doi:10.6041/j.issn.1000-1298.2026.23.031

基于改进 YOLO v5s 的水稻病害图像检测模型研究

杨　宁[1]　张宇慧[1]　刘淑华[2]　毛罕平[3]　王亚飞[3]　程　巍[1]　王爱英[2]

(1.江苏大学电气信息工程学院,镇江 212013;2.中国水稻研究所,杭州 311400;
3.江苏大学农业工程学院,镇江 212013)

摘要:针对基于图像的水稻病害现场检测技术计算能力不足,难以实现快速识别和实时检测的问题,本文提出了一种基于现场可编程门阵列(Field programmable gate array, FPGA)本地部署推理的改进 YOLO v5s 目标检测方法。首先,将主干网络替换为轻量级网络 ShuffleNetV2,以减少冗余计算和内存访问;其次,引入归一化 Wasserstein 距离(Normalized Wasserstein distance, NWD)损失函数,以增强对小目标病害的检测能力;然后,在颈部网络的每个综合卷积模块(Comprehensive convolution block, C3)后添加指数移动平均注意力机制(Exponential moving average, EMA),以增强病害特征提取能力;最后将激活函数替换为修正线性单位(Rectified linear unit, ReLU)激活函数,进一步简化计算。改进后模型的参数量减少 54.3%,计算量减少 62.7%,帧速率提升 16%,精确率提高 0.9 个百分点,平均精度均值为 86.8%。将改进模型映射到 FPGA 平台上,并通过定点量化技术将模型内存占用量缩至 4.2 MB,相较于原模型减少了 71.03%。试验结果表明,在 FPGA 平台上部署后的平均精度均值为 85.6%,移植精度达 98.6%,单幅病害图像平台检测时间为 0.048 5 s,平均功耗为 12.80 W。该研究为农业生产过程中的病害检测提供高效的边缘解析方法。

关键词:水稻病害;图像识别;深度学习;目标检测;轻量化;FPGA

中图分类号:S431　　**文献标识码**:A　　**文章编号**:1000-1298(2026)23-0329-09　　**OSID**:

Rice Disease Image Recognition Model Based on Improved YOLO v5s and FPGA Edge Resolution

Yang Ning[1]　Zhang Yuhui[1]　Liu Shuhua[2]　Mao Hanping[3]　Wang Yafei[3]　Cheng Wei[1]　Wang Aiying[2]

(1. *School of Electrical and Information Engineering*, *Jiangsu University*, *Zhenjiang* 212013, *China*
2. *China National Rice Research Institute*, *Hangzhou* 311400, *China*
3. *School of Agricultural Engineering*, *Jiangsu University*, *Zhenjiang* 212013, *China*)

Abstract: Aiming to address the challenges of insufficient computational power, slow recognition, and the lack of real-time detection in image-based rice disease field detection, an improved YOLO v5s-based (you only look once version 5 small) model was proposed for rice disease image recognition and FPGA-based edge computing. Firstly, the backbone network was replaced with the lightweight ShuffleNetV2 to reduce redundant computation and memory access. Secondly, the normalized Wasserstein distance (NWD) loss function was introduced to enhance detection performance for small-target diseases. Thirdly, an exponential moving average (EMA) attention mechanism was added after each comprehensive convolution block (C3) in the neck network to strengthen feature extraction capabilities. Finally, the activation function was replaced with the rectified linear unit (ReLU) to further simplify calculations. The improved model reduced the number of parameters by 54.3%, computational complexity by 62.7%, increased FPS by 16%, and achieved a precision improvement of 0.9 percentage points, resulting in an mAP of 86.8%. The optimized model was mapped onto the FPGA platform, where fixed-point quantization reduced the model size to 4.2 MB, a 71.03% reduction compared with that of the original model. Experimental results demonstrated that the model achieved an mAP of 85.6%, a migration

收稿日期:2025-08-26　修回日期:2025-12-10

基金项目:国家重点研发计划青年科学家项目(2022YFD2000200)和国家自然科学基金(面上)项目(32171895)

作者简介:杨宁(1984—),男,教授,博士,主要从事智慧农业作物病害预警研究,E-mail:yangning7410@163.com

通信作者:王爱英(1977—),女,助理研究员,博士,主要从事水稻病虫害的智能测报技术研究,E-mail:wangaiying@caas.cn

accuracy of 98.6%, an average detection time of 0.048 5 s per image, and an average power consumption of 12.80 W after being deployed on the FPGA platform. The research result can provide an efficient edge analytics solution for disease detection in agricultural production processes.

Key words: rice diseases; image recognition; deep learning; object detection; lightweight design; FPGA

0 引言

水稻是重要的粮食作物,是我国 60% 以上人口的主食[1]。水稻胡麻叶斑病和稻瘟病作为水稻种植中常见且危害严重的病害,两者视觉症状相似,依靠人眼观察和经验判断往往容易误诊,一旦检测不及时或判断错误,将错失最佳防治时机,进而加剧病害扩散[2]。特别是稻瘟病这一全球性真菌病害,每年导致的水稻产量损失高达 10% ~30%,全球经济损失更是达到数十亿美元[3]。因此,及时、准确地识别这些病害对于控制其传播和保障粮食产量至关重要[4]。

传统的水稻病害检测方法主要包括显微镜检测、光谱分析和聚合酶链式反应(Polymerase chain reaction, PCR)检测等。文献[5-7]方法虽然能够准确识别水稻病害,但存在检测时间长、成本高、智能化水平低且需依赖专门设备和技术人员等问题,难以在大规模农田检测中推广。

相比传统检测方法,基于机器学习的图像分类技术在作物病害图像识别方面得到了广泛的研究[8-9]。Pinki 等[8]采用 K-means 聚类法分割图像,提取颜色、纹理和形状特征,并结合支持向量机(Support vector machines, SVM)识别 3 种水稻病害。Pantazi 等[9]结合 H 通道纹理特征和 SVM,实现了葡萄叶片病害的自动检测。但在特征提取时高度依赖人工设计,且对病害区域的精确定位能力不足。

