卷积神经网络结构优化综述

林景栋 [1] 吴欣怡 [1] 柴 毅 [1] 尹宏鹏 [1]

摘 要 近年来, 卷积神经网络 (Convolutional neural network, CNNs) 在计算机视觉、自然语言处理、语音识别等领域取得了突飞猛进的发展, 其强大的特征学习能力引起了国内外专家学者广泛关注. 然而, 由于深度卷积神经网络普遍规模庞大、计算度复杂, 限制了其在实时要求高和资源受限环境下的应用. 对卷积神经网络的结构进行优化以压缩并加速现有网络有助于深度学习在更大范围的推广应用, 目前已成为深度学习社区的一个研究热点. 本文整理了卷积神经网络结构优化技术的发展历史、研究现状以及典型方法, 将这些工作归纳为网络剪枝与稀疏化、张量分解、知识迁移和精细模块设计 4 个方面并进行了较为全面的探讨. 最后, 本文对当前研究的热点与难点作了分析和总结, 并对网络结构优化领域未来的发展方向和应用前景进行了展望.

关键词 卷积神经网络, 结构优化, 网络剪枝, 张量分解, 知识迁移

引用格式 林景栋, 吴欣怡, 柴毅, 尹宏鹏. 卷积神经网络结构优化综述. 自动化学报, 2020, **46**(1): 24−37

DOI 10.16383/j.aas.c180275

开放科学 (资源服务) 标识码 (OSID):

Structure Optimization of Convolutional Neural Networks: A Survey

LIN Jing-Dong[1] WU Xin-Yi[1] CHAI Yi[1] YIN Hong-Peng[1]

Abstract Recently convolutional neural networks (CNNs) have made great progress in computer vision, natural language processing and speech recognition, which attracts wide attention for their powerful ability of feature learning. However, deep convolutional neural networks usually have large capacity and high computational complexity, hindering their applications in real-time and source-constrained areas. Thus, optimizing the structure of deep model will contribute to rapid deployment of such networks, which has been a hot topic of deep learning community. In this paper, we provide a comprehensive survey of history progress, recent advances and typical approaches in network structure optimization. These approaches are mainly categorized into four schemes, which are pruning & sparsification, tensor factorization, knowledge transferring and compacting module designing. Finally, the remaining problems and potential trend in this topic are concluded and discussed.

Key words Convolutional neural networks (CNNs), structure optimization, network pruning, tensor factorization, knowledge transferring

Citation Lin Jing-Dong, Wu Xin-Yi, Chai Yi, Yin Hong-Peng. Structure optimization of convolutional neural networks: a survey. *Acta Automatica Sinica*, 2020, **46**(1): 24−37

卷积神经网络 (Convolutional neural network, CNNs) 作为最重要的深度模型之一, 由于具有良好的特征提取能力和泛化能力, 在图像处理、目标跟踪与检测、自然语言处理、场景分类、人脸识别、音频检索、医疗诊断诸多领域获得了巨大成功. 卷积神经网络的快速发展一方面得益于计算机性能的大幅提升, 使得构建并训练更大规模的网络不再受到硬件水平的限制; 另一方面得益于大规模标注数据的增长, 增强了网络的泛化能力. 以大规模视觉识别竞赛 (ImageNet large scale visual recognition competition, ILSVRC) 的历届优秀模型为例, AlexNet[1] 在 ILSVRC 2012 上的 Top-5 识别正确率达到 83.6 %, 随后几年卷积神经网络的性能持续提升[2−4], ResNet-50[5] 在 ILSVRC 2015 上的 Top-5 识别正确率达到 96.4 %, 已经超过人类平均水平. 在此之后, 卷积神经网络被进一步应用于

收稿日期 2018-05-03 录用日期 2018-11-05
Manuscript received May 3, 2018; accepted November 5, 2018
国家自然科学基金 (61633005, 61773080), 中央高校基本科研业务费专项资金 (2019CDYGZD001), 重庆市基础科学与研究技术专项 (cstc2015jcyjB0569), 重庆大学科研后备拔尖人才 (cqu2018CDHB1B04), 重庆市重点科技专项子项 (cstc2015shmszdzx30001) 资助
Supported by National Natural Science Foundation of China (61633005, 61773080), Fundamental Research Funds for the Central Universities (2019CDYGZD001), Chongqing Nature Science Foundation of Fundamental Science and Frontier Technologies (cstc2015jcyjB0569), Scientific Reserved Talents of Chongqing University (cqu2018CDHB1B04), Chongqing Nature Science Foundation of Scientific Key Program (cstc2015shmszdzx30001)
本文责任编委 贺威
Recommended by Associate Editor HE Wei
1. 重庆大学自动化学院 重庆 400044
1. College of Automation, Chongqing University, Chongqing 400044

其他领域, 比如由谷歌 DeepMind 公司开发的人工智能围棋程序 AlphaGo 在 2016 年战胜世界围棋冠军李世石.

卷积神经网络的整体架构大体上遵循着一种固定的范式, 即网络前半部分堆叠卷积层, 间或插入若干池化层以组成特征提取器, 最后连上全连接层作为分类器, 构成一个端到端的网络模型, 如图 1 中 LeNet-5[6] 所示. 卷积神经网络一般通过增加卷积层数量以增加网络深度, 用这种方式获得的深度模型在分类任务上有更好的表现[7]. 从表 1 可以看出, 卷积神经网络的性能不断增长, 其在 ImageNet 数据集的识别错误率不断降低, 同时其时间复杂度和空间复杂度也相应上升. 具体地, 卷积神经网络的网络层数呈持续增加态势, 其训练参数数量和乘加操作数量也保持在一个较高的水平, 例如 VGGNet-16 具有高达 138 M 参数量, 其整体模型规模超过 500 M, 需要 155 亿次浮点数操作才能对一张图片进行分类.

深度卷积神经网络通常都包含有几十甚至上百卷积层, 训练参数量动辄上百万, 在 GPU 加速支持下仍然需要花费几天或几周时间才能完成训练 (如 ResNet 需用 8 个 GPU 训练 2～3 周时间), 制约了其在移动设备、嵌入式系统等资源受限场景下的应用. 如表 1 所示, 过去由于卷积层在网络训练阶段和预测阶段的前向推导过程中涉及大量的浮点数计算操作, 而全连接层的神经元之间采用全连接方式, 拥有绝大多数训练参数, 所以卷积神经网络的时间复杂度主要由卷积层决定, 空间复杂度主要由全连接层决定. 随着卷积神经网络逐渐向更深层次发展, 卷积层数量急剧增加, 在前向推导过程中产生的中间变量会占用大量内存空间, 此时卷积层同时决定了网络的时间复杂度和空间复杂度. 因此, 降低卷积层和全连接层的复杂度有助于优化卷积神经网络的结构, 对于网络的压缩与加速也有重要的促进作用.

针对网络结构优化的相关研究在 90 年代已被提出[8−9], 然而由于当时神经网络大多属于浅层网络, 对于结构优化的需求尚不强烈, 因此未能引起广泛关注. 如今卷积神经网络的规模日益庞大, 而大量应用场景都无法提供相应的必需资源, 因此探讨在保证网络精度的前提下压缩并加速模型是网络结构优化领域的前沿热点. 随着对卷积神经网络结构优化研究的逐渐深入, 大量成果不断涌现, 一些学者对这一领域的相关工作进行了归纳与总结, 如文献 [10] 重点讨论了模型压缩与加速各种方法的优缺点, 文献 [11] 从硬件和软件两方面整理了网络加速的研究进展, 文献 [12] 简要介绍了深度网络压缩的典型方法. 本文在这些工作的基础上, 结合最新研究进展和成果, 全面地梳理与总结了卷积神经网络

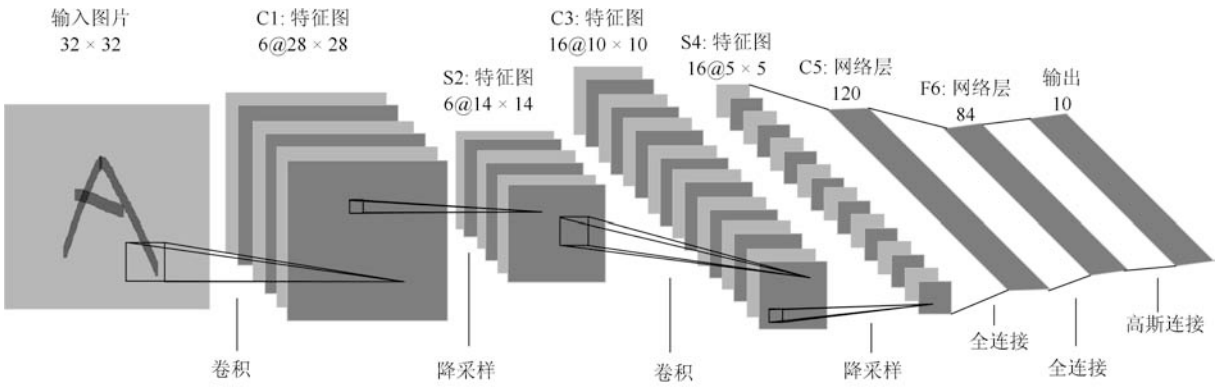


图 1　LeNet-5 网络结构[6]

Fig. 1　Structure of LeNet-5[6]

表 1　经典卷积神经网络的性能及相关参数

Table 1　Classic convolutional neural networks and corresponding parameters

年份	网络名称	网络层数	卷积层数量	参数数量		乘加操作数 (MACs)		Top-5 错误率 (%)
				卷积层	全连接层	卷积层	全连接层	
2012	AlexNet[1]	8	5	2.3 M	58.6 M	666 M	58.6 M	16.4
2014	Overfeat[2]	8	5	16 M	130 M	2.67 G	124 M	14.2
2014	VGGNet-16[[3]	16	13	14.7 M	124 M	15.3 G	130 M	7.4
2015	GoogLeNet[4]	22	21	6 M	1 M	1.43 G	1 M	6.7
2016	ResNet-50[5]	50	49	23.5 M	2 M	3.86 G	2 M	3.6

结构优化方面的研究工作. 其中第 1 节到第 4 节分别从网络剪枝与稀疏化、张量分解、知识迁移和精细化结构设计 4 个方面归纳了相关研究思想和方法, 第 5 节综合卷积神经网络结构优化领域的研究现状, 对其未来研究趋势和应用方向进行了展望.

