综述

高效深度神经网络综述

闵锐

（中国电信股份有限公司智能网络与终端研究院，广东 广州 510630）

摘　要：近年来，深度神经网络（DNN）在计算机视觉、自然语言处理等 AI 领域中取得了巨大的成功。得益于更深更大的网络结构，DNN 的性能正在迅速提升。然而，更深更大的深度神经网络需要巨大的计算和内存资源，在资源受限的场景中，很难部署较大的神经网络模型。如何设计轻量并且高效的深度神经网络来加速其在嵌入式设备上的运行速度，对于推进深度神经网络技术的落地意义巨大。对近年来具有代表性的高效深度神经网络的研究方法和工作进行回顾和总结，包括参数剪枝、模型量化、知识蒸馏、网络搜索和量化。同时分析了不同方法的优点和缺点以及适用场景，并且展望了高效神经网络设计的发展趋势。

关键词：深度神经网络；模型压缩与加速；知识蒸馏

中图分类号：TP393

文献标识码：A

doi: 10.11959/j.issn.1000−0801.2020119

A survey of efficient deep neural network

MIN Rui

Intelligent Network and Terminal Research Institute, China Telecom Co., Ltd., Guangzhou 510630, China

Abstract: Recently, deep neural network (DNN) has achieved great success in the field of AI such as computer vision and natural language processing. Thanks to a deeper and larger network structure, DNN’s performance is rapidly increasing. However, deeper and lager deep neural networks require huge computational and memory resources. In some resource-constrained scenarios, it is difficult to deploy large neural network models. How to design a lightweight and efficient deep neural network to accelerate its running speed on embedded devices is a great research hotspot for advancing deep neural network technology. The research methods and work of representative high-efficiency deep neural networks in recent years were reviewed and summarized, including parameter pruning, model quantification, knowledge distillation, network search and quantification. Also, vadvantages and disadvantages of different methods as well as applicable scenarios were analyzed, and the future development trend of efficient neural network design was forecasted.

Key words: deep neural network, model accelerator and compression, knowledge distillation

收稿日期：2019−11−11；修回日期：2020−03−26

1 引言

近年以来，深度神经网络（deep neural network，DNN）不断应用于计算机视觉、语音识别和自然语言处理等 AI 领域，取得了巨大的成功。得益于更深更大的网络结构，DNN 的性能正在迅速提升：在 ImageNe-1K challenge 上 Top-1 分类准确率从 2012 年的 63.3%[1]提升到 2019 年的 86.4%[2]。然而，伴随着网络性能的提升，网络也变得越来越深，极大地增加了网络所需要的计算开销，为了追求性能而设计更大更深的网络结构使得模型对于存储开销的需求也越来越大。这严重制约着深度神经网络的应用在嵌入式设备中的应用。例如，对于像素大小为 224×224 的图像，ALEXNET[1]有 5 个卷积层和 3 个全连接层，共 6.1×10^8 个参数量。以每个单浮点数占用 4 byte 计算，AlexNet 总共需要占用 240 MB 内存和 7.29×10^9 次浮点运算；VGG19[3]共 24 层，参数量为 1.44×10^9 个 ，所需要的计算量为 1.9×10^{11} 次浮点运算，而 ResNeXt-152[4]更是达到了 152 层，需要 2.26×10^{11} 次浮点运算，所需的运算量比 AlexNet 高出一个数量级。尽管模型可以利用 GPU 的强大并行计算能力来加速模型推理时间，在一些资源有限的嵌入式设备中，很难满足实时性要求。

一般而言，超大深度网络内部存在着大量的冗余信息。Lecun 等[5]实验显示，当训练数据有限时，过多的参数量导致网络的泛化能力下降。将一个深度神经网络删除一半甚至一半以上的权重，对于网络的性能影响甚微，甚至在某些情况下能提升网络的泛化能力。然而，轻量的神经网络很难达到复杂神经网络的性能。因此，如何设计高效的深度神经网络，使模型的参数量和所需的计算量较少，同时在性能上能接近甚至超过较大的神经网络，意义巨大，是目前深度神经网络领域的研究热点之一。

通常而言，设计高效的深度神经网络包含两个目标：模型压缩和加速。两者既有所区别又联系紧密。例如，目前最为普遍的卷积神经网络（convolutional neural network，CNN）包含两种不同类型的计算层：卷积层和全连接层。卷积层所需的计算量非常大，通过逐层抽象来获取高层语义信息，可以通过权值共享来减少参数量；相对于卷积层，全连接层所需的计算量更少，但是通过密集的权值进行连接，因此需要大量的参数。前者能降低对内存的需求，减少常驻内存；后者能降低网络的推理时延，提高响应速度。

