基于模型性能相关性的分级剪枝率剪枝方法

杨 康[1]，郭荣佐[1+]，李 超[2]，许建荣[3]，严阳春[1]，宫禄齐[2]

（1. 四川师范大学 计算机科学学院，四川 成都 610101；2. 中国科学院计算技术研究所，北京 100190；
3. 北京工业大学 信息学部光电子技术教育部重点实验室，北京 100124）

摘 要：目前剪枝方法中还没有对信息量分布不均的神经网络层做不同剪枝率处理的方法，为此提出一种对不同网络层剪枝不同比例的方式。逐层恢复已经被剪枝神经网络模型的各层，得到各层与模型性能的相关性，对神经网络层进行分类级，对不同类级的神经网络标定不同剪枝率。结合 FPGM 剪枝方法在 cifar10 数据集上的实验结果表明，在总体剪枝计算量不变的情况下，不同层级不同剪枝量的方法，模型性能损失更少；在模型性能损失保持良好条件下，可对模型剪枝更高的剪枝量。

关键词：模型压缩；剪枝；性能相关性；剪枝率

中图法分类号：TP399 **文献标识号**：A **文章编号**：1000-7024（2021）04-1109-07

doi：10.16208/j.issn1000-7024.2021.04.030

Hierarchical pruning rate method based on model performance correlation

YANG Kang[1], GUO Rong-zuo[1+], LI Chao[2], XU Jian-rong[3], YAN Yang-chun[1], GONG Lu-qi[2]

(1. College of Computer Science, Sichuan Normal University, Chengdu 610101, China; 2. Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China; 3. Key Laboratory of Opto-Electronics Technology of Ministry of Education, Faculty of Information Technology, Beijing University of Technology, Beijing 100124, China)

Abstract: The current pruning method for processing the neural network layer with uneven distribution of information with different pruning rates is inadequate. Therefore, a method of pruning different amounts of different network layers was proposed. The layers of the neural network model that have been pruned layer by layer was recovered, and the correlation between each layer and the model performance was got. The neural network layer was classified. Neural networks of different classes were calibrated with different pruning rates. Combining the FPGM pruning method with experiments on the cifar10 data set, the results show that, under the condition that the overall pruning calculation amount remains unchanged, and using the method of different levels and different pruning amounts, the model performance loss is less. The model can be pruned with a higher pruning amount under the condition of acceptable model performance loss.

Key words: model compression; pruning; performance correlation; pruning rate

0 引 言

庞大的深度卷积神经网络结构中必然存在着与模型性能相关性不明显的结构[1]，所以剪枝的根本意义在于找出这些相关性不大的结构，将其裁剪，从而简化网络结构。目前的剪枝方法中，例如 Louizos 等、Hao Li 等、Yang He 等[2-4]提出的按照结构正则化大小剪枝网络的方法，He Yang 等[5]、Lin Mingbao 等[6]按照各种结构之间相关性进行裁剪，但是都没有提出对深度卷积神经网络中的不同层设置不同剪枝量方法。尽管已有方法分析深度卷积神经网络各层对剪枝的敏感性[7]，但是此方法使用完整的深度卷积神经网络研究各层对剪枝的敏感性，完整的网络结构与

收稿日期：2019-12-10；修订日期：2021-01-28

基金项目：国家自然科学基金青年基金项目（61701331）

作者简介：杨康（1993-），男，四川雅安人，硕士研究生，研究方向为神经网络模型压缩、嵌入式系统等；+通讯作者：郭荣佐（1973-），男，四川达州人，硕士，教授，硕士生导师，研究方向为嵌入式系统、物联网感知技术、智能控制与智能机器人等；李超（1987-），男，山东聊城人，博士，高级工程师，硕士生导师，研究方向为深度学习模型压缩、人工智能可解释等；许建荣（1995-），男，广东陆丰人，硕士研究生，研究方向为神经网络模型压缩、嵌入式系统等；严阳春（1995-），男，四川达州人，硕士研究生，研究方向为神经网络模型剪枝；宫禄齐（1995-），男，山东威海人，硕士研究生，研究方向为神经网络模型压缩。E-mail：gyz00001@163.com

最终的剪枝网络结构差异较大，剪枝以后各层最终的敏感性变化也相对较大，该方法判断剪枝网络各层结构与模型性能的相关性上有所不足。

本文主要有以下几项突出工作：

（1）提出逐层复原剪枝网络的方法：在各层已被同等比例剪枝的神经网络结构上，逐层复原各层网络，探索神经网络各层与模型性能的相关性。相比于已有剪枝敏感性分析方法[7]，能更加准确、可靠地分析出各层与模型性能的真实相关性；