1 网络剪枝与稀疏化

文献 [13] 的研究表明, 卷积神经网络从卷积层到全连接层存在大量的冗余参数, 大多数神经元被激活后的输出值趋近于 0, 即使将这些神经元剔除也能够表达出模型特征, 这种现象被称为过参数化. 例如 ResNet-50 拥有 50 层卷积层, 整个模型需要 95 MB 存储空间, 在剔除 75 % 的参数后仍然正常工作, 而且运行时间降低多达 50 %[14]. 因此, 在网络训练过程中可以寻求一种评判机制, 剔除掉不重要的连接、节点甚至卷积核, 以达到精简网络结构的目的. 网络结构精简的一个具体表现是网络的稀疏化, 这给模型训练带来了三点好处: 首先是由于网络参数的减少, 有效缓解了过拟合现象的发生[15]; 其次, 稀疏网络在以 CSR (Compressed sparse row format, CSR) 和 CSC (Compressed sparse column format) 等稀疏矩阵存储格式存储于计算机中可大幅降低内存开销; 最后, 训练参数的减少使得网络训练阶段和预测阶段花费时间更少. 由于网络剪枝具有易于实施且效果显著的优点, 目前已成为模型压缩与加速领域最重要的结构优化技术.

根据卷积神经网络训练阶段的不同, 网络剪枝与稀疏化方法主要包含训练中稀疏约束与训练后剪枝两个大类[16]. 对于前者, 通过在优化函数添加稀疏性约束, 诱导网络结构趋于稀疏, 这种端到端的处理方法不需要预先训练好模型, 简化了网络的优化过程. 对于后者, 通过剔除网络中相对冗余、不重要的部分, 同样可以使得网络稀疏化、精简化. 事实上, 无论是在训练中引入稀疏约束还是训练后剪枝网络, 最终目的都是使网络的权重矩阵变得稀疏, 这也是加速网络训练、防止网络过拟合的重要方式.

对于网络损失函数中的稀疏约束, 主要是通过引入 l_0 或 l_1 正则化项实现的. 假设训练数据集 D 包含 N 个数据对 $(x_1, y_1), (x_2, y_2), \cdots, (x_N, y_N)$, 训练参数为 θ, 则网络训练的目标优化函数一般表示为:

$$\arg\ \min_{\theta} \frac{1}{N} \sum_{i=0}^{N} L\left(h\left(x_i; \theta\right), y_i\right) + \lambda ||\theta||_p \qquad (1)$$

其中, $||\theta||_p = \left(\sum_i |x_i|^p\right)^{1/p}, p = 0, 1$. 优化函数的第一项是经验风险, 第二项是正则化项, 带有正则化约束的优化函数在反向传播时驱使不重要权重的数值变为零, 使得训练后的网络具有一定的稀疏性和较好的泛化性能. Collins 等[17] 在参数空间中通过贪婪搜索决定需要稀疏化的隐含层, 能够大幅减少网络中的权重连接, 使模型的存储需求降低了 3 倍, 并且克服了 OBS 与 OBD 处理大型网络面临的精度下降问题. Jin 等[18] 提出的迭代硬阈值 (Iterative hard thresholding, IHT) 方法分两步对网络进行剪枝, 在第一步中剔除隐含节点间权值较小的连接, 然后微调 (Fine-tune) 其他重要的卷积核, 在第二步中激活断掉的连接, 重新训练整个网络以获取更有用的特征. 相比于传统方式训练的网络, 通过 IHT 训练的网络具有更加优越的泛化能力和极低的内存大小. Zeiler 等[19] 利用前向–后向切分法 (Forward-backward splitting method) 处理带有稀疏约束的损失函数, 避免了在反向传播中需要求取二阶导数等计算复杂度较高的运算, 加快了网络训练速度. Wen 等[20] 认为网络结构从卷积核到卷积通道都充斥着冗余无用的信息, 他们提出的结构化稀疏学习 (Structured sparsity learning, SSL) 直接学习到的硬件友好型稀疏网络不仅具有更加紧凑的结构, 而且运行速度可提升 3 倍至 5 倍. Lebedv 等[21] 以分组形式剪枝卷积核输入, 以数据驱动的方式获取最优感受野 (Receptive field), 在 AlexNet 中获得 8.5 倍的速度提升而损失精度不到 1 %. Louizos 等[22] 利用一系列优化措施将不可微分的 l_0 范数正则项加入到目标函数, 学习到的稀疏网络不仅具有良好的泛化性能, 而且极大加速了模型训练和推导过程.

Dropout 作为一种强有力的网络优化方法, 可被视为特殊的正则化方法, 被广泛用于防止网络训练过拟合[23−24]. Dropout 在每次训练时随机使一半神经元暂时失活, 相当于在一定时间内训练了多个不同网络并将其组合, 避免了复杂的共适应现象 (Co-adaptation) 发生, 在图像分类、语音识别、文件分类和生物计算等任务都有较好表现. 然而, 由于 Dropout 在每次训练时都会尝试训练不同的网络, 这将导致训练时间的大幅延长. 因此, 目前也有一些工作针对 Dropout 的加速展开研究, 如 Li 等[25] 提出的自适应 Dropout 根据特征和神经元的分布使用不同的多项式采样方式, 其收敛速度相对于标准 Dropout 提高 50 %.

训练后网络剪枝是从已有模型着手, 消除网络中的冗余信息, 这避免了重新训练网络带来的高昂资源花费. 根据剪枝粒度的不同, 目前主要有层间剪枝、特征图剪枝、$k \times k$ 核剪枝与核内剪枝 4 种方式[26], 如图 2 所示. 层间剪枝一个直接的后果就是减少了网络的深度, 而特征图剪枝则减少了网络的宽度. 这两种粗粒度的剪枝方法在减少网络参数方面效果明显, 但面临网络性能下降严重的问题. $k \times k$ 核剪枝与核内剪枝两种细粒度方法在参数量与模型性能之间取得了一定的平衡, 但提高了方法的复

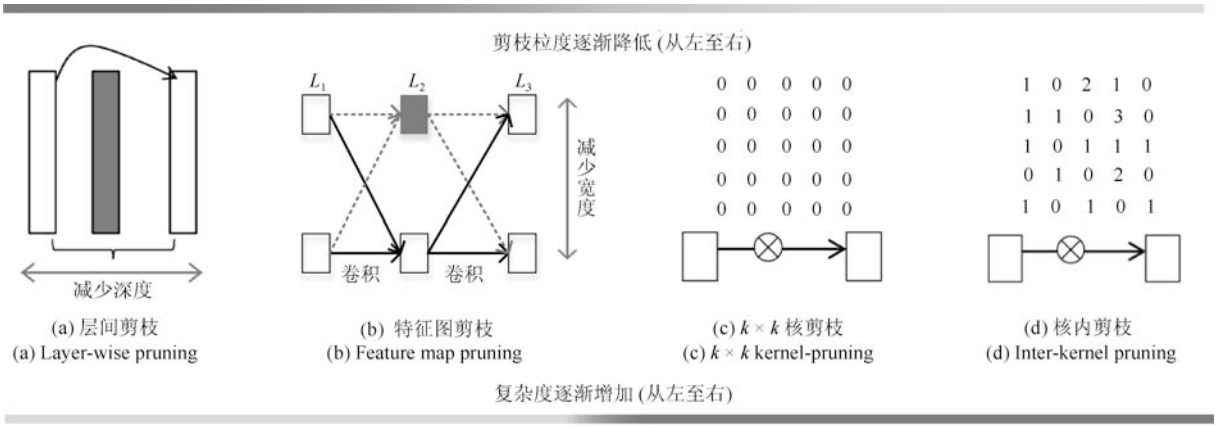


图 2　四种剪枝粒度方式[26]

Fig. 2　Four pruning granularities[26]

杂度.

事实上, 网络剪枝方法在深度学习流行起来就已被提出, 其早在上世纪九十年代即被广泛用于网络的优化问题. Hanson 等[27] 在误差函数中引入权重衰减项使网络趋于稀疏, 即减少隐含节点数目以降低网络复杂度. LeCun 等[8] 提出的最优脑损伤 (Optimal brain damage, OBD) 通过移除网络中不重要的连接, 在网络复杂度和训练误差之间达到一种最优平衡状态, 极大加快了网络的训练过程. Hassibi 等[9] 提出的最优脑手术 (Optimal brain surgeon, OBS) 与 OBD 的最大不同在于损失函数中的 Hessian 矩阵没有约束, 这使得 OBS 在其他网络中具有比 OBD 更普遍的泛化能力. 尽管 OBD 与 OBS 最初取得了较好效果, 但由于其损失函数中需要求取二阶导数, 在处理大型复杂网络结构时计算量巨大, 且面临着网络精度损失严重的问题, 因此探索适合于深度卷积神经网络的网络剪枝与稀疏化方法对于网络结构优化具有重要的研究价值.

网络剪枝方法使得精简后的小型网络继承了原始网络的有用知识, 与此同时具有与其相当的性能表现, 目前已取得一系列卓有成效的成果. Han 等[28] 提出的深度压缩 (Deep compression) 综合应用了剪枝、量化、编码等方法, 在不影响精度的前提下可压缩网络 35～49 倍, 使得深度卷积网络移植到移动设备上成为可能. Srinivas 等[29] 针对全连接层的神经元而非网络连接进行剪枝操作, 提出的方法摆脱了对于训练数据的依赖, 由于避免了多次重复训练, 极大降低了计算资源需求和花费时间. Guo 等[30] 认为参数的重要性会随着网络训练开始而不断变化, 因此恢复被剪枝的重要连接对于改善网络性能具有重要作用. 他们提出的动态网络手术 (Dynamic network surgery) 在剪枝过程中添加了修复操作, 当已被剪枝的网络连接变得重要时可使其重新激活, 这两个操作在每次训练后交替进行, 极大改善了网络学习效率. Liu 等[31] 针对 Winograd 最小滤波算法与网络剪枝方法无法直接组合应用的问题, 提出首先将 ReLU 激活函数移至 Winograd 域, 然后对 Winograd 变换之后的权重进行剪枝, 在 CIFAR-10、CIFAR-100 和 ImageNet 数据集上的乘法操作数分别降低了 10.4 倍、6.8 倍和 10.8 倍.

