文章编号:1671-4598(2025)10-0191-08 DOI:10.16526/j.cnki.11-4762/tp.2025.10.025 中图分类号:TP389.1 文献标识码:A

基于阈值非结构化剪枝的 MobileNetV3 模型优化方法

白鸿冰[1,2]，杨延宁[1,2]，姚 旭[1,2]
(1. 延安大学 物理与电子信息学院，陕西 延安 716000；
2. 陕西省能源大数据智能处理省市共建重点实验室，陕西 延安 716000)

摘要：为解决深度学习模型在移动设备和嵌入式系统中的高效应用问题，对 MobileNetV3 模型进行了优化研究；分析了如何通过剪枝技术减少模型的计算量和参数量，以提高其在资源受限环境中的应用效率；采用了粗粒度通道剪枝与细粒度非结构化剪枝相结合的策略，显著减少了参数量和计算开销，为应对剪枝引起的精度下降，结合深度增强策略通过增加模型深度弥补性能损失；技术创新体现在结合粗粒度与细粒度剪枝的优化策略，有效平衡了模型精度与计算效率；实验在 CIFAR-10 和 CIFAR-100 数据集上验证了该方法，结果显示，优化后的模型显著降低了计算成本并保持了高分类精度，CIFAR-100 数据集精度提升 8.1%，CIFAR-10 数据集精度提升 2.08%；该方法适用于资源受限的设备，满足了对低计算开销和高精度的实际应用需求。

关键词：MobileNetV3；阈值非结构化剪枝；模型优化；计算复杂度；深度增强

Optimization Method for the MobileNetV3 Model Based on Threshold Unstructured Pruning

BAI Hongbing[1,2], YANG Yanning[1,2], YAO Xu[1,2]
(1. School of Physics and Electronic Information, Yan'an University, Yan'an 716000, China;
2. Key Laboratories Jointly Built by Shaanxi Provinces and Cities of Intelligent Processing of Big Energy Data, Yan'an 716000, China)

Abstract: To address the efficient application of deep learning models on mobile devices and embedded systems, study on the MobileNetV3 model is optimized; This paper analyzes how pruning techniques can reduce the computation and parameters of the model to enhance its efficiency in resource-constrained environments; A strategy combining coarse-grained channel pruning and fine-grained unstructured pruning is presented, significantly reducing parameters and computational overhead; To compensate for the accuracy degradation caused by pruning, a depth enhancement strategy is incorporated to restore performance by increasing model depth; The technical innovation lies in the combination of coarse-grained and fine-grained pruning, effectively balancing the accuracy and computational efficiency of the model; Experiments on the CIFAR-10 and CIFAR-100 datasets verify the method, and the results show that the optimized model significantly reduces computational costs while maintaining a high classification accuracy, with accuracy improvements of 8.1% on CIFAR-100 and 2.08% on CIFAR-10, respectively; This method is suitable for resource-constrained devices, meeting practical requirements for low computational cost and high accuracy.

Keywords: MobileNetV3; threshold-based unstructured pruning; model optimization; computational complexity; depth enhancement

收稿日期:2024-08-24； **修回日期**:2024-10-08。
基金项目:国家自然科学基金项目(52365069)。
作者简介:白鸿冰(2000-),男,硕士研究生。
杨延宁(1969-),男,博士,教授。
引用格式:白鸿冰,杨延宁,姚 旭.基于阈值非结构化剪枝的 MobileNetV3 模型优化方法[J].计算机测量与控制,2025,33(10):191-198.

0 引言

在当今数据驱动的世界中，卷积神经网络（CNN，convolutional neural network）的广泛应用推动了计算机视觉技术的发展。MobileNetV3[1]作为谷歌提出的一种高效轻量级神经网络架构，以其优异的性能和较低的计算成本，在移动设备和嵌入式系统中得到了广泛应用。MobileNetV3 在许多标准数据集上的表现虽然令人印象深刻，但其参数量和计算量在处理实际应用中仍然是一个关键挑战。为应对这一问题，本文首先采用 Network Slimming 方法[2]进行粗粒度剪枝，通过稀疏化通道移除不重要的通道，减少模型的初步计算开销和参数量；随后，基于梯度信息，进一步进行非结构化剪枝，精细化地移除网络中个别不重要的权重，生成更加稀疏的矩阵，进一步减少计算量和模型体积。为弥补非结构化剪枝带来的性能下降，本文通过增加模型的深度和复杂度进行优化，确保剪枝后的模型在减少参数量的同时，保持甚至提升精度。通过这一系列优化，不仅有效减轻了计算负担，还在保持或提高模型精度的同时提升了推理效率[3-4]。为了验证所提方法的有效性，在两个数据集（CIFAR-10、CIFAR-100）上进行了实验。实验结果表明，优化后的 MobileNetV3 模型在分类任务中表现出色，减少了参数量和计算开销的同时，保持了高精度。这一研究为轻量级神经网络的进一步优化，资源受限环境中的深度学习应用提供了新的思路。对于嵌入式设备存在的资源受限问题[5]，此研究为解决此问题起到参考作用。

在现有研究中，剪枝技术被广泛应用于神经网络的压缩与加速，但这些方法存在一定的局限性。例如，文献［6］提出的结合权重和连接剪枝的技术，虽然显著减少了神经网络的存储需求和计算量，但这种方法主要基于固定的阈值进行权重剪枝，忽视了模型结构的层次性，难以动态调整网络的复杂性。文献［7］提出的基于梯度的剪枝方法，通过动态评估卷积核的重要性进行剪枝，提升了推理效率，但该方法主要侧重于卷积核的剪枝，对于更细粒度的权重选择缺乏考虑。文献［8］利用 L2 范数对通道的重要性进行评估，并据此进行通道剪枝，这种方法虽然能够加速深层网络的推理速度，但在权重级别上的优化不够精细，可能忽略了一些局部重要的细粒度信息。文献［9］的结构化剪枝与神经网络架构搜索结合，虽取得了一定效果，但其方法过于依赖于预定义的网络结构优化策略，无法灵活应对实际应用中的变化。文献［10］提出的结合稀疏正则化和结构化剪枝的方法虽然有效减少了 MobileNetV3 的参数量，但其对权重和通道的处理较为单一，剪枝粒度不足，精度恢复有限。

