Modeling and Simulation 建模与仿真, 2025, 14(1), 353-365
Published Online January 2025 in Hans. https://www.hanspub.org/journal/mos
https://doi.org/10.12677/mos.2025.141034

Hans汉斯

平衡幅度和相似度的滤波器剪枝算法

闫雅茹

上海理工大学光电信息与计算机工程学院，上海

收稿日期：2024年12月15日；录用日期：2025年1月5日；发布日期：2025年1月15日

摘　要

深度神经网络在计算机视觉任务中广泛应用，但是大规模参数计算导致的高复杂性限制了其在资源有限环境中的部署。本文提出了一种平衡幅度和相似度的滤波器剪枝方法(MASFIP)。在每次剪枝迭代中，通过缩放因子α选择每层临时剪枝的滤波器。根据网络损失进行永久性剪枝，达到预定的浮点运算量后，采取少量的再训练步骤缓解模型精度的急剧下降。在CIFAR-10和CIFAR-100数据集上对VGGNet-16和ResNet模型进行剪枝实验，结果表明在CIFAR-10数据集上，MASFIP分别从VGGNet-16和ResNet-56中删除了60.6%和52.9%的FLOPs，精度提高了0.16%和0.14%。在CIFAR-100数据集上，从ResNet-56中删除了39.1%的FLOPs，仅导致0.05%的精度下降。

关键词

滤波器剪枝，深度神经网络，幅度，相似度

Balancing Magnitude and Similarity for Filter Pruning Algorithm

Yaru Yan

School of Optical-Electrical and Computer Engineering, University of Shanghai for Science and Technology, Shanghai

Received: Dec. 15th, 2024; accepted: Jan. 5th, 2025; published: Jan. 15th, 2025

Abstract

There are extensive applications of Deep Neural Network (DNN) in the field of computer vision tasks. However, the high complexity resulting from the computing large-scale parameters of DNN would hinder its deployment in resource-constrained environments. We propose a filter pruning method, named Balancing Magnitude and Similarity for Filter Pruning (MASFIP), to address this challenge. During each pruning iteration, filters for temporary pruning are selected using a scaling

文章引用: 闫雅茹. 平衡幅度和相似度的滤波器剪枝算法[J]. 建模与仿真, 2025, 14(1): 353-365.
DOI: 10.12677/mos.2025.141034

factor α. Permanent pruning is then performed based on network loss. Upon reaching the designated floating-point operations, a small number of retraining steps are taken to alleviate the sharp decline in model accuracy. Experimental pruning on VGGNet-16 and ResNet models on CIFAR-10 and CIFAR-100 datasets reveals that, on CIFAR-10, MASFIP removes 60.6% and 52.9% of FLOPs from VGGNet-16 and ResNet-56 respectively, resulting in accuracy improvements of 0.16% and 0.14%. On CIFAR-100, pruning from ResNet-56 leads to a reduction of 39.1% in FLOPs with only a marginal accuracy drop of 0.05%.

Keywords

Filter Pruning, Deep Neural Network (DNN), Magnitude, Similarity

1. 引言

深度神经网络(Deep Neural Network, DNN)作为一种强大的模式识别工具，在诸多领域展现出了显著的性能，尤其是在计算机视觉领域中，例如图像分类[1]、人脸识别[2]、目标检测[3]和视频分析[4]等方面。然而，随着模型规模的增长，深度神经网络也带来了巨大的计算和内存需求，使得在资源受限的边缘设备上的部署变得困难。为了缓解该问题，研究者们提出了各种模型压缩技术。这些技术包括网络剪枝[5]-[7]、量化[8]、知识蒸馏[9]和低秩分解[10]等。

剪枝作为一种有效的模型压缩方法受到广泛关注，剪枝方法探索模型中的冗余权重，修剪非关键和冗余的部分。根据剪枝粒度的不同，主要分为非结构化剪枝[11]和结构化剪枝[12]。其中，滤波器剪枝[13]属于结构化剪枝[12]的重要方法，能在减少计算复杂度的同时保持较高的网络性能，得到广泛应用。此外，如何选择待剪枝的滤波器是实现剪枝模型高性能的关键。常见的剪枝方法主要有两类：基于滤波器幅度和基于相似度[13]的剪枝方法。基于幅度的剪枝方法认为，幅度较小的滤波器对网络的贡献较小，因此可以优先剪枝；而基于相似度的方法则通过分析滤波器之间的相似性，剪枝冗余的滤波器。然而，单独使用这两种方法都会存在局限性：仅考虑幅度可能移除幅度稍低但具有独特特征的滤波器；而仅基于相似度则可能移除与其他滤波器相似度高但幅度也高的滤波器。因此，本文提出了一种新的滤波器剪枝方法，通过引入一个缩放因子α，在每次剪枝过程中动态平衡幅度和相似度的影响。实验表明，本文提出的平衡幅度和相似度的滤波器剪枝方法(MASFIP)能够更有效地识别冗余滤波器，并在减少计算复杂度的同时，保持模型的高性能。

综上所述，本文的主要贡献如下：

(1) 分析了滤波器幅度和相似度两个独立准则，发现单独使用前者可能会忽略滤波器之间的特征冗余，而仅考虑相似度可能会忽略对模型性能有重要贡献的滤波器，从而提出了一种平衡幅度和相似度的滤波器剪枝方法。

(2) 本文提出的方法通过控制缩放因子α的取值范围，能够在保留关键滤波器的同时去除冗余，更有效地实现模型的压缩和加速。

(3) 在 CIFAR-10 和 CIFAR-100 基准数据集上对常用的 VGGNet 和 ResNet 模型进行了综合评估，验证了本文所提出的剪枝算法的有效性和先进性。

2. 相关工作

2.1. 结构化剪枝

在深度学习模型优化过程中，虽然非结构化剪枝[14]-[16]能够显著降低网络复杂度，但由于对专用硬件的依赖限制了其应用范围。因此，结构化剪枝[12]方法被深入研究以提升模型效率，Fang 等人[17]提出了一种通用的结构剪枝方法 DepGraph。该方法对层间依赖关系进行建模，并对耦合参数进行综合分组，执行各种神经网络模型的剪枝。Yan 等人[18]提出一种基于权重关联性的结构化剪枝方法，该方法评估滤波器的关联权重，并对评估值全局标准化，剪枝模型中评估值较小的滤波器。