近年来针对更高层级的网络结构剪枝方法层出不穷, 有力推动了模型压缩与加速的发展, 对于卷积神经网络的结构优化也有重要的促进作用. He 等[32] 基于 LASSO 正则化剔除冗余卷积核与其对应的特征图, 然后重构剩余网络, 对于多分支网络也有很好的效果. Li 等[33] 发现基于重要度 (Magnitude-based) 的剪枝方法尽管在全连接层可以取得较好效果, 但是对于卷积层就无能无力了. 他们直接去除对于输出精度影响较小的卷积核以及对应的特征图, 以一种非稀疏化连接的方式降低了百分之三十的计算复杂度. Anwar 等[26] 按照粒度大小将剪枝方法划分为层级剪枝、特征图剪枝、卷积核剪枝、卷积核内部剪枝 4 个层级, 结合特征图剪枝与卷积核剪枝提出的一次性 (One-shot) 优化方法可获得 60 %～70 % 的稀疏度. 同样是针对卷积核剪枝, Luo 等[34] 提出的 ThiNet 在训练和预测阶段同时压缩并加速卷积神经网络, 从下一卷积层而非当前卷积层的概率信息获取卷积核的重要程度, 并决定是否剪枝当前卷积核, 对于紧凑型网络也有不错的压缩效果. 表 2 比较了不同网络剪枝方法对于卷积神经网络的压缩效果, 可以发现这些方法能够大幅减少训练参数而不会显著影响网络精度, 表明网络剪枝与稀疏化是一种强有力的网络结构优化方法.

2　张量分解

由于卷积神经网络规模逐渐向更深、更大层次发展, 卷积操作过程中所需计算资源以及每次卷积

表 2 网络剪枝对不同网络的压缩效果

Table 2 Comparison of different pruned networks

所用方法	选用网络	初始错误率	剪枝后错误率	初始参数量	剪枝后参数量	压缩率
[28]	AlexNet	19.73 %	19.70 %	61 M	6.7 M	6 ×
[29]	CaffeNet	42.16 %	44.4 %	61 M	21.3 M	3 ×
[30]	LeNet-5	0.91 %	0.91 %	431 K	4.0 K	108 ×
[33]	VGGNet-16	6.75 %	6.6 %	150 M	5.4 M	28 ×
[34]	ResNet-50	8.86 %	11.7 %	25.56 M	8.66 M	3 ×

后所需存储资源已成为制约模型小型化、快速化的瓶颈. 比如说, ResNet-152 网络来自于卷积层的参数数量为全部参数的 92 %, 而来自于卷积层的计算量占到总计算量的 97 %. 已有研究结果表明[35], 卷积神经网络仅需很少一部分参数即可准确地预测结果, 这说明卷积核中存在大量的冗余信息. 张量分解对于去除冗余信息、加速卷积计算是一种极为有效的方法, 可以有效压缩网络规模并提升网络运行速度, 有益于深度神经网络在移动嵌入式环境下的高效运行.

一般来说, 向量称为一维张量, 矩阵称为二维张量, 而卷积神经网络中的卷积核可以被视为四维张量, 表示为 $K \in \mathbf{R}^{d\times d\times I\times O}$, 其中, I, d, O 分别表示输入通道, 卷积核尺寸和输出通道. 张量分解的思想即是将原始张量分解为若干低秩张量, 有助于减少卷积操作数量, 加速网络运行过程. 前常见的张量分解方法有 CP 分解、Tucker 分解等, Tucker 分解可将卷积核分解为一个核张量与若干因子矩阵, 是一种高阶的主成分分析方法, 其表达形式为:

$$K \approx C \times U_1 \times U_2 \times U_3 \times U_4 \tag{2}$$

其中, $K \in \mathbf{R}^{d\times d\times I\times O}$ 为分解后的核张量, $U_1 \in \mathbf{R}^{d\times r_1}$、$U_2 \in \mathbf{R}^{d\times r_2}$、$U_3 \in \mathbf{R}^{I\times r_3}$、$U_4 \in \mathbf{R}^{O\times r_4}$ 为因子矩阵. CP 分解的表达形式为:

$$K \approx K_1 \times K_2 \times K_3 \times K_4 \tag{3}$$

其中, $K_1 \in \mathbf{R}^{d\times r}$、$K_2 \in \mathbf{R}^{d\times r}$、$K_3 \in \mathbf{R}^{\mathbf{I}\times r}$、$K_4 \in \mathbf{R}^{\mathbf{O}\times r}$. CP 分解属于 Tucker 分解的一种特殊形式, 其分解过程更为简单, 然而分解矩阵的秩 r 的选取是一个 NP 难问题, 并且可能涉及到分解稳定性问题. 值得注意的是, 由于全连接层也可以视为二维张量, 因此可利用矩阵奇异值分解 (Singular value decomposition, SVD) 去除全连接层的冗余信息, 分解表达式为:

$$W \approx USV^{\mathrm{T}} \tag{4}$$

其中, $W \in \mathbf{R}^{m\times n}$ 为待分解张量, $U \in \mathbf{R}^{m\times m}$ 和 $V \in \mathbf{R}^{n\times n}$ 是正交矩阵, $S \in \mathbf{R}^{m\times n}$ 是对角矩阵. 图 3 展示了将一个 $W \in \mathbf{R}^{d\times d\times I\times O}$ 张量分解为一个 $P \in \mathbf{R}^{O\times K}$ 张量和一个 $W' \in \mathbf{R}^{K\times d\times d\times I}$ 张量的过程. 图 3 (a) 中 W 为原始张量, 复杂度为 $\mathrm{O}(d^2IO)$; 图 3 (b) 中 P 和 W' 为分解后张量, 复杂度为 $\mathrm{O}(OK) + \mathrm{O}(d^2KI)$. 对于大多数网络有 $\mathrm{O}(OK) \ll \mathrm{O}(d^2KI)$, 所以分解后复杂度为原来的 O/K, 并且 K 值越小, 压缩效果越明显.

利用张量分解以加速卷积过程已有很长的一段时间, 最典型的例子就是将高维离散余弦变换 (Discrete cosine transform, DCT) 分解为一系列一维 DCT 变换相乘, 以及将小波系统分解为一系列一维小波的乘积[10]. Rigamonti 等[36] 基于字典学习的思想, 提出的分离卷积核学习方法 (Learning separable filters) 能够将原始卷积核用低秩卷积核表示, 减少所需卷积核数量以降低计算负担. 同时, 作者认为在构建网络时不用再精心设计卷积核结构, 只需通过分离卷积核学习就可以得到最优的卷积核组合. Jaderberg 等[37] 提出了一种逐层分解方法, 每当一个卷积核被分解为若干一阶张量, 则固定此卷积核并基于一种重构误差标准以微调其余卷积核, 研究结果表明在场景文本识别中可加速网络 4.5 倍而准确度仅降低 1 %. Denton 等[38] 认为卷积神经网络的绝大部分冗余参数都位于全连接层, 因此主要针对全连接层展开奇异值分解, 分解后的网络网络参数

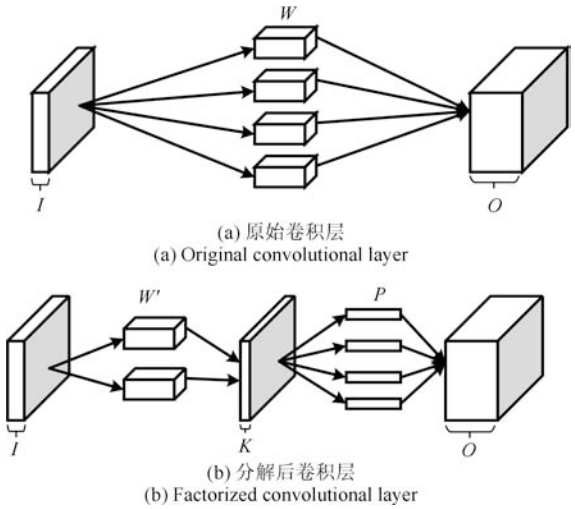


图 3 张量分解过程

Fig. 3 Process of tensor factorization

最多减少 13 倍, 同时其运行速度可提升 2～3 倍. Lebedev 等[39] 提出了基于 CP 分解的卷积核张量分解方法, 通过非线性最小二乘法将卷积核分解为 4 个一阶卷积核张量. 对于 36 类的 ILSVRC 分类实验, 该方法在 CPU 上可获得 8.5 倍加速, 实验结果同时表明张量分解具有正则化效果. Tai 等[40] 提出了一种带有低秩约束的张量分解新算法, 将非凸优化的张量分解转化为凸优化问题, 与同类方法相比提速明显.

以上基于张量分解的方法虽然能够取得一定效果, 然而它们仅仅压缩与加速一层或几层网络, 欠缺对于网络整体的考量. Zhang 等[41] 提出了一种非对称张量分解方法以加速整体网络运行, 例如一个 $D \times D$ 卷积核可被分解为 $1 \times D$、$D \times 1$ 和 1×1 等张量. 此外, 文献 [41] 还提出了基于 PCA 累积能量的低秩选择方法和具有非线性的重构误差优化方法, 在 ImagNet 上训练的大型网络可被整体加速 4 倍. 与文献 [41] 不同, Kim 等[42] 提出了基于变分贝叶斯的低秩选择方法和基于 Tucker 张量分解的整体压缩方法. 由于模型尺寸、运行时间和能量消耗都大幅降低, 使用该方法压缩的网络可以移植到移动设备上运行. Wang 等[43] 认为网络压缩不能仅仅考虑卷积核, 同时要考虑卷积核在网络运行过程中映射的巨量特征图. 文献 [43] 利用循环矩阵剔除特征图中的冗余信息, 获取特征图中最本质的特征, 进一步重构卷积核以匹配压缩后的特征图. 实验结果表明文献 [43] 中的方法尽管只有很少参数, 但具有与原始网络相当的性能. Astrid 等[44] 提出了一种基于优化 CP 分解全部卷积层的网络压缩方法, 在每次分解单层网络后都微调整个网络, 克服了由于 CP 分解不稳定引起的网络精度下降问题.