1 MobileNet 网络模型

MobileNet 是由 Google 提出的轻量级神经网络系列，专门为在移动和嵌入式设备上运行设计。这个系列包括 MobileNetV1、MobileNetV2 和 MobileNetV3，每一代都在前一代的基础上进行了改进，以提高模型的计算效率和准确性。MobileNetV1 显著降低了传统卷积神经网络的计算量和参数量，提出了深度卷积和逐点卷积两种卷积方式。两者结合形成了深度可分离卷积[11]，既保留模型表达能力，又大幅降低计算成本，是轻量级 CNN 的核心技术，深度卷积结构如图 1 所示。

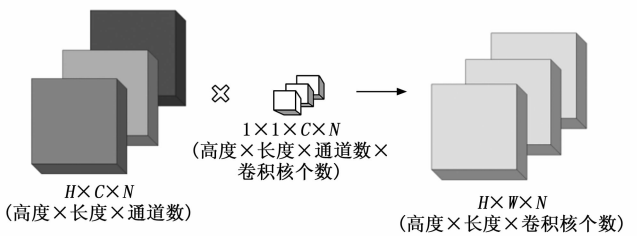

图 1 深度卷积

而逐点卷积通过 1×1 卷积将深度卷积的输出线性组合到新的输出通道中，如图 2 所示。

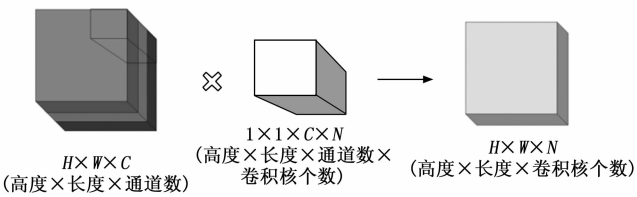

图 2 逐点卷积

MobileNetV2 在 MobileNetV1 的结构上进行了改进，新增了线性瓶颈模块和倒残差结构[12]。结构如图 3 所示。线性瓶颈模块通过使用线性激活函数来减少非线性失真，提高模型表现。

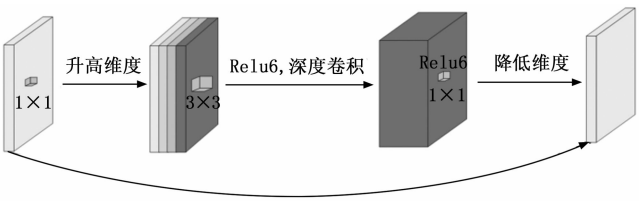

图 3 逆残差网络模型

与传统模型相比较，具体表现出减少了多少参数量以及计算量[13]。根据量化公式如下所示：

$$Params = \frac{K^2 \times C \times F}{K^2 \times C + C \times F} \tag{1}$$

$$Flops = \frac{H \times W \times C \times K^2 \times F}{H \times W \times C \times K^2 + H \times W \times C \times F} \tag{2}$$

式中，分子为传统模型的参数量以及计算量，分母为深

度卷积、逐点卷积的参数量和计算量；H，W 为图像的高和宽；K 为卷积核的尺寸，通常为正方形卷积核；F 为输出特征图的通道数；C 为通道数。

以 3×3 的卷积核为例，通过以上计算可以看出，将计算量和参数量减少了约 8～9 倍。

改进的瓶颈结构如图 4 所示。

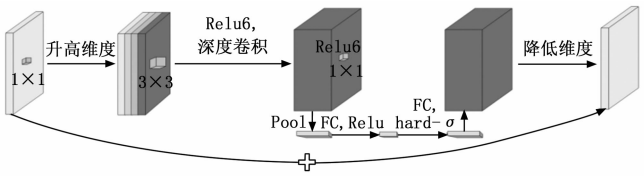

图 4　MobileNetV3 改进的瓶颈结构图

MobileNetV3 在继承了 MobileNetV1 的深度可分离卷积和 MobileNetV2 的线性瓶颈倒残差结构（IR，inverted residual）及压缩激励模块（SE，squeeze and excite）的基础上，还引入了神经网络架构搜索（NAS，neural architecture search）技术，提升了效率和性能。V3 网络包含 4 330 132 个参数，计算量为 233 686 456 次。还采用了 ReLU6 和两种新的激活函数 h-swish（x）与 h-sigmoid（x），这两种函数是进行硬化的版本，更适合在资源受限的移动设备上使用，计算更加简单高效，函数硬化过程表达式：

$$\text{Swish} = x \cdot \sigma(x) \tag{3}$$

$$\text{sigmoid} = \sigma(x) = \frac{1}{1+e^{-x}} \tag{4}$$

$$\text{Rule6}(x) = min(max(0,x),6) \tag{5}$$

$$\text{Hswish}(x) = x \cdot \frac{\text{Rule6}(x+3)}{6} \tag{6}$$

$$\text{Hsigmoid}(x) = \frac{\text{Rule6}(x+3)}{6} \tag{7}$$

其中包括 20 个模块：从输入尺寸为 3 224 224 的二维卷积层开始，经过多个瓶颈层和 3×3、5×5 不同大小的卷积核，以及扩展通道数和使用的激活函数 RE 和 HS，部分层使用注意力机制，最终通过池化层和二维卷积层输出。表 1 是 MobileNetV3-lager 网络结构，图 5 为网络结构图。