结构化剪枝主要包括滤波器剪枝[19] [20]和通道剪枝[21] [22]两种形式。Lin 等人[23]提出了一种 HRank 剪枝方法，通过计算平均秩去除卷积层中不必要的滤波器，避免了传统剪枝方法中单纯依赖权重范数的问题。GFI-AP [24]根据重要性指标与不同层的其他滤波器比较进行剪枝。MSVFP [25]提出了基于幅度和相似度的可变速率滤波器剪枝方法。滤波器剪枝关注卷积层内滤波器的精简，而通道剪枝注重减少网络的层间连接密度。Jiang 等人[22]提出了一种通道剪枝方法 CPGCN，使用全局平均池化提取特征，提高模型压缩效率。

2.1.1. 基于幅度和相似度的剪枝

基于幅度的剪枝方法通过对每个滤波器进行评估，并根据其在网络中的重要性进行排序，确定哪些滤波器应当保留或剪枝。Molchanov 等人[26]提出一种基于泰勒公式展开的剪枝方法，该方法通过迭代移除对网络整体贡献较小的特征映射，高效压缩神经网络且维持其泛化能力。He 等人[27]提出了基于元属性的滤波器剪枝方法。该方法在现有基于幅度信息的剪枝准则基础上，进一步引入了考虑滤波器几何距离的新准则。此外，为了明确评估网络的状态，MFP [27]通过元属性自适应地选择最合适的剪枝标准。PFEC [28]基于滤波器的范数衡量重要性，并剪除那些权重较小的滤波器。

基于相似度的方法旨在识别神经网络中相似或冗余的滤波器。通过比较滤波器之间的相似度，确定具有类似功能或信息的滤波器，并选择性地剪除它们，以降低模型复杂度。FPSSI [29]利用结构相似性指标对滤波器进行软剪枝，有效压缩深度神经网络模型且保持精度。FPGM [30]是一种基于几何中值的滤波器剪枝方法，采用欧几里德距离剪枝冗余滤波器，并考虑滤波器之间的相互关系。

3. 滤波器剪枝算法

首先，在本节中介绍一些符号和注释。网络包含 L 个卷积层，每一层都由多个滤波器组成，定义所有滤波器的集合为 F。其中 F_i 表示第 i 层滤波器。F_i^j 表示第 i 层第 j 个滤波器。将其参数化为 $W_i \in R^{N_{i+1}\times N_i\times K\times K}$，$1\le i\le L$，其中 N_i 为输入通道数，N_{i+1} 为输出通道数，K 为卷积核大小。为了简单起见，将第 i 层中的所有 3-D 滤波器 F_i^j 表示为一维向量 $\boldsymbol{S}_i \in R^{N_{i+1}\times M_i}$，这表示在卷积层 i 中有 N_{i+1} 个一维滤波向量，每个向量的长度为 $M_i = N_i \times K \times K$。

3.1. 滤波器评估准则

3.1.1. 幅度计算

在神经网络的滤波器剪枝中，通常采用 L_p 范数评估滤波器的重要性，其中 L1 范数[24]和 L2 范数是常用的幅度度量准则。如果将第 i 层中的第 j 个滤波器 S_i^j 表示为 $\boldsymbol{s} \in R^{1\times M_i}$，则其 L_p 范数计算为：

$$\left\|S_i^j\right\|_p = \left(\sum_{m=1}^{M_i}\left|S_m\right|^p\right)^{\frac{1}{p}} \tag{1}$$

3.1.2. 相似度计算

常用的相似度度量包括欧几里德距离[26]和余弦相似度。欧几里德距离衡量了两个向量之间的差异程度，距离越小则两个向量越相似。余弦相似度表示两个向量夹角的余弦值，范围是[-1,1]，值越接近 1 则表示两个向量越相似，而值为−1 则表示它们完全相反。对于两个滤波向量 $\boldsymbol{u}\in R^{1\times M_i}$ 和 $\boldsymbol{v}\in R^{1\times M_i}$，它们之间的欧几里德距离计算为：

它们之间的欧几里德距离计算为：

$$D_{eucl}\left(\boldsymbol{u},\boldsymbol{v}\right)=\sqrt{\sum_{m=1}^{M_i}\left|u_m-v_m\right|^2} \tag{2}$$

余弦相似度的计算公式如下：

$$D_{cos}\left(\boldsymbol{u},\boldsymbol{v}\right)=1-\frac{\sum_{m=1}^{M_i}\left(u_m\times v_m\right)}{\sum_{m=1}^{M_i}u_m^2\times\sum_{m=1}^{M_i}v_m^2} \tag{3}$$

两个滤波器在同一层中的相似性可以通过公式(2)和公式(3)进行量化。为了进一步确定某个滤波器相对于该层其它滤波器的整体相似性得分，需要将目标滤波器与该层中其他滤波器的相似性值取平均，作为该滤波器的最终相似性得分。具体而言，第 i 个卷积层中第 j 个滤波器的相似性得分可以表示为：

$$S\left(F_i^j\right)=\frac{\sum_{k=1,k\neq j}^{N_{i+1}}D\left(F_i^j,F_i^k\right)}{N_{i+1}-1} \tag{4}$$

3.2. MASFIP 过程

MASFIP 剪枝过程主要包括以下步骤：(1) 初始化网络；(2) 对原始网络进行指定次数(t_p)的训练，以充分优化网络权重并确定每层的步剪枝率；(3) 检查每一层并根据剪枝排序选择临时剪枝的滤波器集合，如果当前剪枝操作后的模型损失小于之前记录的最小损失，则更新并记录该层的剪枝信息；(4) 选出精度损失最小的模型进行永久性剪枝，之后进行少量轮次的训练，以恢复模型精度；(5) 重复上述步骤，直到达到目标剪枝率；(6) 微调，直至模型收敛。剪枝过程如图 1 所示。