张量分解对于深度网络的压缩与加速具有直接作用, 可以作为网络结构优化设计方法的重要补充. 然而目前大多数的张量分解方法都是逐层分解网络, 缺乏整体性的考虑, 有可能导致不同隐含层之间的信息损失. 此外, 由于涉及到矩阵分解操作, 会造成网络训练过程的计算资源花费高昂. 最后, 由于每次张量分解过后都需要重新训练网络至收敛, 这进一步加剧了网络训练的复杂度.

3　知识迁移

知识迁移是属于迁移学习的一种网络结构优化方法, 即将教师网络 (Teacher networks) 的相关领域知识迁移到学生网络 (Student networks) 以指导学生网络的训练, 完成网络的压缩与加速. 一般地, 教师网络往往是单个复杂网络或者是若干网络的集合, 拥有良好的性能和泛化能力, 而学生网络则具有更小的网络规模, 还未获得充分的训练. 考虑利用教师网络本身的知识或通过教师网络学习到的知识去指导学生网络训练, 使得学生网络具有与教师网络相当的性能, 但是参数数量大幅降低, 同样可以实现网络压缩与加速的效果.

知识迁移主要由教师网络获取和学生网络训练两部分内容构成, 在教师网络获取中, 由于教师网络规模较大, 需要用大量标签数据对其进行训练以获得较高的预测准确率. 在学生网络训练过程中, 首先将未标签数据输入教师网络进行预测, 然后将预测到的结果与输入数据人工合成为标签数据, 最后将这些人工合成的标签数据作为领域知识以指导学生网络的训练. 由于学生网络规模较小, 因此只需少量的标签数据即可完成训练. 知识迁移的整体流程如图 4 所示.

Bucila 等[45] 首先提出了基于知识迁移的模型压缩方法, 通过人工合成数据训练学生网络以完成压缩与加速. 其具体步骤为首先将大型无标签数据集输入教师网络以获得相应的标签, 获得人工合成的标签数据, 然后在人工标签数据集上训练学生网络, 实验结果表明学生网络尺寸减少了 1 000 倍, 同时运行速度提升了 1 000 倍. 最初由大型复杂网络获得的知识可根据 softmax 函数计算的类别概率标

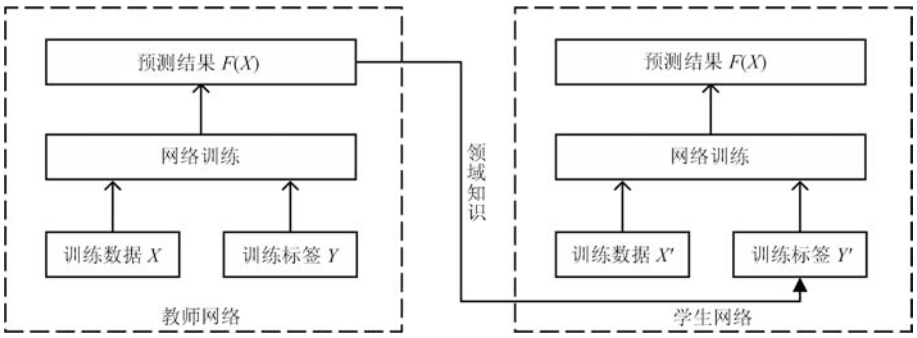


图 4　知识迁移过程

Fig. 4　Process of knowledge transfer

签来表示, 相比于 one-hot 标签, 类别概率标签包含了训练样本中的相关近似程度, 可以更加有效地训练学生网络. 然而类别概率标签的大多数概率值在通过 softmax 函数后都趋近于 0, 损失了大量有效信息. Ba 等[46] 提出利用 logits (通过 softmax 函数前的输入值, 均值为 0) 来表示学习到的知识, 揭露了标签之间的相对关系和样本之间的近似度. 与文献 [45] 类似, Ba 等[46] 将教师网络获得数据集的 logits 标签作为知识指导学生网络的训练, 在 TIMIT 和 CIFAR-10 数据库上都能够达到与深度网络相当的识别精度. Hinton 等[47] 认为类别概率标签和 logits 标签都是 softmax 层的极端输出, 其中 T 分别为 1 和正无穷. 他们提出的知识精馏 (Knowledge distilling, KD) 采用合适的 T 值, 可以产生一个类别概率分布较缓和的输出 (称为软概率标签 (Soft probability labels)). 软概率标签揭示了数据结构间的相似性, 包含大量的有用信息, 可利用软概率标签训练学生网络以模拟复杂的网络集合. Romero 等[48] 提出的 FitNet 不仅利用了教师网络的输出, 同时也将教师网络的隐含层输出作为知识迁移到学生网络中. 通过这种方式训练的学生网络相比于教师网络更深更窄, 因此具有更好的非线性变换能力.

与之前基于类别概率标签的知识迁移不同, Luo 等[49] 利用教师网络的高层神经元输出来表示需要迁移的领域知识. 这种方式不会损失任何信息, 但是学生网络可以获得更高的压缩率. Chen 等[50] 基于函数保留变换 (Function-preserving transformation) 提出的 Net2Net 是加速知识迁移流程的有效工具, 可以快速地将教师网络的有用信息迁移到更深 (或更宽) 的学生网络. Zagoruyko 等[51] 借鉴知识精馏的思想, 提出了一种基于注意力的知识迁移方法. 他们使用教师网络中能够提供视觉相关位置信息的注意力特征图来监督学生网络的学习, 并且从低、中、高三个层次进行注意力迁移, 极大改善了残差网络等深度卷积神经网络的性能. Lucas 等[52] 提出了一种结合 Fisher 剪枝与知识迁移的优化方法, 首先利用预训练的高性能网络生成大量显著性图作为领域知识, 然后利用显著性图训练网络并利用 Fisher 剪枝方法剔除冗余的特征图, 在图像显著度预测中可加速网络运行多达 10 倍. Yim 等[53] 将教师网络隐含层之间的内积矩阵作为领域知识, 不仅能更快更好地指导学生网络的训练, 而且在与教师网络不同的任务中也能获得较好效果. Chen 等[54] 结合文献 [47−48] 的相关方法, 首次提出了基于知识迁移的端到端的多目标检测框架, 解决了目标检测任务中存在的欠拟合问题, 在精度与速度方面都有较大改善.

知识迁移方法能够直接加速网络运行而不需要较高硬件要求, 大幅降低了学生网络学习到不重要信息的比例, 是一种有效的网络结构优化方法. 然而知识迁移需要研究者确定学生网络的具体结构, 对研究者的水平提出了较高的要求. 此外, 目前的知识迁移方法仅仅将网络输出概率值作为一种领域知识进行迁移, 没有考虑到教师网络结构对学生网络结构的影响. 提取教师网络的内部结构知识 (如神经元) 并指导学生网络的训练, 有可能使学生网络获得更高的性能.

4 精细模块设计

网络剪枝与稀疏化、张量分解、知识迁移等方法都是在已有高性能模型基础上, 保证模型性能的前提下降低时间复杂度和空间复杂度. 目前还有一些工作专注于设计高效的精细模块, 同样可以实现优化网络结构的目的. 基于这些精细模块构造的网络具有运行速度快、占用内存少、能耗低下的优点, 此外, 由于采用模块化的网络结构优化方法, 网络的设计与构造流程大幅缩短. 目前具有代表性的精细模块有 Inception 模块、网中网和残差模块, 本节对其进行了详尽讨论与分析.

4.1 Inception 模块

对于如何设计性能更好的卷积神经网络, 目前的主流观点是通过增加网络深度与宽度来扩大模型的规模. 但是这会带来两个无法避免的问题: 1) 随着网络尺寸的增加, 网络的训练参数也会大幅增加, 这在训练数据不足时不可避免地会带来过拟合问题; 2) 网络尺寸和训练参数的增加使得网络模型占用计算资源和内存资源过高的问题加剧, 将会导致训练速度降低, 难以应用于实际工程问题.

为解决以上问题, Szegedy 等[4] 从网中网 (Network in network, NiN)[55] 中得到启发, 提出了如图 5 所示的 Inception-v1 网络结构. 与传统卷积神经网络采用 11×11、9×9 等大尺寸卷积核不同, Inception-v1 大量并行使用 5×5、3×3 卷积核, 有效提升了网络的宽度, 并引入 1×1 卷积核为获取到的特征降维. Inception-v1 结构在增加卷积神经网络深度和宽度的同时, 并没有增加额外的训练参数. 此外, 将不同尺寸的卷积核并行连接能够增加特征提取的多样性, 而引入的 1×1 卷积核则加速了网络运行过程.

Ioffe 等[56] 认为, 卷积神经网络在训练时每层网络的输入分布都会发生改变, 这将会导致模型训练速度降低. 因此, 他们在 Inception-v1 的基础上提出了 Inception-v2 结构, 引入了批标准化 (Batch normalization, BN). 批标准化一般用于激活函数之前, 其最重要的作用是解决反向传播中的梯度问题

(包括梯度消失和梯度爆炸). 此外, 批标准化不仅允许使用更大的学习速率, 而且还简化了网络参数的初始化过程, 将人们从繁重的调参工作中解放出来. 最后, 由于批标准化具有正则化效果, 在某些情况下还可以减少对 Dropout 的需求.