随着深度学习的不断发展,更多先进算法被应用于病害检测,进一步提升识别效果。文献[10-12]方法虽然一定程度上提升了病害检测的精度,但仍伴随较大的计算量,通常需要依赖高算力的服务器进行推理,在网络不稳定或资源受限的农田环境中,难以实现对病斑的实时检测。

随着应用需求的多样化和检测场景的复杂化,将深度学习模型与 FPGA 结合已成为提升检测实时性和能效的有效途径。FPGA 凭借灵活的拓扑结构和强大的并行计算能力,在嵌入式系统中优势显著[13]。Li 等[14]提出了一种基于 Xilinx ZYNQ FPGA 的深度学习 YOLO(You only look once)网络改进算法,通过优化 YOLO 网络模型和定点技术等解决 FPGA 芯片资源有限的问题。Saidani 等[15]在 Xilinx Zynq-7000 片上系统(SoC)上实现了 YOLO v5 算法,在频率 250 MHz 下达到了 120 f/s 的性能,同时效率是原来的两倍,预测时间大大缩短。

当前,水稻病害检测多集中于可控的实验室环境,而非更贴近实际应用的复杂大田场景。为满足病害检测低计算量、低时延和高精度的实际需求,本文基于性能优异的 YOLO v5s 模型,提出一种轻量化的小目标检测方法。该方法旨在开发一种能够高效部署在边缘计算设备 FPGA 上的模型,实现精确度高、检测速度快的水稻病害检测。此外,通过模型量化技术,将浮点运算转换为定点运算,进一步降低计算复杂度和存储开销,使其在资源有限的 FPGA 平台上更加高效,为复杂场景下快速识别水稻病害提供支持。

1 图像采集与数据集构建

1.1 图像采集

稻瘟病和胡麻叶斑病是水稻常见的两种病害,稻瘟病[16]典型病斑呈梭形,中央为灰白色,内圈为褐色,最外层带有淡黄色晕圈。胡麻叶斑病[17]病斑中央为褐色至黑色,周围伴有深浅不一的黄色晕圈。为构建更贴近实际场景的病害检测数据集,在病害专家的指导下,于 2023 年 9—10 月及 2024 年 4—5 月在中国水稻研究所的田间检测基地 C6-2 地块(18 m×75 m)进行大田实地采样工作。使用配备超清广角镜头的 AR 眼镜,在阴天、雨天和晴天 3 种典型天气下的清晨、正午和黄昏时段,从不同角度和距离采集两类水稻病害图像,共计 1 532 幅,图像分辨率为 4 080 像素×3 072 像素。这些数据为进一步优化模型的病害检测能力提供了真实且多样化的样本支持。图 1 为部分水稻叶片病害图像示例。

1.2 数据集构建

为提升模型泛化能力,对原始分辨率为 4 080 像素×3 072 像素的图像进行切片分割处理。通过 Python 脚本将图像切割成 640 像素×640 像素的小块共 4 885 幅,其中包含 836 幅胡麻叶斑病图像和 4 049 幅稻瘟病图像。为提升模型对胡麻叶斑病的检测性能,采用旋转、翻转和亮度

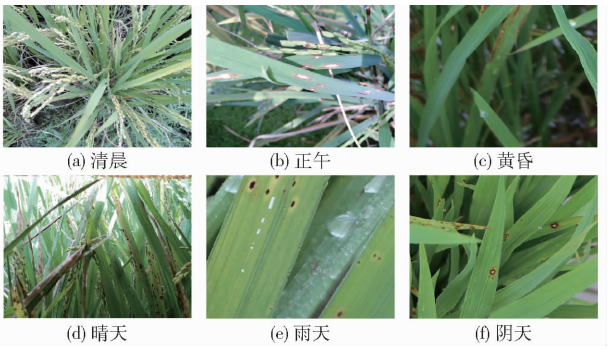

图 1　水稻病害图像示例

Fig. 1　Example of rice disease image

增强等数据增强策略，生成 2 187 幅新增样本，以改善数据分布并优化模型训练。数据标注环节采用在线工具 LabelImg 逐幅标注病斑区域，确保目标框覆盖病斑区域，并进行人工复核。最后，将数据集按 7∶2∶1的比例划分为训练集、验证集和测试集。图 2 展示了胡麻叶斑病图像增强示例，表 1 汇总了数据集的标注信息。

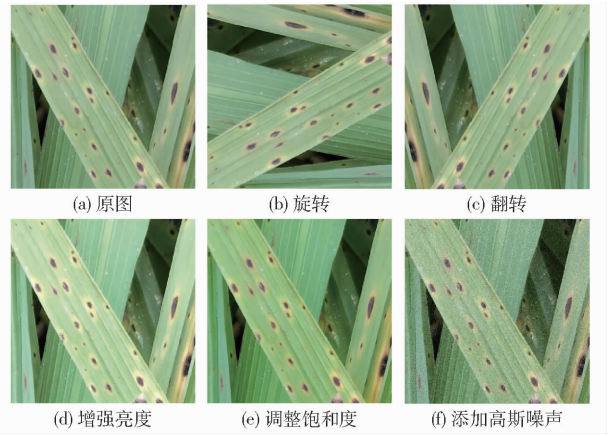

图 2　数据增强示例

Fig. 2　Data enhancement example

表 1　水稻叶片病害图像数据集

Tab. 1　Leaf image data set of rice disease

病害种类	训练集	测试集	验证集	总计	总标记处
胡麻叶斑病	2 116	605	302	3 023	33 462
稻瘟病	2 834	809	406	4 049	12 900

2　改进 YOLO v5s 的水稻病害图像目标检测算法

2.1　YOLO v5s 模型概述

YOLO v5 是一种端到端目标检测模型，相较于二阶段模型（如 Faster R－CNN），具有更快的检测速度和更简便的部署流程[18]。在农业领域，YOLO v5 已广泛应用于作物病害识别，并取得了显著效果[19]。YOLO v5s 是 YOLO v5 系列中轻量级版本，其网络复杂度较低，参数量较少，计算延迟低同时具有良好的检测精度，适合部署在 FPGA 等资源受限的嵌入式设备中[20]。

