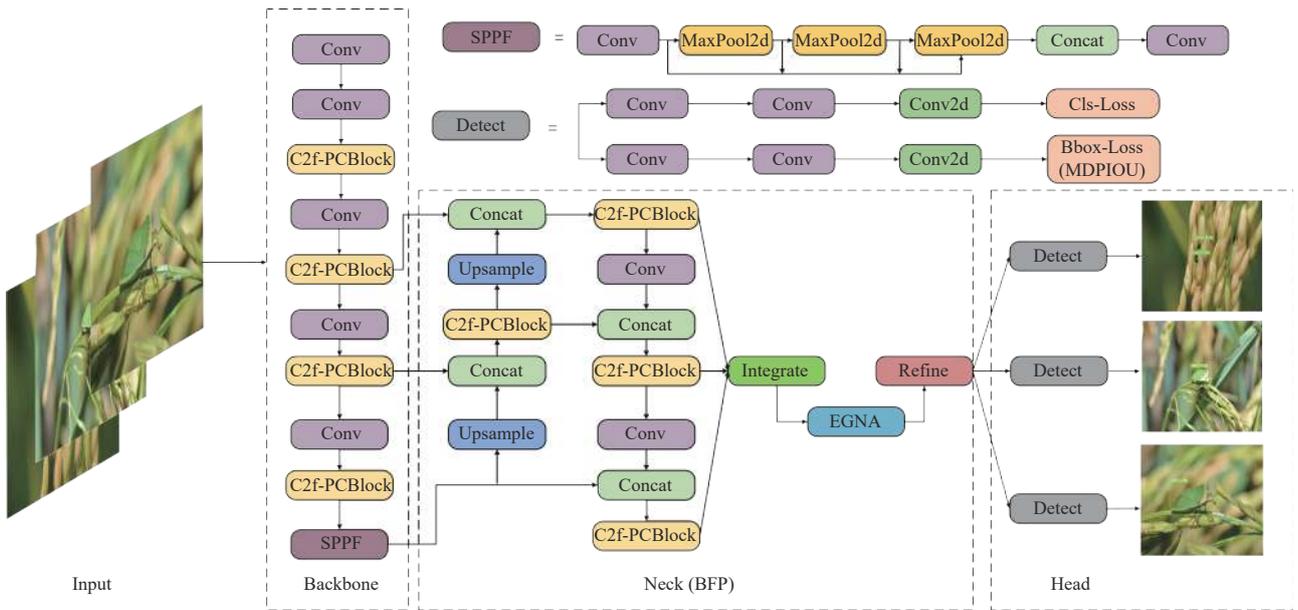
$$d_1^2 = (x_1^B - x_1^A)^2 + (y_1^B - y_1^A)^2 \quad (9)$$

$$d_2^2 = (x_2^B - x_2^A)^2 + (y_2^B - y_2^A)^2 \quad (10)$$

式中  $(x_1^A, y_1^A)$ 、 $(x_2^A, y_2^A)$  和  $(x_1^B, y_1^B)$ 、 $(x_2^B, y_2^B)$  分别表示任意凸形 A 和 B 的左上和右下点坐标,  $d_1$  和  $d_2$  分别表示两点左上角之间和右下角之间的欧式距离。

相较于 CIUO, MPDIUO 的计算更为简便, 实现起来更为直观。MPDIUO 通过对预测边界框和真实边界框的最大池化操作, 更好地处理目标的错位或者尺度变化的情况。使用 MPDIUO 可有效改善因样本差异性大而造成的检测框失真, 提高模型的鲁棒性。

改进后的 PBM-YOLOV8 目标检测模型如图 6 所示。



注：Detect 为检测头。  
Note: Detect is the detection head.

图 6 PBM-YOLOv8 网络结构图  
Fig.6 PBM-YOLOv8 network structure diagram

### 3 结果与分析

#### 3.1 试验环境和参数配置

模型训练环境为 Ubuntu20.04，计算机资源为 NVIDIA RTX 3080Ti 显卡（12GB），Inter(R) Core(TM) i7-6 700 CPU 处理器。深度学习框架为 Pytorch1.10.0，编程语言 Python3.8，CUDA 版本 11.3，CUDNN 版本 8.2.1。具体训练参数设置如下：图像输入尺寸为 640×640 像素，批次为 32，多线程设置为 8，初始学习率为 0.01，最小学习率为 0.000 1，优化器选择 SGD，训练轮数设置为 250 轮，在最后 10 轮关闭马赛克数据增强。

硬件平台为瑞芯微旗下的 RK3588 芯片为核心的 TB-RK3588B1。设备以 Rockchip RK3588 SOC 系统芯片为核心，CPU 中央处理器采用主频高达 2.4 GHz 的 4 个 Cortex-A76 和 4 个 Cortex-A55 内核，通过嵌入式神经网络