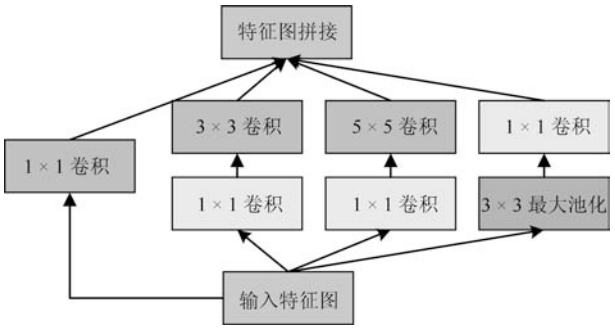


图 5　Inception-v1 结构[4]

Fig. 5　Inception-v1 module[4]

为进一步增加网络深度, Szegedy 等[57] 提出的 Inception-v3 网络借鉴了 VGGNet 的卷积核分解思想, 除了将 7×7、5×5 等较大的卷积核分解为若干连续的 3×3 卷积核, 还将 $n\times n$ 卷积核非对称分解为 $1\times n$ 和 $n\times1$ 两个连续卷积核 (当 $n=7$ 时, 效果最好). Inception-v3 还引入辅助分类器 (Auxiliary classifiers) 以加速卷积神经网络训练的收敛过程, 支持了 Inception-v2 中的批标准化具有正则化作用的观点. 通过卷积核分解, Inception-v3 不仅能够提升网络的深度和宽度, 而且有效降低了时间复杂度和空间复杂度. 此外, Inception-v3 加速训练过程并减轻了过拟合, 同时还强化了网络对不同维度特征的适应能力和非线性表达能力. 图 6 (a) 展示了将一个 5×5 的卷积核分解为两个连续 3×3 的卷积核后的计算过程, 由于一个 5×5 卷积核有 $5\times5=25$ 个参数, 而两个 3×3 卷积核只有 $3\times3+3\times3=18$ 个参数, 因此参数量降低了 28 % 而卷积效果相同; 图 6 (b) 展示了将一个 3×3 卷积核分解为一个 1×3 卷积核和一个 3×1 卷积核后的计算过程, 一个 3×3 卷积核有 $3\times3=9$ 个参数, 而两个分解后卷积核有 $1\times3+3\times1=6$ 个参数, 参数量降低了 33 % 而卷积效果相同.

Szegedy 等[58] 将 Inception 结构与残差结构相结合, 发现了残差结构可以极大地加快网络的训练速度, 提出的 Inception-Resnet-v1 和 Inception-Resnet-v2 模型在 ImageNet 数据集上的 Top-5 错误率分别降低到 4.3 % 和 3.7 %. 他们还提出了 Stem、Inception-A、Inception-B、Inception-C、Reduction-A、Reduction-B 等一系列网络局部结构, 并以此构造出 Inception-v4 网络模型, 极大地增加了网络深度, 提高了网络性能, 同时保证了网络训练参数数量处于可接受的范围之内.

Chollet 等[59] 认为传统的卷积过程同时从二维空间与一维通道进行三维的特征提取, 而 Inception-v3 部分地将空间操作与通道操作分离开, 使得训练过程更加容易且有效率. 从 Inception-v3 中得到启发, Chollet 认为卷积神经网络中特征图的空间维度与通道维度的关联性可以被完全解耦, 基于此他们提出了一种区别于一般卷积 (Regular convolution) 的 Xception (Extremely inception) 模块, 并以此构造出 Xception 网络结构. Xception 模块如图 7 所示, 首先用卷积核对输入特征图进行卷积操作, 对于输出特征图的每个通道都用一个卷积核进行卷积操作, 最后将所有输出拼接起来得到新的特征图. Xception 网络的训练参数比 Inception-v3 网络更少, 但具有与 Inception-v3 网络相当的识别精度和训练速度, 而且在更大的数据集上性能更加优越.

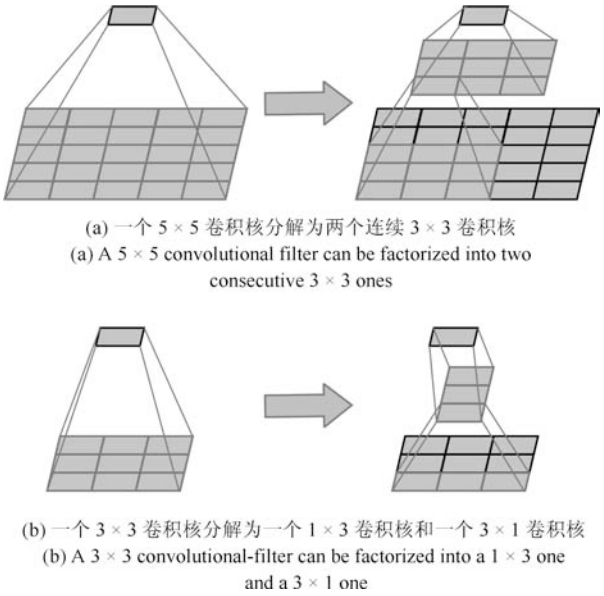


图 6　卷积核分解示意图[57]

Fig. 6　Process of convolutional filter factorization[57]

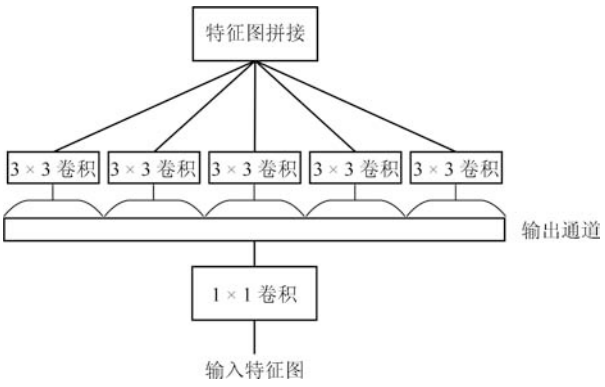


图 7　Xception 模块[59]

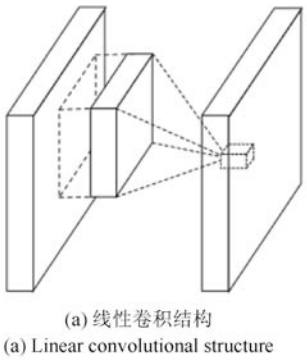
Fig. 7　Xception module[59]

Inception 结构从 Inception-v1 发展到 Xception, 始终致力于增加卷积神经网络的尺寸 (包括深度和宽度) 以提升模型的非线性表达能力. 为了避免训练参数增加而带来的模型训练速度降低、易过拟合等问题, Inception 结构提出了批标准化、卷积核分解等方法来优化更深层次的网络结构, 使得加深后的网络参数量相比于原始网络不变甚至更少, 训练出来的网络模型在各种测试数据集上都取得了领先成绩. Inception 的成功也进一步证明了增加网络尺寸是提升网络性能的可靠方式, 这也是卷积神经网络未来的一种发展方向.

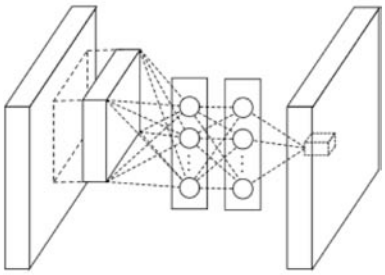
4.2 网中网 (Network in network)

传统卷积神经网络的卷积核作为一种广义线性模型 (Generalized linear model, GLM), 在训练样本的潜在特征是线性可分时能够获取表达能力较强的高维抽象特征. 但在很多任务场景下, 获取到的样本特征是具有较强非线性的, 使用传统的卷积核不能有效地提取更接近本质的抽象特征. Lin 等[55] 提出了一种区别于广义线性模型的非线性结构 — Mlpconv, 即在卷积核后面添加一个多层感知机 (Multilayer perceptron, MLP). 由于多层感知机能够拟合任何函数, 因此 Mlpconv 结构增强了网络对局部感知野的特征辨识能力和非线性表达能力. 通过堆叠 Mlpconv 层构建出的网络被形象地称为网中网 (Network in network, NiN), 如图 8 所示.

网中网不仅用 Mlpconv 结构替代广义线性模型以处理更为复杂的非线性问题, 并且用全局均值池化代替全连接层以减少训练参数, 避免了训练过程中出现过拟合问题. 值得注意的是, Mlpconv 层中的全连接层可以被视为一个 1×1 卷积核, 后来被广泛应用于包括 Inception 在内的各种网络中的 1×1 卷积核都受到了网中网的启发. 在此基础上, 涌现出了大量针对网中网结构的改进措施. Chang 等[60] 认为 Mlpconv 层中的 ReLU 激活函数会带来梯度消失的问题, 因此提出用 Maxout 替代 ReLU 以解决这一问题, 并将这一网络结构称为 Maxout network in network (MIN). Pang 等[61] 认为由于 MLP 本身也包含全连接网络, 这不可避免地会使得训练参数大幅增加, 因此提出用稀疏连接的 MLP 代替原来的 MLP, 并且在通道维度上使用分离卷积 (Unshared convolution) 而在空间维度上使用共享卷积 (Shared convolution), 这种网络结构被称为卷积中的卷积 (Convolution in convolution, CiC). Han 等[62] 提出的 MPNIN (Mlpconv-wise supervised pre-training network in network) 通过监督式预处理方法初始化网络模型的各层训练参数, 并结合批标准化与网中网结构能够训练更深层次的卷积神经网络.



(a) 线性卷积结构
(a) Linear convolutional structure



(b) 多层感知机卷积结构
(b) Mlpconv structure

图 8 线性卷积结构与多层感知机卷积结构[55]
Fig. 8 Linear convolutional structure and Mlpconv structure[55]

网中网结构一经提出就受到了广泛的关注和研究, 包括 GoogLeNet、ResNet 在内的众多卷积神经网络都借鉴了这一结构. 与传统 GLM 卷积核相比, 网中网的 Mlpconv 层可以实现跨通道的特征交互与整合, 由此发展而来的 1×1 卷积核还能实现特征降维与升维的功能, 使得网络模型既能够提取更加抽象的特征以解决复杂的非线性问题, 还可以训练更深层的网络而保持训练参数处于可接受范围. 值得注意的是, 由于 Mlpconv 结构引入了额外的多层感知机, 有可能会导致网络运行速度降低, 对此进行改善将会是未来研究的一个方向.