2.2　改进 YOLO v5s 模型结构

改进后的网络结构如图 3 所示，包含主干网络（Backbone）、颈部网络（Neck）和头部网络（Head）。针对 YOLO v5s 模型在水稻病害检测中参数过多、难以在移动设备上运行的问题，采用了轻量化的 ShuffleNetV2 作为主干网络。在颈部网络中，通过上采样、特征拼接和卷积操作进行病害特征融合，并引入 EMA 注意力模块与 C3 模块协同作用，以提升复杂背景下对水稻病斑的特征提取能力。在主干网络末端集成 SPPF 模块以扩展感受野，并结合 NWD 损失函数以增强对小目标水稻病害的检测性能。ReLU 激活函数被用于优化回归损失，既降低了模

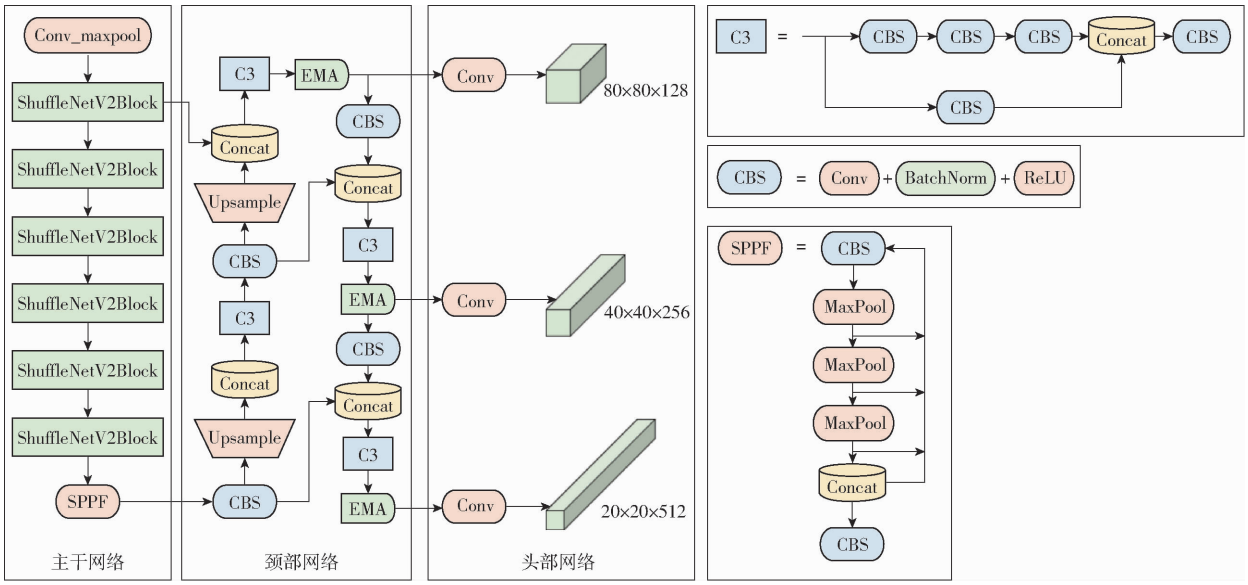

图 3　改进的 YOLO v5s 整体网络结构图

Fig. 3　Overall network structure of improved YOLO v5s

型复杂度,又保持了较高的检测精度。最终,利用检测头进行输出,实现水稻病斑的精准定位与识别,为在 FPGA 上的部署提供有效方案。

2.2.1 网络模型轻量化

图 4 展示了 ShuffleNetV2[21] 的核心单元结构,表 2 为其网络结构。图 4a 为标准的通道分割过程,首先输入特征图通过通道分割分为两个分支:左侧分支直接进行恒等映射并连接到输出,右侧分支则进行 3 个连续的卷积操作。随后,两个分支的输出通过级联操作合并,并进行通道混洗,使得不同通道的病害信息得以交换和融合,在不增加计算量的同时,增强了网络的特征表达能力。对于下采样单元,图 4b 显示了与标准单元的主要区别。下采样单元不再进行通道分割,而是将输入特征图直接复制到两个分支。每个分支分别通过步长为 2 的 3×3 深度可分离卷积实现空间下采样。随后,通过级联操作将两个分支的输出合并,在减少特征图空间尺寸的同时增加通道数,确保下采样后的特征图仍然具有丰富的病斑信息表达能力,从而提升网络的整体性能。

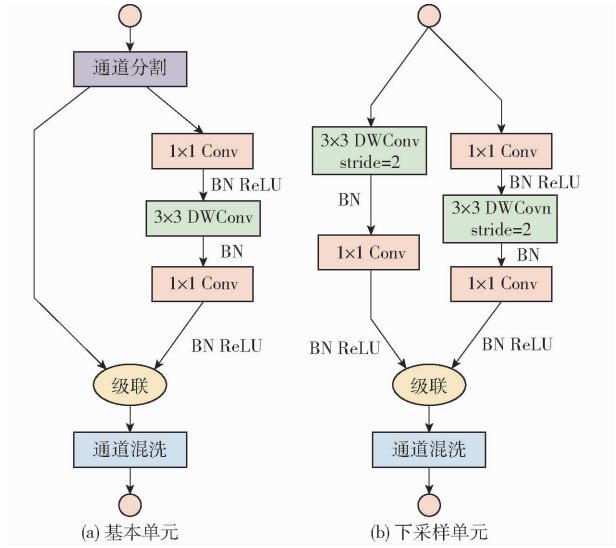

图 4 ShuffleNetV2 核心单元

Fig. 4 Core unit of ShuffleNetV2

表 2 ShuffleNetV2 网络结构

Tab. 2 ShuffleNetV2 network structure

网络层	输出大小	卷积核大小	重复次数	步幅
Image	224×224			
Conv1	112×112	3×3	1	2
MaxPool	56×56	3×3	1	2
Stage2	28×28		1、3	2、1
Stage3	14×14		1、7	2、1
Stage4	7×7		1、3	2、1
Conv5	7×7	1×1	1	1
GlobalPool	1×1	7×7		