（2）根据各层与模型性能的相关性对各层分类级。不同相关性的层分到不同类级中；

（3）标定各层网络模型在不同剪枝比例下的最终剪枝量，为各类级网络设置剪枝量。根据各层的剪枝量，结合 FPGM[5] 方法对模型进行剪枝。

1 相关工作

深度卷积神经网络模型压缩的主流方式有如下几种：①剪枝：对现有的网络结构进行裁剪；②知识蒸馏：用大网络的结构信息，指导小网络的构建与训练[8,9]；③参数共享，量化：多个参数近似、共享一个值，或者降低参数的浮点数的表示位数[10-14]；④矩阵分解：大的网络矩阵，分解为多个小的网络矩阵[15,16]；⑤轻量化网络设计：设计结构更紧凑，计算量更小的网络结构[17-19]。但是轻量化网络的设计相对复杂困难，需要强大的团队基础，且轻量化网络中仍然存在着冗余，仍然可以被剪枝。

深度卷积神经网络模型压缩方式多种多样，且各有优劣。对已有神经网络模型进行剪枝不仅能大幅度压缩深度卷积神经网络结构，且实现性上相对简洁，易操作，在工业及科研领域上有重要的研究价值。所以近年来众多学者对深度卷积神经网络剪枝进行研究，剪枝方法又可以分为两大类：非结构化剪枝和结构化剪枝。

1.1 非结构化剪枝

非结构化剪枝：剪枝过程不拘于某一种结构形式，是最小权重单元的剪枝。非结构化剪化枝的根本只是将网络结构中的某些模型权重值设置为 0，使卷积矩阵稀疏化，因此也被叫作稀疏化剪枝。非结构化剪枝的剪枝过程实质如图 1 所示。许多非结构化剪枝选择在卷积核上做非结构化的剪枝，稀疏化卷积核，例如 Han Song 等[1]提出一种迭代式的方法多次裁剪低于某个阈值的网络权重。Carreira-Perpinán 等[20]将剪枝看作优化问题，非结构化的剪枝对模型性能影响最小的权重。非结构化剪枝在剪枝过程中因不拘于某一种结构，所以相对结构剪枝形式来说较为灵活，剪枝量相对更高。但是非结构化剪枝，只是稀疏化了卷积矩阵，不能直接简化模型实际运算复杂度。所以需要再次采用稀疏矩阵的加速方式来加速网络运算。

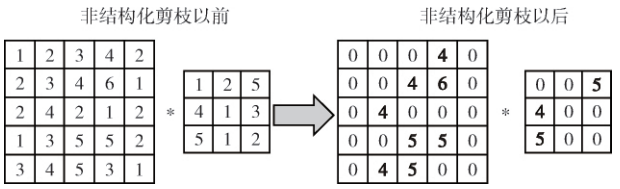

图 1 非结构化剪枝过程实质

1.2 结构化剪枝

结构化剪枝以某种特定结构粒度为基本单元进行裁剪，其过程实质如图 2 所示。在实际剪枝过程中，许多方法如文献［2，21-26］都会结合训练过程多次迭代。Hao Li 等[3]采用 L_1 正则化标准判断卷积核与模型性能相关性。Yang He 等[4]采用 L_2 正则化标准判断卷积核与模型性能相关性，并且创造性提出 SFP 的方法，将剪枝过程与训练过程结合。Liu Zhuang 等[27]将 BN 层 γ（缩放）系数加入到训练 loss 中，γ 引导网络结构稀疏化，以此为基础来剪枝神经网络，该方法在工业领域被广泛使用。Yang He 等[5]，采用同层内卷积核之间的欧几里得距离为判断标准的方法，判断各卷积核与模型性能的相关性，创造性的使用裁剪欧几里得距离中位数卷积核的方法式。Lin Mingbao 等[6]采用卷积核的秩的大小来判断卷积核与模型性能的相关性，在小样本剪枝训练中取得了优异的成绩。结构化剪枝相较于非结构化剪枝剪枝维度受限，所以剪枝量相对较小。但是此方法能够直接加速网络运算。所以近年来更多研究学者关注此维度的剪枝。

