



深度学习的轻量化神经网络结构研究综述

王 军^{1,2,3}, 冯孙铖^{1,2}, 程 勇^{1,3}

(1.南京信息工程大学 计算机与软件学院,南京 210044; 2.南京信息工程大学 数字取证教育部工程研究中心,南京 210044;
3.南京信息工程大学 科技产业处,南京 210044)

摘 要: 随着深度神经网络和智能移动设备的快速发展,网络结构轻量化设计逐渐成为前沿且热门的研究方向,而轻量化的本质是在保持深度神经网络精度的前提下优化存储空间和提升运行速度。阐述深度学习的轻量化网络结构设计方法,对比与分析人工设计的轻量化方法、基于神经网络结构搜索的轻量化方法和基于自动模型压缩的轻量化方法的创新点与优劣势,总结与归纳上述3种主流轻量化方法中性能优异的网络结构并分析各自的优势和局限性。在此基础上,指出轻量化网络结构设计所面临的挑战,同时对其应用方向及未来发展趋势进行展望。

关键词: 深度学习;轻量化设计;深度可分离卷积;Octave卷积;神经网络结构搜索;模型压缩

开放科学(资源服务)标志码(OSID):



中文引用格式:王军,冯孙铖,程勇.深度学习的轻量化神经网络结构研究综述[J].计算机工程,2021,47(8):1-13.

英文引用格式:WANG J, FENG S C, CHENG Y. Survey of research on lightweight neural network structures for deep learning[J]. Computer Engineering, 2021, 47(8): 1-13.

Survey of Research on Lightweight Neural Network Structures for Deep Learning

WANG Jun^{1,2,3}, FENG Suncheng^{1,2}, CHENG Yong^{1,3}

(1.School of Computer and Software, Nanjing University of Information Science and Technology, Nanjing 210044, China;
2.Engineering Research Center of Digital Forensics of Ministry of Education, Nanjing University of Information Science and
Technology, Nanjing 210044, China; 3.Science and Technology Industry Division,
Nanjing University of Information Science and Technology, Nanjing 210044, China)

[Abstract] With the rapid development of deep neural networks and smart mobile devices, the research of lightweight neural network structure has gradually become a hotspot. The essence of lightweight design is to optimize the storage space and improve the running speed without causing any loss to the precision of deep neural networks. Then an introduction to the mainstream methods of lightweight network structure design for deep learning is given, and the innovative features, strengths and weaknesses between the manual design methods, neural network structure search-based design methods and automated model compression-based design methods are compared. The advantages and disadvantages of the high-performance network structures generated by the above methods are also summarized. On this basis, the challenges faced by lightweight network structure design, and its applications and development trends are discussed.

[Key words] deep learning; lightweight design; Depthwise Separable Convolution (DSC); Octave convolution; neural network structure search; model compression

DOI: 10.19678/j.issn.1000-3428.0060931

0 概述

在17世纪,贝叶斯和拉普拉斯完成对最小二乘法的推导并提出马尔科夫链,这两个理论成为机器学习发展的基础理论。在1950年,艾伦·图灵提议建立一个学习机器,之后机器学习进入飞速发展阶

段。在1986年,深度学习被引入机器学习领域,为人工智能的发展提供了极大的动力支持。在2000年,深度学习通过组合多个隐藏层的神经元,并利用非线性函数学习多个具有抽象意义的数据特征,达到模拟神经网络的目的^[1-3],且广泛适用于有监督和无监督特征学习^[4-6]、特征表示^[7]、模式识别等任务。深度卷