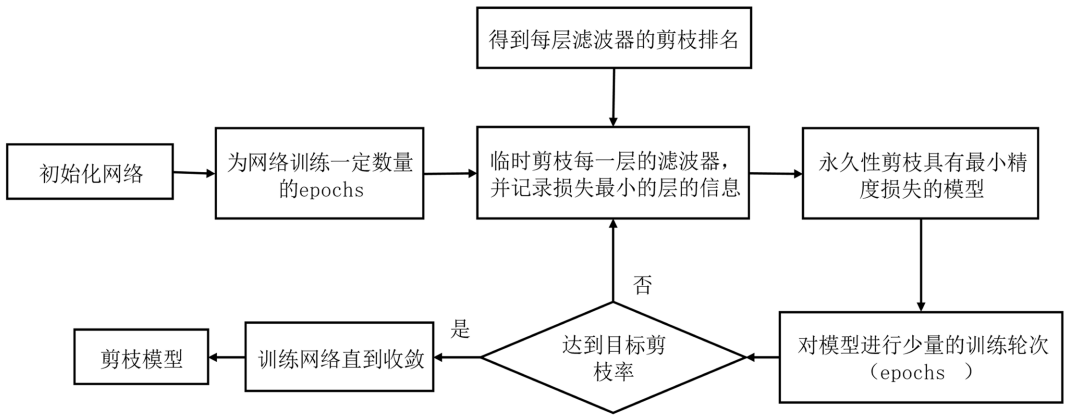

Figure 1. Pruning process flowchart
图 1. 剪枝过程流程图

3.2.1. 剪枝排序

在结构化剪枝过程中，确定合适的剪枝顺序至关重要。显然，对于幅度小且相似度高的滤波器，应

优先进行剪枝，而对于幅度较大而相似度低的滤波器，应优先保留。MSVFP [25]指出幅度信息在滤波器选择的过程中起到至关重要的作用。即使在使用相似度作为衡量标准，滤波器的幅度信息仍然是一个重要因素。根据前人的经验[25]及实验结果，对于较低幅度且相似度较低的滤波器，以及较高幅度且相似较高的滤波器，剪枝时应优先考虑前者，即使前者提供的特征在某种程度上是独特的。这一步主要是为了减少模型的复杂度，同时由于这些滤波器幅度较小，对模型整体性能的影响有限。下一步，可以考虑剪枝后者。尽管这些滤波器较为重要，但因其信息已被其他滤波器涵盖，因此剪枝这些滤波器可以在保持模型性能的同时降低冗余。

为了量化上述剪枝顺序，引入了一个函数，用于评估每个滤波器的剪枝优先级。将公式(1)和公式(4)进行组合，并加入了一个缩放因子 α，得到公式为：

$$rank\left(F_i\right)=w\left(F_i\right)+\alpha\cdot s\left(F_i\right) \tag{5}$$

其中 $w\left(F_i\right)$ 表示第 i 层滤波器的幅度；$s\left(F_i\right)$ 表示第 i 层滤波器的相似度；$rank\left(F_i\right)$ 表示第 i 层滤波器的剪枝排序。实验表明，α 取值范围是[0.2,0.8]。$w\left(F_i\right)$ 和 $s\left(F_i\right)$ 均为归一化后的内容。每层滤波器的剪枝排序算法如表 1 所示。

Table 1. Pruning ordering algorithm for each filter layer
表 1. 每层滤波器的剪枝排序算法

输入：初始化模型 M，步剪枝率 $E=\left\{E_i,1\le i\le L\right\}$
输出：候选剪枝模型 $M^*=\left\{M_i,1\le i\le L\right\}$

for $i=1,\cdots,L$ **do**
calculate the priority F_i using Eq. (5) and sort them in ascending order;
$M_i\leftarrow$prune the filters ranked lower based on the pruning rate E;
end for
record candidate model information;

3.2.2. 最小精度损失

通过使用一个小的随机抽样训练数据集进行损失估计，以最小化计算成本[13]。在给定约束条件 P 的情况下，网络剪枝问题可以表示为一个优化问题。即在满足某些约束条件的情况下，寻找最小化损失函数的模型参数。优化问题可以表示为以下公式：

$$\arg\min\left(Loss\left(W',D\right)\right)\quad \text{s.t. } R\left(f\left(W',a_i\right)\right)\le P \tag{6}$$

其中 $D=\left\{\left(a_i,b_i\right)\right\}_{i=1}^{S}$ 代表训练数据集，其中 S 是训练数据记录的总数。$Loss\left(W',D\right)$ 是剪枝后网络在数据集 D 上的损失函数，$f()$是网络函数，它根据输入 a_i 和参数 W' 产生输出，$R()$是一个映射函数，在文章中表示网络的 FLOPs，P 是目标 FLOPs 剪枝率。被剪枝的网络 W' 相对于原始网络 W 的剪枝速率 P' 可以定义为：

$$P'=1-\frac{R\left(W'\right)}{R\left(W\right)} \tag{7}$$

此外，移除卷积层的滤波器会对网络产生深远影响，不仅会影响相邻的前后层，还可能影响到其它与该层存在连接的部分。为了最大化减少网络损失，采用分层探索步骤[13]，通过迭代地为每一层分配适当的剪枝比例，从而保证每次迭代中 FLOPs 的减少保持近似恒定。通过采用这种方式，可以实现各层剪

枝比例的平衡，从而有效控制剪枝过程中网络性能的下降。

假设W_{-i}表示不包括第 i 层参数的网络权重。若将整个剪枝过程中每次搜索的 FLOPs 减少率设定为 p_s，则第 i 层的探索步长 $E(i)$ 可以近似表示为：

$$E(i)=\max\left(1,\frac{p_s\times R(W)\times N_{i+1}}{R(W)-R(W_{-i})}\right) \tag{8}$$

首先，对原始网络进行指定次数的训练。随后，根据剪枝过程中每次搜索的网络 FLOPs 减少率 p_s，为每层计算步剪枝率，每层的步剪枝率可表示为：

$$E(\cdot)=\{E(1),E(2),\cdots,E(L)\} \tag{9}$$

通过采用一个迭代剪枝方法[13]，使用滤波器幅度或相似度方法来指导剪枝过程。每次迭代对每层进行临时剪枝，得到候选模型。此外，去除每层中低排序的滤波器以计算网络损失。通过比较每一层剪枝后的损失，找到最小精度损失的模型，并进行永久性的滤波器剪枝。MASFIP 算法如表 2 所示。