处理器 NPU 进行深度模型的边缘加速，NPU 可达 6.0T 算力，支持 INT4/INT8/INT16/FP16 运算。

#### 3.2 评价指标

本文选择准确率（precision,  $P$ ）、召回率（recall,  $R$ ）平均精度均值（mean average precision, mAP）衡量模型的检测精度<sup>[26]</sup>，每秒传输帧数（frames per second, FPS）衡量模型的检测速度，模型轻量化效果则通过 FLOPs 及模型参数量（Params）来衡量。

#### 3.3 轻量化网络对比试验

为验证 C2f-PCBlock 结构的有效性，本文在试验中以 YOLOv8 为基线，将特征提取网络更换为近年来典型的主流轻量化模型进行对比，包括 ODCnv<sup>[27]</sup>、Efficientformer<sup>[28]</sup>、EfficientViT<sup>[29]</sup> 和 MobileNetV3<sup>[30]</sup>，全面地评估 C2f-PCBlock 结构的改进效果。试验结果汇总在表 2 所示。

表 2 轻量化网络对比结果  
Table 2 Lightweight network comparison results

编号 No.	特征网络 Feature network	参数量 Params/M	浮点计算数 Floating point operations (FLOPs)/B	平均精度均值 Mean average precision (mAP)/%	帧率 Frame per second (FPS)/(帧·s <sup>-1</sup> )
1	YOLOv8	3.0	8.1	94.1	96
2	ODConv	3.1	5.8	92.3	117
3	Efficientformer	5.3	12.3	95.2	89
4	EfficientViT	4.0	9.5	92.4	93
5	MobileNetV3	3.2	7.6	91.6	104
6	C2f-PCBlock	2.5	6.9	93.7	112

从表 2 中可以看出，ODConv 的参数量较基线算法几乎不变，仅增加 0.1M，FLOPs 降低了 2.3B，FPS 提升了 21 帧/s，极大地提升了模型的检测速度和计算效率。但 mAP 下降了 1.8 个百分点。这说明 ODCnv 的高检测效率是以牺牲部分检测精度换取的，这也导致其轻量化效果最好。MobileNetV3 和 ODCnv 类似，相较于 YOLOv8 参数量和 FLOPs 下降，FPS 提高，但 mAP 同

样下降明显。Efficientformer 和 EfficientViT 虽为轻量化网络模型，但将他们引入 YOLOv8 的骨干网络中，参数量和 FLOPs 升高，FPS 下降，虽然提升了检测精度，但并没有起到对 YOLOv8 的轻量化改进效果。

将本文所设计的 C2f-PCBlock 引入 YOLOv8 的特征提取网络，相较于基线算法，模型的参数量下降了 17%，仅为 2.5 M，FLOPs 下降 15%，FPS 提升了 16 帧/s，在

6组对比试验中仅落后于 ODConv, 轻量化效果显著。在检测精度方面, 本文的轻量化骨干 mAP 相较于 YOLOv8 下降 0.4 个百分点。

综上所述, 本文设计的 C2f-PCBlock 在几乎不改变模型检测精度的同时, 在保持网络性能的同时展现了更低的参数量和浮点运算量, 显著的降低了模型的复杂度和计算成本, 为实时或资源受限的应用场景提供了解决方案, 在几种主流轻量化方案中最适合本文的检测任务。

### 3.4 损失函数对比试验

为验证损失函数 MPDIU 的效果, 本试验以 PBM-YOLOv8 为基准, 将损失函数更换为 SIOU<sup>[31]</sup>、DIOU<sup>[24]</sup>、EIOU<sup>[32]</sup> 进行对比试验, 试验结果如图 7 所示。

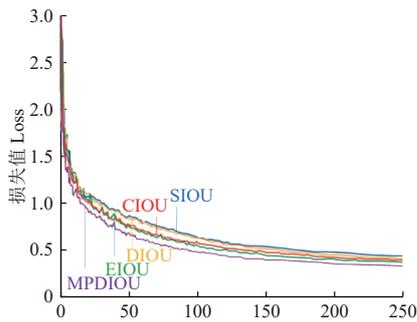


图 7 不同损失函数损失变化

Fig.7 Loss changes with different functions

使用 SIOU 作为边界框损失函数时, 模型的损失回归速度最慢, 整个训练过程中损失值始终保持在较高水平, 这说明 SIOU 在本文的研究任务中效果欠佳。CIOU 和 EIOU 的表现相似, 这两种损失函数在收敛速度和最终损失值方面差异不大, 但均略逊于 DIOU。DIOU 在训练过程中展现出更快的收敛速度和较低损失值, 表明其在边界框定位方面有一定的优势。