目前压缩和加速深度神经网络模型主要可以分成如下 5 种方法：

- 参数剪枝（parameter pruning）；
- 低秩分解（low-rank decomposition）；
- 知识蒸馏 （knowledge distillation）；
- 网络架构搜索（neural architecture search，NAS）；
- 模型量化（model quantization）。

参数剪枝主要通过移除神经网络中不重要的参数，减少网络的参数量。低秩分解则通过对卷积核或全连接层进行矩阵分解来降低参数量。需要注意的是，参数剪枝和低秩分解都需要对已有网络的结构进行一定的改动。知识蒸馏不需要对网络进行任何改动，而是通过性能更高的大网络来指导小网络的训练，提升小网络的性能。知识蒸馏并不会对模型进行加速或者压缩，更像是一种改进的训练方式来提升模型性能。模型量化则通过使用更少的位数来表示模型参数和中间输出值，比如使用 8 bit 来表示浮点数（通常是 32 bit 或 64 bit）。量化并不能减少参数量，但是可以显著减少模型的大小。另外，当与特定的硬件结合（比如高通 DSP 芯片提供了 8 bit 运算）时，低比特的模型能获得显著的加速。网络架构搜索 NAS 则通过自动调优来寻找更好的网络架构，其本质是通过自动化搜索来替代了人工调优过程。对 5 种方法进行了简要介绍见表 1。此外，也可以通过设计精简的结构化卷积核或计

表 1　不同深度神经网络压缩与加速方法总结

方法	原理	代表工作	改动网络	缺点
参数剪枝	判断参数重要与否，移除不重要的参数	Taylor 剪枝	需要	实际加速效果有限
低秩分解	对卷积核或全连接进行矩阵或张量分解	SVD 分解	需要	难以分解精简卷积核和较小的卷积核
		Tucker 分解		
知识蒸馏	利用已训练好的大型网络的输出作为 soft-target 来替代 one-hot 标签	FitNets、DML	不需要	不能直接加速网络
网络架构搜索	自动搜索满足要求的高效网络结构	DARTS、FBNet	不需要	搜索过程漫长
模型量化	利用更少的比特来表示模型的权重和中间值	二值网络	不需要	加速需要特殊硬件支持

算单元，以减少计算和存储开销，这部分代表性的工作有 MobileNet[6-7]和 ShuffleNet[8]。

本文主要对计算机视觉领域中的高效神经网络进行设计，包含网络的压缩和加速，进行回顾并总结，详细介绍不同方法的经典工作和优缺点，并介绍最新的进展，为今后的相关研究提供参考，并展望该领域未来可能存在的发展方向。

2　深度神经网络发展历史

深度神经网络的发展过程历经了理论提出和完善、模型实践和广泛应用阶段。

2.1　理论提出和完善阶段

深度神经网络的发展最早可以追溯到20世纪40 年代。1943 年，Mcculloch 和 Pitts[9]首次提出 M-P 神经元模型。在 M-P 模型中，神经元作为功能逻辑器件来实现算法，开创了神经网络的理论研究。以 M-P 模型为基础，Rosenblatt[10]提出了感知机（Perceptron）模型，这一模型首次将神经网络的学习功能用于模式识别的装置并获得成功。随着研究的深入，Marvin 和 Seymour[11]证明了单层神经网络无法完成简单的“异或”逻辑问题，并指出单层神经网络无法完成复杂的函数功能，随之神经网络领域出现了长达 10 年的低潮期。1982 年 Hopfield[12]提出了 Hopfield 模型，以此为基础求解了非多项式复杂度的旅行商问题。此后，神经网络再次成为了一个研究热点，出现了大量经典的网络模型。基于“隐单元”的概念，Hinton 等[13]设计了一个由随机神经元全连接组成的多层神经网络 Boltzmann 机。Mcclelland[14]针对多层神经网络难以训练和学习时间过长的问题，提出分布并行处理和有效的权值调整算法。

2.2　模型实践阶段

自 20 世纪 40 年代被首次提出以后，神经网络历经了几十年的理论发展并完善，但是一直没有得到实践应用。20 世纪 90 年代，Lecun 等[15]提出了卷积网络（LeNet5）模型，利用该模型顺利地解决了手写体字符识别的问题。LeNet5 网络是卷积神经网络架构的起点，但是由于缺乏有效的训练算法，更多层的神经网络难以训练，制约了神经网络的更进一步的应用。为了解决这个问题，Hinton 等[16]提出可以通过逐层初始化，解决深层网络难以训练这个问题。此外，他们也发现具有多隐层的神经网络学到的特征可以刻画数据的本质属性，并且可以利用特征来对数据进行可视化和分类。在当时，这两个发现再一次掀起了深度学习在学术界和工业界的研究热潮。

