植物病害识别微调模型低秩适配方法

黄锦清[1,2], 叶 进[1,2*], 胡慧琳[1,2], 阳继辉[3], 兰 伟[1,2], 张艳青[1,2]

(1. 广西大学 计算机与电子信息学院,广西南宁 530000,中国; 2. 亚热带农业保护与利用国家重点实验室,广西南宁 530000,中国; 3. 捷佳润科技集团股份有限公司,广西南宁 530000,中国)

摘 要: **[目的/意义]** 深度学习应用于植物病害识别任务时,模型微调面临计算资源受限、参数更新开销较大的挑战。传统低秩适配(Low-Rank Adaptation, LoRA)方法虽能有效减少参数开销,但仍可能导致资源浪费或优化受限。**[方法]** 提出一种动态秩分配算法,旨在提升植物病害分类模型精度的同时,平衡计算资源。通过引入一种基于中心化核对齐的跨层特征相似性度量方法,量化不同层之间的表征相关性;同时基于梯度信息和激活强度构建1个校正因子,用于衡量各层对损失函数的直接影响;考虑训练过程中特征表示变化的影响,提出基于稳定性触发的自适应秩分配表更新策略(Rank Re-Allocation, RRA),在训练过程中监测微调模型参数的收敛状态,在模型趋于稳定时自动更新秩分配表,确保关键层分配更多资源,从而实现参数资源在不同层之间的优化分配。**[结果和讨论]** 在小麦病害数据集、植物病害数据集等公开数据集上使用AlexNet、MobileNetV2、RegNetY和ConvNeXt这4种不同的模型进行测试,该方法与全参数微调相比,在保证平均准确率的基础上,资源消耗分别减少至0.42%,2.46%,3.56%和1.25%。与LoRA方法(R=16)进行对比,AlexNet、MobileNetV2和RegNetY在小麦病害数据集上的准确率分别提高0.38、0.40和0.05个百分点,资源消耗分别减少59.3%、87.4%和50.5%。**[结论]** 低秩自适应微调方法在保证植物病害识别精度的同时,能够显著减少模型训练参数。

关键词: 低秩自适应微调;特征相似性;秩分配算法;秩分配更新策略

中图分类号: S-3 **文献标识码:** A **文章编号:** SA202504003

引用格式:黄锦清, 叶进, 胡慧琳, 阳继辉, 兰伟, 张艳青. 植物病害识别微调模型低秩适配方法[J]. 智慧农业(中英文), 2026, 8(1): 40-51. DOI: 10.12133/j.smartag.SA202504003

HUANG Jinqing, YE Jin, HU Huilin, YANG Jihui, LAN Wei, ZHANG Yanqing. Low-rank Adaptation Method for Fine-tuning Plant Disease Recognition Models[J]. Smart Agriculture, 2026, 8(1): 40-51. DOI: 10.12133/j. smartag. SA202504003 (in Chinese with English abstract)

0 引 言

植物病害是影响农业生产的重要因素,每年造成大量农作物减产,严重威胁粮食安全。传统植物病害识别方法依赖人工观察和专家经验,方法效率低、成本高,且易受人为主观因素干扰,难以满足现代农业高效精准诊断的需求[1]。随着计算机视觉技术的发展,以深度学习为代表的方法被广泛应用于植物病害识别任务[2],在精确性和效率上均表现出巨大潜力。

预训练视觉模型(Pretrained Vision Models, PVMs)在多种机器视觉任务中展现出了卓越的性能,例如图像分类、目标检测和语义分割[3-5]。这些模型在大规模图像数据集上进行预训练,能够捕获丰富的视觉特征,提升目标任务的泛化能力。DONG等[6]的工作提供了针对植物病害识别任务优化的预训练模型集合。诸如AlexNet、MobileNetV2和ShuffleNet等模型具有强大的迁移能力,为在资源受限的环境中开发高效、精准的植物病害诊断系统奠定了坚实基础。然而,这些视觉模型通常具有庞大的参数量与较高的计算复杂度,如经典模型ResNet、ConvNeXt[7]和MobileNetV3[8],ResNet50模型参数超过2 500万,ConvNeXt-B参数

收稿日期:2025-04-02

基金项目:国家自然科学基金(32402495);2024年自治区级大学生创新训练计划(202410593001S)

作者简介:黄锦清,研究方向为大模型微调。E-mail:1766042592@qq.com

*通信作者:叶 进,博士,教授,研究方向为智慧农业与智能计算。E-mail:yejin@gxu.edu.cn

copyright©2026 by the authors (登录www.smartag.net.cn免费获取电子版全文)

更是达到8 800万。

在植物病害检测领域，部署大模型存在较大的挑战。农业应用资源有限且对实时性要求高，亟需开发兼顾准确性与轻量化的诊断模型，提升农业生产效率。因此，研究如何在维持模型识别精度的同时降低训练参数已成为该领域的重要问题[9-11]。近年来，学者提出了多种轻量化模型和改进方法。康继昌和赵连军[12]改进YOLOv11n算法，引入Sim CSP Spatial Pyramid Pooling-Fast（SimCSPSPPF）模块、感知增强卷积（Perception enhancement convolution, PEC）和Adequate Wise IoU（AWIoU）损失函数，有效提升了复杂自然环境下多尺度小麦害虫的检测精度。李想等[13]改进DenseNet的Dense模块，设计S-Dense模块降低计算复杂度，并引入辅助模型增强对不可辨别病害的识别能力，在保持较低推理时延的同时有效提升了苹果叶部多种病害的识别精度。叶进等[14]提出一种基于聚合损失和边界框排斥的密集荔枝花检测方法以提升遮挡场景下的检测精度。邱文杰等[15]提出基于知识蒸馏的轻量化病害识别模型，显著压缩了模型体积。张俊等[16]提出的改进DeepLab v3+模型，结合MobileNet v2轻量化骨干网络和多尺度特征融合模块（Multi-Scale Feature Fusion module, MSFF），提升梯田遥感提取的精度与效率。尽管上述方法在模型轻量化方面取得了进展，但仍然存在参数分配不合理、计算资源消耗大的问题。

近年来，大型预训练语言模型（Pretrained Language Models, PLMs）的微调技术取得了显著进展，为视觉模型参数优化提供了新思路。其中，低秩适配（Low-Rank Adaptation, LoRA）方法冻结预训练模型权重[17]，大幅度减少模型训练参数，其核心思想可以推广到视觉预训练模型的微调上，用于解决植物病害检测任务中的模型适配问题。LoRA方法最初仅对Transformer模型使用相同的秩进行微调，容易造成参数冗余或优化不足，为了进一步提升微调精度和减少训练参数，VALIPOUR等[18]提出DyLoRA方法，在训练时对不同秩的LoRA模块学习到的表示进行排序，一次性训练出支持在推理时动态选择秩的适配器。ZHANG等[19]提出的AdaLoRA方法采用奇异值分解（Singular Value Decomposition, SVD）的方式，依据梯度敏感性来分配秩，实现训练精度和训练参数之间的平衡。但是AdaLoRA方法未考虑梯度信息与具体任务的相关性，限制了其在特定任务中的微调效果。本研究旨在结合植物病害检测这一特定任务，深入研究模型层间特征相似性与参数敏感性，以期提高动态秩分配方法的效果。

中心化核对齐（Centred Kernel Alignment, CKA）基于希尔伯特-施密特独立准则（Hilbert Schmidt Independence Criterion, HSIC），核心思想是利用核矩阵的特性度量两个特征矩阵之间的相关性。KORNBLITH等在研究中证明，CKA能够稳定地衡量神经网络层之间的特征相似性[20,21]，并且与模型的性能表现密切相关。当CKA（X, Y）值较高时，表明两个特征空间之间共享的信息量较多，前一层的特征信息能够被后一层有效继承。因此，本研究利用CKA作为低秩微调中资源分配的一个重要依据，提出一种植物病害识别微调模型低秩适配方法，主要贡献包括：为降低微调模型的训练参数开销，提出一种动态秩分配算法（Dynamic Rank Allocation, DRA），引入CKA衡量视觉模型中各层之间的相似性，同时结合梯度校正计算每层的重要性因子，通过基于重要性因子的动态分配，实现保证性能前提下的低秩分配。为保持秩分配表不偏离最优配置，设计一种自适应秩分配表更新策略（Rank Re-Allocation, RRA），根据训练过程中LoRA层参数的平均相对变化量来判断模型是否趋于收敛。当微调模型进入稳定状态时，自动触发秩分配表的更新，并在多种主流视觉模型如AlexNet、MobileNetV2、RegNetY和ConvNeXt上进行实验验证。