4.3 残差模块

随着卷积神经网络逐渐向更深层次发展, 网络将面临退化问题而不是过拟合问题, 具体表现在网络性能不再随着深度的增加而提升, 甚至在网络深度进一步增加的情况下性能反而快速下降, 此时引入一种称为旁路连接的 (Bypassing connection) 结构优化技术可有效解决这一问题. Srivastava 等[63] 从长短时记忆模型[64] (Long short-term memory, LSTM) 中得到启发, 引入可学习门限机制 (Learned gating mechanism) 以调节网络中的信息传播路径, 允许数据跨越多层网络进行传播, 这一模型被形象

地称为高速网络 (Highway network). 旁路连接使得反向传播中的梯度能够跨越一层或多层传播, 而不至于在逐层运算中扩散甚至消失, 在使用随机梯度下降法 (Stochastic gradient descent, SGD) 训练模型时避免了在平层网络 (Plain network) 中易出现的梯度消失现象. 旁路连接的引入, 突破了深度在达到 40 层时网络将面临退化问题的限制, 进一步促进了网络深度的增加[65].

He 等[5] 提出的残差网络 (Residual network, ResNet) 与 Highway network 类似, 也是允许输入信息可以跨越多个隐含层传播. 区别在于残差网络的门限机制不再是可学习的, 也即始终保持信息畅通状态, 这极大地降低了网络复杂度, 加速了网络训练过程, 同时突破了由网络退化引起的深度限制. 残差模块如图 9 所示, 残差模块的输入定义为 X, 输出定义为 $H(X) = F(X) + X$, 残差定义为 $F(X)$, 在训练过程中网络学习残差 $F(X)$, 这比直接学习输出 $H(X)$ 更加容易.

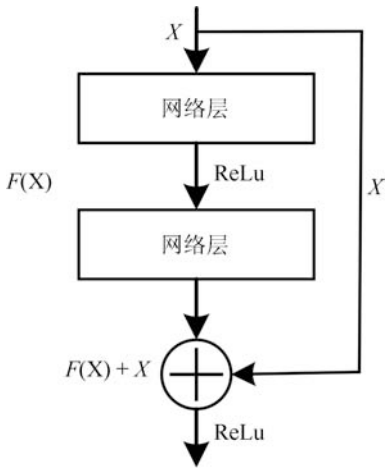


图 9　残差模块[5]

Fig. 9　Residual module[5]

残差网络的提出标志着卷积神经网络发展到了一个新阶段, 之后又有大量研究针对残差结构进行改进. Huang 等[66] 利用随机深度法 (Stochastic depth) 在训练过程中随机地剔除, 某些隐含层并用残差结构连接剩余部分, 训练出一个 1 202 层的极深残差网络, 同时表明原始的残差网络含有大量的冗余结构. He 等[67] 发现前置激活函数 (Pre-activation) 不仅使得模型优化更加容易, 而且, 在一定程度上缓解了过拟合. 作者以此训练了一个 1 001 层的残差网络, 在 CIFAR-10 数据集上的错误率降至 4.62 %. Larsson 等[65] 提出的分形网络 (FractalNet) 在宽度和深度上进一步扩展残差结构, 并用一种称为 Drop-path 的方法优化网络训练, 在图片分类测试中的正确率超过了残差网络. Xie 等[68] 提出的 ResNeXt 借鉴了 Inception 模块的思想, 通过增加旁路连接的数量以进一步扩宽网络, 在不增加网络复杂度的前提下提高识别准确率, 同时还减少了超参数的数量.

文献 [69] 认为残差网络仅仅是若干浅层网络的组合体, 其宽度相比于深度更为重要, 训练超过 50 层的网络是毫无必要的, 因此目前存在大量研究工作从网络宽度出发优化残差网络的结构. Zagoruyko 等[70] 认为 ResNet 在训练时无法充分地重用特征 (Feature reuse), 具体表现在梯度反向传播时不能流经每一个残差模块 (Residual block), 只有很少的残差模块可以学习到有用的特征表示. 作者提出的宽残差网络 (Wide residual network, WRN) 通过增加网络宽度并减少网络深度, 训练速度相较于残差网络提升了 2 倍, 但网络层数减少了 50 倍. Targ 等[71] 提出了一种将残差网络和标准卷积神经网络并行组合的泛化残差网络, 在保留有效特征表达的同时剔除了无效信息, 改善了网络的表达能力, 在 CIFAR-100 数据集上效果显著. Zhang 等[72] 为残差网络添加额外的旁路连接, 通过增加宽度以提高网络的学习能力, 提出的 Residual networks of residual networks (RoR) 可以作为构造网络的通用模块. Abdi 等[73] 通过实验支持了残差网络是若干浅层网络融合得到的假说, 作者提出的模型通过增加残差模块中残差函数的数量以增强模型的表达能力, 得到的多残差网络在 CIFAR-10 和 CIFAR-100 的分类准确率均得到极大改善.

4.4　其他精细模块

在网络结构的设计空间探索方面, 还有大量工作针对精细模块设计展开研究, 取得了一系列成果. 为减少全连接层的训练参数, 文献 [55] 首先提出用全局均值池化 (Global average pooling, GAP) 替代全连接层, 相当于在整个网络结构上做正则化防止过拟合. 全局均值池化在特征图与输出类别标签之间建立联系, 相比于全连接层更具有可解释性, 随后的网中网以及 GoogLeNet 都采用这一结构获得了性能提升.

Huang 等[74] 认为极深网络的成功来源于旁路连接的引入, 他们提出的密集模块 (Dense block) 在任何两层网络之间都有直接连接. 对于任意网络层, 它的输入来源于前面所有网络层的输出, 而它的输出都要作为后面所有网络层的输入. 这种密集连接改善了网络中信息与梯度的流动, 对于网络具有正则化的作用, 避免在小数据集上训练的过拟合问题. 密集连接的另一个优点是允许特征重用, 训练出来的 DenseNet 具有结构紧凑、精度高的优点. 张婷等[75] 提出的跨连卷积神经网络允许第二个池化层

跨过两层直接与全连接层相连接, 在 10 个人脸数据集上的性别分类效果都不低于传统网络. 李勇等[76] 将 LeNet-5 网络的两个池化层与全连接层相结合, 构造的分类器结合了网络结构提取的低层次特征与高层次特征, 在人脸表情识别中取得较好效果.

Howard 等[77] 提出的 MobileNet 将传统卷积过程分解为深度可分离卷积 (Depthwise convolution) 和逐点卷积 (Pointwise convolution) 两步, 在模型大小和计算量上都进行了大量压缩, 由此构造的轻量型网络能够在移动嵌入式设备上运行. Sandler 等[78] 将残差模块与深度可分离卷积相结合, 提出了带有线性瓶颈的反向残差模块 (Inverted residual with linear bottleneck), 由此构造的 MobileNet v2 在速度和准确性上都优于 MobileNet. Zhang 等[79] 在 MobileNet 的基础上进一步提出了基于逐点群卷积 (Pointwise group convolution) 和通道混洗 (Channel shuffle) 的 ShuffleNet, 在图像分类和目标检测任务中均获得极大提速.

5 结束语

随着硬件条件的飞速发展和数据集规模的显著增长, 深度卷积神经网络目前已成为计算机视觉、语音识别、自然语言处理等研究领域的主流方法. 具体地, 更深的网络层数增强了模型的非线性拟合能力, 同时大规模数据增强了模型的泛化能力, 而较高水平的硬件设施条件则保证了模型运行所需要的计算能力和存储要求. 深度卷积神经网络已在诸多领域证明了强大的特征学习和表达能力, 但高昂的时间复杂度和空间复杂度制约其在更广阔领域的实施与应用. 在时间维度上, 大型复杂网络计算量巨大, 在图形处理单元 (Graphic processing unit, GPU) 加速运算的支持下, 仍不能满足自动驾驶汽车等一些强实时场景的要求. 在空间维度上, 随着模型规模日益庞大特别是网络深度剧增, 对模型的存储提出了更高的要求, 这制约了深度卷积神经网络在移动手机、嵌入式设备等资源受限环境的应用.

为加快以卷积神经网络为代表的深度学习技术的推广及应用, 进一步强化在安防、移动设备、自动驾驶等多个行业的优势, 学术界和工业界对其结构的优化展开了大量研究. 现阶段常用的网络结构优化技术包括网络剪枝与稀疏化、张量分解、知识迁移和精细模块设计, 前三种方法通常是在已有高性能模型的基础上改进并加以创新, 在不损害精度甚至有所提高的前提下尽可能降低模型复杂度和计算复杂度. 精细模块设计方法从网络构造的角度出发, 创造性地设计高效模块以提升网络性能, 从根本解决深度卷积神经网络面临的时间复杂度和空间复杂度过高的问题. 笔者整理了近几年的研究成果, 根据自己的理解总结了该领域以下的难点问题以及发展趋势:

1) 网络剪枝与稀疏化能够稳定地优化并调整网络结构, 以较小精度损失的代价压缩网络规模, 是应用最为广泛的网络结构优化设计方法. 目前大多数的方法是剔除网络中冗余的连接或神经元, 这种低层级的剪枝具有非结构化 (Non-structural) 风险, 在计算机运行过程中的非正则化 (Irregular) 内存存取方式反而会阻碍网络进一步加速. 一些特殊的软硬件措施能够缓解这一问题, 然而会给模型的部署带来额外的花销. 另一方面, 尽管一些针对卷积核和卷积图的结构化剪枝方法能够获得硬件友好型网络, 在 CPU 和 GPU 上速度提升明显, 但由于剪枝卷积核和卷积通道会严重影响下一隐含层的输入, 有可能存在网络精度损失严重的问题.

2) 目前主流的精细模块设计方法仍然依赖于设计者的工程经验和理论基础, 在网络构造过程中要考虑到大量因素, 如卷积核尺寸、全连接层数、池化层数等超参数 (Hyper parameter). 不同的选择对于网络最终性能有可能造成完全不同的影响, 需要进行大量的实验来论证不同参数的优劣, 使得网络结构设计耗费大量的人力物力, 不利于深度模型的快速部署及应用. 因此, 研究如何自动设计网络有助于卷积神经网络的设计空间探索 (Design space exploration, DSE), 对于加快网络设计过程和推动深度学习落地于工程化应用具有重要的促进作用.