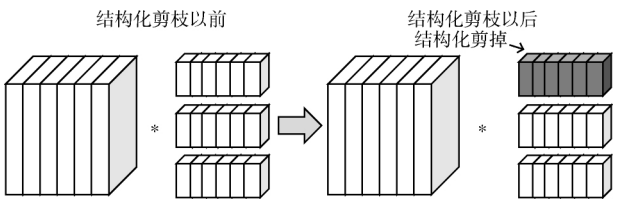

图 2 结构化剪枝过程实质

近年来众多学者都关注于新颖的剪枝方法，甚至多种剪枝方法分步迭代使用，以此达到较高的剪枝量。但是我们在剪枝研究中发现，具有分层结构的深度卷积神经网络模型不同层含有不同的信息量，剪枝中应当对不同层设置不同的剪枝量。通过这种设置，在模型总体剪枝基本相同 FLOPs（浮点数运算量）情况下，模型性能损失更小；或者在保证模型有良好性能表现下，模型能剪枝更高的比例。

2 各层性能相关性分析及层分类级

这一章节将分为 4 部分来阐述如何获得各层与模型性能的相关性及如何对各层分类级。第一部分，阐述通过逐层复原各层结构分析模型性能相关性以及如何对层分级的思路流程；第二部分，通过对比已有层剪枝敏感性分析方

法来详细阐明逐层复原分析各层结构与模型性能相关性的方法；第三部分，根据各层结构与模型性能相关性，运用分类级算法对模型各层进行分类级。第四部分，标定各层实际剪枝量，为各类级设定剪枝率。

2.1　逐层复原分析法和层分类级整体思路流程

深度卷积神经网络模型中的各个结构对模型性能的贡献度是不同的，剪枝的整体思路就是找到各个模型结构与模型性能之间的相关性，按相关性大小对模型实施裁剪。传统的各层等比例剪枝法，各个结构之间相关性排序范围是在各层之内，而层分级剪枝法将性能相关性的排序的范围拓宽到了各层之间。各层对模型性能的相关性的准确度量是层模型层分类级的基础。所以在运用分类级法前需要先获得各层与模型性能相关性的准确度量。基于模型性能相关性的分级剪枝率剪枝方法整体步骤如下：

（1）利用已有剪枝策略对神经网络模型进行各层等剪枝比例的裁剪；

（2）使用逐层复原的方法，依次得到复原每一层以后模型性能。多次实验，取性能的平均值；

（3）根据各层结构与模型性能相关性，运用分类级算法对模型各层进行分类级；

（4）标定神经网络模型在不同剪枝量下各层的实际剪枝量，为每一类网络层设置相同的剪枝量；

（5）使用 FPGM 剪枝方法，按照步骤（4）中的每一层的剪枝量对神经网络每一层做剪枝。

2.2　逐层复原各层结构分析模型性能相关性法

已有的剪枝敏感性分析法[3]，是在完整神经网络基础上，每次剪枝一层网络，逐层进行，得到只剪枝某层网络后模型的性能。剪枝敏感性分析法原理如图 3 所示，其中 ACC 表示完整网络模型的性能，ACC_1 表示只剪枝第一层网络模型的性能，同理有（ACC_2，ACC_3，……）。而逐层复原与逐层剪枝相反，用已经被等剪枝率剪枝了的模型为基础，每次复原一层剪枝网络，逐层进行。逐层复原剪枝网络原理如图 4 所示，其中 ACC' 表示等剪枝率剪枝网络模型性能，ACC'_1 表示只恢复第一层网络结构以后模型的性能。同理有（ACC'_2，ACC'_3……）。使用 ACC'_n 与 ACC' 的差值作为该层与模型性能相关性的标定。即模型各层与模型性能之间的相关性可以表示为

$$X = \{ACC'_1 - ACC', ACC'_2 - ACC', \cdots\cdots, ACC'_n - ACC'\}^T$$

$X_{(i)}$ 越大说明复原这一层与模型性能的相关性越高，可以为该层设置较低的剪枝率。在求各层性能相关性以后，会出现 $ACC'_n - ACC'$ 为负数的情况，说明复原这一层网络模型后性能反而降低了，所以这层应该剪枝更高的比例，当 $X_{(i)} \leqslant 0$ 时 $X_{(i)}$ 取 X 中非零值中的最小值。逐层复原方法得到模型各层与模型性能之间相关性的方法与已有剪枝敏感性分析法比较有以下两个优势：①相关性更加准确：判断性能相关性的网络结构更加接近于最终的剪枝模型；②在实验过程中可以节省大量的实验时间，因为选用已经裁剪的网络作为基础网络，相比较选用完整网络作为基础网络有更小的网络结构。