1 材料与方法

本研究采用公开数据集小麦病害数据集（Wheat Plant Diseases Dataset）和植物病害数据集（Plants Disease Dataset）。小麦病害数据集涵盖15种小麦病害如黑锈病、赤霉病及健康植株的图像共13 104张；植物病害数据集涵盖26种植物病害，如茶藻斑病、玉米锈病及番茄细菌性斑疹病的图像共37 505张，数据来源于Kaggle、Mendeley Data等平台，部分植物病害数据图像见图1。这些数据集具有广泛的病害种类和丰富的样本多样性，包括不同光照、不同背景、不同角度，以及植物生长各个阶段的病害图像。实验中用于训练、验证和测试的数据分别占70%、10%和20%。

基于上述数据样本，本研究提出的植物病害模型微调方法见图2。DRA方法包括选择预训练模型、提取模型信息、生成微调模型这3个阶段：



图1 部分植物病害数据集

Fig. 1 Sample images from the plant disease dataset

(1) 对输入的植物病害训练图像进行标准化预处理(统一224×224分辨率，ImageNet归一化)，根据任务选取预训练模型并加载预训练模型参数。(2) 从两个角度提取模型信息。第一，注册前向钩子(Forward Hook)捕获各层激活输出，使用CKA量化层间特征相关性；第二，利用反向传播钩子(Backward Hook)获取梯度张量，结合梯度范数与特征输出计算各层对损失函数的直接影响。(3) 在微调阶段，采用加权调和方法评估网络层的重要性，将重要性分数线性映射至预设秩范围，生成包含秩配置(r)、缩放因子(α)及Dropout的LoRA参数配置表。各适配层分别初始化可训练的低秩矩阵$\boldsymbol{A}$(随机高斯分布)和$\boldsymbol{B}$(零矩阵)，在微调过程中利用反向传播迭代更新上述矩阵，结合预训练模型参数，生成微调模型。

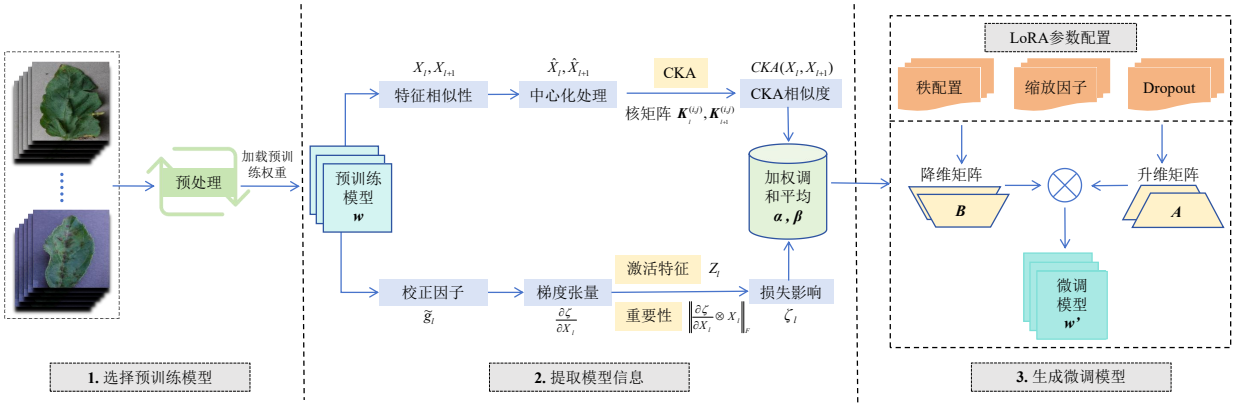


图2 动态秩分配算法整体框架图

Fig. 2 The overall framework of dynamic rank allocation (DRA)

在每个训练轮次监测LoRA参数的平均相对变化量，当该值低于预设的动态阈值时，表明模型进入稳定期，触发上述(2)和(3)阶段再次执行，基于当前模型状态生成新的秩分配表。本节分别对层间特征表示相似性、CKA校正因子、秩分配算法和秩分配更新策略这4个关键部分进行描述。

1.1 层间特征表示相似性

为衡量网络层间的特征表示相似性，本研究采用了基于HSIC的CKA。

1) 给定第l层和第$l+1$层的激活矩阵$\boldsymbol{X}_l$和$\boldsymbol{X}_{l+1}$，首先对它们进行中心化，如公式(1)和公式(2)所示。

$$\tilde{\boldsymbol{X}}_l = \boldsymbol{X}_l - \frac{1}{n}\mathbf{1}\mathbf{1}^T\boldsymbol{X}_l \tag{1}$$

$$\tilde{\boldsymbol{X}}_{l+1} = \boldsymbol{X}_{l+1} - \frac{1}{n}\mathbf{1}\mathbf{1}^T\boldsymbol{X}_{l+1} \tag{2}$$

式中：$\boldsymbol{X}_l$和$\boldsymbol{X}_{l+1}$表示第l层和第$l+1$层的激活矩阵；n表示输入样本的数量；$\mathbf{1}\mathbf{1}^T$表示$n \times n$的全1矩阵；$\tilde{\boldsymbol{X}}_l$和$\tilde{\boldsymbol{X}}_{l+1}$表示第$l$层和第$l+1$层中心化后的激活矩阵。

2）采用径向基函数（Radial Basis Function, RBF）核计算核矩阵。核矩阵的计算如公式（3）和公式（4）所示。

$$\boldsymbol{K}_l^{(i,j)} = e^{-\frac{1}{d_l}\|\tilde{\boldsymbol{x}}_l^{(i)} - \tilde{\boldsymbol{x}}_l^{(j)}\|^2} \tag{3}$$

$$\boldsymbol{K}_{l+1}^{(i,j)} = e^{-\frac{1}{d_{l+1}}\|\tilde{\boldsymbol{x}}_{l+1}^{(i)} - \tilde{\boldsymbol{x}}_{l+1}^{(j)}\|^2} \tag{4}$$

式中：$\boldsymbol{K}_l \in \mathrm{R}^{n \times n}$是第$l$层的核矩阵，其元素$\boldsymbol{K}_l^{(i,j)}$表示第$i$个样本和第$j$个样本在第$l$层的中心化特征空间中的相似性；$d_l$表示第$l$层的特征维度；$\tilde{\boldsymbol{x}}_l^{(i)}$表示第$l$层中第$i$个样本的中心化特征向量；$\tilde{\boldsymbol{x}}_l^{(j)}$表示第$l$层中第$j$个样本的中心化特征向量。

3）对核矩阵进行中心化处理以消除数据尺度的影响，如公式（5）所示。

$$\tilde{\boldsymbol{K}} = \boldsymbol{K} - \frac{1}{n}\mathbf{1}\boldsymbol{K} - \frac{1}{n}\boldsymbol{K}\mathbf{1} + \frac{1}{n^2}\mathbf{1}\boldsymbol{K}\mathbf{1} \tag{5}$$

式中：$\tilde{\boldsymbol{K}}$表示中心化后的核矩阵；$\mathbf{1}$表示全1向量，其维度与核矩阵的行数或列数一致。

4）利用中心化核矩阵的HSIC计算CKA相似度，如公式（6）和公式（7）所示。

$$HSIC(\boldsymbol{X}_l, \boldsymbol{X}_{l+1}) = \sum_{i=1}^{n}\sum_{j=1}^{n}\tilde{\boldsymbol{K}}_l^{(i,j)}\tilde{\boldsymbol{K}}_{l+1}^{(i,j)} \tag{6}$$

$$CKA(\boldsymbol{X}_l, \boldsymbol{X}_{l+1}) = \frac{\sum_{i,j}\tilde{\boldsymbol{K}}_l^{(i,j)}\tilde{\boldsymbol{K}}_{l+1}^{(i,j)}}{\sqrt{\sum_{i,j}(\tilde{\boldsymbol{K}}_l^{(i,j)})^2 \cdot \sum_{i,j}(\tilde{\boldsymbol{K}}_{l+1}^{(i,j)})^2}} \tag{7}$$