Table 2. MASFIP pseudo-code
表 2. MASFIP 伪代码

输入：初始化模型 M，训练数据 D，缩放因子 α，目标 FLOPs 减少率 P，微调 FLOPs 减少间隔 p，剪枝前的训练周期数量 t_p
输出：剪枝模型 M'

$P'\leftarrow 0$； $M'\leftarrow M$； $P'_t\leftarrow 0$；
randomly sample dataset D' from D;
training initial model M for t_p epochs;
for $i=1,\cdots,L$ **do**
    Calculate the pruning rate E_i;
end for
while $P'<P$ **do**
    obtain M^* using algorithm on Table 1.;
    $M'\leftarrow \arg\min(Loss(D'\mid M_i)),1\le i\le L$;
    calculate P' using Eq. (7);
    if $P'-P'_t\ge p$ **then**
        fine-tune pruned model M';
        $P'_t\leftarrow P'$;
    end if
end while
train pruned model M' until convergence;

4. 实验与结果分析

4.1. 实验环境

CIFAR-10 由 60,000 张 32×32 像素的彩色图像组成，总共分为 10 个不同的类别，每个类别包含 6000 张图像。其中 5000 张图像用于训练，1000 张用于测试。CIFAR-100 则分为 100 个类别，每个类别包含 600 张图像。在训练 VGGNet-16 和 ResNet 时，分别应用了与 PFEC [28]、FPSSI [29]和 LAASP [13]中相同的训练设置。算法遵循一种边训练边剪枝[13]的方法，即对原始网络进行一定次数的初始训练，然后暂停训练过程执行剪枝操作。在每次剪枝迭代结束时，将被剪枝的滤波器永久移除。剪枝完成后，继续对

剪枝后的网络进行与原网络相同的训练。在每次剪枝迭代中，每轮设定 FLOPs 减少率约为 1%。在剪枝过程中，每次如果导致网络 FLOPs 减少率大于 3% ($\beta_p = 0.03$)，则需要对剪枝后的网络进行 1~3 轮的微调训练，以恢复模型精度。

4.2. 实验分析

4.2.1. VGGNet 在 CIFAR-10 上的实验结果

表 3 显示了在 CIFAR-10 数据集上修剪 VGGNet-16 的结果。原始网络进行 30 轮训练[13]，随后进行剪枝。为了与其他具有可比剪枝率的方法进行比较，首先对网络进行迭代剪枝，使 FLOPs 减少 34.3%。此外，使用所提出的剪枝技术，网络 FLOPs 可以减少 60%以上，甚至在某些情况下达到比原始未修剪网络更高的精度。

Table 3. The pruning results of VGGNet-16 on CIFAR-10
表 3. VGGNet-16 对 CIFAR-10 的剪枝结果

剪枝算法	是否预训练	剪枝前准确率(%)	剪枝后准确率(%)	准确率减少比例(%)	FLOPs 减少比例(%)
PFEC [28]	是	93.58 ± 0.03	93.31 ± 0.02	0.27	34.2
MFP [27]	否	93.58 ± 0.03	93.54 ± 0.03	0.04	34.2
FPGM [30]	否	93.58 ± 0.03	93.54 ± 0.08	−0.04	34.2
LAASP [13]	否	93.79 ± 0.23	93.90 ± 0.16	−0.11	34.6
MASFIP	否	93.79 ± 0.23	**94.21** ± 0.03	**−0.42**	34.3
Hrank [23]	是	93.96	93.43	0.53	53.5
CPGCN [22]	是	93.2	93.53	−0.51	57.3
LAASP [13]	否	93.79 ± 0.23	93.79 ± 0.11	0	60.5
MASFIP	否	93.79 ± 0.23	**93.95** ± 0.02	**−0.16**	60.6

在表 3 中给出了预训练网络在剪枝前的基线精度。从表 3 中可以看出，在 VGGNet-16 上分别减少了 34.3%和 60.6%的 FLOPs，剪枝后模型的 top-1 精度相比基线分别提高了 0.42%和 0.16%。当 FLOPs 减少率为 34.3%时，本文方法的准确率达到了 94.21%，明显优于其他方法。

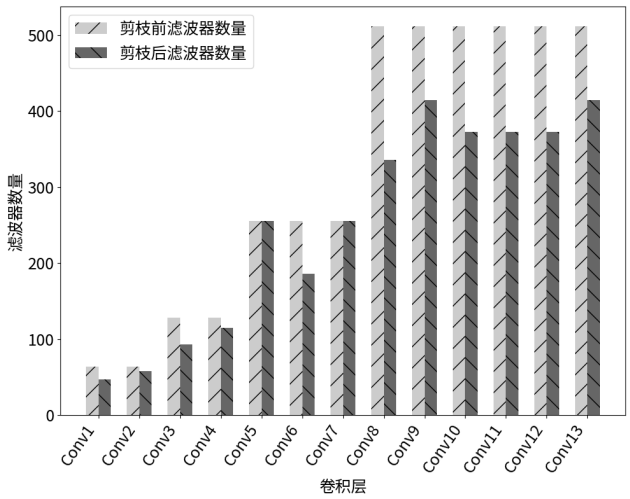

Figure 2. Comparison of filters before and after pruning
图 2. VGGNet-16 模型修剪前后各层滤波器的对比

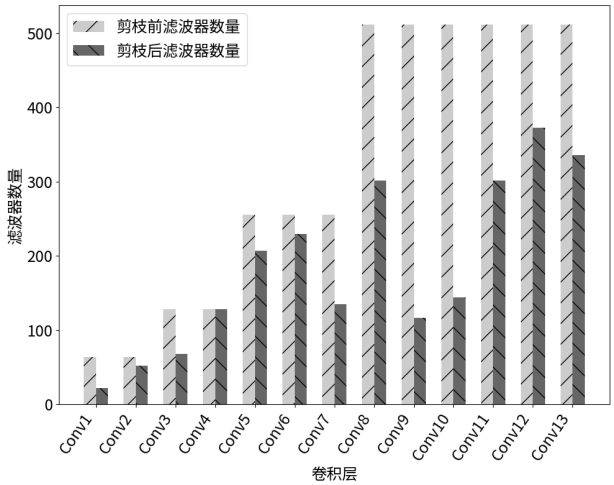

Figure 3. Comparison of filters before and after pruning
图 3. VGGNet-16 模型修剪前后各层滤波器的对比