本文提出的 MPDIU 损失函数在试验中表现最为优异。试验数据显示, 使用 MPDIU 时, 训练开始阶段模型的梯度下降速度明显快于其他损失函数, 大约在 150 轮左右便开始稳定收敛。同时, 在模型收敛后, MPDIU 的类别损失值显著低于其他对比模型, 表现出更好的优化能力和精度。因此, MPDIU 在本试验中的结果表明, 其不仅提高了模型的收敛速度, 还在最终性能上达到了最佳效果, 具备强大的实际应用潜力。

### 3.5 消融试验

本文提出的水稻病虫害检测算法 PBM-YOLOv8, 改进策略主要有以下 3 个方面: 1) 部分卷积特征提取快; 2) 引入平衡特征金字塔; 3) 更换损失函数为 MPDIU。本试验将以 YOLOv8n 作为基线算法对 3 种改进策略进行消融试验, 结果如表 3 所示。

从表中可以看出, C2f-PCBlock 对模型的轻量化效果显著, 对比试验 2 和试验 1, 试验 5 和试验 3 的消融试验结果, 当其他部分不变引入 C2f-PCBlock 时, 模型的参数量分别下降了 0.5 和 0.9 M, FLOPs 下降了 1.2 和 2.8 B, FPS 提升了 16 和 12 帧/s, 可有效降低参数量和

计算复杂度, 并提升模型的检测速度, 虽精度下降但下降幅度均小于 1 个百分点, 对模型的检测精度影响较小, 在定位要求不高时可忽略影响。对比试验 3 和试验 1, 试验 5 和试验 2 的消融试验结果可以得出, 引入 BFP 可有效提升模型的检测精度。引入前后两组试验的 mAP<sub>50</sub> 分别提升了 1.4 和 1.1 个百分点, 但检测精度的提升是以更多的参数量和更为复杂的计算换取的。MPDIU 的引入同样对模型的精度提升作用, 通过试验数据分析, 当仅引入 MPDIU 时 (试验 4 和试验 1、试验 8 和试验 5), 模型的 mAP 分别提升了 0.2 个百分点和 0.4 个百分点。而 MPDIU 对参数量和 FLOPs 没有明显影响, 却略微提升了 FPS。

表 3 消融试验

Table 3 Ablation experiments

编号 No.	改进策略 Improvement			参数量 Params/M	FLOPs/ B	mAP/ %	FPS/ (帧·s <sup>-1</sup> )
	C2f- PCBlock	BFP	MPDIU				
1				3.0	8.1	94.1	96
2	√			2.5	6.9	93.7	112
3		√		5.7	15.4	95.5	78
4			√	3.0	8.1	94.3	109
5	√	√		4.8	12.6	94.8	90
6	√		√	2.5	6.9	94.2	115
7		√	√	5.7	15.4	95.8	81
8	√	√	√	4.8	12.6	95.2	92

消融试验结果表明, C2f-PCBlock 在轻量化方面表现显著, 能够有效降低模型的参数量和计算复杂度, 同时提升检测速度, 对精度的影响较小, 在资源受限的边缘设备中有良好的应用前景。BFP 的引入显著提升了模型的检测精度, 但需要更多的计算资源。MPDIU 则在提升模型定位精度方面效果明显, 在定位要求较高的场景中, 其对参数量和计算量影响较小, 且略微提高了推理速度。消融试验进一步证实了本文的改进策略在不同方面对模型进行了优化, 权衡了模型的检测精度和速度。

### 3.6 检测模型对比试验

为客观证实改进模型 PBM-YOLOv8 的优势, 试验将其与主流检测算法 SSD、Fast R-CNN、YOLOv5n<sup>[33]</sup> 和 YOLOv9 t 在相同的迭代次数内进行对比, 结果见表 4。本文改模型 PBM-YOLOv8 拥有最高的精确度, 召回率和 mAP, 其中准确率相较于 Fast R-CNN、SSD、YOLOv5n、YOLOv8n 和 YOLOv9 t 分别高出 6.1、8.0、4.2、1.3 和 1.9 个百分点, 召回率分别高出 7.1、8.4、5.6、2.8 和 3.2 个百分点, mAP 分别高出 5.7、7.3、3.8、1.1 和 1.5 个百分点, 均大幅度领先于其他模型, 表明其在检测准确性方面具有最佳的性能。PBM-YOLOv8 的参数量为 4.8 M, 远小于二阶段检测算法 Fast R-CNN 的 41.3 M 和 SSD 的 15.6 M, 但相较于 YOLO 系列其他小模型略高。PBM-YOLOv8 的 FPS 虽低于 YOLOv8n 的 96 帧/s 和 YOLOv9 t 的 104 帧/s, 但仍能保持 92 的帧率, 在保持高 mAP 的同时实现了高处理速度。