CKA相似度的归一化特性使其能够稳健地反映层间特征表示的共享程度，CKA值越高，则当前层的输出特征对后续层具有更强的信息支持。

该方法利用CKA相似度的归一化处理，使得特征相似性的度量具有尺度不变性，无论特征维度还是激活幅值的差异，均能保持计算稳定。核矩阵$\boldsymbol{K}_l$的计算复杂度为$O(n^2 d_l)$，在高维特征场景下更具优势。其不足之处是当输入数据分布存在显著噪声时，CKA的计算准确性会受到影响。此外，CKA作为一种特征相似性度量，并未考虑各层特征在优化过程中对损失函数的直接贡献，容易导致资源分配的偏差。因此，本研究在下一小节利用损失函数对各层输出特征的梯度以及CKA的评估结果进行校正，以实现更有效的资源分配。

1.2 CKA校正因子

本研究认为网络层的重要性取决于其输出特征对损失函数的影响，由此引入1个与目标任务直接相关的校正因子，该因子的计算步骤如下。

1）在前向传播中，网络层l的激活特征$\boldsymbol{Z}_l$由前一层的输出$\boldsymbol{X}_{l-1}$与该层权重$\boldsymbol{W}_l$进行卷积运算加上偏置项得到，如公式（8）所示。

$$(\boldsymbol{Z}_l)_{n,c_{\text{out}},h,w} = (b_l)_{c_{out}} + \sum_{c_{in}=1}^{c_{in}}\sum_{i=1}^{M}\sum_{j=1}^{M}(\boldsymbol{W}_l)_{c_{\text{out}},c_{\text{in}},i,j} \times (\boldsymbol{X}_{l-1})_{n,c_{\text{in}},h\cdot s+i,w\cdot s+j} \tag{8}$$

式中：$(\boldsymbol{Z}_l)_{n,\,c_{\text{out}},\,\text{h},\,w}$为激活特征$\boldsymbol{Z}_l$在第$n$个样本，$c_{\text{out}}$个输出通道，高$h$，宽$w$处的值；$(b_l)_{c_{\text{out}}}$为第$l$层第$c_{\text{out}}$个输出通道的偏置项；$M$为卷积核的尺寸；$s$为卷积操作的步长；$i$，$j$表示卷积核的空间索引；$(\boldsymbol{W}_l)_{c_{\text{out}},\,c_{\text{in}},\,i,\,j}$表示第$l$层卷积核中，第$c_{\text{out}}$个输出通道，第$c_{\text{in}}$个输入通道，第$i$行第$j$列处的权重；$(\boldsymbol{X}_{l-1})_{n,\,c_{\text{in}},\,h\cdot s+i,\,w\cdot s+j}$表示第$l-1$层特征向量在第$n$个样本，第$c_{\text{in}}$个输入通道，高$h\cdot s+i$，宽$w\cdot s+j$处的值。

2）激活特征$\boldsymbol{Z}_l$经过非线性激活函数σ得到该层的输出特征$\boldsymbol{X}_l$，如公式（9）所示。

$$\boldsymbol{X}_l = \sigma(\boldsymbol{Z}_l) \tag{9}$$

3）利用反向传播计算输出特征$\boldsymbol{X}_l$对损失函数L的梯度$\frac{\partial \mathrm{L}}{\partial \boldsymbol{X}_l}$，将梯度值与该网络层的输出特征进行Hadamard乘积运算，结果取其Frobeniu范数得到该网络层的重要性度量g_l，如公式（10）所示。

$$g_l = \left\|\frac{\partial \mathrm{L}}{\partial \boldsymbol{X}_l} \odot \boldsymbol{X}_l\right\|_F \tag{10}$$

式中：g_l表示第l层的重要性度量；$\frac{\partial \mathrm{L}}{\partial \boldsymbol{X}_l}$表示损失函数L对第$l$层特征向量$\boldsymbol{X}_l$的梯度；$\odot$是Hadamard乘积，表示两个同维度张量对应位置元素相乘。

4）对g_l进行最大最小归一化处理，得到稳定的校正因子$\widetilde{g_l}$，如公式（11）所示。

$$\widetilde{g_l} = \frac{g_l - \min_k(g_k)}{\max_k(g_k) - \min_k(g_k) + \epsilon} \tag{11}$$

式中：$\widetilde{g_l}$为第l层用于CKA校正的因子；$\max_k(g_k)$和$\min_k(g_k)$分别表示所有网络层中重要性度量的最大值和最小值；ϵ为一个极小的正数。

该因子融合梯度范数与特征激活强度，能够反

映各层对任务的贡献，为后续修正CKA的结构性评估提供有效依据。

1.3 秩分配算法

基于CKA相似度与梯度校正因子，本研究通过加权调和平均数计算每一网络层的重要性S_l，然后根据其重要性动态分配网络层的秩r_l，如公式（12）和公式（13）所示。

$$S_l = \frac{CKA(\boldsymbol{X}_l, \boldsymbol{X}_{l+1}) \times \widetilde{g_l}}{\beta \times CKA(\boldsymbol{X}_l, \boldsymbol{X}_{l+1}) + \alpha \times \widetilde{g_l}} \tag{12}$$

$$r_l = [S_l \times (r_{\max} - r_{\min}) + r_{\min}] \tag{13}$$

式中：α和β为偏向权重，用于平衡结构与任务的重要性；r_l为第l层的分配秩；S_l为每一网络层的重要性；$r_{\max}$和$r_{\min}$分别表示预设秩范围的最大值和最小值。动态秩分配确保对任务贡献较大的层分配更多资源，从而在有限的参数预算下优化模型性能。

为了调节不同网络层级的更新幅度，引入缩放因子γ平衡LoRA参数α_l，如公式（14）所示。

$$\alpha_l = r_l \times \gamma \tag{14}$$

在LoRA参数分配完成后，仅对参数$\boldsymbol{\theta}_{\text{LoRA}}$进行训练，$\boldsymbol{\theta}_{\text{LoRA}}$表示对预训练模型进行微调的低秩参数矩阵，损失函数为交叉熵，如公式（15）所示。

$$L = -\sum_{i=1}^{C} y_i \log(\widehat{y_i}) \tag{15}$$

式中：C为类别数；y_i为真实标签的独热编码；$\widehat{y_i}$为模型的预测概率。训练过程中，参数更新为公式（16）。

$$\boldsymbol{\theta}_{\text{LoRA}} \leftarrow \boldsymbol{\theta}_{\text{LoRA}} - \mu \frac{\partial \text{L}}{\partial \boldsymbol{\theta}_{\text{LoRA}}} \tag{16}$$

式中：μ为学习率，设置为0.000 1；$\frac{\partial \text{L}}{\partial \boldsymbol{\theta}_{\text{LoRA}}}$为可训练LoRA参数的梯度。本研究的方法将更高的秩分配给重要性较高的层，从而确保参数资源的有效利用。秩分配表的计算不引入额外的训练参数，且该过程只需要消耗常数级的计算开销。

1.4 秩分配更新策略

随着训练的进行，特征表示发生变化，秩分配表可能逐渐偏离最优配置。因此本研究设计一种秩分配更新策略，在训练过程中更新秩分配表。

算法RRA描述了本研究的秩分配更新策略，其核心在于采用LoRA层的平均参数变化量来判断是否需要更新秩分配表。首先计算每个LoRA层l在训练轮次k的相对参数变化量$\delta_l^{(k)}$，随后对所有层的变化量求平均，得到整体变化量$\overline{\delta^{(k)}}$，如公式（17）和公式（18）所示。

$$\delta_l^{(k)} = \frac{\|\theta_{\text{LoRA},\,l}^{(k)} - \theta_{\text{LoRA},\,l}^{(k-1)}\|_F}{\|\theta_{\text{LoRA},\,l}^{(k-1)}\|_F + \epsilon} \tag{17}$$

$$\overline{\delta^{(k)}} = \frac{1}{L_{\text{lora}}} \sum_{l=1}^{L_{\text{lora}}} \delta_l^{(k)} \tag{18}$$