2.3　广泛应用阶段

2012 年，深度卷积网络 AlexNet 以大幅超出第二名 11%的巨大优势在 ImageNet 图像分类榜单上，夺得冠军，卷积神经网络成为焦点。此后，

在图像分类任务上，深度卷积网络逐渐占了主导位置，并且在效果上相对于传统方法深度卷积网络优势巨大。2014 年，Taigman 等[17]和 Sun 等[18]在 LFW 上分别得到了 97.35%和 97.45% 的人脸识别精度，这是深度神经网络在人脸识别问题上首次超越人类的识别水平。此后，深度神经网络主导了人脸识别领域，代表性的方法有 FaceNet[19]、ArcFace[20]等。目前人脸识别是深度神经网络商业化落地最为成功的领域之一，孕育出了商汤、旷视、依图等独角兽创业公司。人脸识别领域的巨大成功推动了行人再识别（person re-identification）的研究热潮。行人再识别与人脸识别问题较为类似，很多在人脸识别问题上行之有效的方法，都能应用于行人再识别上，如 center loss[21]、triplet loss[17]等。Girshick 等[22]首次将深度神经网络引入目标检测领域，其提出的 R-CNN 极大地提高了目标检测的效果。此后，有一系列针对 R-CNN 进行改进的工作，如 Fast RCNN[23]、Faster RCNN[24]、Mask-RCNN[25]等。此外，深度神经网络在视频动作识别[26]、自然语言处理[27-28]等诸多领域同样取得了巨大进展。

3 高效深度神经网络

通常可以通过增加深度和参数量，增加神经网络的表达能力并提升性能，但是这也意味着更多的计算负载和内存需求。在一些资源受限的场景中，如智能穿戴设备等嵌入式设备上，难以直接部署大型神经网络模型。为了解决这个问题，过去的几年时间出现了大量解决方法，包括参数剪枝、低秩分解、知识蒸馏、网络搜索和模型量化。

3.1 参数剪枝

一般而言，神经网络存在着大量的冗余信息参数，剪枝通过对训练好的模型进行分析并且移除不重要的参数和连接，可以有效减少模型的复杂度，尤其是模型的参数量。参数剪枝的示意图如图 1 所示。参数剪枝的核心是有效评估模型参数重要性。

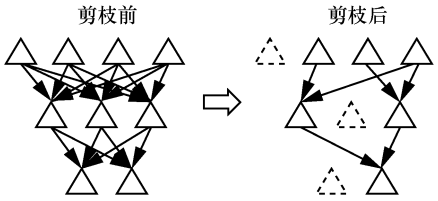

图 1 参数剪枝

（1）非结构化剪枝

早在 20 世纪 90 年代，Lecun 等[5]就提出了一种非均衡权重衰减方法，该方法利用目标损失函数对参数的二阶导数来表示参数的贡献度。同时，采用了一种迭代式的剪枝策略，即每次在预训练好的模型上移除固定比例的参数来逐步减小模型大小。在此基础上，Hassibi 等[29]利用权值的二阶偏导信息（Hessian 矩阵）来评价每个权值的显著性，并删除低的权值。针对二阶偏导难以计算的问题，Srinivas 等[30]提出了一种不需要训练数据和梯度信息的剪枝方法，该方法通过计算每一层内不同节点权重向量的相似性，将相似的向量进行合并，来压缩网络。

Han 等[31]使用参数的绝对值大小来评估其重要性，并将参数的 L1 范数作为优化目标，使更多的参数逼近 0；剪枝时，将这些逼近 0 的参数移除来压缩网络；最后，利用保留的权重作为模型训练模型，重新训练剪枝后的网络，同样采用了迭代式的剪枝—重新训练策略。但是该方法将产生非结构化的稀疏连接，导致不规则的内存访问。非结构化剪枝需要设计专用硬件[32]或函数库[33-34]来进一步加速。

（2）结构化剪枝

结构化剪枝旨在移除整个卷积核，克服非结构化稀疏连接导致的不规则访问内存问题，不需要依赖于特定的硬件和函数库，效率更高、通用性更强，且能较为方便地应用于目前主流的深度学习框架。

Anwar 等[35]通过在卷积核和通道层面引入稀疏性，并移除不重要的卷积核和通道来压缩网络。Lebedev 等[36]通过将结构化的稀疏作为正则项，和深度模型的损失项一起作为优化目标。在网络训练时，采用随机梯度下降法来优化网络，当卷积核的值小于给定阈值时赋值为 0，在测试时，移除值为 0 的卷积核。更进一步，Wen 等[21]将卷积核、通道、卷积核形状、网络深度的结构化稀疏作为正则项加入到优化目标。Hu 等[37]将卷积核所对应的输出特征零值比列作为卷积核重要性的评判标准，零值比例越高，则该卷积核越不重要。该方法背后的思想是依据卷积核的范数来评估其显著性，其目的是移除显著性较低的卷积核。以此为基础，Molchanov 等[38]提出全局搜索策略来找出显著性卷积核，对于需要删除的滤波用 0 值代替，并对目标函数进行泰勒展开，判断使目标函数变换最小的滤波为显著滤波。