图 8 展示了不同算法在真实水稻场景下的检测结果。对于图片左侧的稻螟蛉虫, 仅 SSD 未能有效检测。图片右侧居中位置的纹枯病特征较为明显, 6 种算法均做出

了正确识别。YOLOv5n 和 YOLOv8n 虽然也测出了 3 处水稻病虫害，但都将左下方的杂草误检为纹枯病。图片右下角的纹枯病，由于受到了正常水稻叶片的遮挡，仅本文优化算法 PBM-YOLOv8 正确识别，且未发生误检的现象。

表 4 检测模型对比试验

Table 4 Detection model comparison experiment

模型 Models	精确度 Precision/%	召回率 Recall/%	mAP/ %	参数量 Params/M	FLOPs/ B	FPS/ (帧·s <sup>-1</sup> )
Fast R-CNN	90.5	86.7	89.5	41.3	138.6	24
SSD	88.6	85.4	87.9	15.6	87.2	65
YOLOv5n	92.4	88.2	91.4	1.7	4.3	87
YOLOv8n	95.3	91.0	94.1	3.0	8.1	96
YOLOv9 t	94.7	90.6	93.7	2.0	7.7	104
PBM-YOLOv8	96.6	93.8	95.2	4.8	12.6	92

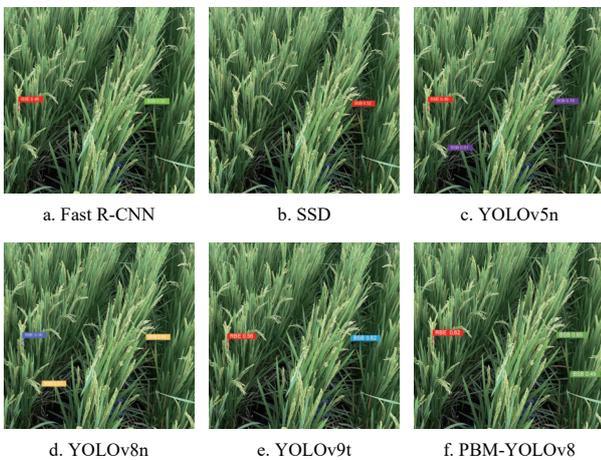


图 8 不同算法可视化结果对比

Fig.8 Comparison of visual results of different algorithm

综上所述，PBM-YOLOv8 模型在保持较高准确性的同时，具有较高的处理速度，并且在计算效率方面也有较好的表现，这些优点使得 PBM-YOLOv8 在各种实际水稻病虫害检测场景中具有很大的潜力。

### 3.7 边缘设备部署

本试验将 PBM-YOLOv8 部署在 RK3588<sup>[34]</sup> 上，用于验证改进模型在实际场景中的应用效果。为使深度学习模型适配 RK3588 AI 芯片并实现 NPU 加速的高效神经网络运算，需要采用模型转化、量化等方式完成从 GPU 到 NPU 的迁移，模型转化与量化流程如图 9 所示。

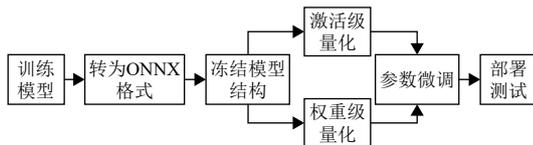


图 9 RK3588 部署流程图

Fig.9 RK3588 deployment flowchart

首先，将在 GPU 服务器上训练好的模型权重文件转化为 ONNX 通用格式，将模型从训练环境迁移到推理环境；接下来，将模型的权重和结构固化，确保在量化过程中模型不会发生变化；本文采用权重级量化联合激活级量化的量化策略，从模型参数和网络中间层输出的存

储精度两方面将模型从 32 位浮点数转换为 8 位整数，最后对量化后的模型进行校准微调，得到用于 NPU 部署加速的 RKNN 模型。

相对于 GPU 服务器上常见的 32 位浮点运算，在 RKNN 平台使用 NPU 进行 8 位定点运算进行算法加速，NPU 的运算方式不仅显著减少计算量，同时还能将能耗降低至原来的 10%，原本以 32 位浮点形式存储的深度神经网络模型权重，通过量化被有效压缩，转换为 8 位的低精度格式，显著提升了硬件资源。

上述过程主要完成了对模型整体的量化，加速了 NPU 运行网络的特征提取骨干与预测子网的回归速度。从实际工程的角度分析，检测算法的运行速率不仅与模型的结构参数有关，也与后处理速度和图像读取速度相关。采用串行的单线程处理方式，在同一时间线上难以做到并行处理多个任务，从而降低了系统的运行帧率。针对这一问题，本文设计了多线程优化方式<sup>[34]</sup>，具体线程池设计如图 10 所示。