式中：$\delta_l^{(k)}$表示第l层低秩矩阵在训练轮次k的相对参数变化量；$\theta_{\text{LoRA},\,l}^{(k)}$表示第$l$层低秩矩阵在训练轮次$k$时的参数；$\theta_{\text{LoRA},\,l}^{(k-1)}$表示第$l$层低秩矩阵在训练轮次$k-1$时的参数；$\epsilon$是一个极小的正数；$\overline{\delta^{(k)}}$表示所有网络层低秩矩阵在训练轮次$k$时的平均参数变化量；$L_{\text{lora}}$表示网络层数。

设置1个动态触发阈值ε_t，t表示已经进行的秩分配表计算次数。当$\overline{\delta^{(k)}} < \varepsilon_t$时，微调模型趋于稳定，将学习到的低秩矩阵与原始矩阵合并，随后重新进行层重要性评估和计算秩分配表的流程。该阈值在每次触发成功后会自动更新，防止模型收敛后产生不必要的计算开销，如公式（19）～公式（21）所示。

$$\boldsymbol{W}_l^{(t+1)} = \boldsymbol{W}_l^{(t)} + \Delta \boldsymbol{W}_l^{(t)} \tag{19}$$

$$\Delta \boldsymbol{W}_l^{(t)} = \boldsymbol{B}_l^{(t)} \boldsymbol{A}_l^{(t)} \tag{20}$$

$$\varepsilon_{t+1} \leftarrow \overline{\delta^{(k)}} \tag{21}$$

式中：$\boldsymbol{W}_l^{(t+1)}$表示第l层在第$t+1$次完成秩分配表重新计算后合并的参数矩阵；$\boldsymbol{W}_l^{(t)}$表示第l层在第t次完成秩分配表重新计算后的参数矩阵；$\Delta \boldsymbol{W}_l^{(t)}$表示第$l$层在第$t$～$t+1$次完成秩分配表重新计算时的参数矩阵变化量；$\boldsymbol{B}_l^{(t)}$和$\boldsymbol{A}_l^{(t)}$分别表示第$l$层在第$t$～$t+1$次完成秩分配表重新计算时的低秩参数矩阵；$\varepsilon_{t+1}$表示第$t+1$次完成秩分配表重新计算时的动态阈值。

将训练前15%的轮次作为窗口期，在此期间仅记录$\overline{\delta^{(k)}}$而不干预。在窗口期结束时，本研究将此刻的$\overline{\delta^{(k)}}$作为初始阈值ε_0，它能在一定程度上代表微调模型在初始训练后达到的首个稳定水平。

算法RRA以预训练模型M，数据集D作为输入。预定义参数包括秩选择范围$r_{\min}$～$r_{\max}$，训练轮次N，校准轮次E_{calib}，重要性权重S，偏置权重α，β。首先通过ComputeInitialRanks方法（公式（1）～公式（14））计算初始秩分配表，为模型M的各目标层l初始化低秩矩阵$\boldsymbol{Al}$和$\boldsymbol{Bl}$，$\boldsymbol{Al}$服从随机高斯分布，$\boldsymbol{Bl}$初始化为零矩阵；并初始化LoRA参数矩阵的优化器θ_{LoRA}以及动态阈值ε。训练过程中，每个轮次k以最小损失L更新θ_{LoRA}；再通过CalculateAvgParamChange方法（公式（17）和公

式（18））计算每一网络层的平均参数变化量；当 $k == E_{\text{calib}}$ 时，将 ε 设置为该平均参数变化量完成校准；当 $k > E_{\text{calib}}$ 并且平均参数变化量小于 ε 时，触发秩分配表重新计算，合并第 t 次分配的模型权重 $W_l^{(t)}$ 与低秩矩阵变化量 $\boldsymbol{B}_l^{(t)}\boldsymbol{A}_l^{(t)}$，基于合并后的模型 M_{merge} 重新计算秩分配表，重置优化器与低秩矩阵 $\boldsymbol{Al}$ 和 $\boldsymbol{Bl}$，同时更新 ε 为该平均参数变化量，训练结束返回微调后的模型 M_{merge} 用于测试。

算法：RRA

Input：Pre-trained model M，dataset D.
Predefined：max rank $r_{\max}$，min rank $r_{\min}$，total epochs N，calibration epochs E_{calib}，Importance weights S，Bias weights α，β.
Initialization the LoRA parameters（r，alpha，dropout）for each target layer.lora_config ← ComputeInitialRanks(M，D_{batch}）by formula（1）~（14）
Initialize the LoRA low-rank matrices according to lora_config，where Al follows a random Gaussian distribution and Bl is initialized as a zero matrix.
$Al \sim N(0, \sigma^2)$，$Bl \leftarrow 0$
Initialize optimizer for LoRA parameters θ_{LoRA}
$\varepsilon \leftarrow \infty$ # Initialize trigger threshold ε
# Training Loop
for epoch $k = 1$ to N do
Update θ_{LoRA} by minimizing loss L on dataset D.
$\overline{\delta^{(k)}} \leftarrow$ CalculateAvgParamChange($\theta_{\text{LoRA}}^{(k)}$，$\theta_{\text{LoRA}}^{(k-1)}$) by formula（17，18）
If $k == E_{\text{calib}}$ then
$\varepsilon \leftarrow \overline{\delta^{(k)}}$. # Set initial threshold.
else if $k > E_{\text{calib}}$ and $\overline{\delta^{(k)}} < \varepsilon$ then # Trigger rank update
// Merge LoRA weights for each target layer l
for each target layer l in M do
$\Delta W_l^{(t)} \leftarrow B_l^{(t)} A_l^{(t)}$
$W_l^{(t+1)} \leftarrow W_l^{(t)} + \Delta W_l^{(t)}$
end for
lora_config′ ← ComputeInitialRanks(M_{merge}，D_{batch})
Initialize the new matrices Al and Bl according to lora_config′.
$Al \sim N(0, \sigma^2)$，$Bl \leftarrow 0$
Re-initialize optimizer for new LoRA parameters θ_{LoRA}
$\varepsilon \leftarrow \overline{\delta^{(k)}}$ # Update threshold
end if
end for
return M_{merge}

2　结果与讨论

使用PyTorch作为框架来实现所有的算法。实验采用AlexNet、MobileNetV2、RegNetY和ConvNeXt等作为主要模型，结合LoRA动态秩分配策略，针对植物病害分类任务进行微调。为了提高模型的初始表现，本研究加载了PDDD-PreTrain提供的预训练权重，该权重集合基于多种广泛应用的模型，优化了植物病害诊断的特定需求。

为增强模型的训练效果，实验采用标准的图像预处理策略，包括图像尺寸调整（224×224）、均值和标准差归一化。训练过程中，损失函数选择交叉熵损失，并使用Adam优化器，学习率设置为0.000 1，训练轮次为40个。实验均在NVIDIA RTX 3060 GPU上运行，批次大小为32，预设秩范围选择1～16。

2.1　与全参数微调的对比

图3以两种典型模型为例，给出本方法的逐层秩分配结果。MobileNetV2[22]是一种基于反向残差结构的轻量级神经网络模型。应用DRA方法进行微调，观察到在不同的卷积层中，秩分配呈现出一定的规律性。其中，conv.2整体秩较高，表明该层在特征提取过程中具有关键作用，比如负责捕捉高阶的特征信息，其输出对后续决策的影响较大；conv.1和conv.0中后期网络层秩较低，反映这些层对优化的贡献相对较小。

RegNetY模型[23]是一种结构规则的网络架构，以其高效的计算性能广泛应用于视觉任务中。对其进行微调，观察到不同类型的层在模型中的重要性存在明显差异。其中，conv1和se.fc1层普遍获得较高的秩，在中后期网络层中表现突出。相比之下，se.fc2中后期网络层整体秩较低，对模型性能的贡献相对较小；而conv2前期网络层整体秩较低，表明该层在模型的中后期对复杂特征捕获的需求更高。