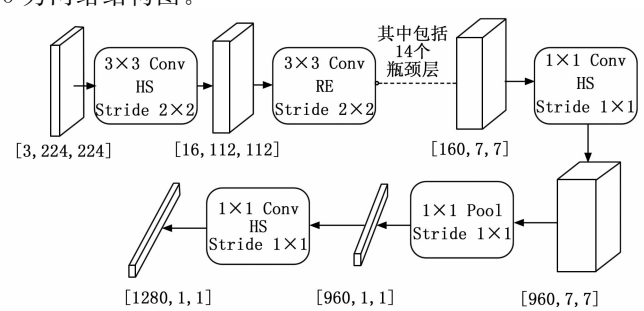

图 5　MobileNetV3-lager 网络结构

表 1　MobileNetV3－lager 网络结构

模块数	组成层	输入尺寸	扩展通道数	激活函数	步长	注意力机制
1	二维卷积层	[3,224,224]	—	HS	2×2	—
2	瓶颈层,3×3	[16,112,112]	16	RE	1×1	—
3	瓶颈层,3×3	[16,112,112]	64	RE	2×2	—
4	瓶颈层,3×3	[24,56,56]	72	RE	1×1	—
5	瓶颈层,5×5	[24,56,56]	72	RE	2×2	+
6	瓶颈层,5×5	[40,28,28]	120	RE	1×1	+
7	瓶颈层,5×5	[40,28,28]	120	RE	1×1	+
8	瓶颈层,3×3	[40,28,28]	240	HS	2×2	—
9	瓶颈层,3×3	[80,14,14]	200	HS	1×1	—
10	瓶颈层,3×3	[80,14,14]	184	HS	1×1	—
11	瓶颈层,3×3	[80,14,14]	184	HS	1×1	—
12	瓶颈层,3×3	[80,14,14]	480	HS	1×1	+
13	瓶颈层,3×3	[112,14,14]	672	HS	1×1	+
14	瓶颈层,5×5	[112,14,14]	672	HS	2×2	+
15	瓶颈层,5×5	[160,7,7]	960	HS	1×1	+
16	瓶颈层,5×5	[160,7,7]	960	HS	1×1	+
17	二维卷积层,1×1	[160,7,7]	—	HS	1×1	—
18	池化层,7×7	[960,7,7]	—	—	1×1	—
19	二维卷积层,1×1	[960,1,1]	—	HS	1×1	—
20	二维卷积层,1×1	[1280,1,1]	—	—	1×1	—

2　基于阈值的非结构化剪枝方法

2.1　剪枝策略

非结构化剪枝是一种用于减少深度学习模型参数和加快推理速度的技术。它通过剪除不重要的权重来稀疏化模型，降低计算复杂度和存储需求。该方法基于权重的重要性进行判断，灵活性高，可应用于任何层，并能精细控制稀疏度。其缺点是剪枝后的稀疏矩阵不规则，难以在现有硬件和软件框架中高效加速计算，并且实现复杂，需专门策略处理稀疏矩阵。非结构化剪枝适用于需要大幅减少模型参数的场景，特别是在存储和传输受限时。

为了解决非结构化剪枝带来的稀疏矩阵不规则性以及硬件和软件框架难以高效加速的问题，本文首先采用粗粒度的通道剪枝进行初步优化。通过移除整块结构来简化网络结构，使得生成的稀疏矩阵具有规律性，能够更好地适应现有的硬件加速器。这种通道剪枝有效减少了模型的计算量和参数量，并确保剪枝后的模型仍然保持较好的硬件兼容性。在此基础上，本文进一步采用细粒度的非结构化剪枝，逐个剪除不重要的权重，进一步提高稀疏度，达到更高的参数压缩和计算复杂度的降低。这种两阶段剪枝策略既保证了剪枝后模型的规则性，使其能够适应现有硬件加速，同时利用非结构化剪枝的精细控制能力，在剪枝后仍能保持高精度与高稀疏度。最终，该方法在减少模型体积和推理时间的同时，充分发挥了剪枝的灵活性和性能优化能力。剪枝流程如

图 6 所示。

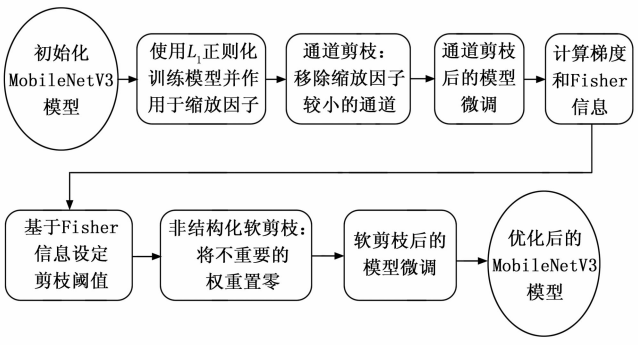

图 6 剪枝策略流程图

2.2 粗粒度通道剪枝

在进行模型剪枝工作时，通常会优先考虑剪枝那些参数量和计算量较大的层，因为这些层的剪枝效果最显著，能够大幅减少模型的参数量和计算量，从而提高模型的运行效率。对 MobileNetV3-Large 模型参数量和计算量的分析，如图 7、图 8 所示。