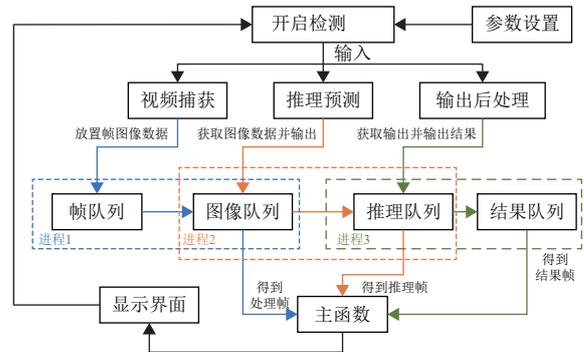


图 10 PBM-YOLOv8 线性池

Fig.10 PBM-YOLOv8 thread pool

针对本文算法，将检测任务划分为视频帧捕获、推理预测和后处理输出 3 部分，设计如图 10 所示的 3 个线程池，按照先入先出的顺序依次向后传播，输出推理结果。多线程优化的方式充分利用多核 NPU 的模型推理与 CPU 后处理的并行加速可极大提升检测的运行速度。

表 5 为模型改进前后在不同硬件环境下的运行结果。试验表明 PBM-YOLOv8 在 GPU 上检测速度最快，可达到 92.8 帧；部署于 NPU 平台进行量化仿真时速率最慢且精度有所下降；实际应用 RK3588 嵌入式系统时，NPU 加速后的模型速率均满足 24 帧以上的实时性要求；部署于嵌入式平台并经过工程上的多线程优化后，多核 NPU 的检测速率可达 71.4 帧/S，检测算法的平均精度可达 95.1%，相同环境下较 YOLOv8n 高出 1.2 个百分点。试验数据充分验证了 PBM-YOLOv8 应用于资源受限的嵌入式硬件设备中的有效性和实时性，能够做到对水稻病虫害的精准识别。

部分 YOLOv8n 和 PBM-YOLOv8 在 RK3588 上的检测结果如图 11 所示。通过对比分析可以得到：在光照强度较高时，改进算法 PBM-YOLOv8 展现出了更高的置信度，检测结果更可靠。对于小尺度目标，PBM-YOLOv8 纠正了原始版本的误检问题，证明本文算法可更准确的

识别多尺度目标, 减少了误报的可能性。而当水稻植株受到杂草及叶片遮挡时, 改进算法正确检测到了原模型未检测到的 2 处病害, 解决了原算法对高相似度目标及植株遮挡的漏检问题, 证明本文算法可有效改善水稻病虫害目标表达特征不充分的问题。

表 5 不同硬件环境下的运行结果

Table 5 Performance results across different hardware environments

模型 Models	硬件环境 Hardware environment	FPS/ (帧·s <sup>-1</sup> )	mAP/%
YOLOv8n	RTX 3080Ti GPU	96.0	94.1
	RKNNToolkit-Simulators	22.3	93.6
	RK3588 NPU	43.7	93.9
PBM-YOLOv8	RTX 3080Ti GPU	92.8	95.2
	RKNNToolkit-Simulators	21.6	94.8
	RK3588 NPU	42.0	95.1
PBM-YOLOv8 (Multithreading)	RK3588 NPU	71.4	95.1

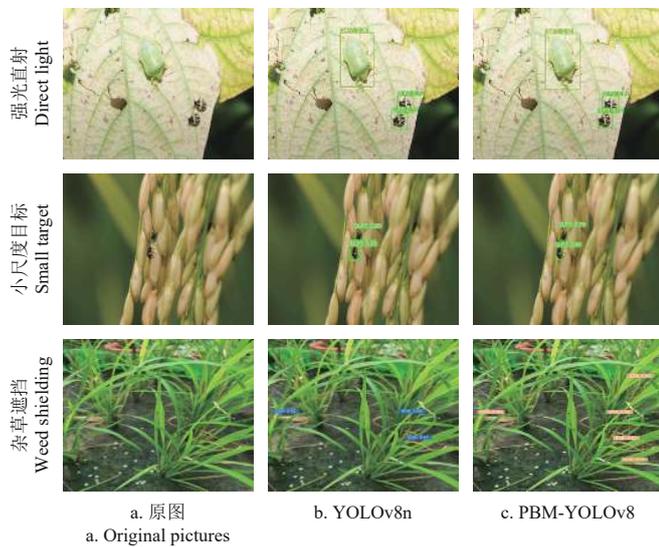


图 11 YOLOv8n 与 PBM-YOLOv8 检测效果

Fig.11 Detection results with YOLOv8n and PBM-YOLOv8

## 4 结论

本文针对水稻病虫害种类多, 尺度差异大, 生长环境复杂致使检测精度不高的问题, 设计一种改进算法 PBM-YOLOv8。首先通过带有嵌入高斯非局部注意力的平衡特征层重构特征金字塔减少特征丢失, 从而更好地识别多尺度目标; 其次基于 PConv 设计 PCBlock, 降低模型的参数量和计算复杂度, 使得模型轻量化, 便于在边缘设备上部署; 最后引入 MPDIU, 改善病害种类繁多造成的检测框失真, 同时减少模型的计算负担。通过试验结果对比分析可得出如下结论:

1) 在相同试验条件下, 改进的 PBM-YOLOv8 通过与 Fast R-CNN、SSD、YOLOv5n 和 YOLOv8n 进行对比, 在水稻病虫害数据集上取得了更为优异的试验效果, 同时在 GPU 上相较于原始 YOLOv8n 基线模型准确率提升了 1.1 个百分点, 可达到 95.2% 的平均准确率均值。

2) 通过将模型部署在 RK3588 嵌入式开发板上, 进一步证实改进算法 PBM-YOLOv8 的准确性和可行性。

本文采用多线程优化的方式充分利用多核 NPU 的模型推理与 CPU 后处理的并行加速, 加速后检测速度达到 71.4 帧/s, 平均准确率均值为 95.1%, 满足实际应用的需求, 可实现对水稻病虫害的实时精准检测。

## [参 考 文 献]

- [1] 姜晓剑, 汤亮, 刘小军, 等. 中国主要稻作区水稻生产气候资源的时空特征[J]. 农业工程学报, 2011, 27(7): 238-245.  
JIANG Xiaojian, TANG Liang, LIU Xiaojun, et al. Spatial and temporal characteristics of rice production climatic resources in main growing regions of China[J]. *Transactions of the Chinese Society of Agricultural Engineering (Transactions of the CSAE)*, 2011, 27(7): 238-245. (in Chinese with English abstract)
- [2] 黄双萍, 孙超, 齐龙等. 基于深度卷积神经网络的水稻穗瘟病检测方法[J]. 农业工程学报, 2017, 33(20): 169-176.  
HUANG Shuangping, SUN Chao, QI Long, et al. Rice panicle blast identification method based on deep convolution neural network[J]. *Transactions of the Chinese Society of Agricultural Engineering (Transactions of the CSAE)*, 2017, 33(20): 169-176. (in Chinese with English abstract)
- [3] 范春全, 何彬彬. 基于迁移学习的水稻病虫害识别[J]. 中国农业信息, 2020, 32(2): 36-44.  
FAN Chunquan, HE Binbin. Rice pest and disease recognition based on transfer learning[J]. *China Agricultural Information*, 2020, 32(2): 36-44. (in Chinese with English abstract)
- [4] FENG J, YANG L, YU C, et al. Image recognition of four rice leaf diseases based on deep learning and support vector machine[J]. *Computers and Electronics in Agriculture*, 2020, 179: 105824.
- [5] 李柯泉, 陈燕, 刘佳晨, 等. 基于深度学习的目标检测算法综述[J]. 计算机工程, 2022, 48(7): 1-12.  
LI Kequan, CHEN Yan, LIU Jiachen, et al. Survey of deep learning-based object detection algorithms[J]. *Computer Engineering*, 2022, 48(7): 1-12. (in Chinese with English abstract)
- [6] WANG X, SHRIVASTAVA A, GUPTA A. A-fast-rcnn: Hard positive generation via adversary for object detection[C] *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Honolulu: IEEE, 2017: 2606-2615.
- [7] ZHAI S, SHANG D, WANG S, et al. DF-SSD: An improved SSD object detection algorithm based on DenseNet and feature fusion[J]. *IEEE Access*, 2020, 8: 24344-24357
- [8] 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: IEEE, 2016: 779-788.
- [9] REDMON J, FARHADI A. YOLO9000: Better, faster, stronger[C] *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Honolulu: IEEE, 2017: 6517-6525.
- [10] REDMON J, FARHADI A. YOLOv3: An incremental