2.2.2 引入损失函数

YOLO v5s 目标检测模型采用完全交并比(Complete-IoU,CIoU)损失函数,该损失函数通过综合考虑交并比(Intersection over union, IoU)、中心点距离和长宽比一致性来提高定位精度,但在小目标定位性能方面表现不足。NWD[22] 的一个主要优点是对微小物体位置偏差不敏感,能够更准确地捕捉和衡量微小物体的特征,适合处理水稻病斑的检测任务,减少误差。

首先使用最优输运理论中的 Wasserstein 距离来计算分布之间的距离。对于两个二维高斯分布 $\mu_1 = N(m_1, \Sigma_1)$ 和 $\mu_2 = N(m_2, \Sigma_2)$,它们之间的二阶 Wasserstein 距离定义为

$$W_2^2(\mu_1,\mu_2) = \| m_1 - m_2 \|_2^2 + \| \Sigma_1^{1/2} - \Sigma_2^{1/2} \|_F^2 \quad (1)$$

边界框 $\boldsymbol{A} = (c_{xa}, c_{ya}, w_a, h_a)$,其中 c_{xa}、c_{ya} 是边界框 $\boldsymbol{A}$ 的中心坐标,w_a 是宽度,h_a 是高度。边界框 $\boldsymbol{B} = (c_{xb}, c_{yb}, w_b, h_b)$,其中 c_{xb}、c_{yb} 是边界框 $\boldsymbol{B}$ 的中心坐标,w_b 是宽度,h_b 是高度。基于这些边界框特征,结合高斯分布 N_a、N_b,可简化为

$$W_2^2(N_a,N_b) = \left\| \left[c_{xa}, c_{ya}, \frac{w_a}{2}, \frac{h_a}{2} \right]^{\mathrm{T}} - \left[c_{xb}, c_{yb}, \frac{w_b}{2}, \frac{h_b}{2} \right]^{\mathrm{T}} \right\|_2^2 \quad (2)$$

虽然 Wasserstein 距离能衡量边界框间的差异,但它不能像 IoU 那样直接作为 0 到 1 之间的相似度度量。因此,对 Wasserstein 距离进行归一化处理,转化为一种新的相似度度量,称为归一化 NWD。这种转换保留了 Wasserstein 距离对边界框细节的敏感性,使其更适用于大田病害检测。其表达式为

$$NWD(N_a,N_b) = \exp\left(-\frac{\sqrt{W_2^2(N_a,N_b)}}{C_1} \right) \quad (3)$$

式中,常数 C_1 用于归一化距离尺度,使 NWD 值在[0,1]区间内。

2.2.3 添加 EMA 注意力机制

为进一步提升性能,在 YOLO v5s 颈部网络的 3 个 C3 模块后添加 EMA 模块[23]。如图 5 所示,输入特征图的尺寸为 $C \times H \times W$,其中 C 为通道数,H、W 分别为高度和宽度。首先,将输入特征图按通道维度划分为 G 个子特征组,并采用 3 个并行分支提取每组的注意力权重。在 1×1 分支中,使用两个一维全局平均池化操作(X Avg Pool 表示水平池化;Y Avg Pool 表示垂直池化)来编码空间方向的跨通道信息。与此同时,3×3 分支通过 3×3 卷积核捕获病害多尺度特征。随后,将两个编码后的特征沿图像高度方向连接,并共享同一个 1×1 卷积。输出被分解为两个向量,并通过非线性 Sigmoid 函数拟

合二维二项分布。为增强全局空间信息的表达，利用二维全局平均池化对 1×1 和 3×3 分支的输出进行编码，并采用 Softmax 函数进行高斯映射处理，进一步拟合线性变换。最后，生成的两个空间注意力权重通过 Sigmoid 函数结合，聚合为每组的输出特征图。

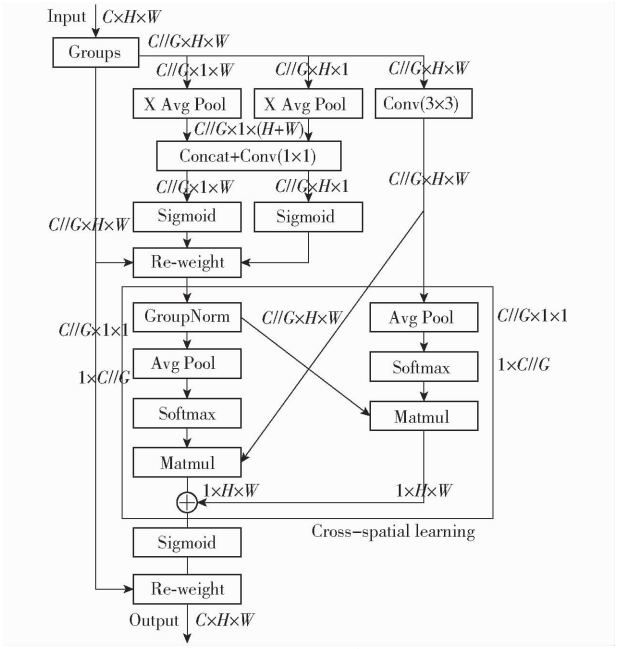

图 5　EMA 模块

Fig. 5　EMA module

2.2.4　引入激活函数

为优化模型在 FPGA 上的部署，将激活函数由 SiLU 替换为计算更简单的 ReLU。SiLU 涉及指数与乘法运算，计算开销较大；而 ReLU 仅需比较操作，在训练中收敛更快，且无需通过归一化来防止饱和[24-25]。该替换显著降低了硬件资源消耗[26]，同时在病害检测中保留了较高的精度，有利于在资源有限场景下实现高效推理。