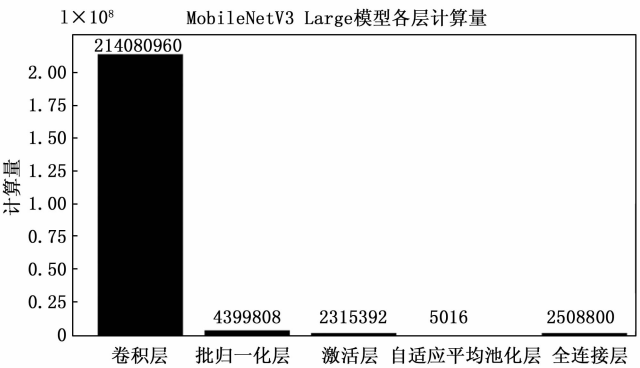

图 7 模型各模块计算量

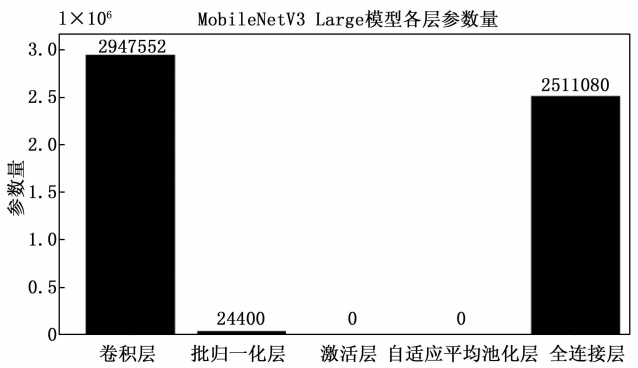

图 8 模型各模块参数量

从图 7 和图 8 可以看出，卷积层和全连接层是模型中参数量和计算量最多的部分，因此是剪枝工作的主要目标，而根据网络模型的结构可以看出，每个卷积层后都会跟着一个批次归一化层，尽管批量归一化层的参数量和计算量相对较低，但由于其与卷积层和全连接层密切相关，剪枝时需要对它们进行相应的调整，以确保模型的一致性和性能。

因此先采用粗粒度通道剪枝方法对 MobileNet-V3 模型进行压缩并在压缩后的模型中增加模型深度，以尽量减少甚至提高模型的准确率。同时通过在两个不同的数据集上进行验证模型的适应性和性能一致性。利用注意力可视化手段，对模型进行性能评估，通过可视化检查注意力模式是否合理，判断模型是否学习到了正确的特征[14-15]。粗粒度通道剪枝方法利用了稀疏化 BN 层的缩放参数 γ，并按缩放参数的大小设定阈值对 BN 层进行剪枝[16]。为了解决不一致性，BN 层的每个通道对应卷积层的一个输出通道，如果剪掉 BN 层的一个通道，但保留卷积层对应的滤波器，这个滤波器的输出将失去归一化和缩放的过程，导致卷积层输出和 BN 层输入的不一致，可能会引起模型性能的下降，所以在剪枝 BN 层时也要剪枝掉对应的卷积层滤波器。根据 BN 层批次归一化的原理，如下式所示：

$$\hat{x}_i = \frac{x_i - \mu_B}{\sqrt{\sigma_B + \varepsilon}} \tag{8}$$

$$\mu_B = \frac{1}{B}\sum_{i=1}^{B} x_i \tag{9}$$

$$\sigma_B^2 = \frac{1}{B}\sum_{i=1}^{B}(x_i - \mu_B)^2 \tag{10}$$

式中，$\hat{x}_i$ 为归一化结果，μ_B 为均值，σ_B^2 为方差。

BN 层在网络模型中的作用是为了稳定每一层的输入分布，将输入数据稳定在激活函数的线性部分，提高模型的训练效果。但如果总是将模型的每一层稳定在线性区间，模型可能会失去部分非线性特性，这会导致模型的表达能力受限，无法充分利用激活函数的非线性特性来捕捉复杂的数据模式。所以要通过 BN 层的缩放和平移机制，引入可学习的参数 γ 和 β：

$$y_i = \gamma x_i + \beta \tag{11}$$

对参数进行逆变换，恢复一定的非线性部分，并且这些参数在训练过程中通过反向传播进行优化。在训练过程中，通过反向传播优化 γ 参数，使其值能够反映每个通道的重要性。量化 γ 值，并根据其大小进行剪枝，可以有效地识别和移除那些冗余的、不重要的通道。因此，可以通过量化的 γ 值，并且设置阈值，对 BN 层进行筛选，剪掉小于阈值的 BN 层通道，保留大于阈值的通道。

可学习的参数 γ 和 β 通过反向传播计算梯度，并其利用优化器进行更新，对 β 的梯度：

$$\frac{dL}{d\beta} = \sum_{i=1}^{m}\frac{dL}{dy_i} \tag{12}$$

对 γ 的梯度：

$$\frac{dL}{d\gamma} = \sum_{i=1}^{m}\frac{dL}{dy_i}\cdot\bar{x_i} \tag{13}$$

梯度公式在反向传播过程中逐层计算，最终更新模型的参数。

对于 γ 和 β 值量化以后，再对其 L1 正则化处理，正则化鼓励模型参数变为 0，从而引入稀疏性[17]。这使得模型中许多不重要的 γ 值趋向于 0，识别出冗余的通道，处理结果如下：

$$L = L_{\text{original}} + \lambda \cdot \sum_{i=1}^{C} | \gamma_i | \tag{14}$$

为交叉熵损失；λ 是正则化系数，控制稀疏性约束项的权重。较大的 λ 值会导致更强的稀疏性；是 BN 层第 i 个通道的缩放参数；C 是 BN 层的通道数。阈值设定法非结构剪枝结构如图 9 所示。

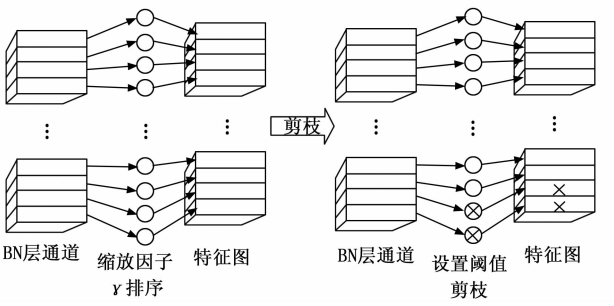

图 9　阈值设定法非结构剪枝结构图

2.3　细粒度非结构化剪枝

在通道剪枝之后，为进一步优化模型的计算效率和参数量，引入了基于 Fisher 信息矩阵的非结构化剪枝方法[17]，并采用了软剪枝策略。通过软剪枝，我们并不改变模型的整体结构，而是仅将不重要的权重置零，从而保持模型的完整性，简化网络结构并确保剪枝后的模型能够高效适应现有硬件加速器。利用 Fisher 信息矩阵评估每个权重对模型性能的影响，通过计算每个权重对损失函数的敏感性，筛选出不重要的权重进行逐个剪除，从而实现更加精细的稀疏化。这种结合了通道结构化剪枝和非结构化剪枝的双重策略，不仅提升了模型压缩效果，还保持了高精度，适合在资源受限的设备上进行部署。