如图3所示，对比两个模型在植物病害检测任务中的秩分配规律可见，两种架构在识别病害特征时体现出不同的关注模式，可以得到如下结论：

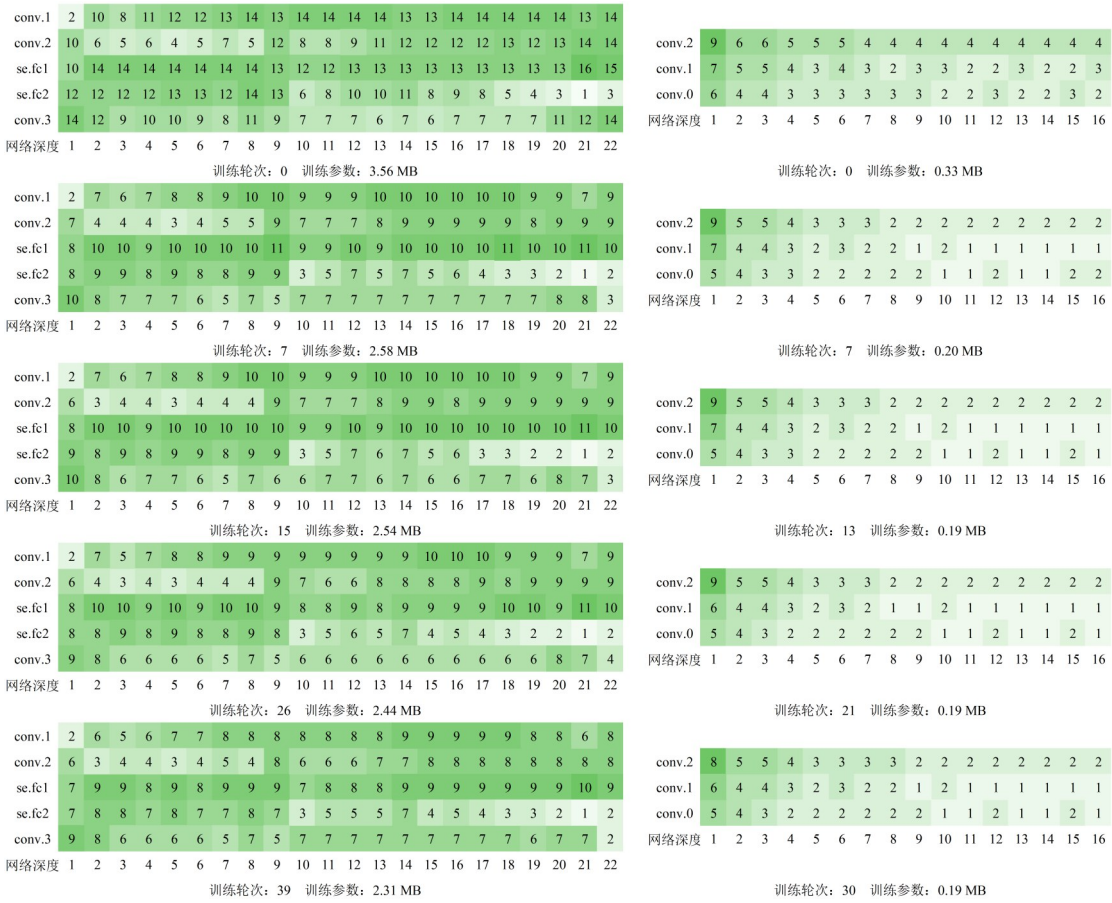
1）模型训练过程中，层间特征表示出现变化时各层级贡献度会改变，层级秩分配表相应调整以适应特征变化。

2）模型对高秩的需求主要集中在训练初期，随着训练深入，层间特征表示一致性逐渐增强，其对秩的需求均随着特征学习的收敛逐步降低。

3）MobileNetV2模型强调对病害高级语义特征的深度提取，在后期层的秩较高，各训练轮次的秩变化幅度较大；而RegNetY模型更注重层间特征融合，在前中期层就具备较高的特征表达能力，不同轮次秩需求的变化幅度较小。

因此，在训练过程中根据学习状态调整秩分配表，能够有效减少模型训练参数，实现对计算资源的持续优化。

在AlexNet、MobileNetV2、RegNetY和ConvNeXt模型中应用DRA方法与全参数微调进行对比。如表1所示，Avg_Parm表示训练过程中的平均训练参数，从整体结果来看，应用DRA方法在保



a. RegNetY 模型　　b. MobileNetV2 模型

图 3　RegNetY 和 MobileNetV2 模型层级秩分配表在训练过程中的变化

Fig. 3　Changes in layer-wise rank allocation tables of RegNetY and MobileNetV2 models during training

证模型准确率的前提下，显著降低了训练参数。其中 AlexNet、MobileNetV2、RegNetY 和 ConvNeXt 模型在小麦病害数据集上的可训练参数分别减少至 0.42%、2.46%、3.56% 和 1.25%；在植物病害数据集上的可训练参数分别减少至 0.31%、2.93%、2.81% 和 0.97%。表明 DRA 方法能够保持低秩适应方法的优势，同时更合理地给模型的各个层分配合适的秩，从而在提高准确率的同时节省开销。

表 1　AlexNet、MobileNetV2、RegNetY 和 ConvNeXt 模型在各数据集上的准确率与训练参数对比

Table 1　Comparison of the accuracy and the training parameters of AlexNet, MobileNetV2, RegNetY and ConvNeXt on each dataset

模型	小麦病害数据集				植物病害数据集			
	Full fine-tuning		DRA		Full fine-tuning		DRA	
	mAP/%	Parm/M	mAP/%	Avg_Parm/M	mAP/%	Parm/M	mAP/%	Avg_Parm/M
AlexNet	81.75	217.62	80.09	0.90	90.41	217.62	89.42	0.66
MobileNetV2	77.02	8.54	75.80	0.21	87.04	8.54	86.21	0.25
RegNetY	82.07	74.67	80.67	2.66	90.83	74.67	90.96	2.10
ConvNeXt	74.90	1 290	72.13	16.17	86.89	1 290	85.12	12.55

表 2 给出了 RRA 策略在两个植物病害数据集上的训练参数变化。T_i 表示在第 i 个轮次重新计算秩分配表。以 ConvNeXt 为例，在小麦病害数据集上，训练参数从初始的 19.62 MB 逐步减少至第 35 个轮次时的 13.73 MB；在植物病害数据集上，训练参数从 18.34 MB 减少至第 33 个轮次时的 9.26 MB，降幅接近 50%。该实验结果表明，随着训练的进行，模型逐渐收敛，所需训练参数可以持续优化。

表2　AlexNet、MobileNetV2、RegNetY和ConvNeXt模型应用RRA策略在各数据集微调过程中的训练参数大小对比

Table 2　Comparison of trainable parameter sizes during fine-tuning of AlexNet, MobileNetV2, RegNetY and ConvNeXt models with the RRA strategy on various datasets

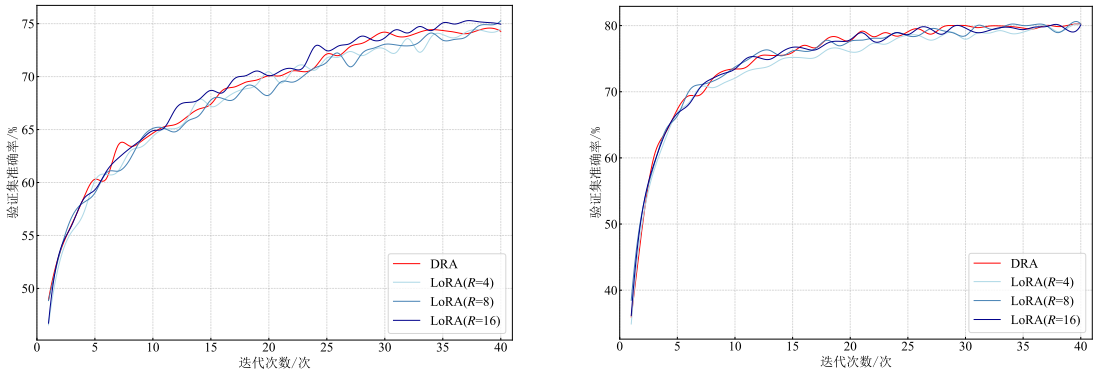
数据集	模型	Epoch	Parm/M	Epoch	Parm/M	Epoch	Parm/M	Epoch	Parm/M	Epoch	Parm/M
小麦病害数据集	AlexNet	T_0	1.35	T_7	1.01	T_{17}	0.76	T_{29}	0.72	/	/
	MobileNetV2	T_0	0.33	T_7	0.20	T_{13}	0.19	T_{21}	0.19	T_{30}	0.19
	RegNetY	T_0	3.56	T_7	2.58	T_{15}	2.54	T_{26}	2.44	T_{39}	2.31
	ConvNeXt	T_0	19.62	T_7	18.21	T_{14}	16.71	T_{24}	13.80	T_{35}	13.73
植物病害数据集	AlexNet	T_0	1.38	T_7	0.62	T_{19}	0.49	/	/	/	/
	MobileNetV2	T_0	0.27	T_7	0.26	T_{15}	0.25	T_{25}	0.23	T_{37}	0.23
	RegNetY	T_0	3.64	T_7	1.83	/	/	/	/	/	/
	ConvNeXt	T_0	18.34	T_7	13.58	T_{14}	13.47	T_{22}	10.45	T_{33}	9.26

