# 集成改进 YOLOv8n 与通道剪枝的轻量化番茄叶片病虫害识别方法

杨 森[1,2]，张鹏超[1,2※]，王 磊[1,2]，唐梁彬[1,2]，王树声[1,2]，贺 兴[1,2]
（1. 陕西理工大学机械工程学院，汉中 723001；2. 陕西省工业自动化重点实验室，汉中 723001）

**摘 要**：针对当前番茄叶片病害检测模型参数量、计算量过大的问题，该研究提出了一种基于 YOLOv8n 的轻量化高精度网络模型。通过 StarBlock 模块对原始的 C2f（CSP bottleneck with 2 convolutions）进行重构，大幅降低参数量的同时增强模型表达能力；其次引入混合局部通道注意力机制（mixed local channel attention，MLCA），以捕捉更多的上下文信息和多尺度特征；同时，通过多级通道压缩方式改进了原有检测头，减少了沿通道维度的特征；最后通过融合通道剪枝算法对模型二次压缩，使其更加轻量化。试验结果表明，经处理的模型参数量、浮点计算量、模型权重大小分别降低了 63.3%、72.8%、61.9%，模型精确率、召回率和平均精度均值（mean average precision（IoU=0.5），$mAP_{0.5}$）分别为 97.5%、96.2% 和 98.5%，性能方面，移动端设备检测帧率达到 358.5 帧/s，番茄叶片病虫害图像单幅推理时间平均为 4.4 ms。证明了该算法可在大幅降低网络计算量的同时保持较高的检测性能，能够满足移动端和嵌入式设备的部署要求。

**关键词**：病虫害检测；YOLOv8n；轻量化模型；通道剪枝
doi：10.11975/j.issn.1002-6819.202409008
**中图分类号**：S24 **文献标志码**：A **文章编号**：1002-6819(2025)-02-0206-09

**杨森，张鹏超，王磊，等. 集成改进 YOLOv8n 与通道剪枝的轻量化番茄叶片病虫害识别方法[J]. 农业工程学报，2025，41(2)：206-214. doi: 10.11975/j.issn.1002-6819.202409008 http://www.tcsae.org**
YANG Sen, ZHANG Pengchao, WANG Lei, et al. Identifying tomato leaf diseases and pests using lightweight improved YOLOv8n and channel pruning[J]. Transactions of the Chinese Society of Agricultural Engineering (Transactions of the CSAE), 2025, 41(2): 206-214. (in Chinese with English abstract) doi: 10.11975/j.issn.1002-6819.202409008 http://www.tcsae.org

## 0 引 言

番茄作为一种重要的经济作物，在全球范围内广泛种植，番茄喜光、喜温并且对栽培的条件要求不严格、环境适应能力强、生长周期长、结果期长、产量高、风味好，市场需求量大，是中国各地设施栽培主要作物之一。然而，番茄生长周期中面临的各种常见病虫害问题，且诸多病害都是从叶部开始发病，进而蔓延到整个植株。这些病害一旦暴发，不仅会导致大量减产，还会增加农药使用量，影响生态环境和食品安全。因此，及时准确地识别出叶部病害类型是病害综合防治的关键[1-2]。

随着深度学习的快速发展和计算能力的急剧提升，目标检测算法已经从基于手动特征的传统算法转向基于深度学习的目标检测算法[3]，其算法普遍使用卷积神经网络（convolutional neural network，CNN）提取特征。如 ZENG 等[4] 使用卷积神经网络模型来提取图像特征并实施训练水果和蔬菜分类，在自己的数据库上进行了分类试验，实现了 95.6% 的准确率；毛锐等[5] 采用卷积核拆解和下采样延迟策略优化深度残差网络（deep residual neural network，ResNet-50），并采用 ROI (region of interest) Align 改进 ROI 以降低特征量化误差，结果表明改进后的 Faster-RCNN 识别方法对小麦条锈病、黄矮病、健康小麦和其他黄化症状小麦识别的平均精度均值可达 98.74%。

但目前多数卷积神经网络架构倾向于通过堆叠更多层次来增强模型的表现力，以此追求更高的预测准确率。然而，这种方法是以消耗显著的计算能力和存储资源为代价的，容易导致训练速度慢、网络参数过多等问题，超出了许多移动和嵌入式应用程序的能力。

作为前沿的单阶段目标检测算法，SSD（single shot multibox detecto）[6] 与 YOLO（you only look once）[7] 系列算法因其实时性和高效性，在自然环境下的水果识别与定位任务中得到了广泛应用。如 CHEN 等[8] 使用 MobileNetv3 替换主干网络、并使用 Soft-NMS（soft non-maximum suppression）对原 SSD 进行改进，在其数据集上 mAP 达到 95.50%，单副图像推理时间为 30 ms；针对细小病斑检测不明显问题，如 KARTHIK 等[9] 引入了一种新颖的双轨网络对葡萄叶片病害进行分类，它采用了 Swin Transformer 和残差变形网络的组合，在其数据集上的准确率达到 98.6%。

为使模型更好地应用于实际场景，近年来相关研究都向轻量级网络的方向发展。如 LI 等[10] 使用 ShuffleNetv2 重建骨干网络减少模型的计算次数和参数，并将网络中的 Concat 操作替换为参数较少的 Add 操作，试验结果表明该模型具有 98.8% 的准确率，且计算量、大小和参数分别减少了 65.18%、56.55% 和 57.59%；XIAO 等[11] 替

收稿日期：2024-09-02 修订日期：2024-12-28
基金项目：国家自然科学基金项目（62176146）
作者简介：杨森，研究方向为计算机视觉。Email：janssen0908@163.com
※通信作者：张鹏超，教授，博士生导师，研究方向为智能控制理论与应用、计算机视觉。Email：snutzpc@126.com

换骨干网络，并使用通道剪枝技术修剪冗余特征层，此外引入注意力机制增强模型的特征提取能力，使得模型大小从 36.5 M 大幅减少到 7.8 M，同时保持了 88.6% 的准确率；谭厚森等[12] 使用简化的残差与卷积模块优化部分 C2f 进行特征融合，对空间金字塔池化（spatial pyramid pooling fast，SPPF）进行优化；引入了 partial convolution（PConv）卷积，在自建的香梨数据集上平均精度均值比原模型分别提升 0.4 和 0.5 个百分点，达到 94.7% 和 88.3%，检测速度分别提升了 34% 和 24.4%。LI 等[13] 引入轻量级特征提取模块，并采用解耦头结构对 YOLOv7-tiny 模型进行了改进，提高了其检测精度和效率。此外，应用修剪方法，对模型进行了二次压缩，结果表明优化后的模型在计算参数数量、复杂度、模型大小方面分别是原始 YOLOv7-tiny 网络的 37.8%、34.1% 和 40.7%。