3.2　低秩分解

低秩分解通过对网络的参数矩阵进行张量分解，利用其分解得到的低秩矩阵对元参数矩阵进行近似，减少网络的参数量，主要包括奇异值分解（SVD）、Tucker 分解[39]、块分解（block decomposition）[34]。这一过程如图 2 所示。

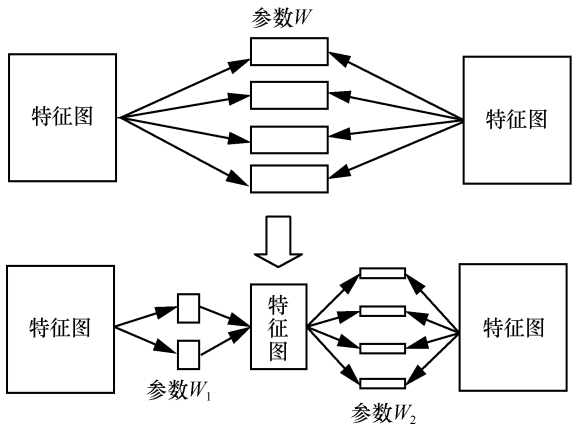

图 2　低秩分解过程

低秩分解出现的时间较早，在未广泛应用卷积网络之前，网络主要是由多层的全连接层构成的。虽然全连接运算量就少，但是所需要的参数量非常多。例如，对于一个秩为 r 的全连接层的参数矩阵 W，其大小为 $m\times n$（通常 r 远小于 m 或 n）。将 W 分解一个 $m\times r$ 矩阵 W_1 和一个 $r\times n$ 矩阵 W_2 之积，W_1 和 W_2 所占用的存储空间为 $r\times(m+n)$，远小于 W 所需的存储空间 $m\times n$。为了减小全连接层的参数量，大幅度压缩含有全连接层的 CNN 模型，Alexander 等[40]针对全连接层提出了 Tensor Train 分解方法，该方法能对 VGGNet 的全连接层压缩至原来的 1/20 000，极大降低了 VGGNet 的参数量。

对于全连接参数矩阵进行低秩分解，虽然能显著减少参数量，但是卷积网络中卷积操作占了主要的计算量。由于卷积神经网络存在着冗余卷积核，故而能够使用低秩分解来对卷积运算进行加速。例如，卷积神经网络可以通过 SVD 来对卷积核进行低秩近似，对卷积核进行聚类等[41- 42]。Kim 等[39]使用 Tucker 分解对卷积核张量进行分解，并在多重经典网络结构如 Alexnet、VGGNet 和 GoogLeNet 等进行验证。与 SVD 分解相比，Tucker 分解可以得到更精简的模型、更快的推理速度和更低的性能损失。需要注意的是，随着精简卷积核和小卷积核的广泛应用，低秩分解不再是模型压缩和加速的主流方法。目前，主要网络的卷积核基本是 3×3，甚至是 1×1 的卷积核，而这类小卷积核并不适合低秩分解。

3.3　模型量化

量化的主要目的是减小模型的存储大小，主要包含两种方法：权值共享和使用更低比特来表示模型的参数。

（1）权值共享

通过让网络的多个参数共用一个权值来减少参数存储大小。Han 等[31]通过对模型的参数进行 k-means 聚类，将权重量化为距离最近的聚类中心，同时采用霍夫曼编码来对权重进行编码，从而进一步压缩网络。Chen 等[43]提出的 HashedNets

通过哈希函数来将参数映射到不同的哈希桶中，每个哈希桶内所有参数共享一个权重。

（2）参数的低比特表示

通常模型参数的类型为单精度浮点型，需要占用 32 bit，共 4 bit。参数的低比特方法通过使用更低比特来表示参数，达到减小模型大小的目的。Courbariaux 等[44]将 32 bit 的梯度和激活值用 8 bit 来近似，可以在保持模型精度时，降低一半的数据传输加速，加快深度神经网络在集群上的训练数据。Dettmers 等[45]分析和评估了浮点型、定点型和动态定点型 3 种运算方式对网络训练误差的影响，结果表明低比特不仅可用于网络的推理中，也可以用于网络的训练中，即反传的梯度同样可以用较低比特来表示而不影响训练效果。Han 等[31]通过实验验证，在保证性能损失较少的情况下，通过 6 bit 或 8 bit 来量化网络可以满足大部分任务要求。