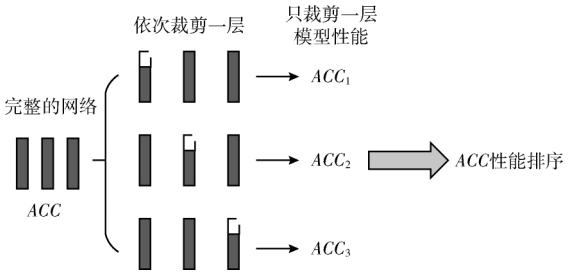

图 3　剪枝敏感性分析法原理

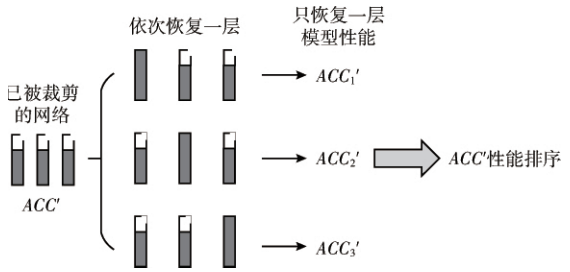

图 4　逐层复原剪枝网络原理

2.3　模型性能相关性分级

用集合 G 表示所有层的所有结构，用 M 表示神经网络的层数，用 N 表示神经网络每一层结构数量。则有 $G_{pq} \in G$，表示神经网络的任一结构。其中 $p \in \{1,2,3,\cdots,M\}$，$q \in \{1,2,3,\cdots,N\}$。剪枝的理想状态是在整个 G 中按顺序剪枝模型性能相关性较小的结构 G_{ij}。因为神经网络模型具有黑盒性，所以很难对整个网络的所有结构做与模型性能相关性的精准标定。只能通过一些表象反推结构与模型性能之间的相关性，做结构与模型性能相关性的相对标定。且在实际训练过程中，模型初始化等会带来随机性，也不可能做到结构与模型性能相关性的绝对标定。所以本文提出将深度神经网络层分类级剪枝法，将待剪枝的神经网络层进行分类，为不同类的层结构设置不同的剪枝量。此方法将会带来以下两个好处：①网络训练过程中的随机初始化虽然会造成网络性能相关性的波动，但是在一定范围内波动且在多次实验取平均值以后的网络层会被分到同一类级当中；②分类级方法还能在一定程度上保证原始网络结构的特性，减少每层不同剪枝量对原始模型结构特点的改变。

使用上述方法分析得到 VGG-16 网络各层模型性能相关性折线图，如图 5 所示，从图中可以看出相邻两层之间的模型性能相关性不会出现突变式的变化。这是因为当某一层网络在复原以后，下一层网络的裁剪比例也会有相对应的网络结构复原（2.4 小节详细分析这一原因）。所以当对模型性能相关性分类级以后，在神经网络结构中上下相

邻的两层一般设置相同的剪枝比例。这样就可以使剪枝网络保留更多原始网络的结构信息。

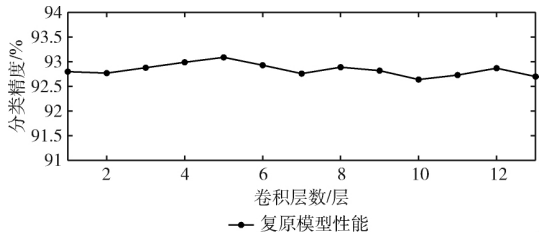

图 5 VGG-16 模型各层性能相关性

2.4 剪枝量的标定

各层剪实际枝量剖析如图 6 所示，设卷积层第 n 层卷积核的剪枝量为 $P_n\%$，如图中，模型第 n 层中浅色阴影部分，那么会造成 n 层卷积层的输出通道数相应减少 $P_n\%$，此输出通道也是 $n+1$ 层的输入通道，如图中第 $n+1$ 层输入通道中用浅阴影表示的部分，为了卷积维度能够对应起来，那么在 $n+1$ 层的卷积核中相应的通道也会相应剪枝 $P_n\%$。如图中，$n+1$ 层的卷积核中浅色的阴影部分。因此当n层卷积核剪枝$P_n\%$时，实际上会同时造成下一层卷积层输入通道数和卷积核通道数都剪枝 $P_n\%$，但第 n 层卷积核的剪枝对第 $n+2$ 层的输入通道数和卷积核数都不会造成直接变化。所以某一层剪枝只会使与其直接相连的下一层网络结构发生变化。又假设 $n+1$ 层剪枝量为 $P_{n+1}\%$，如图 6 中，第 $n+1$ 层卷积核中深色阴影部分，结合图 6 分析，可以得出此时第 $n+1$ 层的实际剪枝量为：$p_n\% + p_{n+1}\% - p_n\% \times p_{n+1}\%$。深度卷积神经网络的 FLOPs 统计中，卷积运算所包含 FLOPs 远远大于其它层运算量例如：BN 层运算、池化层运算等，为了简化分析，本文忽略这些层的 FLOPs。所以当 n 层剪枝 $P_n\%$ 和 $n+1$ 层剪枝 $P_{n+1}\%$ 时，深度神经网络模型 $n+1$ 层实际上减少的计算量为