注：“/”表示直至训练结束，秩分配表未再进一步更新。

2.2　与LoRA方法的对比

本研究使用AlexNet、MobileNetV2、RegNetY和ConvNeXt模型在数据环境较为复杂的小麦病害数据集上对比优化后的DRA方法与原始LoRA方法，原始LoRA方法分别取秩为4、8和16。

图4给出AlexNet和RegNetY模型训练过程的验证集准确率，可以发现DRA方法的曲线表现出更为稳定的上升趋势。测试结果见表3，与LoRA方法（R=16）对比，AlexNet、MobileNetV2和RegNetY在小麦病害数据集上的准确率分别提高0.38、0.40和0.05个百分点，资源消耗分别减少59.3%、87.4%和50.5%。DRA不仅优于秩分配较少LoRA方法，还略高于秩分配较高的LoRA方法，表明DRA能更好地适应模型不同层级对资源的需求，实现模型精度与资源消耗的平衡。



AlexNet　　RegNetY

图4　AlexNet和RegNetY模型DRA与LoRA微调验证集准确率

Fig. 4　Comparison of validation accuracy for AlexNet and RegNetY models under DRA and LoRA fine-tuning

2.3　消融实验

本研究在AlexNet、MobileNetV2、RegNetY和ConvNeXt这4种模型上进行消融实验，调整层重要性计算中的偏向权重α和β来评估不同指标组合的效果。实验共有5组，均在小麦病害数据集上进行，其中α = 0表示仅使用基于梯度的校正因子，β = 0表示仅使用CKA相似度。

记录各组实验的准确率和可训练参数大小，结果如表4和表5所示。对于所有测试模型，仅使用基于梯度的校正因子或仅使用CKA相似度作为秩分配依据，准确率均低于使用复合指标。例如，在RegNetY模型上，复合指标α = 0.25和β = 0.75的准确率达到80.99%，相对于仅使用校正因子或仅使用CKA相似度分别提升7.23和1.75个百分点。同时可以发现仅使用CKA相似度往往导致较高的训练参数，而仅使用校正因子则可训练参数最少，但性能也相应较差。复合指标结合两者的优势，在有效利用层间特征相似性指导参数分配的同时，兼顾

表3 AlexNet、MobileNetV2、RegNetY和ConvNeXt模型DRA与LoRA微调性能对比

Table 3 Performance comparison of DRA and LoRA fine-tuning on various models

模型	LoRA(R=4)		LoRA(R=8)		LoRA(R=16)		DRA	
	mAP/%	Parm/M	mAP/%	Parm/M	mAP/%	Parm/M	mAP/%	Parm/M
AlexNet	77.64	0.55	79.93	1.11	79.71	2.21	80.09	0.90
MobileNetV2	75.54	0.41	75.51	0.83	75.40	1.66	75.80	0.21
RegNetY	79.67	1.34	80.79	2.69	80.86	5.37	80.91	2.66
ConvNeXt	70.26	6.29	71.42	12.57	72.64	25.14	72.13	16.17

梯度信息对任务的直接贡献，从而在不同的模型中，都能以较低的训练参数实现较好的性能。

表4 不同α和β权重组合下AlexNet、MobileNetV2、RegNetY和ConvNeXt模型在小麦病害数据集上的消融实验准确率

Table 4 Accuracy of ablation experiments on AlexNet, MobileNetV2, RegNetY and ConvNeXt models under different α and β weight combinations on wheat disease dataset

单位：%

模型	$\alpha=0$, $\beta=1$	$\alpha=0.15$, $\beta=0.85$	$\alpha=0.20$, $\beta=0.80$	$\alpha=0.25$, $\beta=0.75$	$\alpha=1$, $\beta=0$
AlexNet	76.87	80.09	79.46	78.55	77.63
MobileNetV2	74.58	75.80	76.79	76.94	74.73
RegNetY	73.76	80.67	80.38	80.99	79.24
ConvNeXt	65.65	72.13	71.26	70.81	69.66

表5 不同α和β权重组合下AlexNet、MobileNetV2、RegNetY和ConvNeXt模型在小麦病害数据集上的可训练参数量

Table 5 Trainable parameter count of AlexNet, MobileNetV2, RegNetY and ConvNeXt models under different α and β weight combinations on wheat disease dataset

单位：M

模型	$\alpha=0$, $\beta=1$	$\alpha=0.15$, $\beta=0.85$	$\alpha=0.20$, $\beta=0.80$	$\alpha=0.25$, $\beta=0.75$	$\alpha=1$, $\beta=0$
AlexNet	0.70	0.90	0.95	1.17	1.89
MobileNetV2	0.15	0.18	0.18	0.21	1.16
RegNetY	0.67	2.66	2.88	3.42	4.30
ConvNeXt	10.26	16.17	17.53	19.3	24.12

2.4 鲁棒性分析

为了检验DRA方法的鲁棒性和适应性，本研究使用ConvNeXt模型在两个数据集上进行实验。本研究采用4种数据增强方法对测试集数据进行干扰，分别为向图像添加强度因子为0.05的高斯噪声；对图像进行随机区域裁剪，裁剪面积最大为原图的20%；对图像进行色彩抖动，随机调整其亮度、对比度、饱和度与色调，调整幅度最大为原图的10%；在0～90º对图像进行随机旋转。结果见表6。可以发现对于植物病害数据集，色彩抖动与随机旋转对测试结果的影响较大，准确率分别下降6.02和5.11个百分点。而对于小麦病害数据集，随机裁剪和随机旋转对测试结果的影响较大，准确率分别下降4.33和4.40个百分点。由于承载任务核心判别信息的关键层被赋予高秩，即使输入数据存在显著扰动，依然能够稳定地提取特征表示。整体下降幅度在可接受范围内，可以说明模型在面对复杂多样的数据环境时具有一定的鲁棒性。

表6 ConvNeXt模型应用DRA方法在4种数据干扰下进行鲁棒性测试的准确率

Table 6 Accuracy of the ConvNeXt model with DRA method in robustness test under four data perturbations

单位：%

数据集	DRA	Gaussian noise	Random crop	Color jitter	Random rotation
植物病害数据集	85.12	83.02	83.98	79.10	80.01
小麦病害数据集	72.13	68.93	67.80	70.42	67.73

2.5 与先进低秩方法的对比

为了全面评估DRA方法的有效性及其在众多先进方法中的特点，本研究设计并进行DRA方法与AdaLoRA和DyLoRA方法的对比实验。AdaLoRA方法采用SVD参数化，通过对单个奇异值的重要性进行评估和剪枝，以在训练中自适应参数预算。DyLoRA方法则追求推理期的灵活性，通过单次训练生成1个能适应多种秩配置的通用模型，然后在推理时根据不同的硬件或延迟需求动态切换秩配置。本研究对这两种方法进行原理性复现，在AlexNet和RegNetY模型，以及两个病害数据集上，从模型性能、训练效率和推理灵活性等维度对这些方法进行综合对比。