图 2 和图 3 显示了在 CIFAR-10 数据集上，当 FLOPs 分别减少 34.3%和 60.6%时，VGGNet-16 模型剪枝前后的滤波器数量。可以观察到：随着卷积层在整个网络结构中的层次逐渐加深，滤波器的剪枝比例显著增加，剪枝掉的滤波器数量多于靠近输入端的卷积层。这意味着在 VGGNet-16 模型的深层部分的特征冗余性较高，剪枝它们不会对模型的整体性能产生重大影响，因而具有更高可剪枝性。

4.2.2. ResNet 在 CIFAR-10 上的实验结果

如表 4 所示，我们在 ResNet-32、ResNet-56 和 ResNet-110 模型上使用 CIFAR-10 数据集进行实验。结果显示，本文方法在实现相似 FLOPs 减少率时，取得了更高的 top-1 精度。此外，本文针对不同的 FLOPs 减少率进行了多次实验，以验证其有效性。SFP [12]、MPF [27]和 FPGM [30]等方法采用软滤波剪枝技术，其特点是在每个训练周期结束时进行剪枝，并在下一个训练周期中更新剪枝后的滤波器，从而保持网络结构的一致性。相比之下，采用硬滤波器剪枝[13]在每次剪枝迭代中永久移除被剪枝的滤波器，并重新组织网络连接，以提升计算效率。硬剪枝的优点是剪枝完成后，精简后的模型可以继续训练直至收敛，从而进一步减少计算负担。由于本文重点关注无需预训练的剪枝方法结果，为了展示该方法的竞争力，我们还列出了一些其它常用方法，如 Rethink [19]、GFI-AP [24]、NPPM [21]、Hrank [23]和 MSVFP [25]。

如表 4 所示，在 ResNet-56 模型上尽管已经使用了硬滤波剪枝，FLOPs 的减少率仍然达到了 52.9%。然而，本文所提出的剪枝方法比基线模型的精度提高了 0.14%。在 CIFAR-10 上，MASFIP 从 ResNet-32 和 ResNet-110 模型中分别删除了 53.2%和 52.1%的 FLOPs 操作，但是精度仅比基线降低了 0.19%和 0.15%。

Table 4. The pruning results of ResNet on CIFAR-10
表 4. ResNet 对 CIFAR-10 的剪枝结果

ResNet 模型	剪枝算法	是否预训练	剪枝前准确率(%)	剪枝后准确率(%)	准确率减少比例(%)	FLOPs 减少比例(%)
32	SFP [12]	否	92.63 ± 0.70	92.08 ± 0.08	0.55	41.5
	GFI-AP [24]	是	92.54	92.09 ± 0.15	0.45	42.5
	FPGM [30]	否	92.63 ± 0.70	91.93 ± 0.03	0.7	53.2
	MFP [27]	否	92.63 ± 0.70	91.85 ± 0.09	0.78	53.2
	LAASP [13]	否	93.12 ± 0.04	92.64 ± 0.09	0.48	53.3
	MASFIP	否	93.12 ± 0.04	**92.93** ± 0.05	**0.19**	53.2

续表

56	HRank [23]	是	93.26	93.17	0.09	50
	NPPM [21]	是	93.04	93.4	−0.36	50
	SFP [12]	否	93.59 ± 0.58	92.26 ± 0.31	1.33	52.6
	FPGM [30]	否	93.59 ± 0.58	92.93 ± 0.49	0.66	52.6
	MFP [27]	否	93.59 ± 0.58	92.76 ± 0.03	0.83	52.6
	DepGraph [17]	是	93.53	93.46	0.07	52.6
	LAASP [13]	否	93.61 ± 0.11	93.49 ± 0.00	0.12	52.6
	MASFIP	否	93.61 ± 0.11	**93.75** ± 0.05	**−0.14**	52.9
110	PFEC [28]	是	93.53	93.3	0.23	38.6
	SFP [12]	否	93.68 ± 0.32	93.38 ± 0.30	0.3	40.8
	Rethink [19]	是	93.77 ± 0.23	93.70 ± 0.16	0.07	40.8
	FPGM [30]	否	93.68 ± 0.32	93.73 ± 0.23	−0.05	52.3
	MFP [27]	否	93.68 ± 0.32	93.69 ± 0.31	−0.01	52.3
	MSVFP [25]	是	93.69 ± 0.22	93.92 ± 0.52	−0.23	52.4
	LAASP [13]	否	94.41 ± 0.07	94.17 ± 0.16	0.24	52.5
	MASFIP	否	94.41 ± 0.07	94.26 ± 0.10	0.15	52.1

4.2.3. 在 CIFAR-100 上的实验结果

实验使用 CIFAR-100 数据集，在 VGGNet-16、ResNet-56 和 ResNet-110 模型上测试剪枝方法。实验使用 CIFAR-100 数据集，在 VGGNet-16、ResNet-56 和 ResNet-110 模型上测试剪枝方法。如表 5 所示，在相似剪枝率的条件下，本文方法在所有列出的方法中均表现出更优的性能。例如，当 FLOPs 减少率达到 37.5%时，本文方法在 VGGNet-16 模型上实现了比基线模型更高的精度，提升了 0.02%。此外，在 ResNet-56 模型上，当 FLOPs 减少率达到 39.1%时，精度仅下降了 0.05%。

Table 5. The pruning results of VGGNet and ResNet on CIFAR-100
表 5. VGGNet 和 ResNet 对 CIFAR-100 的剪枝结果