$$p_n\% + p_{n+1}\% - p_n\% \times p_{n+1}\%$$

在为每一类级的神经网络设置不同剪枝率的时候，神经网络的剪枝的 FLOPs 的计算需要逐层进行计算，假设网络等比例剪枝 30%，结合以上公式，将各层网络分为 3 个类级时，可以给各类级设置剪枝量为 20%、30%、40%。结合网络中各层所包含的计算量，可以使网络整体剪枝量依然保持在一个基本相同的水平。

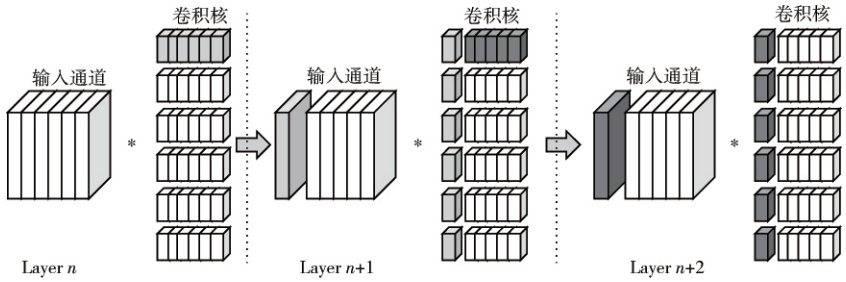

图 6 各层剪枝量分析

2.5 本章小结

综上所述，本文提出通过逐层复原各层网络结构分析该层结构与模型性能相关性的方法，在标定每一层与模型性能相关性时，网络结构更接近最终的剪枝网络结构，能够更加准确标定每一层与模型性能的相关性。并且使用神经网络层分类级的算法，将不同层分类，能够减少模型训练时参数初始化带来的随机性。为不同类级的层设置不同剪枝量的方法，相对于已有各层同等剪枝率剪枝的方法，此方法可以更灵活的为每一层网络结构设置剪枝量，进而可以更准确剪枝与模型性能相关性更小的网络结构。

3 实验分析

本研究实验代码，均在 Pytorch 编程框架上编程开发。实验所使用的硬件资源主要是 NVIDIA RTX 2080ti 显卡，本节将分为 4 部分来阐述实验情况及结果。第一部分：阐述实验的基本情况，基本设置。第二部分：阐述在 ResNet[28] 网络下的实验情况及结果。第三部分：阐述在 VGG 网络下的实验结果。第四部分：实验总结。

3.1 实验基本设置

3.1.1 实验网络结构和数据集

为了验证层分类级剪枝方法的实验效果，本实验选取被计算机视觉领域科研工作者广泛采用的 cifar10 数据集，一个用于普适物体分类的计算机视觉数据集。该数据集一共分为 10 个类别，每一类有 60 000 张 32×32 大小的 3 通道 RGB 彩色照片，其中 50 000 张用来训练，10 000 张用来测试。本实验首先剪枝没有分支结构的 VGG 网络结构，然后剪枝网络层数更深的带有分支结构的 ResNet 网络。其中 VGG 网络选择常见的 VGG-16；ResNet 按照 He Kaiming 等[28]的研究，在 cifar 数据集上实验室选择深度为 20，32，56，110 的网络结构。

3.1.2 训练方法

本研究实验遵循 He Yang 等[4]实验的训练方法设置，

学习率设置范围为动态 0.1-0.0008，bitch _size 设置为 128，其它参数均按照 SFP 实验默认设置。本实验选择软剪枝（SFP）训练方法，在不使用预训练网络结构参数的情况下从随机初始化开始训练，训练和剪枝同时进行，训练过程中已剪枝的网络结构可以参与下一次训练过程中的参数更新。在剪枝完成以后，可以不用对网络进行进一步的微调训练，模型也能有很好的实验效果。