- improvement[EB/OL]. (2018-04-08) [2023-06-10] <https://arxiv.org/abs/1804.02767>.
- [11] ALEXEY B, WANG C, LIAO H. YOLOv4: Optimal speed and accuracy of object detection [EB/OL]. (2020-04-23) [2023-06-10] <https://arxiv.org/abs/2004.10934>.
- [12] 郭文娟, 冯全, 李相周. 基于农作物病害检测与识别的卷积神经网络模型研究进展[J]. 中国农机化学报, 2022, 43(10): 157-166.  
GUO Wenjuan, FENG Quan, LI Xiangzhou. Research progress of convolutional neural network model based on crop disease detection and recognition[J]. Journal of Chinese Agricultural Mechanization, 2022, 43(10): 157-166. (in Chinese with English abstract)
- [13] RAHMAN C R, Arko P S, Ali M E, et al. Identification and recognition of rice diseases and pests using convolutional neural networks[J]. *Biosystems Engineering*, 2020, 194: 112-120.
- [14] 余颢, 吴伶, 单鲁泉. 基于 SSD 网络模型改进的水稻害虫识别方法[J]. 郑州大学学报 (理学版), 2020, 52(3): 49-54.  
SHE Hao, WU Ling, SHAN Luquan. Improved rice pest recognition based on SSD network model[J]. Journal of Zhengzhou University (Natural Science Edition), 2020, 52(3): 49-54. (in Chinese with English abstract)
- [15] 吴子炜, 夏芳, 陆林峰, 等. 基于改进 YOLOv5 的水稻主要害虫识别方法[J]. 江苏农业科学, 2023, 51(21): 218-224.  
WU Ziwei, XIA Fang, LU Linfeng, et al. Rice major pest identification method based on improved YOLO v5[J]. Jiangsu Agricultural Sciences, 2023, 51(21): 218-224. (in Chinese with English abstract)
- [16] 马超伟, 张浩, 马新明, 等. 基于改进 YOLOv8 的轻量化小麦病害检测方法[J]. 农业工程学报, 2024, 40(5): 187-195.  
MA Chaowei, ZHANG Hao, MA Xinming, et al. Method for the lightweight detection of wheat disease using improved YOLOv8[J]. *Transactions of the Chinese Society of Agricultural Engineering (Transactions of the CSAE)*, 2024, 40(5): 187-195. (in Chinese with English abstract)
- [17] CHEN J, KAO S, HE H, et al. Run, Don't walk: Chasing higher FLOPS for faster neural networks[C]Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Canda: IEEE, 2023: 12021-12031.
- [18] GO J, RYU J. Spatial bias for attention-free non-local neural networks[J]. *Expert Systems with Applications*, 2024, 238: 122053.
- [19] 王玉, 何文清, 靳拓, 等. 基于生命周期评价的东北地区覆膜水稻环境影响分析[J]. 农业资源与环境学报, 2024, 41(4): 918-925.  
WANG Yu, HE Wenqing, JIN Tuo, et al. Environmental impact assessment of the life cycle of rice cultivation by plastic film mulching in northeast China[J]. Journal of Agricultural Resources and Environment, 2024, 41(4): 918-925. (in Chinese with English abstract)
- [20] FAN J, CUI L, FEI S. Waste detection system based on data augmentation and YOLO\_EC[J]. *Sensors*, 2023, 23(7): 3646
- [21] XIE S, SUN H. Tea-YOLOv8s: A tea bud detection model based on deep learning and computer vision[J]. *Sensors*, 2023, 23(14): 6576.
- [22] PANG J, CHEN K, SHI J, et al. Libra R-CNN: Towards balanced learning for object detection[C]Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach: IEEE, 2019: 821-830.
- [23] SILIANG M, YONG X. Mpdjou: a loss for efficient and accurate bounding box regression[J]. arXiv preprint arXiv: 2307.07662, 2023.
- [24] LIU S, QI L, QIN H, et al. Path Aggregation Network for Instance Segmentation [C] Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City: IEEE, 2018: 8759-8768.
- [25] 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: IEEE, 2020: 12993-13000.
- [26] 鲍文霞, 谢文杰, 胡根生, 等. 基于 TPH-YOLO 的无人机图像麦穗计数方法[J]. 农业工程学报, 2023, 39(1): 155-161.  
BAO Wenxia, XIE Wenjie, HU Gensheng, et al. Wheat ear counting method in UAV images based on TPH-YOLO[J]. *Transactions of the Chinese Society of Agricultural Engineering (Transactions of the CSAE)*, 2023, 39(1): 155-161. (in Chinese with English abstract)
- [27] LI C, ZHOU A, YAO A. Omni-dimensional dynamic convolution[J]. arXiv preprint arXiv:2209.07947, 2022.
- [28] LI Y, YUAN G, WEN Y, et al. Efficientformer: Vision transformers at mobilenet speed[J]. *Advances in Neural Information Processing Systems*, 2022, 35: 12934-12949.
- [29] ZHAI X, KOLESNIKOV A, Houlby N, et al. Scaling vision transformers[C]Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans: IEEE, 2022: 12104-12113.
- [30] 黄铝文, 关非凡, 谦博, 等. 基于 2D DWT 与 MobileNetV3 融合的轻量化茶叶病害识别[J]. 农业工程学报, 2023, 39(24): 207-214.  
HUANG Lyuwen, GUAN Feifan, QIAN Bo, et al. Recognizing tea diseases with fusion on 2D DWT and MobileNetV3[J]. *Transactions of the Chinese Society of Agricultural Engineering (Transactions of the CSAE)*, 2023, 39(24): 207-214. (in Chinese with English abstract)
- [31] TYCHSEN-SMITH L, PETERSSON L. Improving object localization with fitness nms and bounded iou loss[C] Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City: IEEE, 2018: 6877-6885.
- [32] YANG Z, WANG X, LI J. ElIoU: An improved vehicle detection algorithm based on vehiclenet neural network[C] Journal of Physics: Conference Series. IOP Publishing, 2021: 012001.