2.3　模型压缩与 FPGA 推理

为了减小模型的复杂性并适应 FPGA 硬件的限制，需要对模型进行后端压缩[27]。通过参数量化将 32 位浮点数据替换为 8 位定点数据类型。在减少逻辑单元和存储资源占用的同时，提高计算效率并降低功耗[28]。数据转换公式为

$$x_{\text{int}} = \text{round}\left(\frac{x_{float}}{S}\right) \tag{4}$$

式中　x_{int}——转换后的 8 位定点数的值

x_{float}——原始的 32 位浮点数的值

S——缩放因子

round(·)——四舍五入函数，用于将结果取整到最接近的整数值

该公式将浮点数缩放并量化为定点数，使其能够在更小的位宽内表示，同时尽量保留原始数据的精度。

系统实现的完整流程涵盖开发端与硬件端的核心模块。在开发端，首先利用 Vitis－AI 工具对训练好的浮点神经网络模型进行定点量化和剪枝优化，得到 8 位定点数模型，从而降低模型复杂度并提升硬件运行效率。量化完成后，将模型对应的计算图编译成硬件执行所需的一系列二进制指令，获得 Xmodel 文件。随后，通过调用 Xilinx 的 VART 库及相关模型库将模型部署到硬件端的 DPU 上运行。在硬件部署阶段，设备通过网线与主机连接，加载生成计算图以执行推理任务。推理过程中，首先对输入病害图像进行预处理，处理后的数据被输入到 DPU 计算图中，完成加速计算。最后，对推理结果进行后处理，生成病害检测的识别框坐标，并绘制到原图像上，完成水稻病害检测任务。

3　结果与分析

3.1　试验环境和参数配置

本试验模型基于 Linux 平台的深度学习框架搭建，操作系统为 Ubuntu 23.04，硬件配置包括 Intel(R)Xeon(R)Gold 6248R CPU@3.00 GHz 处理器和 NVIDIA RTX A4000 显卡，使用的软件为 CUDA 11.8 和 cuDNN 8.9.07。FPGA 选用的是 Xilinx 公司的 AXU2CGB 开发板，虚拟机采用 Ubuntu 18.04，试验程序主要在赛灵思 Vitis 2022.2 平台上使用 Python 编写。模型训练参数如表 3 所示。

表 3　训练参数设置

Tab. 3　Training parameters setting

参数	数值	参数	数值
图像大小/(像素×像素)	640×640	初始学习率	0.01
类别数	2	动量参数	0.937
批次大小	16	权重衰减参数	0.000 5
训练轮次	150	预热周期	3

3.2　评价指标

为全面评估模型的性能，采用多个关键指标：精确率，用于衡量模型在检测病害区域中的准确性；平均精度均值(mAP)反映模型在多类别病害检测中的综合表现；帧速率(Frames per second, FPS)评估模型对水稻病害图像的实时处理能力。此外，模型的复杂度通过模型内存占用量、参数量、计算量等指标进行评估。

3.3　模型训练

3.3.1　模型改进前后对比

由表 4 可知，模型改进后精确率提升 0.9 个百分点，且整体性能表现优异。其中，FPS 提升 16%，显著提高了病害检测效率；模型参数量减少

54.3%,存储需求大幅降低;计算量减少 62.7%,有效降低了硬件资源依赖;模型内存占用量减少 55.5%,为在 FPGA 上的部署提供了有利条件。改进前后的性能曲线分别如图 6a 和图 6b 所示,改进后的模型在精度上有所提升,且模型在训练过程中具有良好的收敛性和稳定性。

表 4 改进前后对比

Tab.4 Comparison before and after model improvement

模型	精确率/%	平均精度均值/%	参数量	计算量	模型内存占用量/MB	帧速率/(f·s^{-1})
YOLO v5s	81.6	89.2	7.0×10^6	1.58×10^{10}	14.5	270.0
改进 YOLO v5s	82.5	86.8	3.2×10^6	5.90×10^9	6.45	312.5

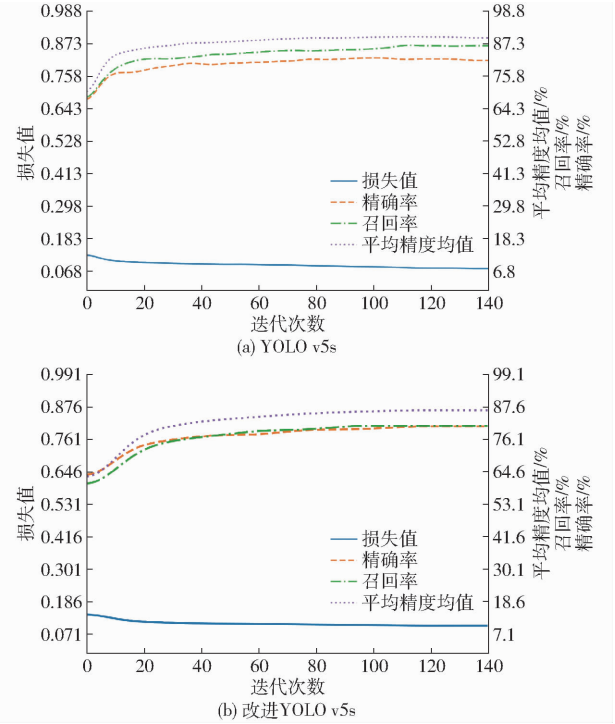

图 6 改进前后模型性能曲线

Fig.6 Model performance curves before and after improvement

3.3.2 不同轻量级网络性能对比

结合表 5 和图 7 可知,将主干网络替换为 ShuffleNetV2 后,模型精度虽略低于 CSPDarkNet53,但在模型内存占用量、参数量和计算量方面均显著降低,分别减少了 53.1%、54.3% 和 63.3%。此外,ShuffleNetV2 的 FPS 为 417 f/s,相比原模型提升了 54.4%,显著增强了病害实时处理能力。FasterNet 虽在精度上略有提高,但资源消耗较高。EfficientRep 在精度上优于其他主干网络,在胡麻叶斑病和稻瘟病的精确率上分别提高了 3.2、2.2 个百分点,但其 FPS 仅为 143 f/s,限制了在资源受限场景中的应用。因此,综合考虑精度、资源消耗和实时性,ShuffleNetV2 在 FPGA 等资源受限环境中的部署优势更加突出。

表 5 不同轻量级网络性能对比

Tab.5 Performance comparison of different lightweight networks

主干网络	精确率/%		平均精度均值/%		帧速率/(f·s^{-1})
	胡麻叶斑病	稻瘟病	胡麻叶斑病	稻瘟病	
CSPDarkNet53	75.4	87.7	88.5	89.8	270
GhostNetV2	77.6	88.7	88.9	89.6	227
FasterNet	76.4	87.7	89.2	89.9	256
EfficientRep	78.6	89.9	88.8	89.5	143
ShuffleNetV2	75.6	86.8	85.1	87.0	417