3.2 ResNet 网络剪枝实验分析

He Yang 等[5]提出的 FPGM 剪枝方法，相比于大部分传统剪枝某些小数的剪枝方法，该方法创造性地提出剪枝各个卷积核之间欧几里得距离中位数的方法，目前在剪枝领域有很好的剪枝效果。所以本实验选用此方法作为逐层复原分析法的基础模型，在上一章节中已经对不同层设置不同剪枝量情况下，模型每一层的实际剪枝量已有一个明确的标定。虽然模型的实际剪枝量很难精确把控，但是在本实验中会保持模型剪枝计算量或者剪枝后模型的性能最大可能一致。ResNet 网络结构涉及到支路结构，对于支路结构的处理：在支路聚合时，当分支路剪枝量大于主路剪枝量时，分支路不足的卷积结构用 0 补足；当分支路的剪枝量小于主路时，分支路多余的卷积结构舍弃。对于 ResNet 网络逐层复原，通过分类级的方法，将各层与模型性能相关性进行分类分析，从而将各层分类级，各层分级详细结果见表 1。

在表 1 基础上对各类层做剪枝量的标定，性能相关性高的第一类设置剪枝量 30%，相关性次高的第二类设置剪枝量 40%，相关性最低的第三类为 50%。在本实验中还在保证模型性能损失很小情况下，结合 FPGM 方法尝试进行更高的剪枝量，分别设置为：35%、45%、55%，在与 FPGM 剪枝差不多相同浮点数运算量条件下，模型性能有一定的提升，实验结果见表 2。实验结果更高剪枝比例表，见表 3。

表 1 各层分类级结果

网络名称	类级 1	类级 2	类级 3
ResNet20	1、2、5、6、8、15	0、3、4、11、12、14、17、18	7、9、10、13、16、19
ResNet32	2、4、6、12、20、22、23、24、26、27、28	0、1、5、7、9、11、13、14、17、21、25、30	3、8、10、15、16、18、19、29、31
ResNet56	2、4、6、7、8、13、14、20、21、23、26、27、33、35、37、44、47、53	1、11、12、15、16、18、24、25、28、30、31、32、34、36、40、41、42、50、52	0、3、5、9、10、17、19、22、29、38、39、43、45、46、48、49、51、54
ResNet110	2、3、6、8、13、14、16、20、21、25、31、32、36、38、41、43、48、51、53、57、58、68、69、71、75、77、82、86、91、93、94、96、100、102、103、106	0、1、9、10、11、18、22、24、26、27、28、33、35、37、39、40、42、44、45、47、50、54、55、60、61、62、63、67、78、79、81、83、92、97、98、105、107、109	4、5、7、12、15、17、19、23、29、30、34、46、49、52、56、59、64、65、66、70、72、73、74、76、80、84、85、87、88、89、90、95、99、101、104、108

注：表 1 来源于杨康等[29]。

表 2 相似剪枝比例下模型性能对比

模型名称（ResNet）	剪枝方法	剪枝以后模型性能/%	裁剪浮点数计算量/%	性能提升/%
20	FPGM	90.62	54.00	**0.35**
	分类级剪枝方法	**90.97**	52.28	
32	FPGM	91.91	53.20	**0.55**
	分类级剪枝方法	**92.46**	52.85	
56	FPGM	92.93	52.6	**0.47**
	分类级剪枝方法	**93.40**	53.69	
110	FPGM	93.85	52.3	**0.35**
	分类级剪枝方法	**94.20**	52.73	

注：表 2 数据于杨康等[29]。

表 3 更高剪枝比例

模型名称（ResNet）	剪枝方法	裁剪浮点数计算量/%	剪枝以后模型性能/%	裁剪比例提升/%
20	FPGM	54.00	90.62	**0.35**
	分类级剪枝方法	**57.89**	90.71	
32	FPGM	53.20	91.91	**0.55**
	分类级剪枝方法	**60.38**	91.87	
56	FPGM	52.6	92.93	**0.47**
	分类级剪枝方法	**60.66**	92.86	
110	FPGM	52.3	93.85	**0.35**
	分类级剪枝方法	**58.17**	93.83	

注：表 3 来源于杨康等[29]。

3.3 VGG 网络剪枝实验

VGG-16 网络至上而下只有一条通路，没有任何的支路结构。在目前传统的剪枝实验中 VGG-16 网络结构一般只剪枝 30%左右，本实验中尝试了做更多剪枝比例，模型性能依然能保持良好的性能。首先是逐层获得各层与模型性能的相关性，然后使用分类级方法将模型各层分类。结合模型各层实际剪枝量标定方法，设置各类层剪枝量分别是 10%、20%、30%，则有模型最终减少 FLOPs 为 32.86%，VGG-16 实验结果结果见表 4。