此外，有不少工作尝试了更低比特的量化方法。Courbariaux 等[44]提出的二值权重网络，针对参数进行二值化，即把浮点数强行设置为–1 和+1 两个值，网络的中间输入值和输出值仍然使用正常的全精度来表示。Rastegari 等[46]提出的引入了近似因子来弥补低比特导致的精度损失，在保持二值网络速度的同时提高了网络的精度。在 BNN 中，无论对权重进行二值化还是对中间值进行二值化，都会给全精度数据造成精度损失。Li 等[47]在二值网络基础上提出了三值网络（ternary weights network，TWN），其核心在于把处于前期设定的阈值内的数据强制设为 0，其他值强制设为–1 或+1，同时，作者给出了分析了阈值设定的策略。

3.4 知识蒸馏

通常大型网络相对于小网络具有更高的性能。一个预训练好的大网络，其输出的结果可以表示模型预测的类别分布概率，这一过程如图 3 所示。

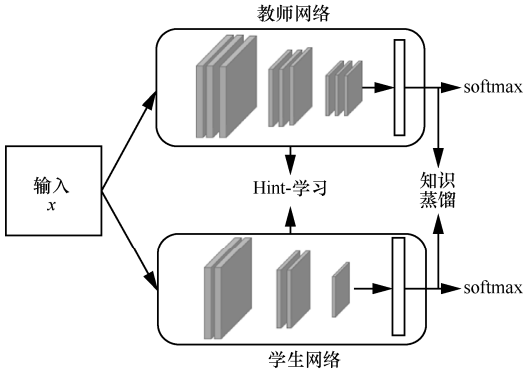

图 3 知识蒸馏过程

与 one-hot 类别标签相比，网络的预测结果所包含的信息更加丰富。知识蒸馏[48]就是将已训练好的大模型（通称教师网络）输出值（soft-label，通称软标签）作为额外的监督信号，辅助小模型（通称学生网络）的训练过程。其本质是将大网络学到的知识迁移到小网络中，具体的做法是让小网络模仿大网络的预测结果，从而提升小型网络性能。知识蒸馏由 Bucilua 等[49]提出，其主要目的是对模型进行压缩。然而，在当时知识蒸馏只局限于浅层网络，影响力较小。直到 2014 年 Hinton 等[48]通过最小化教师网络和学生网络输出值的 KL 散度来训练学生网络，使得学生网络可以接近甚至超过教师网络的精度。同时，文章也引入了一个超参数“温度”来松弛教师网络的输出值。此后，知识蒸馏成为提升小网络性能的主流方法之一。Ba[50]则直接利用优化教师网络和学生网络 logits 之间的 L2 距离，将较深较大的网络进行压缩，得到较浅较小的网络。

Hint-based Learning 在参考文献[48] 和参考文献[49]中，只有教师网络的最终输出值才被用来指导小模型的训练，完全忽视了网络的中间值。Remero 等[51]提出的 FitNets 不仅使用教师网络最终输出的 logits，同时也利用了中间层的激活响应，来训练对应的学生网络中间层。FitNets 旨在将较大且层数较少（larger and shallower）的网络压缩成层数较多且通道数较少（deeper and thin-

ner）的网络，来提升学生网络的表达能力和性能。这种利用大网络中间层的信息来训练小网络的方法被称为 Hint-based learning。Zagoruyko 等[52]提出了一种基于注意力的方法来匹配基于教师网络和学生网络中间层的激活值和注意力图（attention feature map）。Li 等[53]和 Luo 等[54]分别针对物体检测和人脸识别任务，利用大网络用来做识别的特征层来辅助小模型的训练，具体做法是最小化教师网络和学生网络特征的欧氏距离。这种方法被称为 mimicking。为了进一步提升小网络的性能，有部分工作[55-56]也将输入数据间的关系（correlation or relationship）看作教师网络学到的知识，并让学生网络同时学习教师网络学到数据间的关系。

与以上先离线训练教师网络再训练学生网络的方法不同，Zhang 等[8]提出的 DML（deep mutual learning）采用协同培训策略同时训练两个不同的网络，通过最小化两个网络输出值之间的 KL 散度来使得两个网络可以相互促进。而 Lan 等[57]则更进一步，提出了在没有教师网络的情况下利用知识蒸馏来提升小网络性能。该方法利用多分枝构造多个网络并采用集成的方法来构造教师网络的输出，并采用在线学习的方式同时训练分支网络及多个分支网络的集成。

3.5　网络搜索

依靠人工去设计神经网络要求有较强的专业知识并在特定数据集上训练模型并验证，这导致神经网络的使用和实现成本较高。网络架构搜索（network architecture search，NAS）方法可有效解决这一问题。NAS 借助模型自动搜索网络结构，从而找出高效的网络结构，有效地降低了神经网络的使用成本。