3) 网络结构优化设计的评价指标. 目前对于深度卷积神经网络的结构优化设计主要侧重于准确率、运行时间、模型大小等方面的评价, 但使用更加全面的评价指标对于发现不同网络的优点和缺点是大有裨益的. 除了准确率、运行时间、模型大小等传统指标, 有必要将乘加 (Multiply-and-accumulate) 操作量、推导时间、数据吞吐量、硬件能耗等指标纳入评价体系, 这为从不同方面评价优化模型提供了更加完备的信息, 也有助于解决了不同网络性能评价指标不统一的问题.

4) 在过去, 深度卷积神经网络的结构优化更多着眼于算法的设计与实现, 而对于模型的具体部署平台和硬件设施欠缺考虑. 考虑到硬件条件仍是制约着深度模型部署于移动手机、机器人、自动驾驶等资源受限场景下的主要因素, 若统筹兼顾网络模型和硬件设施的优化与设计, 使算法与硬件相匹配, 不仅能够进一步提高数据吞吐量与运行速度, 还可以减少网络规模与能耗. 因此, 设计硬件友好型深度模型将有助于加速推进深度学习的工程化实现, 也是网络结构优化的重点研究方向.

5) 本文归纳与总结的网络结构优化方法有不同的侧重点和局限性, 其中网络剪枝与稀疏化方法能

够获得较大的压缩比, 同时对于网络精度的影响较小, 在需要模型稳定运行的场景下较为适用. 张量分解能够极大加速模型的运行过程, 而且端到端的逐层优化方式也使其容易实施, 然而该方法不能较好地压缩模型规模, 而且在卷积核尺寸较小时加速效果不明显. 知识迁移方法能够利用教师网络的领域知识指导学生网络的训练, 在小样本环境下有较高的使用价值. 同时, 知识迁移和精细模块设计都面临网络结构如何构造的问题, 要求设计者具有较高的理论基础和工程经验, 与其他方法相比其调试周期较长. 因此, 在使用网络结构优化技术时应考虑实际情况, 综合应用以上方法以压缩并加速网络.

6) 深度神经网络结构优化的迁移应用. 本文分析了卷积神经网络目前存在的挑战和问题, 并且探讨了卷积神经网络结构优化领域的主流方法、思想及其应用. 由于目前其他主流的深度网络 (如循环神经网络、生成对抗网络) 同样面临模型规模大、运行速度慢的问题, 因此借鉴卷积神经网络结构优化的思想以优化其模型是一种有效的解决方式. 此外, 目前很多优化方法一般都是针对图像分类问题, 若将其应用于目标检测、语义分割等领域也应取得较好效果.

References

1 Krizhevsky A, Sutskever I, Hinton G E. ImageNet classification with deep convolutional neural networks. In: Proceedings of the 25th International Conference on Neural Information Processing Systems. Lake Tahoe, Nevada, USA: Curran Associates Inc., 2012. 1097−1105

2 Zeiler M D, Fergus R. Visualizing and understanding convolutional networks. In: Proceedings of 13th European Conference on Computer Vision. Zurich, Switzerland: Springer, 2014. 818−833

3 Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. arXiv: 1409.1556, 2014.

4 Szegedy C, Liu W, Jia Y Q, Sermanet P, Reed S, Anguelov D, et al. Going deeper with convolutions. In: Proceedings of 2015 IEEE Conference on Computer Vision and Pattern Recognition. Boston, MA, USA: IEEE, 2015. 1−9

5 He K M, Zhang X Y, Ren S Q, Sun J. Deep residual learning for image recognition. In: Proceedings of 2016 IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas, NV, USA: IEEE, 2016. 770−778

6 LeCun Y, Bottou L, Bengio Y, Haffner P. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 1998, **86**(11): 2278−2324

7 He K M, Sun J. Convolutional neural networks at constrained time cost. In: Proceedings of 2015 IEEE Conference on Computer Vision and Pattern Recognition. Boston, USA: IEEE, 2015. 5353−5360

8 Le Cun Y, Denker J S, Solla S A. Optimal brain damage. In: Proceedings of the 2nd International Conference on Neural Information Processing Systems. Denver, Colorado, USA: MIT Press, 1989. 598−605

9 Hassibi B, Stork D G, Wolff G, Watanabe T. Optimal brain surgeon: extensions and performance comparisons. In: Proceedings of the 6th International Conference on Neural Information Processing Systems. Denver, Colorado, USA: Morgan Kaufmann Publishers Inc., 1993. 263−270

10 Cheng Y, Wang D, Zhou P, Zhang T. A survey of model compression and acceleration for deep neural networks. arXiv: 1710.09282, 2017.

11 Cheng J, Wang P S, Li G, Hu Q H, Lu H Q. Recent advances in efficient computation of deep convolutional neural networks. *Frontiers of Information Technology & Electronic Engineering*, 2018, **19**(1): 64−77

12 Lei Jie, Gao Xin, Song Jie, Wang Xing-Lu, Song Ming-Li. Survey of deep neural network model compression. *Journal of Software*, 2018, **29**(2): 251−266
(雷杰, 高鑫, 宋杰, 王兴路, 宋明黎. 深度网络模型压缩综述. 软件学报, 2018, **29**(2): 251−266)

13 Hu H Y, Peng R, Tai Y W, Tang C K. Network trimming: a data-driven neuron pruning approach towards efficient deep architectures. arXiv: 1607.03250, 2016.

14 Cheng Y, Wang D, Zhou P, Zhang T. Model compression and acceleration for deep neural networks: the principles, progress, and challenges. *IEEE Signal Processing Magazine*, 2018, **35**(1): 126−136

15 Gong Y C, Liu L, Yang M, Bourdev L. Compressing deep convolutional networks using vector quantization. arXiv: 1412.6115, 2014.

16 Reed R. Pruning algorithms-a survey. *IEEE Transactions on Neural Networks*, 1993, **4**(5): 740−747

17 Collins M D, Kohli P. Memory bounded deep convolutional networks. arXiv: 1412.1442, 2014.

18 Jin X J, Yuan X T, Feng J S, Yan S C. Training skinny deep neural networks with iterative hard thresholding methods. arXiv: 1607.05423, 2016.

19 Zhou H, Alvarez J M, Porikli F. Less is more: towards compact CNNs. In: Proceedings of the 14th European Conference on Computer Vision. Amsterdam, The Netherlands: Springer, 2016. 662−677

20 Wen W, Wu C P, Wang Y D, Chen Y R, Li H. Learning structured sparsity in deep neural networks. In: Proceedings of the 30th Conference on Neural Information Processing Systems. Barcelona, Spain: MIT Press, 2016. 2074−2082

21 Lebedev V, Lempitsky V. Fast convnets using group-wise brain damage. In: Proceedings of 2016 IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas, USA: IEEE, 2016. 2554−2564

22 Louizos C, Welling M, Kingma D P. Learning sparse neural networks through L_0regularization. arXiv: 1712.01312, 2017.

23 Hinton G E, Srivastava N, Krizhevsky A, Sutskever I, Salakhutdinov R R. Improving neural networks by preventing co-adaptation of feature detectors. arXiv: 1207.0580, 2012.

24 Srivastava N, Hinton G, Krizhevsky A, Sutskever I, Salakhutdinov R. Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 2014, **15**(1): 1929−1958

25 Li Z, Gong B Q, Yang T B. Improved dropout for shallow and deep learning. In: Proceedings of the 30th Conference on Neural Information Processing Systems. Barcelona, Spain: MIT Press, 2016. 2523−2531

26 Anwar S, Sung W. Coarse pruning of convolutional neural networks with random masks. In: Proceedings of 2017 International Conference on Learning Representations. Toulon, France: 2017. 134−145

27 Hanson S J, Pratt L Y. Comparing biases for minimal network construction with back-propagation. In: Proceedings of the 1st International Conference on Neural Information Processing Systems. Denver, Colorado, USA: MIT Press, 1988. 177−185

28 Han S, Mao H Z, Dally W J. Deep compression: compressing deep neural networks with pruning, trained quantization and huffman coding. arXiv: 1510.00149, 2015.

29 Srinivas S, Babu R V. Data-free parameter pruning for deep neural networks. arXiv: 1507.06149, 2015.

30 Guo Y W, Yao A B, Chen Y R. Dynamic network surgery for efficient DNNs. In: Proceedings of the 30th Conference on Neural Information Processing Systems. Barcelona, Spain: MIT Press, 2016. 1379−1387

31 Liu X Y, Pool J, Han S, Dally W J. Efficient sparse-winograd convolutional neural networks. In: Proceedings of 2017 International Conference on Learning Representation. France: 2017.

32 He Y H, Zhang X Y, Sun J. Channel pruning for accelerating very deep neural networks. In: Proceedings of 2017 IEEE International Conference on Computer Vision. Venice, Italy: IEEE, 2017. 1398−1406

33 Li H, Kadav A, Durdanovic I, Samet H, Graf H P. Pruning filters for efficient convNets. arXiv: 1608.08710, 2016.

34 Luo J H, Wu J X, Lin W Y. Thinet: a filter level pruning method for deep neural network compression. In: Proceedings of 2017 IEEE International Conference on Computer Vision. Venice, Italy: IEEE, 2017. 5068−5076

35 Denil M, Shakibi B, Dinh L, Ranzato M, de Freitas N. Predicting parameters in deep learning. In: Proceedings of the 26th International Conference on Neural Information Processing Systems. Lake Tahoe, Nevada, USA: Curran Associates Inc., 2013. 2148−2156

36 Rigamonti R, Sironi A, Lepetit V, Fua P. Learning separable filters. In: Proceedings of 2013 IEEE Conference on Computer Vision and Pattern Recognition. Portland, OR, USA: IEEE, 2013. 2754−2761

37 Jaderberg M, Vedaldi A, Zisserman A. Speeding up convolutional neural networks with low rank expansions. arXiv: 1405.3866, 2014.