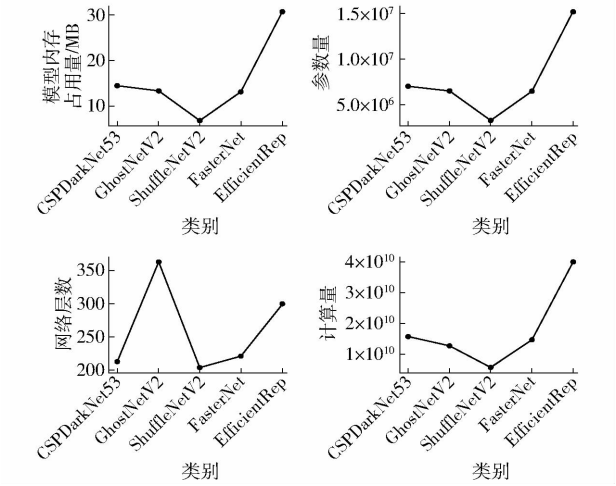

图 7 模型复杂度对比

Fig.7 Comparison of model complexity

3.3.3 ShuffleNetV2 基础上不同损失函数对比

表 6 比较了不同损失函数在胡麻叶斑病和稻瘟病检测中的性能。结果显示,Focal-EIoU Loss 在病害检测中的精确率略有优势,QualityFocalLoss 在稻瘟病检测中的综合性能较好,但精度不及 NWD。替换原损失函数 CIoU 为 NWD 后,对比仅替换主干网络的模型,对稻瘟病和胡麻叶斑病的检测精确率分别提升了 1.7、1.8 个百分点。此外,损失函数的替换并未增加模型的权重参数量,也无需额外的计算与存储开销。

表 6 不同损失函数对比

Tab.6 Comparison of different loss functions

损失函数	精确率/%		平均精度均值/%	
	胡麻叶斑病	稻瘟病	胡麻叶斑病	稻瘟病
CIoU	75.6	86.8	85.1	87.0
WIoU	73.8	85.7	84.7	86.9
Focal-EIoU	79.4	73.3	85.0	86.7
Focal-SIoU	74.4	84.6	85.0	87.1
QualityFocalLoss	74.0	86.4	85.0	87.8
NWD	77.4	88.5	85.0	87.5

3.3.4　添加 EMA 前后及替换激活函数性能对比

如图 8a 所示，在训练初期，加入和未加入 EMA 结构的模型，其召回率曲线基本重合，且均呈快速上升趋势。训练超过 100 轮后，添加 EMA 结构的模型表现出更高的平稳性，召回率波动减少，且整体水平略高于未添加 EMA 的模型。这表明 EMA 结构在模型收敛阶段能有效抑制召回率波动，提升模型的稳定性，从而提高水稻病害检测的准确性。

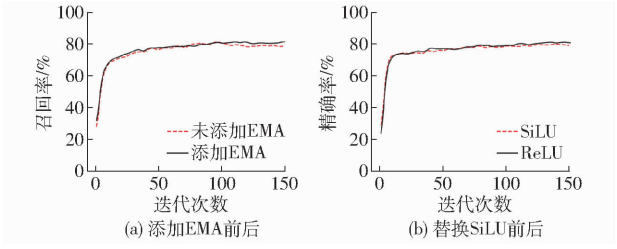

图 8　改进前后性能对比

Fig. 8　Performance comparison before and after improvement

图 8b 中，在训练初期，SiLU 因其非线性特性表现出更高精度。随着训练的深入，两者的精度逐渐趋于一致，并在约 25 轮后，ReLU 逐渐超越 SiLU，表现出更高的精度。这表明，SiLU 在初期有一定优势，而 ReLU 的简单非线性特性在后期表现更优，且具有更好的泛化能力。

3.3.5　不同检测模型对比试验

为验证改进 YOLO v5s 在水稻病害检测中的综合性能，将其与 YOLO v5s、YOLO v5 - lite、YOLO 11n 和 Faster R - CNN 进行对比分析。由表 7 可知，改进模型的精确率较上述模型分别提升 0.9、10.3、3.5、11.5 个百分点。模型内存占用量较原始 YOLO v5s 减少 55.5%，FPS 提升 16%，仅次于 YOLO v5 - lite。YOLO v5s 虽在平均精度均值上表现最佳，但模型较大；YOLO v5 - lite 轻量且高速，但精度偏低。综合评估表明，改进模型在精度、速度与轻量化间实现了更优平衡，可更高效完成水稻病害检测。

表 7　不同检测模型对比试验结果

Tab. 7　Comparison tests results of different detection models

模型	精确率/%	平均精度均值/%	模型内存占用量/MB	帧速率/(f·s^{-1})
YOLO v5s	81.6	89.2	14.50	270.00
YOLO v5 - lite	72.2	69.3	1.60	385.00
YOLO v11n	79.0	87.0	5.50	20.88
Faster R - CNN	71.0	54.0	113.50	28.50
改进 YOLO v5s	82.5	86.8	6.45	312.50

3.4　模型推理

3.4.1　FPGA 资源利用率

本设计在 FPGA 部署中的资源利用率体现了硬件加速与资源分配的优化，如图 9 所示。BRAM 的使用率为 53%，适用于中等规模数据的快速存储和读取；LUTRAM 满足小容量数据存储需求，展现了高效的存储划分策略。DSP 和 URAM 的利用率分别达 82% 和 96%，充分支持计算密集型任务和大容量数据存储，硬件性能利用最大化。同时，LUT 和 LUTRAM 的较低利用率分别为 45% 和 13%，预留了硬件扩展空间，有助于后续模型优化。此外，时钟管理模块 BUFG 和 MMCM 的资源利用率仅为 1% 和 13%，体现了时钟配置的高效与简洁。