模型	剪枝算法	是否预训练	剪枝前准确率(%)	剪枝后准确率(%)	准确率减少比例(%)	FLOPs 减少比例(%)
VGG-16	PFEC [28]	是	73.01	71.11	1.90	34.2
	CPGMI [31]	是	73.26	73.53	−0.27	37.1
	NS [32]	否	73.26	73.48	−0.22	37.1
	MASFIP	否	73.71	**73.73**	**−0.02**	37.5
ResNet-32	PFEC [28]	是	70.38	70.42	−0.04	10.4
	MASFIP	否	71.94	**72.11**	**−0.17**	12.8
	PFEC [28]	是	70.38	69.95	0.43	27.6
	MIL [33]	是	71.33	68.37	2.96	39.3
	SDN [34]	否	70.01	69.78	0.23	38.3
	MASFIP	否	71.94	**71.89**	**0.05**	39.1
ResNet-110	PFEC [28]	是	72.92	70.88	1.41	27.6
	MIL [33]	是	72.79	70.78	2.01	31.3
	MASFIP	否	73.71	73.13	0.58	34.6

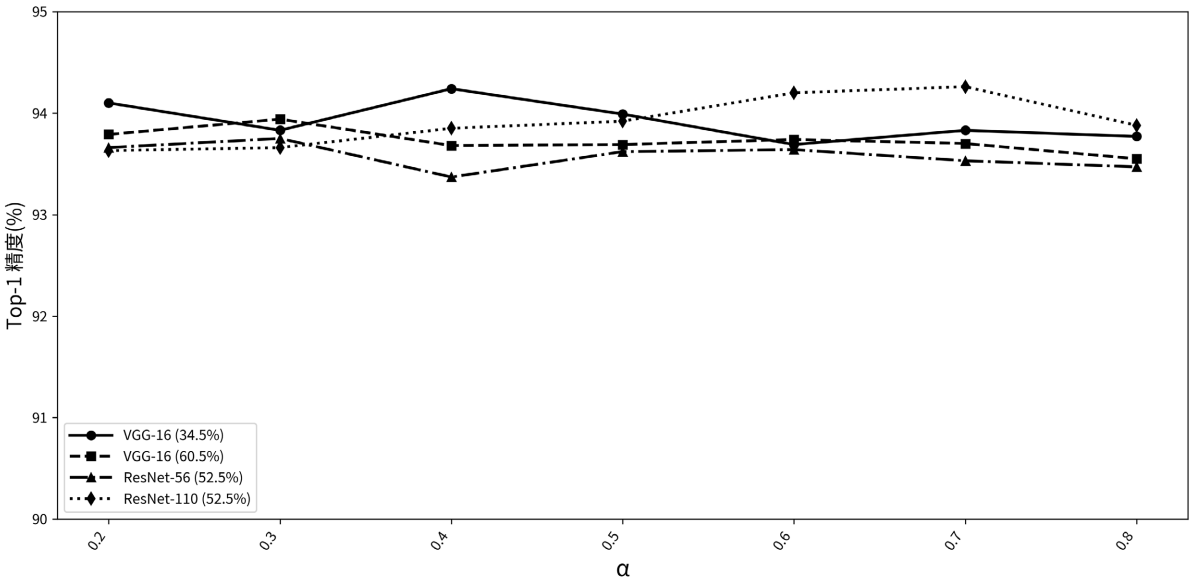

Figure 4. Effect of different α on model accuracy
图 4. 不同 α 对模型精度的影响

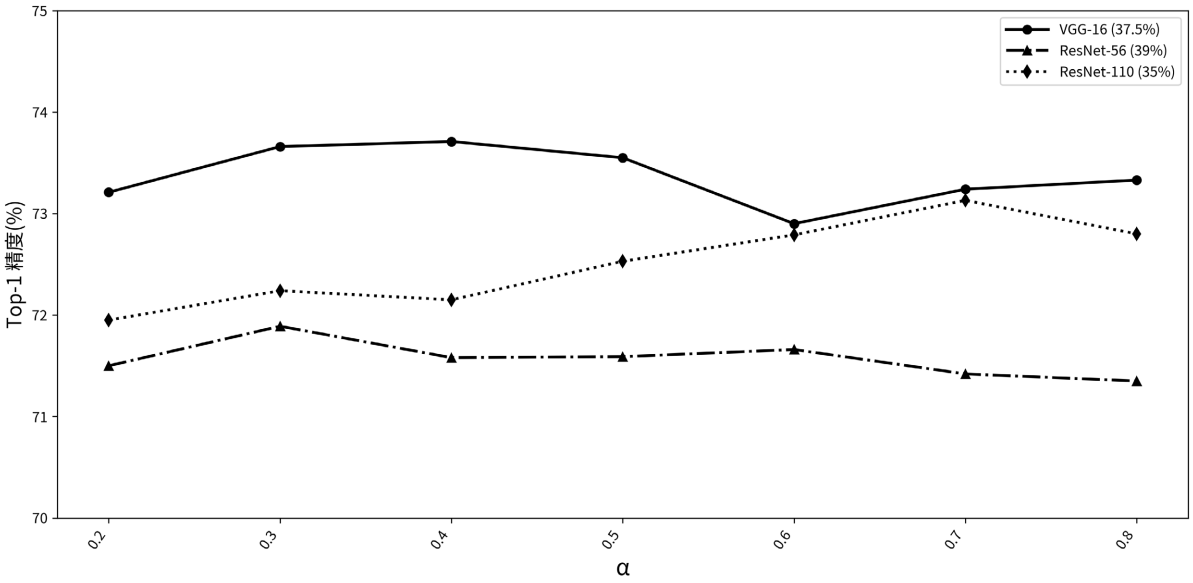

Figure 5. Effect of different α on model accuracy
图 5. 不同 α 对模型精度的影响

如图 4 和图 5 所示，为了验证公式(5)中缩放因子 α 对模型剪枝的影响。本文在 CIFAR-10 和 CIFAR-100 数据集上，针对 VGGNet-16 和 ResNet 模型，选择了多个不同的 α 值进行剪枝实验。起始时，α 设为 0.2，并以 0.1 的增量逐步增加。实验结果表明，对于 ResNet-56 模型，α 值在 0.3 左右时即可获得较好的精度，表明网络中可能存在较多的低幅度滤波器。相比之下，ResNet-110 模型在 α 值为 0.7 左右时表现最佳，表明相似度在滤波器选择中的作用增大，这也暗示该模型中存在较多冗余的滤波器，因此需要更高的 α 值以增强相似度的影响，进而更有效地去除冗余的滤波器。因此，在进行剪枝时，通过选择选择 α 值，可以显著提升模型精度。