计算 Fisher 信息矩阵是进行非结构化剪枝的重要步骤之一，如下式：

$$I(\theta) = E\left[\left(\frac{\partial \log p(y|x;\|\theta)}{\partial \theta}\right)^2\right] \tag{15}$$

其中：I 为网络权重，p 是模型在给定输入 x 时的输出概率分布，E 为表示对输入数据的期望值。Fisher 信息矩阵描述了当参数 θ 发生微小变化时，对模型输出的概率分布带来的影响。

直接计算完整的 Fisher 信息矩阵通常是不可行的，因为涉及大量的计算和存储开销。因此，通常采用近似方法来计算 Fisher 信息。对于每个参数 θ，首先计算损失函数关于该参数的梯度：

$$g_\theta = \frac{\partial L(y,x)}{\partial \theta} \tag{16}$$

其中：L 为模型的损失函数，y 为真实标签，x 为模型预测输出。为了简化计算，通常不计算完整的 Fisher 信息矩阵，而是基于参数的梯度平方进行累积，以近似计算 Fisher 信息，其中 N 为数据样本的数量：

$$I(\theta) = \sum_{i=1}^{N} g_\theta^2 \tag{17}$$

根据 Fisher 信息矩阵的值，按照百分比设定阈值 T，如果某个权重的 Fisher 信息小于阈值，则该权重被置零：

$$I(\theta) < T, \theta \leftarrow 0 \tag{18}$$

3　模型优化实验

3.1　实验环境

实验采用 cifar-10、cifar-100 两种数据集，分别是 10 分类、100 分类的数据集。这两个数据集涵盖了从低到高不同的分类难度和图像分辨率，能全面评估模型的性能。CIFAR-10 简单易用，适合快速测试；CIFAR-100 更加复杂，能测试模型在细分类别上的性能，适合评估模型在更复杂场景下的表现，有助于促进相关研究。使用 4 张 NVIDIA GeForce GTX1080 Ti 搭建的服务器进行实验，在 Pytorch 架构下进行模型的剪枝和训练、验证、测试。对于完整模型训练和剪枝后模型训练，初始学习率都为 0.001，使用的优化器均为 AdamW。完整模型训练为 150 轮，剪枝后模型训练为 100 轮。在训练过程中使用 StepLR 学习率调度器，设定每 30 轮进行一次学习率更新，更新为上一次的 1/10。在训练过程中，采用早停（Early Stopping）策略，设定一个耐心系数（Patience），耐心系数通过设定等待的 epoch 数来监控模型性能的变化，如果在这些 epoch 中性能没有改善，则采取措施如调整学习率或提前停止训练，以避免过拟合。采用的深度增强手段包括数据增强和 Dropout 正则化技术。

3.2　实验结果分析

3.2.1　BN 层 γ 值分析

在实验中，剪枝策略是基于 BN 层的缩放系数 γ 值进行筛选，通过设定阈值来实现剪枝操作。具体来说，对于小于设定阈值的 γ 值所对应的通道，执行剪枝，以此达到模型压缩的目的，进而减少参数量和计算量。由于较小的 γ 值通常表明该通道对模型输出的贡献较小，因此剪除这些通道对模型准确性的影响也较小。

在稀疏化 BN 层的过程中，缩放参数 γ 和可学习参数 β 的设置原则至关重要。BN 层的缩放参数 γ 反映了每个通道的相对重要性。在训练过程中，γ 作为可学习参数，通过反向传播进行优化。γ 值会动态调整，使其能够反映出每个通道在网络中的重要性。对于剪枝操

作，设定阈值时需要特别关注 γ 值的分布，确保剪除对模型性能影响较小的通道。为了避免过度剪枝影响模型的表达能力，通常选择的阈值应该兼顾模型精度和压缩效果，阈值设定需要结合实验调优。选取了 3 个不同的阈值：0.1、0.3、0.5，分别对模型进行剪枝，目的是探讨不同程度的剪枝对模型准确率的影响。图 10 展示了以 MobileNetV3 网络在 CIFAR-100 数据集上训练后的模型为例，3 组不同阈值下 BN 层 γ 值的分布情况。从上至下分别为阈值 0.1、0.3、0.5。通过对比不同阈值下的 γ 值分布情况，我们可以分析在不同剪枝程度下，哪些通道被剪除，以及这些操作对模型性能的影响。在设定这些阈值时，我们还需考虑 γ 值的分布特性，确保在剪掉不重要通道的同时，保留住那些重要的、有贡献的通道，从而维持模型的非线性表达能力和性能。通过设定不同的阈值，可以更好地平衡模型的剪枝比例和最终性能，确保剪枝后的模型仍具有良好的泛化能力和准确率。