上述研究促进了水果识别与检测技术的革新，各研究在提升检测精确度、优化处理速度以及实现模型轻量化等方面贡献了独特的视角与解决方案。但很少有研究针对番茄叶片病害分类识别问题提出轻量化模型，因此本文对番茄叶片最常见的 6 种病害（早疫病、晚疫病、叶霉病、花叶病、斑枯病、黄化曲叶病）及 2 种虫害（潜叶虫、叶螨）与健康叶片共计 9 种叶片类型开展研究。为了保证检测精度的同时大幅压缩模型大小，提出了一种基于改进 YOLOv8n 的模型轻量化方法，以期所提出算法在能够保持较高模型预测精度的同时大幅降低模型的计算复杂度，为后续部署在移动端设备中提供理论基础。

## 1　材料与方法

### 1.1　材　料

#### 1.1.1　数据来源

本研究的数据集含有 9 个叶片类别，分别为健康叶片、早疫病、晚疫病、叶霉病、花叶病、斑枯病、黄化曲叶病、潜叶虫、叶螨（如图 1）。为了全面覆盖番茄叶片病害的各种表现形式，本研究数据集来源于两类，数据集 1 是由 AI_Challenger_2018 公共数据源[14] 随机挑选而出的 300 张病虫害图像，由于公共数据集中的病害图像主要聚焦于单一叶片的病状，这虽然有利于精确识别单一病叶，但在复杂场景中，如多叶片混叠等自然情况下面临识别准确率下降的问题。鉴于此，于 2024 年的 5－6 月，在陕西省汉中市汉台区梧凤设施蔬菜基地（33°10'N，106°97'W）进行了现场采集，使用 iphone13 pro 作为采集设备，分别在不同位置随机拍摄共采集 600 张不同病况的番茄叶片图像，格式为 JPG，除掉相似、模糊、重复等低质量图像后，剩余 526 张图像，像素大小为 1 920×1 080，获取了温室中多种患病与未患病番茄叶片的图像，对其病态进行分类后，形成了补充数据集 2。为了减少网络训练过程中的资源消耗，所有数据集中的图像均经过压缩处理，统一调整至 640×640 像素的分辨率。


图 1　采集到的番茄叶片病虫害与健康叶片图像

Fig.1　Collected images of tomato leaf diseases, pests, and healthy leaves

#### 1.1.2　数据集构建

由于获取的数据集大多数都没有标注，因此本文使用 LabelImg 软件在 RGB（Red，Green，Blue）图像上对目标所在像素区域进行人工手动标注。病害部分采用最大外接矩形框，且尽可能保证矩形框贴近目标像素边缘，标注格式采用 YOLO格式。此外，为了增强训练模型的适应性和鲁棒性，并防止其在特定数据集上表现过拟合，对原始图像集合随机噪声、局部裁剪、随机旋转，最终将其扩充至 4 130 张图像。为了保持试验数据的随机性，本试验将数据集划分为训练集、验证集和测试集，比例为 7.5:1.5:1。训练集包含 3 095 张图片，测试集和验证集的图片总数1 035 张。

#### 1.1.3　试验平台

本次试验训练环境采用 Ubuntu 18.04.6 LTS 操作系统、62.5GB 运存、Intel®Xeon(R) Silver 4214R CPU 080Ti/PCIe/SSE2 图形处理器，采用 Pytorch 1.12 深度学习框架基于 Python 3.8 编写 Python 语言程序。

在模型训练过程中，为减小模型陷入局部最优的可能性，使用随机梯度下降 (stochastic gradient descent，SGD) 优化器，将网络训练的初始学习率设置为 0.01，动量因子设为 0.937，权重衰减设置为 0.000 5，批次大小为 16，共训练 300 轮。在模型训练中加入 Mosaic 数据增强，通过对输入图像进行各种图像增强后将 4 张图片随机拼接，提高模型泛化能力。因为经过 Mosaic 数据增强，数据集已经足够丰富，因此选择在后 10 轮训练中关闭 Mosaic 数据增强来提高模型性能。

#### 1.1.4　试验评价指标

为综合表示轻量化处理对模型的影响，选取了 4 个

模型识别性能指标、2 个计算性能指标以及模型权重大小来评估模型。其中模型识别性能指标包括精确率（precision，P）、召回率（recall，R）、IoU 阈值为 0.5 时的平均精度均值，记作 $mAP_{0.5}$、IoU 阈值 0.5～0.95 之间 10 个 mAP 值的平均值，记作 $mAP_{0.5\text{-}0.95}$；计算性能指标包括参数量（parameters）、浮点数计算量（Giga floating-point operations per second，GFLOPs）。计算如式（1）～（3）所示：

$$P = \frac{T_P}{T_P + F_P} \tag{1}$$

$$R = \frac{T_P}{T_P + F_N} \tag{2}$$

$$mAP = \frac{\Sigma_{i=1}^{n} AP_i}{n} \tag{3}$$

式中 $F_P$ 为错误预测的正样本数，$F_N$ 表示错误预测的负样本数，$T_P$ 表示正确预测的正样本数。$AP_i$ 反映目标检测对单个类别的准确性。

### 1.2 基于改进 YOLOv8n 的番茄叶片病虫害检测模型

#### 1.2.1 YOLOv8n 算法模型

YOLOv8（you only look once version 8）是 YOLO 系列目标检测模型的第 8 个主要版本，由 Ultralytics 团队开发[15]。它继承了 YOLO 系列模型的一阶段检测机制，能够在一次前向传播中同时预测目标的位置和类别，这使得 YOLOv8 在实时检测场景中能够保持高效性和准确性。