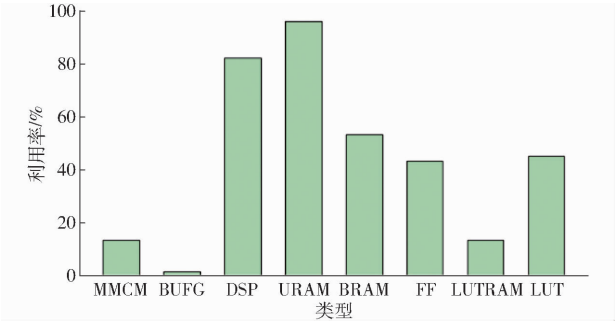

图 9　FPGA 资源利用率

Fig. 9　FPGA resource utilization

3.4.2　不同硬件平台推理性能对比

通过表 8 数据可知模型在不同硬件设备上的性能表现。FPGA 的平均精度均值达到 85.6%，与 CPU 下的 86.8% 和 CPU + GPU 的 86.8% 基本持平，移植精度达 98.6%，能够满足精度需求。其推理时间为 0.048 5 s，接近高性能 CPU。功耗方面，FPGA 的平均功耗仅为 12.80 W，比 CPU 低 21%，比 CPU + GPU 组合低 64%，充分体现了其低功耗优势。因此，FPGA 在保证精度和实时性的同时，功耗显著下降，是低功耗场景下的理想选择。

表 8　不同硬件平台上推理性能对比

Tab. 8　Performance comparison of different hardware platform models

硬件平台	平均精度均值/%	推理时间/s	平均功耗/W
CPU	86.8	0.045 7	16.20
CPU + GPU	86.8	0.003 2	35.54
FPGA	85.6	0.048 5	12.80

从图 10 中可以看出，GPU 和 FPGA 的推理结果均无明显漏检或误检。胡麻叶斑病和稻瘟病的病斑均被准确检测，检测框位置一致，且类别正确。相比之下，FPGA 推理的置信度略低于 GPU，但整体检测效果保持一致，为农业生产中的病害检测提供了一

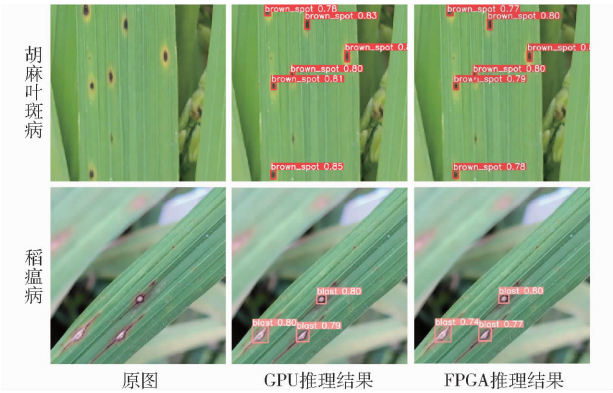

图 10 不同硬件平台水稻病害识别结果

Fig. 10 Rice disease identification results on different hardware platforms

种高效、实用的边缘计算方案。

4 结论

(1)将 YOLO v5s 主干网络替换为轻量级 ShuffleNetV2;其次,引入对小目标病斑检测有效的 NWD 损失函数;此外,在颈部网络中添加 EMA 结构以增强特征提取能力;最后将激活函数从 SiLU 替换为简单高效的 ReLU。改进后模型的参数量减少 54.3%,计算量减少 62.7%,FPS 提升 16%,精度提高 0.9 个百分点,平均精度均值为 86.8%。改进后的模型分别与 YOLO v5s、YOLO v5 - lite、YOLO 11n、Faster R - CNN 进行对比试验,精确率分别提升 0.9、10.3、3.5、11.5 个百分点,具有良好的检测性能。

(2)通过 8 位定点量化操作,模型内存占用量进一步缩减至 4.2 MB,相较于原模型减少 71.03%。在 FPGA 平台上部署后,推理单幅图像的时间为 0.048 5 s;平均功耗为 12.80 W,相较于 CPU 低 21%,比 CPU + GPU 组合降低 64%,充分体现了 FPGA 低功耗高效率的优势。

参 考 文 献

[1] 袁培森,欧阳柳江,翟肇裕,等. 基于 MobileNetV3Small - ECA 的水稻病害轻量级识别研究[J]. 农业机械学报,2024,55(1):253 - 262.
Yuan Peisen, Ouyang Liujiang, Zhai Zhaoyu, et al. Lightweight identification of rice diseases based on improved ECA and MobileNetV3Small[J]. Transactions of the Chinese Society for Agricultural Machinery, 2024, 55(1): 253 - 262. (in Chinese)

[2] Suvanthini T, Nishadi F, Chandrika P, et al. Morphological and molecular analysis of fungal species associated with blast and brownSpot diseases of oryzae sativa[J]. Plant Disease, 2022, 106(6): 9.

[3] International Rice Research Institute (IRRI). Rice blast[EB/OL]. [2025 - 12 - 10]. https://www.irri.org/rice - blast.

[4] 杨宁,程巍,张钊源,等. 基于 FPGA 加速的 Mask R - CNN 稻瘟病高通量自适应识别模型研究[J]. 农业机械学报,2024,55(7):298 - 304,314.
Yang Ning, Cheng Wei, Zhang Zhaoyuan, et al. Research on high-througput adaptive recognition Mask R - CNN model for rice blast disease based on FPGA acceleration[J]. Transactions of the Chinese Society for Agricultural Machinery, 2024, 55(7): 298 - 304, 314. (inChinese)

[5] Yang N, Yu J, Wang A, et al. A rapid rice blast detection and identification method based on crop disease spores' diffraction fingerprint texture[J]. Journal of the Science of Food and Agriculture, 2020, 100(9): 3608 - 3621.

[6] Feng L, Wu B, Zhu S, et al. Investigation on data fusion of multisource spectral data for rice leaf diseases identification using machine learning methods[J]. Frontiers in Plant Science, 2020, 11: 577063.

[7] Dong Liying, Liu Shufang, Li Jing, et al. A rapid and simple method for DNA preparation of magnaporthe oryzae from single rice blast lesions for PCR-Based molecular analysis[J]. The Plant Pathology Journal, 2023, 39(2): 234.