NAS 算法的核心要素是搜索空间、搜索策略和评估策略。搜索空间确定了解的空间，即提供了可用于搜索的神经网络结构的集合，搜索策略确定了寻找最优网络结构的方法策略，性能评估策略确定了评估网络结构性能的方法。通过对搜索空间的候选神经网络结构进行遍历，产生“子网络”，并通过评估策略对神经网络结构的性能并返回给搜索策略来确定下一步的搜索方向。这样，不断地在训练样本集上训练子网络，在验证集上评估其性能，逐步优化，最终找到最优的网络结构，这一过程如图 4 所示。

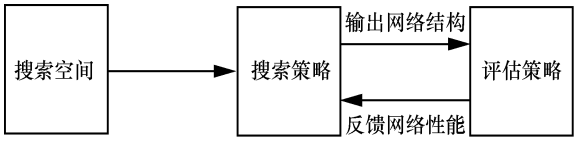

图 4　网络结构搜索示意图

目前 NAS 的工作多集中在搜索策略上，主要基于如下 3 种方法。

（1）强化学习（reinforcement learning）

Baker 等[58]提出的 MetaQNN 最早引入了强化学习。MetaQNN 将网络搜索看作马尔可夫决策过程，网络每一层的类型和相应的参数（如卷积层的通道数、卷积核大小、padding、stride 等）作为待搜索的空间，通过使用 Q-learning 算法来生成深度网络结构。Zoph 等[59]将 RNN 作为控制器，并通过采样来生成网络结构。强化学习方法被用来学习控制器的参数。

（2）进化算法（evolutionary algorithm）

较为经典的工作是 Google 提出的[60]。进化算法通过添增添或删除特定层，调整某些层的超参数，或改变层与层之间的跳连接和训练的超参来对网络进行变异，并在验证集上评估变异后的网络的有效性。进化算法在一些较小的数据集，如 CIFAR-10 和 CIFAR-100 上，已验证了其有效性。但是在数据规模较大时，进化算法实用性较差。

（3）基于梯度方法（gradient-based method）

强化学习和进化算法本质上是在离散空间中搜索，这导致了搜索的效率更低。基于梯度的 NAS 方法将搜索空间看作一个连续空间，通

过定义一个可微的目标函数，并使用基于梯度更新的方法来对更新搜索方向，大大提升了网络搜索的效率。基于梯度的方法较为开创性的工作是 DARTS[61]。

目前，NAS 仍处于快速发展阶段，各种不同的新方法不断涌现，并且出现了更为通用的自动机器学习方法——AutoML。在可预见的一段时间内，NAS 仍将是一个研究热点。

4 结束语

本文对主流的高效深度神经网络的研究方法和工作——模型加速、模型剪枝、量化、神经网络搜索、知识蒸馏进行了回顾并总结，详细介绍不同方法的经典工作和优缺点，并介绍最新的进展，为今后的相关研究提供参考，并展望该领域未来可能存在的发展方向。

参考文献：

[1] KRIZHEVSKY A, SUTSKEVER I, HINTON G E. Imagenet classification with deep convolutional neural networks[C]//Proceedings of the Advances in Neural Information Processing Systems. Lake Tahoe: MIT Press, 2012.

[2] HU J, SHEN L, SUN G. Squeeze-and-excitation networks[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE Press, 2018.

[3] SIMONYAN K, ZISSERMAN A. Very deep convolutional networks for large-scale image recognition[C]//Proceedings of International Conference on Learning Representations. San Diego: Elsevier, 2015.

[4] XIE S, GIRSHICK R, DOLL R P, et al. Aggregated residual transformations for deep neural networks[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE Press, 2017.

[5] LECUN Y, DENKER J S, SOLLA S A. Optimal brain damage[C]//Proceedings of the Advances in Neural Information Processing Systems. Piscataway: IEEE Press, 1990.

[6] HOWARD A G, ZHU M, CHEN B, et al. Mobilenets: efficient convolutional neural networks for mobile vision applications[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE Press, 2017,

[7] SANDLER M, HOWARD A, ZHU M, et al. Mobilenetv2: inverted residuals and linear bottlenecks[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE Press, 2018.

[8] ZHANG X, ZHOU X, LIN M, et al. Shufflenet: an extremely efficient convolutional neural network for mobile devices[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE Press, 2018.

[9] MCCULLOCH W S, PITTS W. A logical calculus of the ideas immanent in nervous activity[J]. The Bulletin of Mathematical Biophysics, 1943, 5(4): 115-133.

[10] ROSENBLATT F. The perceptron: a probabilistic model for information storage and organization in the brain[J]. Psychological Review, 1958, 65(6): 386.