在 CIFAR-10 和 CIFAR-100 数据集上，MASFIP 剪枝效果差异较大。CIFAR-100 由于包含更多类别，模型需要学习并区分更多的特征，在剪枝过程中移除滤波器对模型性能的影响更为显著。而相较之下，CIFAR-10 数据集的类别较少，剪枝对精度的影响较小。因此，CIFAR-100 数据集上的剪枝精度变化更大，剪枝率也相对较低。

5. 结束语

本文提出了一种新的滤波器剪枝方法，用于加速卷积神经网络的推理过程。通过平衡滤波器的幅度和相似度，首先筛选出候选滤波器集。在此基础上，记录剪枝后的精度损失，选择在精度损失最小的卷积层中进行永久性滤波器剪枝，并通过多次迭代剪枝，直至达到预定剪枝率。剪枝完成后，对模型进行训练，直到收敛。大量实验结果表明，MASFIP 在不同数据集及主流网络模型上，剪枝效果显著优于现有方法，例如，在 CIFAR-10 上，从 VGGNet-16 中删除了 60.6%FLOPs，精度提高了 0.16%。

未来的工作将致力于评估 MASFIP 方法在更复杂的神经网络架构和应用场景(如目标检测、图像分割等)中的有效性。除此之外，我们还计划将本文方法与其他模型压缩方法(参数量化[8]、权重剪枝[15]等)结合，进一步提高压缩效率。

参考文献

[1] Huang, S., Pareek, A., Jensen, M., Lungren, M.P., Yeung, S. and Chaudhari, A.S. (2023) Self-Supervised Learning for Medical Image Classification: A Systematic Review and Implementation Guidelines. *NPJ Digital Medicine*, **6**, Article No. 74. https://doi.org/10.1038/s41746-023-00811-0

[2] Liu, Z., Gu, C., Xie, Y., Zhang, H. and Pei, S. (2023) Realistic Sketch Face Generation via Sketch-Guided Incomplete Restoration. 2023 *IEEE 29th International Conference on Parallel and Distributed Systems* (*ICPADS*), Ocean Flower Island, 17-21 December 2023, 32-37. https://doi.org/10.1109/icpads60453.2023.00014

[3] Amit, Y., Felzenszwalb, P. and Girshick, R. (2021) Object Detection. In: *Computer Vision*, Springer, 875-883. https://doi.org/10.1007/978-3-030-63416-2_660

[4] Liu, W., Kang, G., Huang, P., Chang, X., Yu, L., Qian, Y., *et al*. (2020) Argus: Efficient Activity Detection System for Extended Video Analysis. 2020 *IEEE Winter Applications of Computer Vision Workshops* (*WACVW*), Snowmass, 1-5 March 2020, 126-133. https://doi.org/10.1109/wacvw50321.2020.9096929

[5] Pei, S., Wu, Y. and Qiu, M. (2020) Neural Network Compression and Acceleration by Federated Pruning. In: *Lecture Notes in Computer Science*, Springer, 173-183. https://doi.org/10.1007/978-3-030-60239-0_12

[6] Pei, S., Luo, J. and Liang, S. (2022) DRP: Discrete Rank Pruning for Neural Network. In: *Lecture Notes in Computer Science*, Springer, 168-179. https://doi.org/10.1007/978-3-031-21395-3_16

[7] 陈胤杰, 裴颂文. 面向 FPGA 的二值神经网络模型压缩方法研究[J]. 小型微型计算机系统, 2024, 45(6): 1356-1362.

[8] Shang, Y., Yuan, Z., Xie, B., Wu, B. and Yan, Y. (2023) Post-Training Quantization on Diffusion Models. 2023 *IEEE/CVF Conference on Computer Vision and Pattern Recognition* (*CVPR*), Vancouver, 17-24 June 2023, 1972-1981. https://doi.org/10.1109/cvpr52729.2023.00196

[9] Phuong, M. and Lampert, C. (2019) Towards Understanding Knowledge Distillation. *International Conference on Machine Learning*, Long Beach, 9-15 June 2019, 5142-5151.

[10] Phan, A., Sobolev, K., Sozykin, K., Ermilov, D., Gusak, J., Tichavský, P., *et al*. (2020) Stable Low-Rank Tensor Decomposition for Compression of Convolutional Neural Network. In: *Lecture Notes in Computer Science*, Springer, 522-539. https://doi.org/10.1007/978-3-030-58526-6_31

[11] Zhang, T., Ye, S., Zhang, K., Tang, J., Wen, W., Fardad, M., *et al*. (2018) A Systematic DNN Weight Pruning Framework Using Alternating Direction Method of Multipliers. In: *Lecture Notes in Computer Science*, Springer, 191-207. https://doi.org/10.1007/978-3-030-01237-3_12

[12] He, Y., Kang, G., Dong, X., Fu, Y. and Yang, Y. (2018) Soft Filter Pruning for Accelerating Deep Convolutional Neural Networks. *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, Stockholm, 13-19 July 2018, 2234-2240. https://doi.org/10.24963/ijcai.2018/309

[13] Ghimire, D., Lee, K. and Kim, S. (2023) Loss-Aware Automatic Selection of Structured Pruning Criteria for Deep Neural

Network Acceleration. *Image and Vision Computing*, **136**, Article 104745. https://doi.org/10.1016/j.imavis.2023.104745

[14] Shi, X., Ding, J., Hao, Z., *et al*. (2024) Towards Energy Efficient Spiking Neural Networks: An Unstructured Pruning Framework. *The* 12*th International Conference on Learning Representations*, Vienna Austria, 7-11 May 2024, 1-12.