- [33] JOCHER G. Ultralytics/yolov5 [EB/OL]. (2022-11-22) [2023-06-10]<https://github.com/ultralytics/yolov5>.
- [34] 陈立群, 邹旭, 张磊, 等. 基于国产商用器件的星载智能目标检测技术[J]. 航空学报, 2023, 44(S2): 6-15.
- CHEN Liqun, ZOU Xu, ZHANG Lei, et al. Intelligent target detection technology based on domestic commercial devices[J]. Acta Aeronautica et Astronautica Sinica, 2023, 44(S2): 6-15. (in Chinese with English abstract)

## Detecting rice disease using PBM-YOLOv8

LIU Peng , ZHANG Tianyi , RAN Xin , SHI Jialin , BI Yuxuan , WANG Caixia

(College of Electronic Information Engineering, Changchun University of Science and Technology, Changchun 130022, China)

**Abstract:** Rice diseases and pests have posed a serious threat to agricultural production in recent years. Particularly, false and missed detections have been caused by the diversity, scale, and complex growth environments of these pests. This study aims to enhance the detection accuracy of rice diseases and pests, in order to facilitate the deployment of detection models on edge devices. An improved YOLOv8-based model was developed to detect rice diseases and pests. Several innovations were also incorporated to optimize the performance of the model. Firstly, partial convolution (PConv) was introduced to design the PCBlock structure and then integrated into the feature extraction module of YOLOv8. As such, the module of lightweight feature extraction known as C2f-PCBlock significantly reduced the number of model parameters, thereby enhancing the detection speed without compromising accuracy. The complexity of the model was then reduced to deploy on the resource-constrained edge devices. Secondly, the semantic information was diluted to cause the feature fusion across non-adjacent layers. A balanced Feature fusion layer was then added to the neck network. feature pyramid networks (FPN) was restructured into balanced feature pyramid (BFP). The preservation of critical features was improved during fusion. Additionally, the embedded Gaussian non-local attention (EGNA) mechanism was incorporated to mitigate the aliasing behavior from the multi-layer feature fusion. The important features were retained to reduce the loss of information, leading to more accurate detections. The loss function of the model was also enhanced to replace the standard IoU-based loss with MPDIoU. The distortions were avoided in the detection bounding boxes, due to the significant variations in sample sizes. MPDIoU was used to reduce these distortions for less computational burden, indicating more efficient training. Extensive experiments were conducted on a rice pest and disease dataset. The results demonstrate that the improved model, named PBM-YOLOv8, outperformed the rest. Specifically, PBM-YOLOv8 was achieved in the precision rate of 6.1, 8.0, 4.2, 1.3, and 1.9 percentage points higher than Fast R-CNN, SSD, YOLOv5n, YOLOv8n, and YOLOv9t, respectively. The recall rates were also significantly improved, with an increase of 7.1, 8.4, 5.6, 2.8, and 3.2 percentage points, respectively. Furthermore, the mean average precision (mAP) values were 5.7, 7.3, 3.8, 1.1, and 1.5 percentage points higher than those of the compared models, respectively. PBM-YOLOv8 was selected as a leading model, in terms of detection accuracy. Moreover, the PBM-YOLOv8 model was highly efficient with only 4.8M parameters, which was far less than the 41.3M of Fast R-CNN and the 15.6 M of SSD. A high processing speed was maintained with an FPS of about 92, in order to balance the accuracy with computational efficiency. The effectiveness of the improved model was further validated. PBM-YOLOv8 was deployed on the RK3588 embedded board. The deployment showed an accelerated detection speed of 71.4 frames per second, with an average accuracy of 95.1%. This performance can fully meet the demands of practical applications, thus enabling real-time and accurate detection of rice pests and diseases. The multi-thread optimization was adopted in the deployment. The parallel acceleration of multi-core NPU was also realized for the model inference and CPU for post-processing. The model has further enhanced the efficiency and feasibility for real-world use.

**Keywords:** image recognition; deep learning; target detection; rice diseases; YOLOv8; BFP; grand central dispatch