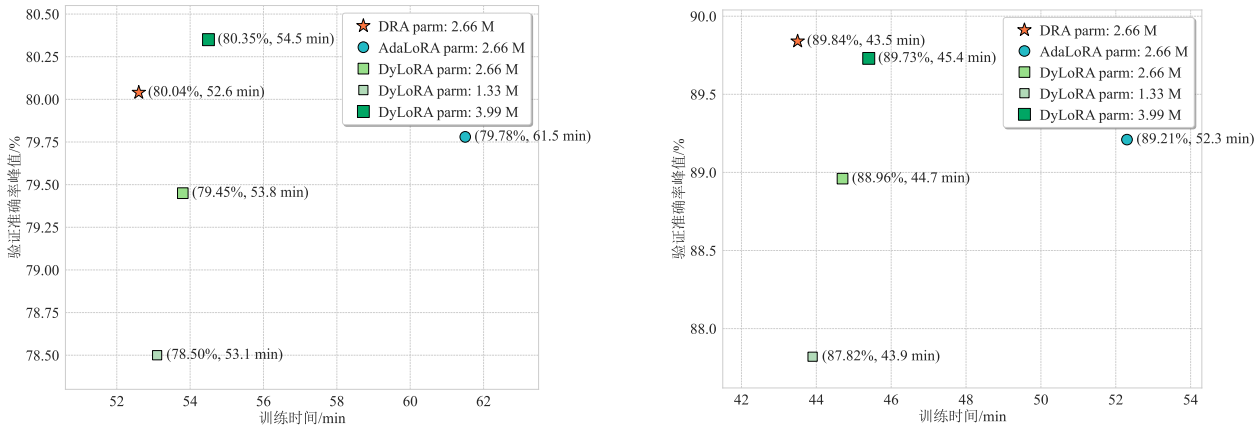
实验结果见表7，与AdaLoRA方法对比，在控制参数预算相同的情况下，DRA方法的准确率稳定地高于AdaLoRA。例如，在RegNetY模型上，DRA方法在小麦病害数据集上的准确率比AdaLoRA高出0.12个百分点，在植物病害数据集上高出0.55个百分点。DyLoRA方法的价值在于其推理的灵活性，在同等预算下，DRA方法的性能同样高于DyLoRA。例如，在AlexNet模型上，DRA方法在小麦病害数据集上的准确率比DyLoRA高出1.38个百分点，在植物病害数据集上高出1.39个百分点。

表 7　DRA、AdaLoRA 和 DyLoRA 方法的微调性能与参数预算对比

Table 7　Comparison of fine-tuning performance and parameter budget among DRA, AdaLoRA, and DyLoRA methods

模型	方法	小麦病害数据集		植物病害数据集	
		Avg_Parm/M	mAP/%	Avg_Parm/M	mAP/%
AlexNet	DRA	0.90	80.09	0.66	89.42
	AdaLoRA	0.90	79.95	0.66	88.65
	DyLoRA	1.35	80.16	0.99	89.25
	DyLoRA	0.90	78.71	0.66	88.03
	DyLoRA	0.45	76.88	0.33	86.57
RegNetY	DRA	2.66	80.67	2.10	90.96
	AdaLoRA	2.66	80.55	2.10	90.41
	DyLoRA	3.99	80.81	3.15	90.75
	DyLoRA	2.66	79.83	2.10	90.02
	DyLoRA	1.33	78.52	1.05	88.94

图 5 通过验证集准确率峰值与训练时间的权衡图直观地展示不同方法的综合效率，左上角代表用时少，精度高的区域。从图 5 中可以发现，在相同条件下，DRA 方法在达到验证集峰值准确率和所需训练时间上均表现出一定优势。在 RegNetY 模型上，DRA 方法在植物病害数据集上的准确率为 90.96%，比 AdaLoRA 方法高出 0.55 个百分点，比 DyLoRA 方法高出 0.94 个百分点。训练效率方面，在植物病害数据集上，DRA 方法需要 43.5 分钟达到 89.84% 的验证集峰值准确率，而 AdaLoRA 方法需要 52.3 分钟，训练时间增加约 20.23%。实验结果表明，DRA 方法在追求特定任务性能与训练参数的平衡上具有优势。相比之下，AdaLoRA 方法牺牲了部分训练效率，而 DyLoRA 方法更具推理的灵活性。



a. 小麦病害数据集　　b. 植物病害数据集

图 5　DRA、AdaLoRA 和 DyLoRA 方法的验证集准确率峰值与训练时间对比

Fig. 5　Comparison of validation set accuracy peaks and training time among DRA, AdaLoRA, and DyLoRA methods

3　结　论

针对植物病害识别任务中模型微调面临的计算资源受限与参数更新开销大的问题，本研究提出了一种 DRA。引入 CKA 量化跨层特征相似性，并结合基于梯度的校正因子实现动态分配秩，提出基于稳定性触发的 RRA，在训练过程中监测模型收敛状态，动态更新各层的秩。在植物病害数据集和小麦病害数据集等公开数据集上的实验表明，该方法在 AlexNet、MobileNetV2、RegNetY 和 ConvNeXt 等多种模型上均取得优异表现。该方法在保持接近全微调准确率的同时，显著减少了模型训练参数，为资源受限场景下的植物病害识别提供了高效解决方案。未来工作将进一步探索 DRA 在广泛视觉任务中的适用性，研究其在边缘计算设备上的部署潜力，以实现更高效的农业智能化应用，并结合知识蒸馏与动态稀疏化等技术进一步减少模型计算开销。

利益冲突声明：本研究不存在研究者以及与公开研究成果有关的利益冲突。

参考文献：

[1] 李淼, 王敬贤, 李华龙, 等. 基于CNN和迁移学习的农作物病害识别方法研究[J]. 智慧农业, 2019, 1(3): 46-55.
LI M, WANG J X, LI H L, et al. Method for identifying crop disease based on CNN and transfer learning[J]. Smart Agriculture, 2019, 1(3): 46-55.

[2] 乔世成, 潘春宇, 白明宇, 等. 基于改进YOLOv10n的石榴病害检测[J]. 沈阳农业大学学报, 2025, 56(4): 93-102.
QIAO S C, PAN C Y, BAI M Y, et al. Pomegranate disease detec-