[15] Liao, Z., Quétu, V., Nguyen, V. and Tartaglione, E. (2023) Can Unstructured Pruning Reduce the Depth in Deep Neural Networks? 2023 *IEEE*/*CVF International Conference on Computer Vision Workshops* (*ICCVW*), Paris, 2-3 October 2023, 1402-1406. https://doi.org/10.1109/iccvw60793.2023.00151

[16] Wang, H. and Zhang, W. (2024) Unstructured Pruning and Low Rank Factorisation of Self-Supervised Pre-Trained Speech Models. *IEEE Journal of Selected Topics in Signal Processing*, 1-14. https://doi.org/10.1109/jstsp.2024.3433616

[17] Fang, G., Ma, X., Song, M., Bi Mi, M. and Wang, X. (2023) Depgraph: Towards Any Structural Pruning. 2023 *IEEE*/*CVF Conference on Computer Vision and Pattern Recognition* (*CVPR*), Vancouver, 17-24 June 2023, 16091-16101. https://doi.org/10.1109/cvpr52729.2023.01544

[18] Yan, Y.-C., Guo, R.-Z. and Yang, J.-X. (2021) Model Pruning Based on Weight Dependency for Convolutional Neural Network. *Journal of Chinese Computer Systems*, **42**, 1500-1504.

[19] Liu, Z., Sun, M., Zhou, T., *et al*. (2019) Rethinking the Value of Network Pruning. *International Conference on Learning Representations*, United States, 6-9 May 2019, 113-133.

[20] Luo, J., Wu, J. and Lin, W. (2017). ThiNet: A Filter Level Pruning Method for Deep Neural Network Compression. 2017 *IEEE International Conference on Computer Vision* (*ICCV*), Venice, 22-29 October 2017, 5058-5066. https://doi.org/10.1109/iccv.2017.541

[21] Gao, S., Huang, F., Cai, W. and Huang, H. (2021) Network Pruning via Performance Maximization. 2021 *IEEE*/*CVF Conference on Computer Vision and Pattern Recognition* (*CVPR*), Nashville, 20-25 June 2021, 9266-9276. https://doi.org/10.1109/cvpr46437.2021.00915

[22] Jiang, D., Cao, Y. and Yang, Q. (2022) On the Channel Pruning Using Graph Convolution Network for Convolutional Neural Network Acceleration. *Proceedings of the* 31*st International Joint Conference on Artificial Intelligence*, Vienna, 23-29 July 2022, 3107-3113. https://doi.org/10.24963/ijcai.2022/431

[23] Lin, M., Ji, R., Wang, Y., Zhang, Y., Zhang, B., Tian, Y., *et al*. (2020) HRank: Filter Pruning Using High-Rank Feature Map. 2020 *IEEE*/*CVF Conference on Computer Vision and Pattern Recognition* (*CVPR*), Seattle, 13-19 June 2020, 1526-1535. https://doi.org/10.1109/cvpr42600.2020.00160

[24] Mondal, M., Das, B., Roy, S.D., Singh, P., Lall, B. and Joshi, S.D. (2022) Adaptive CNN Filter Pruning Using Global Importance Metric. *Computer Vision and Image Understanding*, **222**, Article 103511. https://doi.org/10.1016/j.cviu.2022.103511

[25] Ghimire, D. and Kim, S. (2022) Magnitude and Similarity Based Variable Rate Filter Pruning for Efficient Convolution Neural Networks. *Applied Sciences*, **13**, Article 316. https://doi.org/10.3390/app13010316

[26] Molchanov, P., Tyree, S., Karras, T., *et al*. (2017) Pruning Convolutional Neural Networks for Resource Efficient Inference. *International Conference on Learning Representations*, Toulon, 24-26 April 2017, 30-46.

[27] He, Y., Liu, P., Zhu, L. and Yang, Y. (2023) Filter Pruning by Switching to Neighboring CNNs with Good Attributes. *IEEE Transactions on Neural Networks and Learning Systems*, **34**, 8044-8056. https://doi.org/10.1109/tnnls.2022.3149332

[28] Li, H., Kadav, A., Durdanovic, I., *et al*. (2017) Pruning Filters for Efficient ConvNets. *International Conference on Learning Representations*, Toulon, 24-26 April 2017, 1683-1695.

[29] Zhu, J. and Pei, J. (2019) Filter Pruning via Structural Similarity Index for Deep Convolutional Neural Networks Acceleration. 2019 *IEEE* 14*th International Conference on Intelligent Systems and Knowledge Engineering* (*ISKE*), Dalian, 14-16 November 2019, 730-734. https://doi.org/10.1109/iske47853.2019.9170362

[30] He, Y., Liu, P., Wang, Z., Hu, Z. and Yang, Y. (2019) Filter Pruning via Geometric Median for Deep Convolutional Neural Networks Acceleration. 2019 *IEEE*/*CVF Conference on Computer Vision and Pattern Recognition* (*CVPR*), Long Beach, 15-20 June 2019, 4335-4344. https://doi.org/10.1109/cvpr.2019.00447

[31] Lee, M.K., Lee, S., Lee, S.H. and Song, B.C. (2020) Channel Pruning via Gradient of Mutual Information for Light-Weight Convolutional Neural Networks. 2020 *IEEE International Conference on Image Processing* (*ICIP*), Abu Dhabi, 25-28 October 2020, 1751-1755. https://doi.org/10.1109/icip40778.2020.9190803

[32] Liu, Z., Li, J., Shen, Z., Huang, G., Yan, S. and Zhang, C. (2017) Learning Efficient Convolutional Networks through Network Slimming. 2017 *IEEE International Conference on Computer Vision* (*ICCV*), Venice, 22-29 October 2017, 2755-2763. https://doi.org/10.1109/iccv.2017.298

[33] Dong, X., Huang, J., Yang, Y. and Yan, S. (2017) More Is Less: A More Complicated Network with Less Inference Complexity. 2017 *IEEE Conference on Computer Vision and Pattern Recognition* (*CVPR*), Honolulu, 21-26 July 2017,

1895-1903. https://doi.org/10.1109/cvpr.2017.205

[34] Chen, S. and Zhao, Q. (2019) Shallowing Deep Networks: Layer-Wise Pruning Based on Feature Representations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **41**, 3048-3056. https://doi.org/10.1109/tpami.2018.2874634