基金项目:国家自然科学基金(41875184);江苏省“六大人才高峰”创新人才团队项目(TD-XYDXX-004)。

作者简介:王 军(1970—),男,教授,主研方向为深度学习、物联网、图像处理;冯孙铖,硕士研究生;程 勇,高级工程师、博士。

收稿日期:2021-02-24 修回日期:2021-03-24 E-mail:512710092@qq.com

积神经网络(Convolutional Neural Networks, CNN)在图像识别^[8-9]、目标检测^[10-11]、语义分割^[12]等计算机视觉的前沿领域展现出巨大的潜力,但是常规的卷积神经网络在达到较高分类精度的同时需要较快的运算速度和占用大量存储空间^[13]。目前,智能移动设备的发展趋向于边缘化和移动化发展,但却受限于设备本身的硬件条件,而深度卷积神经网络轻量化设计的目标就是在低硬件条件的设备上仍能保持良好的网络性能来适应智能设备的发展趋势。

轻量化的本质是在硬件不足的设备上解决存储空间和能耗对于传统神经网络性能的限制,在保持传统神经网络精度的基础上,通过人工设计、神经结构搜索或自动化机器学习等方法降低对存储空间的要求,提高运行速度。LECUN等^[14]在信息论的基础上,通过对网络中不重要的权重进行剪除,增强神经网络的泛化性,提高学习速率,最终实现模型压缩。HAN等^[15]发表了一篇关于模型压缩方法的综述型文章,该文作为ICLR 2016的最佳论文,受到了学术界的广泛关注。CHENG等^[16-17]对近年来提出的模型压缩方法进行了综述。轻量化网络结构设计是深度学习中的热点研究方向。2012年, AlexNet^[18]由于受到硬件设备的限制,创新性地使用组卷积并将一个网络在两个硬件设备上训练,对于轻量化网络结构具有一定的借鉴意义。目前,针对轻量化深度学习网络的研究主要集中于人工设计的轻量化网络和基于神经网络结构搜索的自动轻量化网络。在人工设计的轻量化网络方面,包括基于深度可分离卷积(Depthwise Separable Convolution, DSC)的SqueezeNet^[19]、MobileNet^[20]、MobileNet V2^[21]、ShuffleNet^[22]、ShuffleNet V2^[23]、基于Octave卷积^[24]的改进基线网络、基于Ghost特征的GhostNet^[25]等轻量化网络。在神经网络结构搜索的自动轻量化网络设计方面有NasNet^[26]、MnasNet^[27]等轻量化网络。本文对深度学习的轻量化网络结构设计方法进行详细介绍和优势分析,阐述人工设计的轻量化方法、基于神经网络结构搜索的轻量化方法、基于自动模型压缩的轻量化方法的应用现状和发展趋势。

1 人工设计的轻量化方法

1.1 组卷积

组卷积^[8]对于输入特征图按通道进行分组卷积,再将分组卷积得到的结果按通道进行连接(concat)得到最终的输出特征,具有轻量化效果。例如,将输入特征分成 G 组进行组卷积,仅需原有 $1/G$ 的参数。因此,组卷积对深度神经网络具有一定的正则化作用。但是,组卷积也有局限性,会导致特征图之间的信息不流畅,输出的特征图没有包含所有输入特征图的信息。因此,在组卷积的基础上,使用

深度可分离卷积中的Pointwise卷积和ShuffleNet中的通道变换来改善该问题。

1.2 深度可分离卷积

深度可分离卷积主要由Depthwise卷积和Pointwise卷积组成,如图1和图2所示。Depthwise卷积使用卷积核对输入特征按通道进行分别卷积,即第一通道的卷积核与第一通道的输入特征进行卷积。Depthwise卷积在获得特征的空间信息后,将得到的输出特征进行Pointwise卷积,即利用 1×1 的卷积核对Depthwise卷积的输出进行卷积,以获取特征中不同通道之间的信息,通过该组合方式达到轻量化效果。