[8] Pinki F T, Khatun N, Islam S M. Content based paddy leaf disease recognition and remedy prediction using support vector machine[C]//2017 20th International Conference on Computer And Information Technology (ICCIT). IEEE, 2017: 1 - 5.

[9] Pantazi X E, Moshou D, Tamouridou A A. Automated leaf disease detection in different crop species through image features analysis and one class classifiers[J]. Comput. Electron. Agric., 2019, 156: 96 - 104.

[10] 文斌,曹仁轩,杨启良,等. 改进 YOLOv3 算法检测三七叶片病害[J]. 农业工程学报,2022, 38(3): 164 - 172.
Wen Bin, Cao Renxuan, Yang Qiliang, et al. Detection of *Panax notoginseng* leaf diseases using improved YOLOv3 algorithm [J]. Transactions of the CSAE, 2022, 38(3): 164 - 172. (in Chinese)

[11] 兰玉彬,孙斌书,张乐春,等. 基于改进 YOLOv5s 的自然场景下生姜叶片病虫害识别[J]. 农业工程学报,2024,40(1): 210 - 216.
Lan Yubin, Sun Binshu, Zhang Lechun, et al. Identification of ginger leaf diseases and pests in natural scenes based on improved YOLOv5s [J]. Transactions of the CSAE, 2024, 40(1): 210 - 216. (in Chinese)

[12] 公徐路,张淑娟. 基于改进 YOLOv5s 的苹果叶片小目标病害轻量化检测方法[J]. 农业工程学报,2023,39(19):175 - 184.
Gong Xulu, Zhang Shujuan. Lightweight detection method for small-target apple leaf diseases based on improved YOLOv5s [J]. Transactions of the CSAE, 2023, 39(19): 175 - 184. (in Chinese)

[13] Boutros A, Yazdanshenas S, Betz V. You cannot improve what you do not measure: FPGA vs. ASIC efficiency gaps for convolutional neural network inference[J]. ACM Transactions on Reconfigurable Technology and Systems, 2018, 11(3): 3242898.

[14] Li Z, Wang J. An improved algorithm for deep learning YOLO network based on Xilinx ZYNQ FPGA[C]//2020 International Conference on Culture-oriented Science & Technology(ICCST), 2020:447 - 451.

[15] Saidani T, Ghodhbani R, Alhomoud A, et al. Hardware acceleration for object detection using YOLOv5 deep learning algorithm on Xilinx Zynq FPGA platform[J]. Eng. Technol. Appl. Sci. Res., 2024, 14(1): 13066 - 13071.

[16] Blast(leaf and collar)IRRI Rice KnowledgeBank[EB/OL]. (2021 - 10 - 02)[2025 - 12 - 10]. http://www.knowledgebank.irri.org/training/fact-sheets/pestmanagement/diseases/item/blast-leaf-collar.

[17] Brown spot-IRRI Rice Knowledge Bank[EB/OL]. (2021 - 10 - 02)[2025 - 12 - 10]. http://www.knowledgebank.irri.org/training/fact-sheets/pestmanagement/diseases/item/brown-spot.

[18] Ultralytics. YOLOv5[EB/OL]. (2020 - 06 - 26)[2025 - 12 - 10]. http://github.com/ultralytics/yolov5.

[19] 周成,陈章彬,杜雅刚,等.面向边缘计算的水稻病害检测方法与装置研究[J].农业机械学报,2025,56(4):353 - 362.
Zhou Cheng, Chen Zhangbin, Du Yagang, et al. Development of rice disease detection methods and devices for edge computing[J]. Transactions of the Chinese Society for Agricultural Machinery, 2025, 56(4): 353 - 362. (in Chinese)

[20] Howard A G, Zhu M, Chen B, et al. MobileNets: efficient convolutional neural networks for mobile vision applications[PP/OL]. V1. arXiv(2017 - 04 - 17)[2025 - 12 - 10]. https://doi.org/10.48550/arXiv:1704.04861.

[21] Ma N, Zhang X, Zheng H T, et al. ShuffleNet V2: practical guidelines for efficient CNN architecture design[C]//Ferrari V, Hebert M, Sminchisescu C, et al. Computer Vision-ECCV 2018. ECCV, 2018.

[22] Wang Jinwang, Xu Chang, Yang Wen, et al. A normalized Gaussian Wasserstein distance for tiny object detection[PP/OL]. V1. arXiv(2021 - 10 - 26)[2025 - 12 - 10]. https://doi.org/10.48550/arXiv:2110.13389.

[23] Ouyang D L, He S, Zhang G Z, et al. Efficient multi-scale attention module with cross-spatial learning[C]//2023 IEEE International Conference on Acoustics, Speech and Signal Processing(ICASSP). Rhodes Island: IEEE, 2023: 1 - 5.

[24] Krizhevsky A, Sutskever I, Hinton G E. ImageNet classification with deep convolutional neural networks[C]//Proc. Adv. Neural Inf. Process. Syst., 2012: 1097 - 1105.

[25] Suda N. Throughput-optimized opencl-based FPGA accelerator for large-scale convolutional neural networks[C]//Proc. ACM/SIGDA Int. Symp. Field-Program. Gate Arrays, 2016: 16 - 25.

[26] Si J, Harris S L, Yfantis E. Neural networks on an FPGA and hardware-friendly activation functions[J]. Journal of Computer and Communications, 2020, 8: 251 - 277.

[27] 张丽丽,陈真,刘雨轩,等.基于 ZYNQ 的 Yolo v3 - SPP 实时目标检测系统[J].光学精密工程,2023,31(4):543 - 551.
Zhang Lili, Chen Zhen, Liu Yuxuan, et al. Yolo v3 - SPP real-time target detection system based on ZYNQ[J]. Optics and Precision Engineering, 2023, 31(4): 543 - 551. (in Chinese)

[28] Khalifa D B, Martel M. Rigorous floating-point to fixed-point quantization of deepNeural networks on STM32 micro-controllers[C]//2024 10th International Conference on Control, Decision and Information Technologies(CoDIT). Vallette: Malta, 2024: 1201 - 1206.