[11] MARVIN M, SEYMOUR P. Perceptrons[M]. Cambridge: MIT Press, 1969.

[12] HOPFIELD J J. Neural networks and physical systems with emergent collective computational abilities[J]. Proceedings of the National Academy of Sciences, 1982, 79(8): 2554-2558.

[13] HINTON G E, SEJNOWSKI T J. Learning and relearning in Boltzmann machines[J]. Parallel Distributed Processing: Explorations in the Microstructure of Cognition, 1986, 1(282-317): 2.

[14] MCCLELLAND J L, RUMELHART D E, GROUP P R. Parallel distributed processing[M]. Cambridge: MIT Press , 1987.

[15] LECUN Y, BOTTOU L, BENGIO Y, et al. Gradient-based learning applied to document recognition[J]. Proceedings of the IEEE, 1998, 86(11): 2278-324.

[16] HINTON G E, SALAKHUTDINOV R R. Reducing the dimensionality of data with neural networks[J]. Science, 2006, 313(5786): 504-507.

[17] TAIGMAN Y, YANG M, RANZATO M A, et al. Deepface: closing the gap to human-level performance in face verification[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE Press, 2014.

[18] SUN Y, WANG X, TANG X. Deep learning face representation from predicting 10,000 classes[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE Press, 2014.

[19] SCHROFF F, KALENICHENKO D, PHILBIN J. Facenet: A unified embedding for face recognition and clustering[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE Press, 2015.

[20] DENG J, GUO J, XUE N, et al. Arcface: additive angular margin loss for deep face recognition[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE Press, 2019.

[21] WEN Y, ZHANG K, LI Z, et al. A discriminative feature learning approach for deep face recognition[C]//Proceedings of the European Conference on Computer Vision. Glasgow: Springer, 2016.

[22] GIRSHICK R, DONAHUE J, DARRELL T, et al. Rich feature hierarchies for accurate object detection and semantic segmentation[C]//Proceedings of the IEEE Conference on Computer

Vision and Pattern Recognition. Piscataway: IEEE Press, 2014.

[23] GIRSHICK R. Fast R-CNN[C]//Proceedings of the IEEE International Conference on Computer Vision. Piscataway: IEEE Press, 2015.

[24] REN S, HE K, GIRSHICK R, et al. Faster R-CNN: towards real-time object detection with region proposal networks[C]//Proceedings of the Advances in Neural Information Processing Systems. Cambridge: MIT Press, 2015.

[25] HE K, GKIOXARI G, DOLL R P, et al. Mask R-CNN[C]// Proceedings of the IEEE International Conference on Computer Vision. Piscataway: IEEE Press, 2017.

[26] WANG L, XIONG Y, WANG Z, et al. Temporal segment networks: towards good practices for deep action recognition[C]//Proceedings of the European Conference on Computer Vision. Amsterdam: Springer, 2016.

[27] MIKOLOV T, YIH W-T, ZWEIG G. Linguistic regularities in continuous space word representations[C]//Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Piscataway: IEEE Press, 2013.

[28] KALCHBRENNER N, GREFENSTETTE E, BLUNSOM P. A convolutional neural network for modelling sentences[J]. arXiv: 14042188, 2014,

[29] HASSIBI B, STORK D G, WOLFF G J. Optimal brain surgeon and general network pruning[C]//Proceedings of the IEEE International Conference on Neural Networks. Piscataway: IEEE Press, 1993.

[30] SRINIVAS S, BABU R V. Data-free parameter pruning for deep neural networks[C]//Proceedings of the British Machine Vision Conference. Swansea: BMVA Press, 2015,

[31] HAN S, POOL J, TRAN J, et al. Learning both weights and connections for efficient neural network[C]//Proceedings of the Advances in Neural Information Processing Systems. Cambridge: MIT Press , 2015.

[32] HAN S, LIU X, MAO H, et al. EIE: efficient inference engine on compressed deep neural network[C]//Proceedings of the 2016 ACM/IEEE 43rd Annual International Symposium on Computer Architecture (ISCA). Piscataway: IEEE Press, 2016.

[33] PARK J, LI S, WEN W, et al. Faster cnns with direct sparse convolutions and guided pruning[C]//International Conference on Learning Representations. San Diego: Elsevier, 2016,

[34] LIU B, WANG M, FOROOSH H, et al. Sparse convolutional neural networks[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE Press, 2015.

[35] ANWAR S, HWANG K, SUNG W. Structured pruning of deep convolutional neural networks[J]. ACM Journal on Emerging Technologies in Computing Systems (JETC), 2017, 13(3): 32.

[36] LEBEDEV V, LEMPITSKY V. Fast convnets using group-wise brain damage[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE Press, 2016.