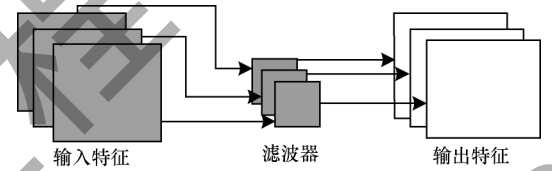


图1 Depthwise 卷积

Fig.1 Depthwise convolution

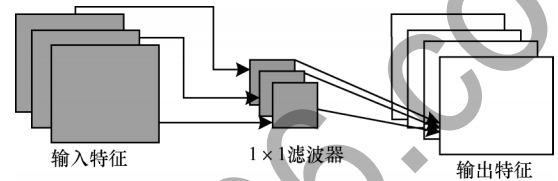


图2 Pointwise 卷积

Fig.2 Pointwise convolution

对网络参数数量和运算量进行分析,假设输入的特征图大小为 $h_{in} \times w_{in} \times c_{in}$,卷积核大小为 $k \times k \times c_{out}$,输出特征为 $h_{out} \times w_{out} \times c_{out}$,深度可分离卷积的运算量和参数量计算如下:

$$F_{FLOPS} = k \times k \times c_{in} \times h_{out} \times w_{out} \times c_{in} \times c_{out} \times h_{out} \times w_{out} \quad (1)$$

$$P_{parameter} = k \times k \times c_{in} + 1 \times 1 \times c_{in} \times c_{out} \quad (2)$$

其中: h_{in} 和 w_{in} 分别表示输入特征的高度和宽度; c_{in} 表示输入特征图的通道数; k 表示卷积核尺寸; c_{out} 表示输出特征图的通道数; h_{out} 和 w_{out} 分别表示输出特征的高度和宽度。

深度可分离卷积在保持较高分类精度的前提下,仅需常规卷积 $1/3$ 的参数数量,但在训练过程中计算零散是影响其实际应用性能的主要阻力。

1.3 基于深度可分离卷积的MobileNet

现在主流的轻量化网络结构MobileNet就是基于深度可分离卷积。MobileNet^[20]的基本思想是使用深度可分离卷积代替常规卷积,利用深度卷积代替传统卷积中的滤波器进行特征提取,并采用点卷积来代替滤波器对特征进行组合,同时减少参数量和运算量,最终将MobileNet^[28]堆叠成深度神经网络。常规卷积和深度可分离卷积如图3所示,其中,BN表示批量正则化,ReLU表示激活函数。

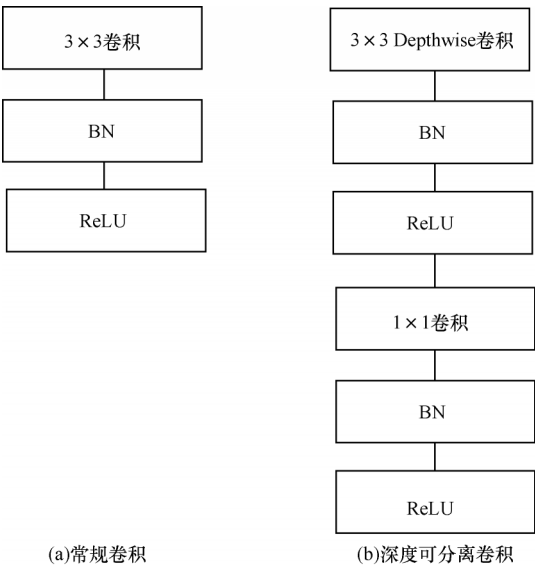


图 3 常规卷积和深度可分离卷积

Fig.3 Conventional convolution and depthwise separable convolution

MobileNet V2^[21]作为 MobileNet的改进网络,引入 ResNet 网络中的残差思想,同时为了解决常规 ResNet 中大量使用 ReLU 激活函数导致神经元失活的缺陷,通过高维的特征结合 ReLU 激活层尽可能地保留低维输入信息。MobileNet V2 的核心为反向残差块,如图 4 所示。与常规残差块不同,反向残差块采用两边窄中间宽的通道结构,首先对输入进行特征扩展,将低维特征映射到高维空间,然后对高维特征使用 Depthwise 卷积代替常规 3×3 卷积,这样既可以保留信息且不失非线性,最后去除最后一个 ReLU 激活层并使用投影层代替,即使用 1×1 的网络结构将高维特征映射到低维空间。