表 4 VGG-16 实验结果

网络名称	剪枝方法	未裁剪模型性能/%	裁剪浮点数运算量/%	裁剪模型性能/%
VGG-16	FPGM	93.58	34.2	93.54
	分类级剪枝法		32.86	93.58

注：表 4 来源于杨康等[29]

Li，Hao 等[3]已经验证了逐层剪枝敏感性分析的方法，此方法在 VGG-16 模型上得到 VGG-16 模型的第 3～7 卷积层修剪与模型性能的相关性比较大，所以在剪枝过程中对 3～7 卷积层不做任何的剪枝，对剩余的卷积层均剪枝 50%。很显然这种简单的方式很难灵活的给模型各层设置剪枝量。而逐层复原分析方法可以对神经网络模型的每一层灵活地设置剪枝量，进而让模型总体上实现更高的剪枝比例。从表 4 中可以看出，FPGM 剪枝方法剪枝 VGG-16 模型 34.2%的 FLOPs 的时候，模型性能几乎没有损失，所以在对该模型剪枝 34.2%左右时，再比较模型性能损失意义不大。但是使用分类级法设置剪枝比例时，可以对模型进行更高比例的剪枝，模型性能损失依然能控制在很低的水平上。

将已有的剪枝敏感性分析方法得出的模型性能相关性高的层和逐层复原法得到的剪枝比例高的层对比。两种方法下模型性能相关性相对较高层对比见表 5，可以看出两种方法得到的剪枝量相对高的层基本对应，4-6 层的复原能相对大幅提升模型性能。但是第 3，7 层两种方法在剪枝量的设置上有一点区别。

表 5 两种方法下模型性能相关性相对较高层对比

网络结构	级 1	级 2	级 3
VGG-16	4,5,6,8	1,2,3,9	7,10,11,13

注：表 5 来源于杨康等[29]和 Li，Hao 等[3]

3.4 实验总结

本研究实验选用 VGG 和 ResNet 神经网络模型，通过本研究提出的逐层复原分析法，分析模型各层结构与性能相关性，然后对模型进行分类级设置剪枝比例的剪枝，最终两种模型下实验都取得了良好的效果。最终剪枝模型都对于 cifar10 数据集中验证集的分类性能良好。

4 结束语

本文提出逐层复原模型结构，分析各层与模型性能相关性，然后根据相关性分类级设置各层剪枝比例的方法。能够解决已有各层相同剪枝比例剪枝方法造成模型剪枝不均的问题。FPGM 剪枝方法结合分类级设置剪枝比例法，可以在保证对模型设置基本相同剪枝量的前提下，剪枝模型能够有更好的模型性能，也可以在保证模型性能损失很小的情况下，模型能剪枝更多的剪枝量。在未来的研究中，有很多方面还值得进一步的研究，比如可以剪枝更复杂、更新颖的深度卷积神经网络结构，例如 GoogLeNet[17]，MobileNet[18]等。或者在此方法的基础上结合更多的模型压缩的方法，例如元学习等方法，探究剪枝量与模型性能损失平衡的问题。

参考文献：

[1] Han Song，Mao Huizi，Dally W J. Deep compression：Compressing deep neural networks with pruning，trained quantization and huffman coding [J]. arXiv Preprint arXiv：1510.00149，2015.

[2] Louizos C，Welling M，Kingma D P. Learning sparse neural networks through $ L _0 $ regularization [J]. arXiv Preprint arXiv：1712.01312，2017.

[3] Li Hao，Kadav A，Durdanovic I，et al. Pruning filters for efficient convnets [J]. arXiv Preprint arXiv：1608.08710，2016.

[4] He Yang，Kang G，Dong X，et al. Soft filter pruning for accelerating deep convolutional neural networks [J]. arXiv Preprint arXiv：1808.06866，2018.

[5] He Yang，Liu Ping，Wang Ziwei，et al. Filter pruning via geometric median for deep convolutional neural networks acceleration [C] //Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Los Angeles：IEEE Computer Society，2019：4340-4349.

[6] Lin Mingbao，Ji Rongrong，Wang Yan，et al. HRank：Filter pruning using High-Rank feature map [C] //IEEE/CVF Conference on Computer Vision and Pattern Recognition. Los Angeles：IEEE Computer Society，2020：1529-1538.