[37] HU H, PENG R, TAI Y-W, et al. Network trimming: a data-driven neuron pruning approach towards efficient deep architectures[C]//Proceedings of International Conference on Learning Representations. San Diego: Elsevier, 2016.

[38] MOLCHANOV P, TYREE S, KARRAS T, et al. Pruning convolutional neural networks for resource efficient transfer learning[C]//Proceedings of International Conference on Learning Representations. San Diego: Elsevier, 2017.

[39] KIM Y-D, PARK E, YOO S, et al. Compression of deep convolutional neural networks for fast and low power mobile applications[C]//International Conference on Learning Representations. San Diego: Elsevier, 2016.

[40] ALEXANDER N, PODOPRIKHIN D, OSOKIN A, et al. Tensorizing neural networks[C]//Proceedings of the Advances in Neural Information Processing Systems. Montreal: MIT Press , 2015.

[41] DENTON E L, ZAREMBA W, BRUNA J, et al. Exploiting linear structure within convolutional networks for efficient evaluation[C]//Proceedings of the Advances in Neural Information Processing Systems. Montreal: MIT Press, 2014.

[42] JADERBERG M, VEDALDI A, ZISSERMAN A. Speeding up convolutional neural networks with low rank expansions[C]//Proceedings of the British Machine Vision Conference. Nottingham: BMVA Press, 2014,

[43] CHEN W, WILSON J, TYREE S, et al. Compressing neural networks with the hashing trick[C]//Proceedings of the International Conference on Machine Learning. [S.l.:s.n.], 2015.

[44] COURBARIAUX M, BENGIO Y, DAVID J-P. Training deep neural networks with low precision multiplications[C]//International Conference on Learning Representations. San Diego: Elsevier, 2015.

[45] DETTMERS T. 8-bit approximations for parallelism in deep learning[C]//4th International Conference on Learning Representations. San Diego: Elsevier, 2016.

[46] RASTEGARI M, ORDONEZ V, REDMON J, et al. Xnor-net: Imagenet classification using binary convolutional neural networks[C]//Proceedings of the European Conference on Computer Vision. Amsterdam: Springer, 2016.

[47] LI F, ZHANG B, LIU B. Ternary weight networks[J]. arXiv preprint arXiv: 160504711, 2016,

[48] HINTON G, VINYALS O, DEAN J. Distilling the knowledge in a neural network[C]//Proceedings of the Advances in Neural Information Processing Systems. Montreal: MIT Press, 2015.

[49] BUCILUǍ C, CARUANA R, NICULESCU-MIZIL A. Model compression[C]//Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York: ACM Press, 2006.

[50] BA J, CARUANA R. Do deep nets really need to be

deep?[C]//Proceedings of the Advances in Neural Information Processing Systems. Montreal: MIT Press , 2014.

[51] ROMERO A, BALLAS N, KAHOU S E, et al. Fitnets: hints for thin deep nets[C]//International Conference on Learning Representations. San Diego: Elsevier, 2015.

[52] ZAGORUYKO S, KOMODAKIS N. Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer[C]///International Conference on Learning Representations. San Diego: Elsevier, 2016.

[53] LI Q, JIN S, YAN J. Mimicking very efficient network for object detection[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE Press, 2017.

[54] LUO P, ZHU Z, LIU Z, et al. Face model compression by distilling knowledge from neurons[C]//Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence. [S.l.:s.n.], 2016.

[55] LIU Y, CAO J, LI B, et al. Knowledge distillation via instance relationship graph[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE Press, 2019.

[56] PENG B, JIN X, LIU J, et al. Correlation congruence for knowledge distillation[C]//Proceedings of the IEEE International Conference on Computer Vision. Seoul: Elsevier, 2019.

[57] LAN X, ZHU X, GONG S. Knowledge distillation by on-the-fly native ensemble[C]//Proceedings of the 32nd International Conference on Neural Information Processing Systems. Montreal: MIT Press, 2018.

[58] BAKER B, GUPTA O, NAIK N, et al. Designing neural network architectures using reinforcement learning[J]. arXiv: 1611.02167, 2016.

[59] ZOPH B, LE Q V. Neural architecture search with reinforcement learning[C]//Proceedings of International Conference on Learning Representations. Toulon: Elsevier, 2017.

[60] REAL E, MOORE S, SELLE A, et al. Large-scale evolution of image classifiers[C]//Proceedings of the 34th International Conference on Machine Learning. [S.l.:s.n.], 2017: 2902-2911.

[61] LIU H, SIMONYAN K, YANG Y. Darts: differentiable architecture search[C]//Proceedings of International Conference on Learning Representations. New Orleans: Elsevier, 2019.

[作者简介]

闵锐（1978− ），男，中国电信股份有限公司智能网络与终端研究院高级工程师，主要研究方向为大数据、图像识别及人工智能。