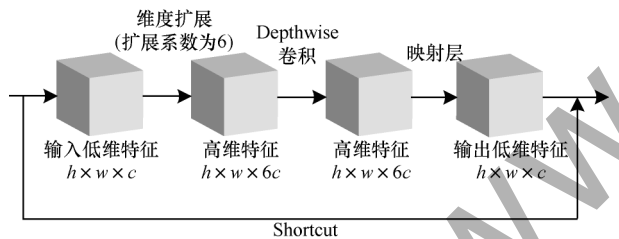


图 4 反向残差块

Fig.4 Inverse residual block

改进的 MobileNet V2 网络结构如表 1 所示,其中, Conv 2d 表示卷积操作, bottleneck 表示 MobileNet V2 中的瓶颈结构, 1×1 Conv 2d 表示点卷积, 7×7 Avgpool 表示平均池化, K 表示根据实际需求设置的输出通道数, t 表示对于操作输入特征图的扩展系数, n 表示该操作的重复操作次数, c 表示输出通道数, s 表示该操作模块第 1 次使用的卷积步长,之后重复的卷积默认步长为 1。

表 1 MobileNet V2 整体结构

Table 1 The overall structure of MobileNet V2

输入尺寸	操作	t	c	n	s
224×224×3	Conv 2d	—	32	1	2
112×112×32	bottleneck	1	16	1	1
112×112×16	bottleneck	6	24	2	2
56×56×24	bottleneck	6	32	3	2
28×28×32	bottleneck	6	64	4	2
14×14×64	bottleneck	6	96	3	1
14×14×96	bottleneck	6	160	3	2
7×7×160	bottleneck	6	320	1	1
7×7×320	1×1 Conv 2d	—	1 280	1	1
7×7×1 280	7×7 Avgpool	—	—	1	—
1×1×1 280	1×1 Conv 2d	—	K	—	—

MobileNet V2 有效解决了 ReLU 函数导致的神经元失活问题,实验效果相比 MobileNet 更优异。但是,基于深度可分离卷积的 MobileNet V2 存在局限性。在实际训练过程中,由于深度可分离卷积的卷积核和常规卷积相比较小,在激活函数的非线性激活作用下使得输出易趋近 0,因此通常会出现卷积核失活的问题。

1.4 基于深度可分离卷积的 ShuffleNet

1.4.1 ShuffleNet

ShuffleNet^[22]是一个效率极高且可运行在手机等移动设备上的网络。常规组卷积最大的局限性为在训练过程中不同分组之间没有信息交换,这样会大幅降低深度神经网络的特征提取能力。因此,在 MobileNet 中使用大量的 1×1 Pointwise 卷积来弥补这一缺陷,而 ShuffleNet 采用通道变换来解决该问题。通道变换的核心思想是对组卷积之后得到的特征图在通道上进行随机均匀打乱,再进行组卷积操作,这样就保证了执行下一个组卷积操作的输入特征来自上一个组卷积中的不同组,如图 5 所示。基于深度可分离卷积、通道变换和组卷积得到 ShuffleNet 结构,如图 6 所示。通过堆叠 ShuffleNet 的基本单元来构建轻量化的 ShuffleNet 结构。