38 Denton E, Zaremba W, Bruna J, LeCun Y, Fergus R. Exploiting linear structure within convolutional networks for efficient evaluation. In: Proceedings of the 27th International Conference on Neural Information Processing Systems. Montreal, Canada: MIT Press, 2014. 1269−1277

39 Lebedev V, Ganin Y, Rakhuba M, Oseledets I, Lempitsky V. Speeding-up convolutional neural networks using fine-tuned CP-decomposition. arXiv: 1412.6553, 2014.

40 Tai C, Xiao T, Zhang Y, Wang X G, E W N. Convolutional neural networks with low-rank regularization. arXiv: 1511.06067, 2015.

41 Zhang X Y, Zou J H, Ming X, He K M, Sun J. Efficient and accurate approximations of nonlinear convolutional networks. In: Proceedings of 2015 IEEE Conference on Computer Vision and Pattern Recognition. Boston, USA: IEEE, 2015. 1984−1992

42 Kim Y D, Park E, Yoo S, Choi T, Yang L, Shin D. Compression of deep convolutional neural networks for fast and low power mobile applications. arXiv: 1511.06530, 2015.

43 Wang Y H, Xu C, Xu C, Tao D C. Beyond filters: compact feature map for portable deep model. In: Proceedings of the 34th International Conference on Machine Learning. Sydney, Australia: JMLR.org, 2017. 3703−3711

44 Astrid M, Lee S I. CP-decomposition with tensor power method for convolutional neural networks compression. In: Proceedings of 2017 IEEE International Conference on Big Data and Smart Computing. Jeju, South Korea: IEEE, 2017. 115−118

45 Buciluǎ C, Caruana R, Niculescu-Mizil A. Model compression. In: Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Philadelphia, USA: ACM, 2006. 535−541

46 Ba J, Caruana R. Do deep nets really need to be deep? In: Proceedings of Advances in Neural Information Processing Systems. Montreal, Quebec, Canada: MIT Press, 2014. 2654−2662

47 Hinton G, Vinyals O, Dean J. Distilling the knowledge in a neural network. arXiv: 1503.02531, 2015.

48 Romero A, Ballas N, Kahou S E, Chassang A, Gatta C, Bengio Y. Fitnets: hints for thin deep nets. arXiv: 1412.6550, 2014.

49 Luo P, Zhu Z Y, Liu Z W, Wang X G, Tang X O. Face model compression by distilling knowledge from neurons. In: Proceedings of the 30th AAAI Conference on Artificial Intelligence. Phoenix, Arizona, USA: AAAI, 2016. 3560−3566

50 Chen T Q, Goodfellow I, Shlens J. Net2Net: accelerating learning via knowledge transfer. arXiv: 1511.05641, 2015.

51 Zagoruyko S, Komodakis N. Paying more attention to attention: improving the performance of convolutional neural networks via attention transfer. In: Proceedings of 2017 International Conference on Learning Representations. France: 2017.

52 Theis L, Korshunova I, Tejani A, Huszár F. Faster gaze prediction with dense networks and Fisher pruning. arXiv: 1801.05787, 2018.

53 Yim J, Joo D, Bae J, Kim J. A gift from knowledge distillation: fast optimization, network minimization and transfer learning. In: Proceedings of 2017 IEEE Conference on Computer Vision and Pattern Recognition. Honolulu, HI, USA: IEEE, 2017.

54 Chen G B, Choi W, Yu X, Han T, Chandraker M. Learning efficient object detection models with knowledge distillation. In: Proceedings of the 31st International Conference on Neural Information Processing Systems. Long Beach, USA: Curran Associates Inc., 2017. 742−751

55 Lin M, Chen Q, Yan S C. Network in network. arXiv: 1312.4400, 2013.

56 Ioffe S, Szegedy C. Batch normalization: accelerating deep network training by reducing internal covariate shift. arXiv: 1502.03167, 2015.

57 Szegedy C, Vanhoucke V, Ioffe S, Shlens J, Wojna Z. Rethinking the inception architecture for computer vision. In: Proceedings of 2016 IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas, NV, USA: IEEE, 2016. 2818−2826

58 Szegedy C, Ioffe S, Vanhoucke V, Alemi A A. Inception-v4, inception-resnet and the impact of residual connections on learning. In: Proceedings of the 31st AAAI Conference on Artificial Intelligence. San Francisco, USA: AAAI, 2017. 12

59 Chollet F. Xception: deep learning with depthwise separable convolutions. In: Proceedings of 2017 IEEE Conference on Computer Vision and Pattern Recognition. Honolulu, HI, USA: IEEE, 2017.

60 Chang J R, Chen Y S. Batch-normalized maxout network in network. arXiv: 1511.02583, 2015.

61 Pang Y W, Sun M L, Jiang X H, Li X L. Convolution in convolution for network in network. *IEEE transactions on neural networks and learning systems*, 2018, **29**(5): 1587−1597

62 Han X M, Dai Q. Batch-normalized mlpconv-wise supervised pre-training network in network. *Applied Intelligence*, 2018, **48**(1): 142−155

63 Srivastava R K, Greff K, Schmidhuber J. Highway networks. arXiv: 1505.00387, 2015.

64 Hochreiter S, Schmidhuber J. Long short-term memory. *Neural Computation*, 1997, **9**(8): 1735−1780

65 Larsson G, Maire M, Shakhnarovich G. Fractalnet: ultra-deep neural networks without residuals. arXiv: 1605.07648, 2016.

66 Huang G, Sun Y, Liu Z, Sedra D, Weinberger K Q. Deep networks with stochastic depth. In: Proceedings of the 14th European Conference on Computer Vision. Amsterdam, The Netherlands: Springer, 2016. 646−661

67 He K M, Zhang X Y, Ren S Q, Sun J. Identity mappings in deep residual networks. In: Proceedings of the 14th European Conference on Computer Vision. Amsterdam, The Netherlands: Springer, 2016. 630−645

68 Xie S N, Girshick R, Dollár P, Tu Z W, He K M. Aggregated residual transformations for deep neural networks. In: Proceedings of 2017 IEEE Conference on Computer Vision and Pattern Recognition. Honolulu, HI, USA: IEEE, 2017. 5987−5995

69 LeCun Y, Bengio Y, Hinton G. Deep learning. *Nature*, 2015, **521**(7553): 436−444

70 Zagoruyko S, Komodakis N. Wide residual networks. arXiv: 1605.07146, 2016.

71 Targ S, Almeida D, Lyman K. Resnet in resnet: generalizing residual architectures. arXiv: 1603.08029, 2016.

72 Zhang K, Sun M, Han T X, Yuan X F, Guo L R, Liu T. Residual networks of residual networks: multilevel residual networks. *IEEE Transactions on Circuits and Systems for Video Technology*, 2018, **28**(6): 1303−1314

73 Abdi M, Nahavandi S. Multi-residual networks: improving the speed and accuracy of residual networks. arXiv: 1609.05672, 2016.

74 Huang G, Liu Z, van der Maaten L, Weinberger K Q. Densely connected convolutional networks. In: Proceedings of 2017 IEEE Conference on Computer Vision and Pattern Recognition. Honolulu, HI, USA: IEEE, 2017.

75 Zhang Ting, Li Yu-Jian, Hu Hai-He, Zhang Ya-Hong. A gender classification model based on cross-connected convolutional neural networks. *Acta Automatica Sinica*, 2016, **42**(6): 858−865
(张婷, 李玉鑑, 胡海鹤, 张亚红. 基于跨连卷积神经网络的性别分类模型. 自动化学报, 2016, **42**(6): 858−865)

76 Li Yong, Lin Xiao-Zhu, Jiang Meng-Ying. Facial expression recognition with cross-connect LeNet-5 network. *Acta Automatica Sinica*, 2018, **44**(1): 176−182
(李勇, 林小竹, 蒋梦莹. 基于跨连接 LeNet-5 网络的面部表情识别. 自动化学报, 2018, **44**(1): 176−182)

77 Howard A G, Zhu M L, Chen B, Kalenichenko D, Wang W J, Weyand T, et al. Mobilenets: efficient convolutional neural networks for mobile vision applications. arXiv: 1704.04861, 2017.

78 Sandler M, Howard A, Zhu M L, Zhmoginov A, Chen L C. MobileNetV2: inverted residuals and linear bottlenecks. In: Proceedings of 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Salt Lake City, USA: IEEE, 2018. 4510−4520

79 Zhang X Y, Zhou X Y, Lin M X, Sun J. ShuffleNet: an extremely efficient convolutional neural network for mobile devices. In: Proceedings of 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Salt Lake City, USA: IEEE, 2018.

林景栋 重庆大学自动化学院副教授. 2002 年获得重庆大学博士学位. 主要研究方向为工业自动化生产线设计, 智能家居控制系统的设计.
E-mail: linzhanding@163.com
(**LIN Jing-Dong** Associate professor at the College of Automation, Chongqing University. He received his Ph. D. degree from Chongqing University in 2002. His research interest covers industrial automation line design, and smart home control system design.)

吴欣怡 重庆大学自动化学院硕士研究生. 2016 年获得重庆大学学士学位. 主要研究方向为深度学习, 计算机视觉.
E-mail: wuxinyi12358@gmail.com
(**WU Xin-Yi** Master student at the College of Automation, Chongqing University. He received his bachelor degree from Chongqing University in 2016. His research interest covers deep learning and computer vision.)

柴 毅 重庆大学自动化学院教授. 2001 年获得重庆大学博士学位. 主要研究方向为信息处理, 融合与控制, 计算机网络与系统控制. E-mail: chaiyi@cqu.edu.cn
(**CHAI Yi** Professor at the College of Automation, Chongqing University. He received his Ph. D. degree from Chongqing University in 2001. His research interest covers information processing, integration and control, and computer network and system control.)

尹宏鹏 重庆大学自动化学院教授. 2009 年获得重庆大学博士学位. 主要研究方向为模式识别与智能系统. 本文通信作者.
E-mail: yinhongpeng@gmail.com
(**YIN Hong-Peng** Professor at the College of Automation, Chongqing University. He received his Ph. D. degree from Chongqing University in 2009. His research interest covers pattern recognition, image processing, and computer vision. Corresponding author of this paper.)