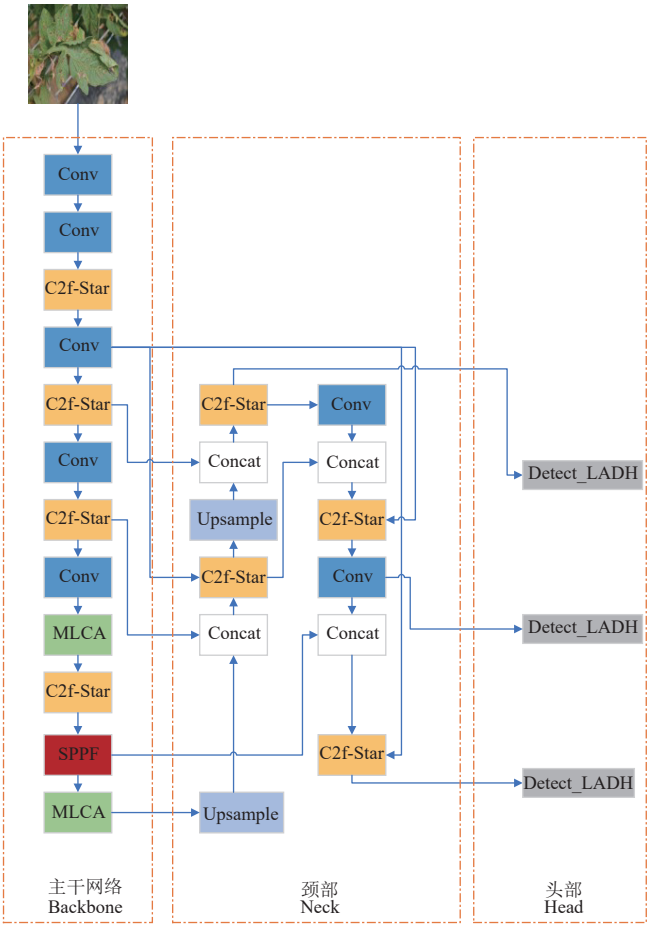
YOLOv8 算法主要包括输入端、骨干网络（backbone）、颈部网络（neck）和预测头部（head）4 个部分。作为单阶段目标检测架构的一员，其设计涵盖了 5 个配置版本，分别是 YOLOv8n、YOLOv8s、YOLOv8m、YOLOv8l 以及 YOLOv8x。这些版本的区分主要在于模型的复杂度，具体体现在网络的深度和宽度上，即残差模块的数量逐级递增，从而赋予了模型更强的特征提取与融合能力。随着版本的升级，检测精度也随之提升，然而，代价是处理时间的延长。其中，YOLOv8n 作为最轻量级的版本，以其卓越的检测速度脱颖而出，但相应的，其在精度上略逊一筹。 而 YOLOv8s 至 YOLOv8x 的系列，虽然在精度上有显著改善，但过多的残差结构引入了额外的计算负担，导致检测周期延长。在现实农业实时监测场景下，这种时间上的延迟可能构成不利因素，影响系统的响应效率和实用性。

鉴于此，本研究决定以 YOLOv8n 为基础进行针对性优化，提出一种同时兼顾速度与精度的检测算法。

#### 1.2.2 改进 YOLOv8n 算法模型

本文基于 YOLOv8n 目标检测算法提出的轻量化番茄叶片病害检测模型 SLMP-YOLOv8n 结构如图 2 所示。

1）轻量化特征增强模块 C2f-Star：原始的 YOLOv8 骨干网络使用了 4 个传统 C2f [16] 模块以保证图像特征提取的优质性，但鉴于一般卷积网络在获得番茄叶片全面信息时易产生大量相似特征图，进而导致模型参数量大、计算时间长等问题，因此，本文使用深度卷积（depthwise convolution，DWConv）[17] 替换了传统卷积网络，与传统卷积网络相比，每个输入通道都有自己的卷积核，卷积核仅在该通道上滑动，不混合通道信息。如图 3 所示。将输入特征图通过多个不同大小的卷积核进行处理，以捕捉到不同尺度的特征，这相当于在每个通道上独立进行卷积操作，因此计算量大大减少。




注：Conv 为架构中最基本的卷积块，用来提取输入特征图的局部特征；C2f-Star 为轻量化特征增强模块；MLCA 为混合局部通道注意力机制；SPPF 为快速空间金字塔池化操作；Upsample 为上采样操作，将特征图的空间尺寸放大；Concat 为特征图链接操作，沿通道维度拼接特征图，Detect-LADH 为轻量级非对称检测头。

Note: Conv is the most basic convolution block in the architecture, which is used to extract the local features of the input feature map. C2f-Star is a lightweight feature enhancement module; MLCA is a mixed local channel attention mechanism; SPPF is a fast spatial pyramid pooling operation; Upsample is an upsampling operation that enlarges the spatial size of the feature map. Concat is the feature map link operation, which stitches the feature map along the channel dimension, and Detect-LADH is a lightweight asymmetric detection head.

图 2 SLMP-YOLOv8n 网络结构图

Fig.2 SLMP-YOLOv8n network structure diagram

此外，本文采用 StarNet[18] 中的 StarBlock 进行网络重构，其整体结构如图 4 所示。经由 DWconv 初步处理的尺度信息在每条路径上再次使用相同大小的 Conv 进一步提取和强化特征，然后将不同路径的特征图进行星形操作拼接（该模块融合星型运算替换原有的乘积运算，使其能够以一种新的方式将输入特征隐式转换为极高的非线性维度），以整合多尺度信息，最终输出经过多路径处理和融合的特征图到后续的层进一步处理。通过特殊的连接模块和高效的特征提取机制，在保证计算效率的同时，增强了模型的表达能力，使模型更适合在移动

端等算力有限的设备上部署。

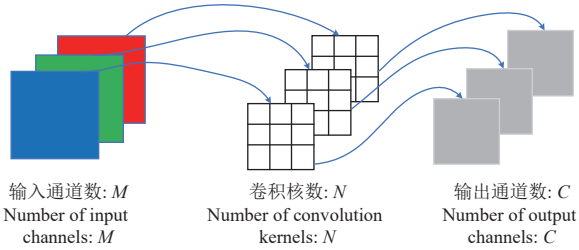
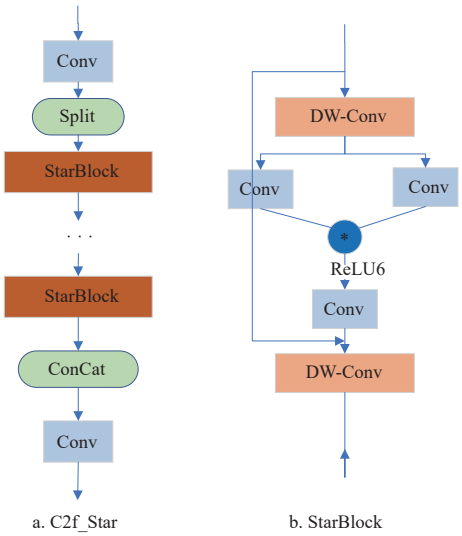



图 3　深度卷积实现过程

Fig.3　Depthwise convolution implementation procedure




注：Split 为分割特征图操作；StarBlock 为创新的星型模块；Dwconv 为替换后的深度卷积块，ReLU6 为设置了输出上界的激活函数，“*”号代表星型操作。

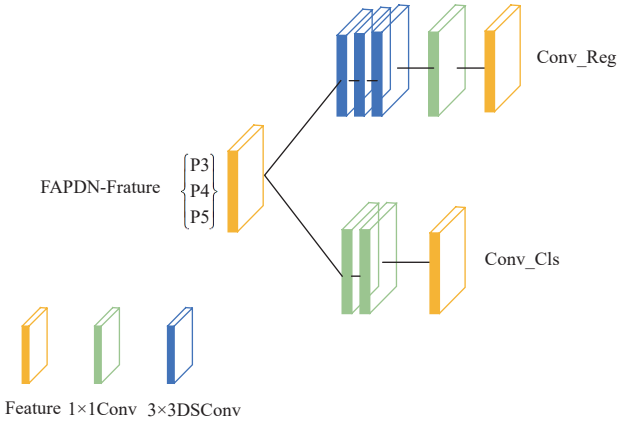
Note: Split is the operation of segmenting the feature map; StarBlock is an innovative star module; Dwconv is the replaced deep convolution block, ReLU6 is the activation function with the output upper bound, and the "*" sign represents the star operation.