tion based on improved YOLOv10n[J]. Journal of Shenyang Agricultural University, 2025, 56(4): 93-102.
[3] KRIZHEVSKY A, SUTSKEVER I, HINTON G E. ImageNet classification with deep convolutional neural networks[J]. Communications of The Acm, 2017, 60(6): 84-90.
[4] BADGUJAR C M, POULOSE A, GAN H. Agricultural object detection with You Only Look Once (YOLO) Algorithm: A bibliometric and systematic literature review[J]. Computers and Electronics in Agriculture, 2024, 223: 109090.
[5] HE K M, ZHANG X Y, REN S Q, et al. Deep residual learning for image recognition[C]// 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Piscataway, New Jersey, USA: IEEE, 2016: 770-778.
[6] DONG X Y, WANG Q, HUANG Q D, et al. PDDD-PreTrain: A series of commonly used pre-trained models support image-based plant disease diagnosis[J]. Plant Phenomics, 2023, 5: 0054.
[7] LIU Z, MAO H Z, WU C Y, et al. A ConvNet for the 2020s[C]// 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Piscataway, New Jersey, USA: IEEE, 2022: 11966-11976.
[8] HOWARD A, SANDLER M, CHEN B, et al. Searching for MobileNetV3[C]// 2019 IEEE/CVF International Conference on Computer Vision (ICCV). Piscataway, New Jersey, USA: IEEE, 2019: 1314-1324.
[9] FERENTINOS K P. Deep learning models for plant disease detection and diagnosis[J]. Computers and Electronics in Agriculture, 2018, 145: 311-318.
[10] 于明, 郭志永, 王岩. 基于计算机视觉的植物病害识别方法综述[J]. 科学技术与工程, 2024, 24(12): 4811-4823.
YU M, GUO Z Y, WANG Y. Review of computer vision-based plant disease identification techniques[J]. Science Technology and Engineering, 2024, 24(12): 4811-4823.
[11] 傅卓军, 胡政, 邓阳君, 等. 基于Deep-Semi-NMF的苹果斑点落叶病检测方法[J]. 智慧农业(中英文), 2024, 6(6): 144-154.
FU Z J, HU Z, DENG Y J, et al. Detection method of apple Alternaria leaf spot based on deep-semi-NMF[J]. Smart Agriculture, 2024, 6(6): 144-154.
[12] 康继昌, 赵连军. 基于PSA-YOLO11n的小麦害虫检测[J]. 农业大数据学报, 2025, 7(3): 294-306.
KANG J C, ZHAO L J. Wheat pest detection based on PSA-YOLO11n[J]. Journal of Agricultural Big Data, 2025, 7(3): 294-306.
[13] 李想, 胡肖楠, 李方一, 等. 苹果树叶多病害及不可辨别病害的轻量识别算法[J]. 农业工程学报, 2023, 39(14): 184-190.
LI X, HU X N, LI F Y, et al. Lightweight recognition for multiple and indistinguishable diseases of apple tree leaf[J]. Transactions of the Chinese Society of Agricultural Engineering, 2023, 39(14): 184-190.
[14] 叶进, 吴梦岚, 邱文杰, 等. 基于polyphyletic损失函数的荔枝花检测方法[J]. 农业机械学报, 2023, 54(5): 253-260.
YE J, WU M L, QIU W J, et al. Litchi flower detection method based on polyphyletic loss function[J]. Transactions of the Chinese Society for Agricultural Machinery, 2023, 54(5): 253-260.
[15] 邱文杰, 叶进, 胡亮青, 等. 面向植物病害识别的卷积神经网络精简结构Distilled-MobileNet模型[J]. 智慧农业(中英文), 2021, 3(1): 109-117.
QIU W J, YE J, HU L Q, et al. Distilled-MobileNet model of convolutional neural network simplified structure for plant disease recognition[J]. Smart Agriculture, 2021, 3(1): 109-117.
[16] 张俊, 陈雨艳, 秦震宇, 等. 改进DeepLab v3+模型下的梯田遥感提取研究[J]. 智慧农业(中英文), 2024, 6(3): 46-57.
ZHANG J, CHEN Y Y, QIN Z Y, et al. Remote sensing extraction method of terraced fields based on improved DeepLab v3+[J]. Smart Agriculture, 2024, 6(3): 46-57.
[17] HU E J, SHEN Y L, WALLIS P, et al. LoRA: Low-rank adaptation of large language models[EB/OL]. arXiv: 2106.09685, 2021.
[18] VALIPOUR M, REZAGHOLIZADEH M, KOBYZEV I, et al. DyLoRA: Parameter-efficient tuning of pre-trained models using dynamic search-free low-rank adaptation[C]// Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics. Stroudsburg, PA, USA: ACL, 2023: 3274-3287.
[19] ZHANG Q R, CHEN M S, BUKHARIN A, et al. AdaLoRA: Adaptive budget allocation for parameter-efficient fine-tuning [EB/OL]. arXiv: 2303.10512, 2023.
[20] KORNBLITH S, NOROUZI M, LEE H, et al. Similarity of neural network representations revisited[EB/OL]. arXiv: 1905.00414, 2019.
[21] DAVARI M, HOROI S, NATIK A, et al. Reliability of CKA as a similarity measure in deep learning[EB/OL]. arXiv: 2210.16156, 2022.
[22] SANDLER M, HOWARD A, ZHU M L, et al. MobileNetV2: Inverted residuals and linear bottlenecks[C]// 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway, New Jersey, USA: IEEE, 2018: 4510-4520.
[23] RADOSAVOVIC I, KOSARAJU R P, GIRSHICK R, et al. Designing network design spaces[C]// 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Piscataway, New Jersey, USA: IEEE, 2020: 10425-10433.

Low-rank Adaptation Method for Fine-tuning Plant Disease Recognition Models

HUANG Jinqing[1,2], YE Jin[1,2*], HU Huilin[1,2], YANG Jihui[3], LAN Wei[1,2], ZHANG Yanqing[1,2]

(1. *School of Computer, Electronics and Information, Guangxi University, Nanning* 530000, *China*; 2. *State Key Laboratory for Conservation and Utilization of Subtropical Agrobiological Resources, Nanning* 530000, *China*; 3. *JJR Science and Technology Group Co. , Ltd. , Nanning* 530000, *China*)

Abstract: [Objective] When deep learning is applied to plant disease recognition tasks, model fine-tuning faces significant challenges, including limited computational resources and high parameter update overhead. Although traditional low-rank adaptation (LoRA) methods effectively reduce parameter overhead, their strategy of assigning a uniform, fixed rank to all layers often overlooks the varying importance of different layers. This approach may still lead to constrained optimization in critical layers or resource waste in less significant ones. To address this limitation, a dynamic rank allocation (DRA) algorithm is proposed in this research. The DRA algorithm is designed to evaluate and adjust the required parameter resources for each layer during training, enhance the accuracy of plant disease classification models while more efficiently balancing computational resources. **[Methods]** Public datasets of the Wheat Plant Diseases Dataset and the Plants Disease Dataset were utilized in the experiments. The Wheat Plant Diseases Dataset comprised 13 104

images covering 15 types of wheat diseases such as black rust and fusarium head blight, while the Plants Disease Dataset included 37 505 images of 26 types of plant diseases such as algal leaf spot, corn rust, and bacterial spot of tomato. These datasets were captured under varied lighting, different backgrounds, diverse angles, and at various stages of plant growth. A cross-layer feature similarity metric based on centred kernel alignment (CKA) was introduced to quantify the representational correlation between different layers. Concurrently, a correction factor was constructed based on gradient information and activation intensity to measure the direct impact of each layer on the loss function. These two metrics were then fused using a weighted harmonic mean to generate a comprehensive importance score, which was subsequently used for the initial rank allocation. Furthermore, considering the effect of feature representation changes during training, a stability-triggered adaptive rank update strategy rank re-allocation (RRA) was proposed. This strategy monitored the average parameter change of the low-rank adapters during the training process to determine the convergence state. When this change fell below a specific threshold, the low-rank matrices were merged into the original weights, and the rank allocation table was then re-calculated and updated. This process ensured that more resources were allocated to critical layers, thereby achieving an optimized allocation of parameter resources across different layers. **[Results and Discussions]** Tests on four models (AlexNet, MobileNetV2, RegNetY, and ConvNeXt) indicated that, compared to full-parameter fine-tuning, the proposed method reduced resource consumption to 0.42%, 2.46%, 3.56%, and 1.25%, respectively, while maintaining a comparable average accuracy. The RRA strategy demonstrated continuous parameter optimization throughout the model's training. On the ConvNeXt model, the trainable parameters on the plants disease dataset were progressively reduced from 18.34 to 9.26 M, a reduction of nearly 50%. In comparison with the standard LoRA method (R=16), the method reduced the accuracy by 0.38, 0.40 and 0.05 percentage points on the wheat plant diseases dataset for AlexNet, MobileNetV2, and RegNetY, respectively, while resource consumption was reduced by 59.3%, 87.4% and 50.5%. Robustness was tested by applying perturbations to the test set, including Gaussian noise, random cropping, color jitter, and random rotation. The results showed that the model was most affected by color jitter and random rotation on the Plants Disease Dataset, with accuracy decreasing by 6.02 and 5.11 percentage points, respectively. On the wheat plant diseases dataset, the model was more sensitive to random cropping and random rotation, with accuracy decreasing by 4.33 and 4.40 percentage points, respectively; the overall performance degradation remained within an acceptable range. When compared to other advanced low-rank methods such as AdaLoRA and DyLoRA under the same parameter budget, the DRA method exhibited higher accuracy. On the RegNetY model, the DRA method achieved an accuracy of 90.96% on the Plants Disease Dataset, which was 0.55 percentage points higher than AdaLoRA and 0.94 percentage points higher than DyLoRA. In terms of training efficiency on the Plants Disease Dataset, the DRA method required 43.5 minutes to reach its peak validation accuracy of 89.84%, whereas AdaLoRA required 52.3 minutes, representing a training time increase of approximately 20.23%. Regarding inference flexibility, the DyLoRA method was designed to generate a universal model capable of adapting to multiple rank configurations after a single training run, allowing for dynamic rank switching during inference based on hardware or latency requirements. The DRA method, however, did not possess this inference-time flexibility. It was focused on converging to a single, high-performance rank configuration for a specific task during the training phase. **[Conclusions]** The low-rank adaptive fine-tuning method proposed in this research significantly reduced the number of model training parameters while ensuring plant disease recognition accuracy. Compared to traditional fixed-rank LoRA and other advanced low-rank optimization methods, it demonstrated distinct advantages, providing an effective pathway for efficient model deployment on resource-constrained devices.

Key words: low-rank adaptive fine-tuning; feature similarity; dynamic rank allocation algorithm; rank allocation update strategy

Foundation items: National Natural Science Foundation of China (32402495); 2024 Autonomous Region-Level Student Innovation Training Program Project (202410593001S)

First author: HUANG Jinqing, E-mail: 1766042592@qq.com

***Corresponding author:** YE Jin, E-mail: yejin@gxu.edu.cn