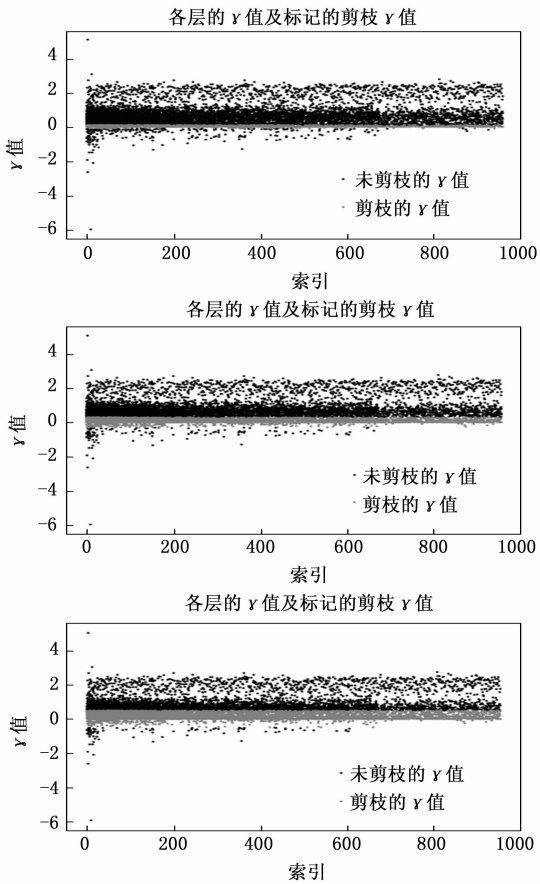

图 10 设定阈值在 BN 层整体 γ 值的分布

由图 10 可以看出，选用的 3 组阈值在模型 BN 层整体结构的 γ 值中占比不同，进而对模型剪枝的程度也就不同，设定的阈值越高，对模型剪枝的程度越深。

3.2.2 完整模型训练

实验选取 MobileNetV3-Large 作为模型结构，对数据集 cifar-10 和 cifar100 两个数据集进行训练，随后对不同程度的剪枝模型进行了同样的数据集训练，起到对照组的作用，以观察剪枝前后的模型训练的精确值变化。实验引入了 Top-1、Top-3、Top-5 共 3 种准确值，从训练集、验证集以及测试集 3 个角度对数据进行分析，有利于多角度评估模型性能，衡量模型的泛化能力并且深入理解模型表现。图 11、图 12 为完整模型在两种数据集上的具体数据。

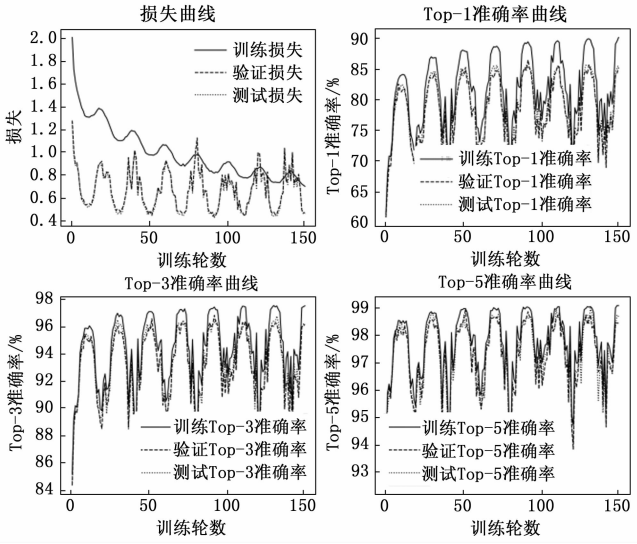

图 11 cifar-10 数据集下的模型训练数据图

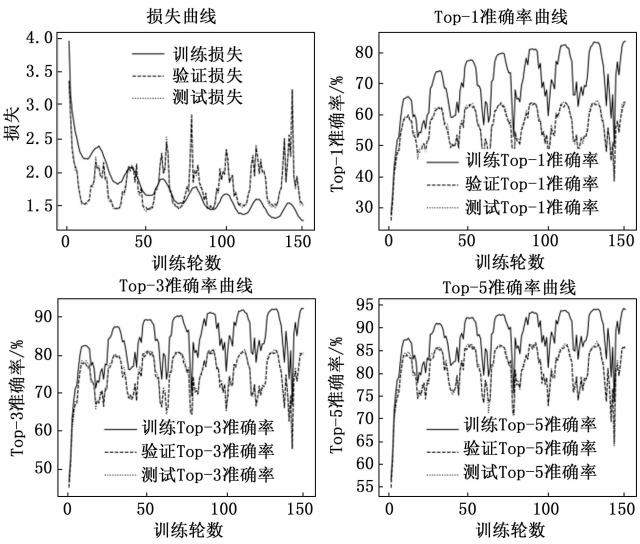

图 12 cifar-100 数据集下的模型训练数据

3.2.3 可视化注意力机制实验

可视化注意力机制[18]可以帮助研究者对模型在做出决策时关注图像的某些区域，提供了一种使模型更加透明和可解释的方法。可以看到模型在做出预测时实际关注的图像部分，从而增加对模型的信任。

实验中使用 Grad-CAM 技术[19]对剪枝后的模型进行多次可视化，每次添加不同的随机噪声以观察模型对

不同扰动的反应，将原始图像与生成的 Grad-CAM 热力图叠加并进行可视化。通过叠加原始图像和 Grad-CAM 热力图，直观展示模型在处理图像时的注意力分布，有助于识别模型可能存在的偏差或问题[20-21]。图 13 和图 14 是在 MobileNetV3-Large 模型上，分别对 cifar-100 和 cifar-10 数据集进行训练，将一张照片输入进行可视化注意力机制实验结果的热力图。