图 4　C2f_Star 与 StarBlock 模块结构图

Fig.4　C2f_Star and StarBlock module structure diagram

2）轻量级非对称检测头：对于本文数据集而言，病虫害种类的数量会直接导致不同类别之间误检概率的变化，最初的 YOLO 算法采用耦合头[19] 进行检测任务，其大多是利用网络顶部的相同卷积层进行分类和回归。但是，这些任务具有不同的重点，可能会导致检测过程中发生冲突。在后续改进工作中，引入解耦头方法虽显著提高了模型的检测能力，但也大大增加了网络的参数量，导致推理速度降低。因此，为解决这些问题，本文借鉴 ZHANG 等[19] 改进非对称解耦头（asymmetric decoupling head, ADH）的思想，使用多级通道压缩来改进 YOLOv8 网络中的检测头，将改进模块命名为 Detect-LADH，其结构如图 5 所示。

通过任务类型隔离网络，利用 2 个不同的渠道来执行相关任务，为了扩展感受野并增加 Reg 分支的任务参数，采用了 3 个深度可分离卷积（depthwise separable convolution，DSConv）[18] 来减少沿通道维度的特征。取代了每个分支中传统的 3 × 3 卷积。与标准卷积相比，DSConv 的优势在于它通过将卷积运算分解为深度卷积和逐点卷积（pointwise convolution，PWConv）[18] 来显著减少参数数量，PWConv 实现流程如图 6 所示，这里的卷积运算会将上一步输出的通道 channels C 在深度方向上进行加权组合，生成新的 Feature channels C，其生成数量等于卷积核数，该模块能够有效解决原耦合头引入的分类和回归任务之间的冲突，降低了不同病害之间误检的概率。




注：FAPDN-Frature 为多尺度特征图，其中，P3 对应的检测图大小为 80×80，P4 对应的检测特征图大小为 40×40，P5 对应的检测特征图大小为 20×20；Conv_Reg 为回归通道，Conv_Cls 为分类通道。

Note: FAPDN-Frature is a multi-scale feature map, in which the size of the detection map corresponding to P3 is 80×80, the size of the detection feature map corresponding to P4 is 40×40, and the size of the detection feature map corresponding to P5 is 20×20; Conv_Reg is the regression channel, and Conv_Cls is the classification channel.

图 5　Detect-LADH 模块结构图

Fig.5　Detect-LADH module structure chart

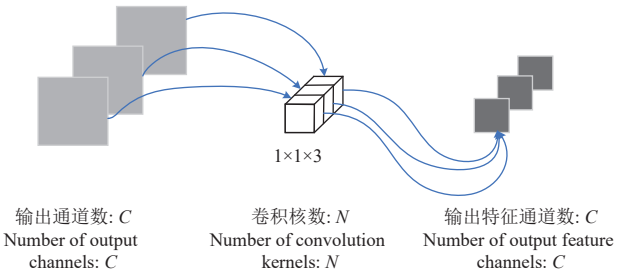
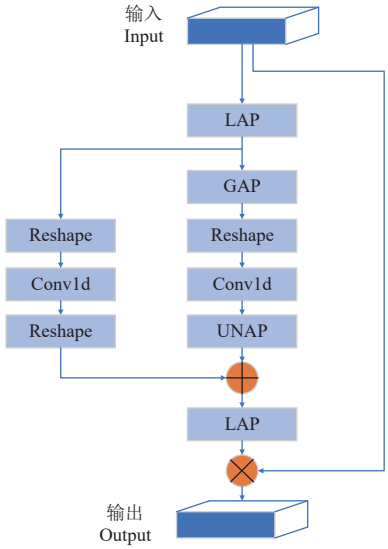



图 6　逐点卷积实现过程

Fig.6　Pointwise convolution implementation procedure

3）混合局部通道注意力机制：由于真实果园中对于番茄叶片的识别往往受到场景的影响，为了模型理论上能更好专注于病害识别，提高检测精度，遂引入注意力机制。注意力机制最近几年在深度学习的各个领域被广泛使用[20]，其核心思想是在计算能力有限的前提下，合理分配计算资源，使得神经网络能够利用有限的注意力资源从海量的输入信息中快速筛选出高价值的信息。目前构建轻量级神经网络通常采用通道注意力 (channel attention, CA) 模块与空间注意力（spatial attention, SA）模块[21-22]，其通常在单一尺度上进行注意力计算，可能会忽略不同层次的特征导致重要特征信息的丢失。鉴于此，本文采用混合局部通道注意力（mixed local channel attention，MLCA）[23]，其结构如图 7 所示，该注意力能够整合通道信息、空间信息、局部通道信息和全局通道信息，在多个层次上对输入特征进行交叉注意力处理，以捕捉更多的上下文信息和多尺度特征。这种全面的方

法使得模型理论上一定程度能够适应不同样本，不同场景，增强了本文模型的泛化能力与检测效率，而不会过度增加计算复杂性。




注：LAP 为局部平均池化，GAP 为全局平均池化，Reshape 为重新排列特征操作，UNAP 为反池化操作。
Note: LAP refers to local average pooling, GAP refers to global average pooling operation, Reshape refers to feature rearrangement operation, and UNAP refers to anti polling operation.