[7] Li Hao，Kadav A，Durdanovic I，et al. Pruning filters for efficient convnets [J]. arXiv Preprint arXiv：1608.08710，2016.

[8] Hinton G，Vinyals O，Dean J. Distilling the knowledge in a neural network [J]. arXiv Preprint arXiv：1503.02531，2015.

[9] Kim J，Park S，Kwak N. Paraphrasing complex network：Network compression via factor transfer [C] //Proceedings of the 32nd International Conference on Neural Information Processing Systems. Montréal：IEEE，2018：2765-2774.

[10] Courbariaux M，Hubara I，Soudry D，et al. Binarized neural

networks：Training deep neural networks with weights and activations constrained to+1 or−1 [J]. arXiv Preprint arXiv：1602.02830，2016.

[11] Rastegari M，Ordonez V，Redmon J，et al. XNOR-Net：ImageNet classification using binary convolutional neural networks [J]. Computer Vision-ECCV. Springer：Cham，2016：525-542.

[12] Zhu Chenzhuo，Han Song，Mao Huizi，et al. Trained ternary quantization [J]. arXiv Preprint arXiv：1612.01064，2016.

[13] Zhou Aojun，Yao Anbang，Guo Yiwen，et al. Incremental network quantization：Towards lossless cnns with low-precision weights [J]. arXiv Preprint arXiv：1702.03044，2017.

[14] Son Sang，Nah S，Mu Lee K. Clustering convolutional kernels to compress deep neural networks [C] //Proceedings of the European Conference on Computer Vision. Munich：IEEE，2018：216-232.

[15] ZHANG Xingyu，ZOU Jianhua，HE Kaiming，et al. Accelerating very deep convolutional networks for classification and detection [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence，2015，38 (10)：1943-1955.

[16] Tai Cheng，Xiao Tong，Zhang Yi，et al. Convolutional neural networks with low-rank regularization [J]. arXiv Preprint arXiv：1511.06067，2015.

[17] Szegedy C，Liu Wei，Jia Yangqing，et al. Going deeper with convolutions [C] //Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Boston：IEEE，2015：1-9.

[18] Howard A G，Zhu Menglong，Chen Bo，et al. Mobilenets：Efficient convolutional neural networks for mobile vision applications [J]. arXiv Preprint arXiv：1704.04861，2017.

[19] Zhang Xiangyu，Zhou Xinyu，Lin Mengxiao，et al. Shufflenet：An extremely efficient convolutional neural network for mobile devices [C] //Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Hawaii：IEEE，2018：6848-6856.

[20] Carreira-Perpinán M A，Idelbayev Y. “Learning-Compression” algorithms for neural net pruning [C] //Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas：IEEE，2018：8532-8541.

[21] Liu Zhenhua，Xu Jizheng，Peng Xiulian，et al. Frequency-domain dynamic pruning for convolutional neural networks [C] //Advances in Neural Information Processing Systems. Montréal：IEEE，2018：1051-1061.

[22] He Yihui，Zhang Xiangyu，Sun Jian. Channel pruning for accelerating very deep neural networks [C] //Proceedings of the IEEE International Conference on Computer Vision. Venice：IEEE，2017：1389-1397.

[23] Molchanov P，Tyree S，Karras T，et al. Pruning convolutional neural networks for resource efficient inference [J]. arXiv Preprint arXiv：1611.06440，2016.

[24] Dubey A，Chatterjee M，Ahuja N. Coreset-based neural network compression [C] //Proceedings of the European Conference on Computer Vision. Munich：IEEE，2018：454-470.

[25] Suau X，Zappella L，Palakkode V，et al. Principal filter analysis for guided network compression [J]. arXiv Preprint arXiv：1807.10585，2018.

[26] Yu Ruichui，Li Ang，Chen Chunfu，et al. Nisp：Pruning networks using neuron importance score propagation [C] //Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition，2018：9194-9203.

[27] Liu Zhuang，Li Jianguo，Shen Zhiqiang，et al. Learning efficient convolutional networks through network slimming [C] //Proceedings of the IEEE International Conference on Computer Vision. Venice：IEEE，2017：2736-2744.

[28] He Kaiming，Zhang Xiangyu，Renhaoqing S，et al. Deep residual learning for image recognition [C] //Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition，2016：770-778.

[29] YANG Kang. Research and deployment of deep convolutional neural network model compression algorithm [D]. Chengdu：Sichuan Normal University，2020：14-21 (in Chinese). [**杨康. 深度卷积神经网络模型压缩算法研究与部署实现** [D]. **成都：四川师范大学**，2020：14-21.]