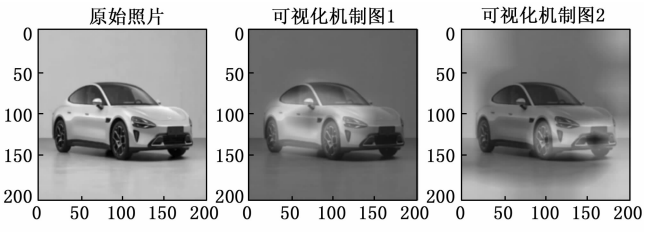

图 13　cifar-10 数据集下的模型可视化注意力机制实验

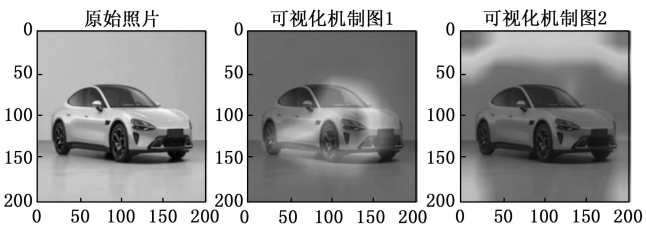

图 14　cifar-100 数据集下的模型可视化注意力机制实验

Grad-CAM 热力图显示，模型在多个扰动下对图像的不同区域进行了关注。注意力主要集中在车辆的前部和车轮区域，这些区域可能是模型用来进行分类的关键特征。热力图随着扰动变化有所不同，但总体上模型能够保持对车辆主要特征区域的关注，说明模型具有一定的鲁棒性。

3.2.4　剪枝模型训练

实验通过对 BN 层的缩放因子 γ 值进行阈值设定，对 $|\gamma|<0.1$，$|\gamma|<0.3$，$|\gamma|<0.5$ 三组剪枝进行对照试验，分析不同 γ 值下进行模型剪枝操作对模型的影响。并且将细粒度剪枝中的阈值 T 设定为 10%，皆作用于以上三组实验，这是因为在第一阶段的通道剪枝之后，模型已经损失了一部分通道，因此不能再设置太高的阈值。对 cifar-100 数据集进行不同剪枝后的数据展示，图 15 为 $|\gamma|<0.1$ 剪枝，图 16 为 $|\gamma|<0.3$ 剪枝，图 17 为 $|\gamma|<0.5$ 剪枝。

表 2 展示了不同数据集在不同程度剪枝下的效果，列出了剪枝后减少的参数量，计算量以及验证集的准确度。优化后的模型在 CIFAR-10 数据集上保持了较高的分类精度，与原模型相比 $|\gamma|<0.1$ 剪枝后的验证集精确度提升了 2.08%，但是 $|\gamma|<0.3$ 下降了 1.28%，$|\gamma|<0.5$ 下降了，$|\gamma|<0.5$ 下降到 49.4%不足原模型精度的一半；在 CIFAR-100 数据集上，剪枝和深

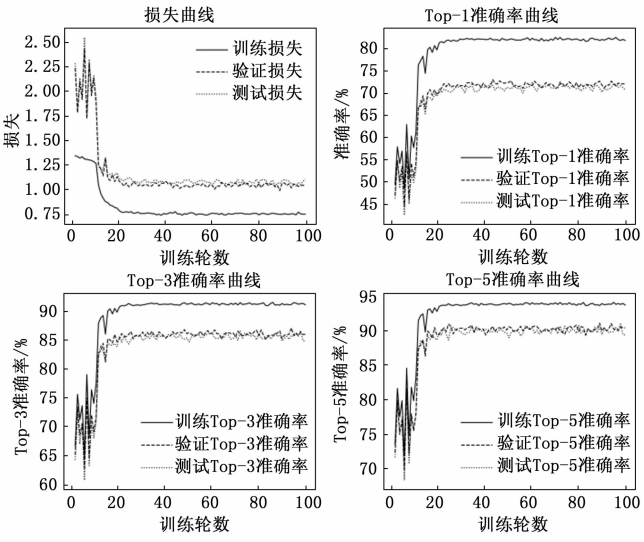

图 15　基于 cifar-100 数据集的 $|\gamma|<0.1$ 剪枝

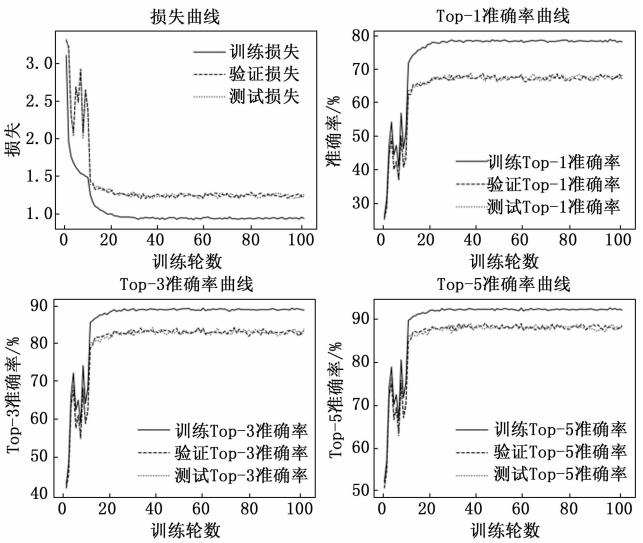

图 16　基于 cifar-100 数据集的 $|\gamma|<0.3$ 剪枝

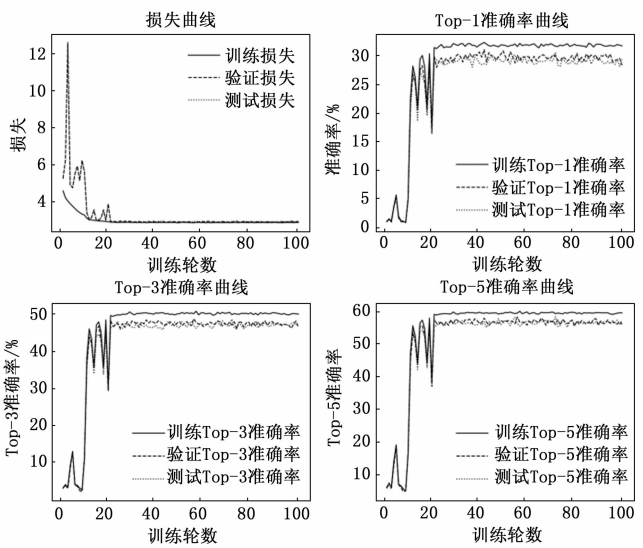

图 17　基于 cifar-100 数据集的 $|\gamma|<0.5$ 剪枝

度增强后的模型表现出更好的泛化能力，精度显著提高，$|\gamma|<0.1$ 剪枝后在原模型验证集精度的基础上提高了 8.1%，$|\gamma|<0.3$ 剪枝后也提高了 3.72%，而 $|\gamma|<0.5$ 剪枝下降了 33.44%；因此可以看出 $|\gamma|<0.3$ 和 $|\gamma|<0.5$ 剪枝属于过度剪枝。