图 7 MLCA 模块结构图

Fig.7 Structure diagram of mixed local channel attention module

### 1.2.3 模型剪枝

1）剪枝原理

上述对番茄叶片病害检测的 YOLOv8n 模型所使用的改进方法在一定程度上保留了检测精度，同时压缩了模型大小以及计算量，但由于模型本身存在大量的卷积结构，使得模型的结构仍然存在冗余[24]，对未来部署到嵌入式设备上来说仍然是个很大的资源负担。为进一步对模型进行轻量化改进，加快推理速度，遂采用结构化剪枝中的 Slim 剪枝方法[25] 对模型继续压缩。

在通道剪枝的方法中，需要先对网络模型中的 BN 层进行稀疏训练来筛选出一些不重要的通道，目前多采用批量归一化（batch normalization，BN）层来加快模型的收敛速度。通道剪枝时，BN 层使用小批量统计对内部激活进行归一化处理，设$z_{in}$和$z_{out}$为 BN 层的输入和输出，$B$ 表示当前的小批量，$\varepsilon$ 为避免除数为 0 而使用的微小正数，并为每个通道的 BN 层引入缩放因子 $\gamma$ 和偏置 $\beta$。如式（4）～（5）所示对通道数据进行归一化处理 BN 层作以下归一化处理：

$$\hat{z}=\frac{z_{in}-\mu_B}{\sqrt{\sigma_B^2+\varepsilon}} \tag{4}$$

$$z_{out}=\gamma\hat{z}+\beta \tag{5}$$

式中$\hat{z}$表示归一化后的通道数据；$\mu_B$和$\sigma_B$分别表示批次 $B$ 中输入激活的均值和标准差。

在 CNN 中，常见的做法是在卷积层后插入 BN（批量归一化）层，并利用通道缩放参数进行调整。基于此，Slim 为每个通道引入了一个缩放因子 $\gamma$，将该因子与通道的输出相乘，之后再将网络权重与这些缩放因子结合，并对 $\gamma$ 进行稀疏正则化处理[24]。损失函数 $L$ 如式（6）所示。

$$L=\sum_{(x,\ y)}l(f(x,\ W),\ y)+\lambda\sum_{\gamma\in\Gamma}g\cdot(\gamma) \tag{6}$$

式中$\sum_{(x,\ y)}l(f(x,\ W),\ y)$为正常训练损失函数，$x$ 和 $y$ 表示训练的输入与输出，$W$ 表示可训练权重，$\lambda\sum_{\gamma\in\Gamma}g\cdot(\gamma)$为正则化项，$g(\gamma)$ 是对缩放因子的惩罚函数，本文选择 L1[26] 范数：$g(\gamma)=|\gamma|$，权重系数 $\lambda$ 是两项的平衡因子[24]，即稀疏率。

2）模型剪枝步骤

①设置不同稀疏正则项 λ 时，模型 BN 层的权重及平均精度均会出现相应的变动，若 λ 值过小，稀疏过程太慢，区分不出通道重要性；若 λ 值过大，往往会使精度下降过快。为在模型进行稀疏训练的同时保持较好的识别性能，确定最佳稀疏率，结合预试验数据，将 λ 分别设置为 0.000 5，0.001，0.005 及 0.01[27] 后对原始模型进行稀疏化训练。不同系数选择下的 BN 层缩放因子分布变化情况经可视化工具输出后如图 8 所示。

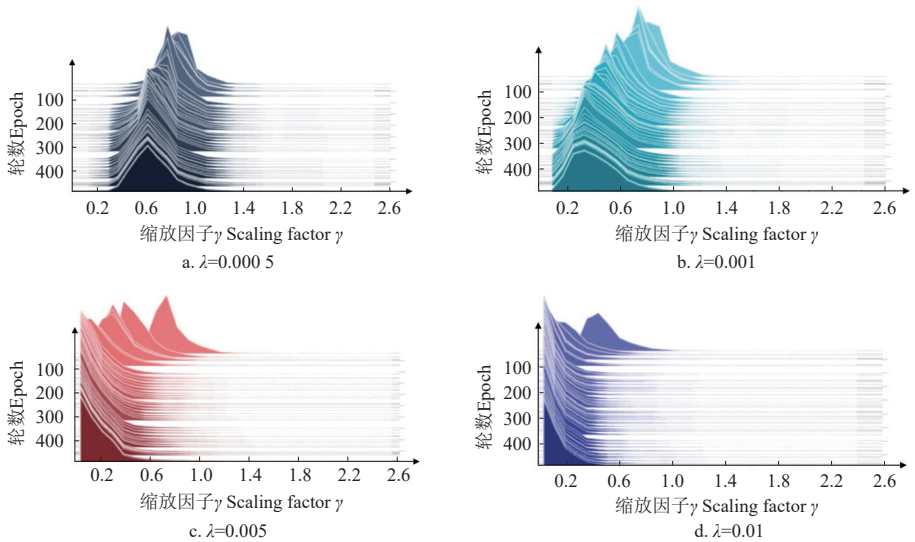



图 8 设置不同正则项系数 λ 下的缩放因子γ的变化

Fig.8 The variation of scaling factor γ under different regularization coefficients λ

可以看出缩放因子系数 γ 随着训练的进行逐渐趋近于 0，并且稀疏率越大，γ 趋近于 0 的速度越快。

②通道剪枝与微调

经过稀疏训练之后，模型变得更加紧凑，此时很多缩放因子已经接近于零，具体的通道剪枝示意图如图 9 所示。随后，针对那些缩放因子接近零的通道，移除其所有的输入和输出连接及其对应的权重。依据不同的比例对模型进行剪枝。由于被删除通道数量的增加（即百分比增大），在剪枝过程中需要选择合适的剪枝率。

设置不同剪枝率进行 10 次试验后，结果见图 10。随着剪枝率的提升，模型参数量逐渐减少，在剪枝率为 20%～60% 间较为明显，当剪枝率低于 50% 时，模型平均精度均值在 97% 左右有较小波动，当大于 50% 时，模型精度大幅减小，已不适合实时检测叶片病害场景。因此，为平衡模型的准确性与内存占用的减少程度，本文选择 50% 作为试验剪枝率。进行试验后，除了部分重要通道无法剪枝外，剩余大部分通道均进行了一定程度上的压缩，该结果表明剪枝算法对本模型有效。

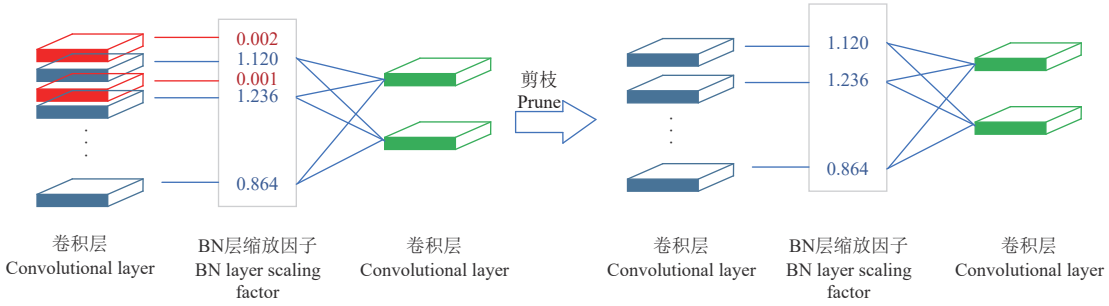



图 9　通道剪枝示意图

Fig.9　Diagram of channel pruning

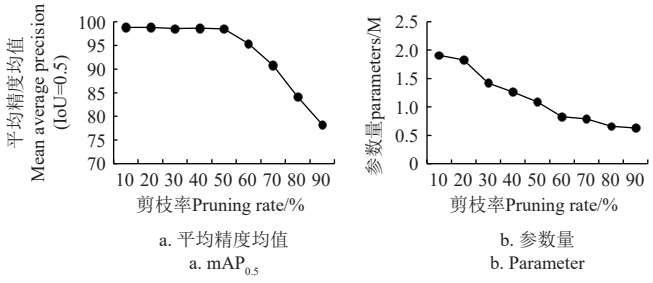



图 10　不同剪枝率下模型参数变化趋势

Fig.10　Model parameters under different pruning rates

## 2　试验结果分析

### 2.1　训练结果

本研究对改进后的 YOLOv8n 模型进行 0.005 稀疏率的稀疏训练，训练轮次为 500 轮，并且对稀疏训练后的模型进行微调，将改进后并且稀疏化训练后的模型命名为 SLMP-YOLOv8n，该模型在自建数据集上参数量为 1.1 M，计算量为 2.2 G，模型大小为 2.4 M，$mAP_{0.5}$ 达到 98. 5%，准确率达到 97.5%，召回率达到 96.2%，移动端设备检测帧率达到 358.5 帧/s，单幅推理时间平均为 4.4 ms。

### 2.2　不同改进消融试验对比

为验证模型在番茄叶片病虫害目标检测任务中的轻量化效果，研究设计的消融试验及结果如表 1 所示。各模型均在自建数据集上进行了试验，并与原始 YOLOv8n 网络模型进行对比。模型②、③、④依次使用了 StarBlock 模块、Detect-LADH 改进传统检测头以及引入 MLCA 注意力机制来对主干网络进行重构及改进，可以看出与原始算法相比，C2f-Star 与 Detect-LADH 方法通过优化模型结构和计算方式，直接减少了参数量和计算量。其中最为明显的是引入了轻量化检测头 Detect-LADH，平均精度均值（IoU=0.5）只降低约 0.1 个百分点的同时网络模型参数量、浮点计算量分别降低了 36.67 、40.74 个百分点，大幅减少模型参数量的同时基本保证了检测精度值。

表 1　消融试验结果

Table 1　Results of ablation experiment

| 模型 Model | C2f-Star | Detect-LADH | MLCA | 参数量 Params/M | 浮点计算量 Floating point operations/G | 模型权重大小 Model weight size/MB | 准确率 Precision/% | 召回率 Recall/% | 平均精度均值 $mAP_{0.5}$/% |
|---|---|---|---|---|---|---|---|---|---|
| ① | × | × | × | 3.0 | 8.1 | 6.3 | 97.8 | 96.3 | 98.8 |
| ② | √ | × | × | 2.5 | 6.9 | 5.4 | 97.7 | 96 | 98.8 |
| ③ | × | √ | × | 1.9 | 4.8 | 4.9 | 97.5 | 96.6 | 98.7 |
| ④ | × | × | √ | 3.0 | 8.1 | 6.3 | 97.2 | 96.3 | 98.7 |
| ⑤ | √ | √ | × | 2.1 | 4.8 | 5.7 | 98.4 | 96 | 98.9 |
| ⑥ | √ | × | √ | 2.7 | 7.3 | 5.8 | 97.3 | 96.3 | 98.8 |
| ⑦ | × | √ | √ | 2.5 | 6.9 | 5.3 | 96.2 | 96.1 | 98.7 |
| ⑧ | √ | √ | √ | 1.9 | 4.5 | 4.2 | 97.9 | 95.8 | 98.8 |
| ⑨ | √ | √ | √ | 1.1 | 2.2 | 2.4 | 97.5 | 96.2 | 98.5 |

注：①为 YOLOv8n 模型；②为 YOLOv8n+C2f-Star 模型；③为 YOLOv8n+Detect-LADH 模型；④为 YOLOv8n+MLCA 模型；⑤YOLOv8n+C2f-Star+Detect-LADH 模型；⑥为 YOLOv8n+C2f-Star+MLCA 模型；⑦为 YOLOv8n+Detect-LADH+MLCA 模型；⑧为 YOLOv8n+C2f-Star+Detect-LADH+MLCA；⑨为 YOLOv8+C2f-Star+Detect-LADH+MLCA 模型进行剪枝操作后的模型。

Note：①YOLOv8n model; ②YOLOv8n+C2f-Star model; ③YOLOv8n+Detect-LADH model; ④YOLOv8n+MLCA model; ⑤ YOLOv8n+C2f-Star+Detect-LADH model; ⑥YOLOv8n+C2f-Star+MLCA model; ⑦YOLOv8n+Detect-LADH+MLCA model; ⑧ YOLOv8n+C2f-Star+Detect-LADH+MLCA model; ⑨ is the model of YOLOv8+C2f-Star+Detect LADH+MLCA after pruning operation.

对于模型④，MLCA 的主要作用是提升特征的表征能力，而不是减少参数量和计算量，因此单独引入 MLCA 时，参数量和计算量没有明显变化。模型⑤～⑦对 C2f-Star、Detect-LADH 与 MLCA 方法两两分组进行试验，结果表明其结合方式均不能获得较好的效果，在此基础上，将三者混合后得到模型⑧，此时参数量、计算量、模型权重大小降低至 1.9 M、4.5 G、4.2 MB，准确率、召回率、mAP0.5 分别为 97.9%、95.8%、98.8%。C2f-Star 和 Detect-LADH 为压缩模型，MLCA 弥补了性能损失，从而实现了参数量、计算量和性能的平衡优化。这种协同作用是改进 YOLOv8n 模型时取得显著效果的关键原因。

最后，对模型⑧使用剪枝算法进行二次压缩后得到模型⑨，改进后的网络模型在基本保持精度的同时大幅减少了模型复杂度，参数量、浮点计算量、模型权重大小相较 YOLOv8n 降低了 63.3%、72.8%、61.9%，同时，本模型准确率、召回率、$mAP_{0.5}$ 分别为 97.5%、96.2%、98.5%。因此，采用 Slim 剪枝算法能在轻量化的同时兼顾模型的检测性能和推理速度。

通过上述消融试验的全面评估，验证了在原模型中改进的不同网络模块均表现出优异的性能。

### 2.3 对比试验

为验证本文算法的优势，将本文提出的算法与 Faster R-CNN[28]、SSD[6]、YOLOX-Nano[29]、YOLOv5-Nano[7] 以及 YOLOV8n[15]5 种高性能模型在自建番茄叶片病虫数据集上进行训练与测试，结果见表 2，可以看出，与 Faster R-CNN 和 SSD 相比，本文提出的算法在参数量、浮点计算量方面表现远低于二者，其参数量分别比 Faster R-CNN 和 SSD 减少了 207.3 与 34.9 M，浮点计算量分别减少 497.2、280.6 G；与 YOLOX-Nano 与 YOLOv5-Nano 相比，本文算法在精度基本持平的前提下，参数量分别降低 20% 和 59.3%，浮点计算量分别降低 17% 和 49.5%。