表 2 不同数据集和剪枝阈值下的模型剪枝效果

数据集类型	剪枝阈值	剪枝掉的参数量	剪枝掉的计算量	剪枝后模型验证集准确值/%
Cifar-100	$\gamma<0.1$	2 116	2 218 260	72.80
	$\gamma<0.3$	5 068	5 420 252	68.42
	$\gamma<0.5$	7 639	8 208 951	31.26
	None	0	0	64.70
Cifar-10	$\gamma<0.1$	4 542	5 854 350	87.52
	$\gamma<0.3$	7 360	8 570 000	84.16
	$\gamma<0.5$	9 430	10 829 334	36.04
	None	0	0	85.44

4 结束语

对 MobileNetV3 模型进行了基于阈值的非结构化剪枝和深度增强策略的优化，剪枝过程结合了细粒度和粗粒度两个阶段的剪枝，并在 CIFAR-10 和 CIFAR-100 数据集上进行了验证。通过设定剪枝阈值，成功移除了网络中不重要的权重，显著减少了模型的参数量和计算量。为补偿剪枝带来的性能下降，剪枝后增加了模型深度，使模型在更深层次上学习数据特征，从而恢复并提升了性能。结果证明，基于阈值的非结构化剪枝在减少模型参数量和计算复杂度方面有效，同时结合深度增强策略，可以在降低计算负担的前提下提升模型性能。这为嵌入式设备上部署 CNN 提供有益参考。

参考文献：

[1] HOWARD A，SANDLER M，CHU G，et al. Searching for MobileNetV3 [J]. Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)，2019：1314－1324.

[2] LIU Z，LI J，SHEN Z，et al. Network slimming：Learning efficient neural networks with sparse scaling factors [J]. Proceedings of the IEEE International Conference on Computer Vision (ICCV)，2017：2736－2744.

[3] 李 屹，魏建国，刘贯伟. 模型剪枝算法综述 [J]. 计算机与现代化，2022 (9)：51－59.

[4] 刘崇阳，刘勤让. 一种神经网络模型剪枝后泛化能力的验证方法 [J]. 计算机工程，2019，45 (10)：234－238.

[5] 刘之禹，李 述，王英鹤. 基于 ZYNQ 的深度学习卷积神经网络加速平台设计 [J]. 计算机测量与控制，2022，30 (12)：264－269.

[6] HOWARD A G，ZHU M，CHEN B，et al. MobileNets：efficient convolutional neural networks for mobile vision applications [EB/OL]. ArXiv Preprint ArXiv：1704.04861，2017.

[7] MOLCHANOV P，TYREE S，KARRAS T. Pruning convolutional neural networks for resource efficient inference [EB/OL]. Arxiv Preprint Arxiv:1611.06440，2016.

[8] HE Y，ZHANG X，SUN J. Channel pruning for accelerating very deep neural networks [EB/OL]. Arxiv Preprint Arxiv：1707.06168，2017.

[9] 蒲 亮，石 毅. 基于神经网络结构搜索的卷积神经网络剪枝与压缩方法 [J]. 自动化与仪表，2023，38 (2)：15－18.

[10] 韦 越，陈世超，朱凤华，等. 基于稀疏正则化的卷积神经网络模型剪枝方法 [J]. 计算机工程，2021，47 (10)：61－66.

[11] SANDLER M，HOWARD A，ZHU M，et al. MobileNetV2：inverted residuals and linear bottlenecks [J]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)，2018：4510－4520.

[12] 李 艳，谌雨章，郭煜玮，等. 基于蓝图可分离卷积的轻量级水下图像超分辨率重建 [J]. 计算机测量与控制，2023，31 (6)：191－197.

[13] HOWARD A G，ZHU M，CHEN B，et al. MobileNets：efficient convolutional neural networks for mobile vision applications [EB/OL]. Arxiv Preprint Arxiv：1704.04861，2017.

[14] 刘 宇，雷雪梅. 融合 MobileNetV3 特征的结构化剪枝方法 [J]. 上海交通大学学报，2023，57 (9)：1203－1213.

[15] 严春满，张 翔，王青朋. 基于改进 MobileNetV2 的人脸表情识别 [J]. 计算机工程与科学，2023，45 (6)：1071－1078.

[16] 顾轶寅，王鸿奎，殷海兵. 基于上下文自适应阈值剪枝的快速依赖量化算法 [J]. 计算机工程，2023，49 (7)：143－149.

[17] YANG Z，ZHANG H. Comparative analysis of structured pruning and unstructured pruning [J]. Frontier Computing，FC 2021. Lecture Notes in Electrical Engineering，Springer，Singapore，2022，827：112.

[18] 司念文，常禾雨，张文林，等. 基于注意力机制的卷积神经网络可视化方法 [J]. 信息工程大学学报，2021，22 (3)：257－263.

[19] 朱学岩，陈锋军，郑一力，等. 融合双线性网络和注意力机制的油橄榄品种识别 [J]. 农业工程学报，2023，39 (10)：183－192.

[20] 张海刚，鲁伽祎，匡国文，等. 基于孪生网络的工业缺陷弱监督视觉检测算法 [J]. 深圳大学学报 (理工版)，2023，40 (6)：657－664.

[21] 周 扬，张瑞宾. 基于迁移学习的驾驶分心行为识别及模型解释 [J]. 科学技术与工程，2021，21 (7)：2967－2973.