表 2 不同模型对比试验结果

Table 2 Comparison test results of different models

| 模型 Model | 参数量 Params/M | 浮点计算量 FLOPs/G | 准确率 Precision/% | 召回率 Recall/% | 平均精度均值 $mAP_{0.5}$/% |
|---|---|---|---|---|---|
| Faster R-CNN | 208.4 | 499.4 | 97.8 | 96.5 | 96.0 |
| SSD | 36.0 | 282.8 | 94.9 | 98.3 | 99.3 |
| YOLOX-Nano | 1.4 | 2.7 | 93.5 | 94.2 | 97.6 |
| YOLOv5-Nano | 2.7 | 4.4 | 97.2 | 96.9 | 98.8 |
| YOLOv8n | 3.0 | 8.1 | 97.8 | 96.3 | 98.8 |
| Our | 1.1 | 2.2 | 97.5 | 96.2 | 98.5 |

综上所述，本文提出的 SLMP-YOLOv8n 算法在 6 种模型中表现效果最佳，证明本研究模型在番茄叶片病虫害的检测任务中切实可行，特别是在降低参数量、浮点计算量以及模型权重大小上所具备的优势，使得在嵌入式设备上部署高效的检测模型成为可能。

## 3 结 论

本研究旨在解决在检测番茄叶片病虫害研究中平衡网络复杂度与准确度的问题，经过以上数据集准备、算法搭建与试验，得出以下结论：

1）本文提出了一种基于 YOLOv8n 的轻量化番茄叶片病害检测算法，通过融合星型运算模块 StarBlock 对原始 C2f 模块进行重构，引入 Detect-LADH 对检测头进行改进，实现了初步网络的轻量化设计，为了使模型对数据鲁棒性更好，添加了 MLCA 注意力机制，在降低复杂度的同时，不会使模型精度大幅降低。最后，对模型应用剪枝算法，进一步压缩模型。

2）在自建数据集上验证了改进后的算法可以在大幅降低网络复杂度与参数量的同时保持高检测精度，本文算法在自建番茄叶片病虫害数据集上平均精度均值 $mAP_{0.5}$ 达到 98.5%，准确率达到 97.5%，召回率达到 96.2%，参数量为 1.1 M，浮点计算量为 2.2 G，模型权重大小仅为 2.4 MB，移动端设备检测帧率达到 358.5 帧/s，单幅推理时间平均为 4.4 ms。基本达到原始 YOLOv8 网络模型的检测结果，为后续番茄叶片病害识别检测系统的移动端部署提供了理论依据。

后续会收集遮挡或不同光照下的番茄叶片病虫害图像丰富数据集，并进一步分析输入数据的特征和观察网络结构。此外，会对比使用多种剪枝算法对模型进行二次压缩，使得在实现模型的进一步轻量化同时提升目标检测的准确性。

[参 考 文 献]

[1] 刘玉霞. 番茄在中国的传播及其影响研究[D]. 南京：南京农业大学，2007：1-2.
LIU Yuxia. Research on the Spread of Tomato and its Influence in China[D]. Nanjing: Nanjing Agricultural University, 2007: 1-2. (in Chinese with English abstract)

[2] 李英梅，杨艺炜，刘晨，等. 陕西番茄黄化曲叶病毒病绿色防控新思路[J]. 陕西农业科学，2021，67(11)：110-114.
LI Yingmei, YANG Yiwei, LIU Chen, et al. New ideas on integrated control of tomato yellow leaf curl virus disease in Shaanxi[J]. Shaanxi Journal of Agricultural Sciences, 2021, 67(11): 110-114. (in Chinese with English abstract)

[3] 蒋雪松，计恺豪，姜洪喆，等. 深度学习在林果品质无损检测中的研究进展[J]. 农业工程学报，2024，40(17)：1-16.
JIANG Xuesong, JI Kaihao, JIANG Hongzhe, et al. Search progress of non-destructive detection of forest fruit quality using deep learning[J]. Transactions of the Chinese Society of Agricultural Engineering (Transactions of the CSAE), 2024, 40(17): 1-16. (in Chinese with English abstract)

[4] ZENG G. Fruit and vegetables classification system using image saliency and convolutional neural network[C]//2017 IEEE 3rd Information technology and mechatronics engineering conference (ITOEC). China: IEEE, 2017: 613-617.

[5] 毛锐，张宇晨，王泽玺，等. 利用改进 Faster-RCNN 识别小麦条锈病和黄矮病[J]. 农业工程学报，2022，38(17)：176-185.
MAO Rui, ZHANG Yuchen, WANG Zexi, et al. Recognizing stripe rust and yellow dwarf of wheat using improved Faster-

RCNN[J]. Transactions of the Chinese Society of Agricultural Engineering (Transactions of the CSAE), 2022, 38(17): 176-185. (in Chinese with English abstract)

[6] LIU W. SSD: Single shot multibox detector[C] //Computer Vision & Pattern Recognition. USA: IEEE, 2016: 21-37.

[7] REDMON J, DIVVALA S, GIRSHICK R, et al. You only look once: Unified, real-time object detection[EB/OL]. (2015-06-02) [2023-12-01]. https://doi.org/10.48550/arXiv.1506.02640.

[8] CHEN Y，JIAO M, PENG X, et al. Study on positioning and detection of crayfish body parts based on machine vision[J]. Journal of Food Measurement and Characterization, 2024，18(7): 1-13.

[9] KARTHIK R, VARDHAN G V, KHAITAN S, et al. A dual-track feature fusion model utilizing Group Shuffle Residual DeformNet and swin transformer for the classification of grape leaf diseases[J]. Scientific Reports, 2024, 14(1): 14510.

[10] LI Z, KANG L, RAO H, et al. *Camellia oleifera* fruit detection algorithm in natural environment based on lightweight convolutional neural network[J]. Applied Sciences, 2023, 13(18): 10394.

[11] XIAO J, KANG G, WANG L, et al. Real-time lightweight detection of lychee diseases with enhanced YOLOV7 and edge computing[J]. Agronomy 2023, 13, 2866.

[12] 谭厚森，马文宏，田原，等. 基于改进 YOLOv8n 的香梨目标检测方法[J]. 农业工程学报，2024，40(11)：178-185.
TAN Housen, MA Wenhong, TIAN Yuan, et al. Improved YOLOv8n object detection of fragrant pears[J]. Transactions of the Chinese Society of Agricultural Engineering (Transactions of the CSAE), 2024, 40(11): 178-185. (in Chinese with English abstract)

[13] LI J, LIU Y, LI C, et al. Pineapple detection with yolov7-tiny network model improved via pruning and a lightweight backbone sub-network[J]. Remote Sensing, 2024, 16(15): 2805.

[14] ZHAO C. plants_disease_detection[EB/OL]. (2018-10-30)[2020-04-14]．https://github.com/spytensor/plants_disease_detection.

[15] VARGHESE R, SAMBATH M. YOLOv8: A novel object detection algorithm with enhanced performance and robustness[C]//2024 International Conference on Advances in Data Engineering and Intelligent Computing Systems (ADICS). Chennai: IEEE, 2024: 1-6.

[16] SUN Y, CHEN G, ZHOU T, et al. Context-aware cross-level fusion network for camouflaged object detection[J]. Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence (IJCIA), 2021, 31(4): 1025-1031.

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

[18] MA X, DAI X, BAI Y, et al. Rewrite the stars[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Columbus, USA: IEEE, 2024: 5694-5703.

[19] ZHANG J, CHEN Z, YAN G, et al. Faster and lightweight: an Improved YOLOv5 object detector for remote sensing images[J]. Remote Sensing, 2023, 15(20): 4974.

[20] ZHANG M. Neural attention: Enhancing QKV calculation in self-attention mechanism with neural networks[J]. Computation and Language (cs. CL), 2023, 33(10): 2310.

[21] MENG H G, TIAN X X, LIU J J, et al. Attention mechanisms in computer vision: A survey[J]. Computational Visual Media, 2022, 8(3): 331-368.

[22] ZHU X, CHENG D, ZHANG Z, et al. An empirical study ofspatial attention mechanisms in deep networks[C]// Proceedingsof the IEEE/CVF International Conference on Computer Vision, Korea (South), 2019: 6688-6697.

[23] WAN D, LU R, SHEN S, et al. Mixed local channel attentionfor object detection[J]. Engineering Applications of ArtificialIntelligence, 2023, 123: 106442.

[24] VADERA S, AMEEN S. Methods for pruning deep neural networks[J]. IEEE Access, 2022, 10: 63280-63300.

[25] LIU Z, LI J, SHEN Z, et al. Learning efficient convolutional networks through network slimming[C]//Proceedings of the IEEE International Conference on Computer Vision, Venice, 2017: 2736-2744.

[26] DANESHVAR A, MOUSA G. Regression shrinkage and selection via least quantile shrinkage and selection operator[J]. PLoS One, 2023, 18(2): e0266267.

[27] 梁晓婷，庞琦，杨一，等. 基于 YOLOv4 模型剪枝的番茄缺陷在线检测[J]. 农业工程学报，2022，38(6)：283-292.
LIANG Xiaoting, PANG Qi, YANG Yi, et al. Online detection of tomato defects based on YOLOv4 model pruning[J]. Transactions of the Chinese Society of Agricultural Engineering (Transactions of the CSAE), 2022, 38(6): 283-292. (in Chinese with English abstract)

[28] REN S, HE K, GIRSHICK R, et al. Faster R-CNN: Towards real-time object detection with region proposal networks[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2016, 39(6): 1137-1149.

[29] GE Z. Yolox: Exceeding yolo series in 2021[J]. Computer Research Repository(CoRR), 2021, 31(7): 2107.

# Identifying tomato leaf diseases and pests using lightweight improved YOLOv8n and channel pruning

YANG Sen[1,2] , ZHANG Pengchao[1,2※] , WANG Lei[1,2] , TANG Liangbin[1,2] , WANG Shusheng[1,2] , HE Xing[1,2]

(1. *School of Mechanical Engineering*, *Shaanxi University of Technology*, *Hanzhong* 723001, *China*; 2. *Shaanxi Key Laboratory of Industrial Automation*, *Hanzhong* 723001, *China*)

**Abstract:** Timely and accurate identification of leaf diseases can greatly contribute to the effective pest prevention and control in the tomato growth cycle. In this study, a lightweight detection approach was proposed to balance between detection accuracy and computational efficiency. An enhanced version of the YOLOv8n (you only look once) model was augmented with a pruning algorithm, specifically designed to identify the various types of tomato leaf diseases. The YOLOv8n architecture was used to replace the conventional C2f (concatenated feature fusion) module with the more efficient StarBlock module. The number of parameters were was significantly reduced within the network. The complex features were represented to improve the overall expressive power of the improved model. Additionally, a mixed local channel attention mechanism (MLCA) was integrated to capture the richer set of contextual information and multi-scale features, which were critical to distinguishing among different types of leaf diseases. Furthermore, multi-level channel compression was used to reengineer the original detection head. The performance of the improved model was refined to reduce the dimensionality of the feature maps along the channel axis. Thus, the a more streamlined and computationally efficient structure was obtained after these architectural adjustments. Sparse training was then performed on the high-precision and lightweight model. Some weights were selectively eliminated within the network, according to their importance. A specified level of sparsity was achieved after training. Experimental results indicate that there was the best trade-off between data sparsity and model performance at a sparsity rate of 0.005. The redundant or less significant channels were removed using channel pruning. The final high-precision and lightweight models were obtained after training. A series of tests were carried out to validate the improved model. A dataset was comprised of 4,130 images. Nine distinct types of tomato leaf diseases were compiled. The developed model was then tested against this dataset. Compared with the baseline YOLOv8n, the improved model exhibited a substantial reduction in the parameter count (63.3%), floating point operations (72.8%), and model weight size (61.9%). The better performance of the improved model was achieved in the accuracy (97.5%), recall (96.2%), and average precision (mAP@0.5: 98.5%), with the a minimal average drop in the performance metrics of just 0.23 percentage points. Furthermore, the improved model was deployed on mobile devices, indicating the a remarkable detection frame rate of 358.5 frames per second, with an average inference time of 4.4 milliseconds per image. Once benchmarked against the popular object detection frameworks, such as Faster R-CNN and SSD, the improved model demonstrated a dramatic decrease in both the number of parameters (by 207.3 and 34.9 M, respectively) and computational complexity (by 497.2 and 280.6 G, respectively). The YOLOv8n was reduced by 20.0% and 59.3% in the parameters, corresponding to a 17% and 49.54% decrease in the computational complexity, compared with the more recent and lightweight models, like YOLOX Nano and YOLOv5Nano. The competitive accuracy levels were all preserved. In conclusion, the improved YOLOv8-SLMP model can offer a highly viable solution to the real-time detection of tomato leaf diseases. Particularly, the footprint was also minimized, in terms of the parameters, computations, and model size. An ideal candidate was deployed on the resource-constrained embedded systems, thus enabling more widespread and efficient monitoring of crop health.

**Keywords:** disease and pest detection; YOLOv8n; lightweight model; channel pruning