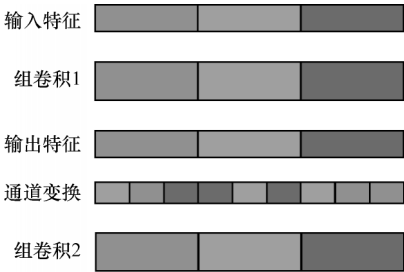


图 5 通道变换在组卷积中的应用

Fig.5 Application of channel shuffle in group convolution

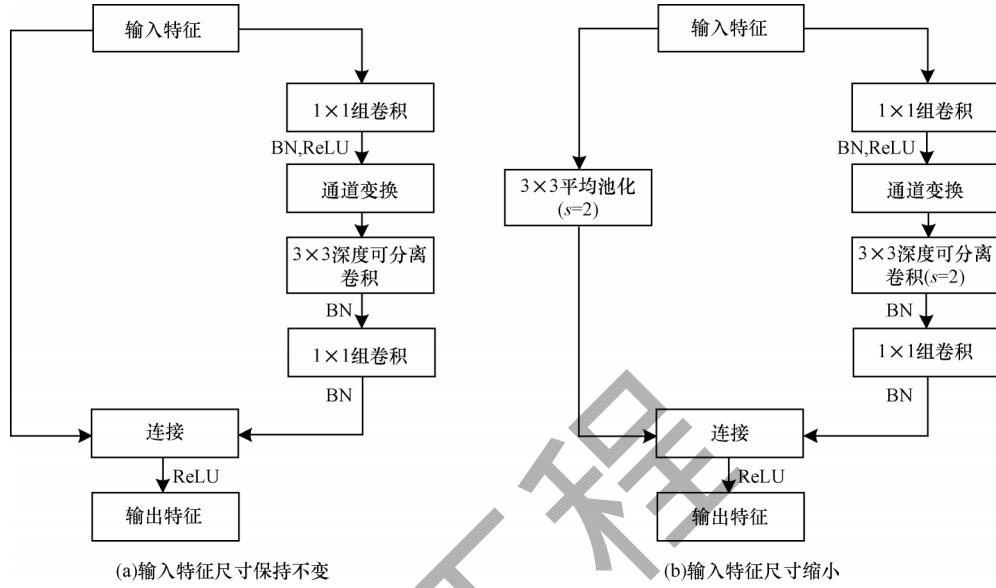


图6 ShuffleNet结构

Fig.6 Structure of ShuffleNet

1.4.2 ShuffleNet V2

在 ShuffleNet V2^[23]被提出之前,轻量化网络中衡量模型复杂度的通用指标为每秒浮点运算次数(Floating-point Operations Per Second, FLOPS)。FLOPS代表运算力,对于网络性能评估是一个间接指标,因为运算力不完全等同于运行速度。通过实验可以发现,相同FLOPS的两个模型的运行速度却存在差异,导致该差异的原因包括GPU、内存使用量(Memory Access Cost, MAC)等因素。因此,MA等^[23]对于轻量化网络提出了4条更实用的指导原则:

1) 尽量使用和输入特征通道数相同的卷积核个数来最小化内存使用量。以上文提及的深度可分离卷积中的Pointwise卷积为例,假设输入特征尺寸为 $h \times w \times c_{in}$,输出通道数为 c_{out} ,于是在Pointwise卷积中可得:

$$F_{FLOPS} = h \times w \times c_{in} \times c_{out} \quad (3)$$

$$M_{MAC} = h \times w \times (c_{in} \times c_{out}) + c_{in} \times c_{out} \quad (4)$$

当固定 F_{FLOPS} 时,根据均值不等式得到:

$$M_{MAC} \geq 2 \times \sqrt{h \times w \times F_{FLOPS}} + \frac{F_{FLOPS}}{h \times w} \quad (5)$$

当 $c_{in} = c_{out}$ 时,MAC取最小值,此时内存使用量最小。

2) 适量使用组卷积以降低内存使用量。在组卷积中,假设输入特征尺寸为 $h \times w \times c_{in}$,输出通道数为 c_{out} ,得到:

$$F_{FLOPS} = h \times w \times c_{in} \times c_{out} / G \quad (6)$$

$$M_{MAC} = h \times w \times (c_{in} \times c_{out}) + c_{in} \times c_{out} / G \quad (7)$$

当固定 F_{FLOPS} 时,得到:

$$M_{MAC} = h \times w \times c_{in} + F_{FLOPS} \times \frac{G}{c_{in}} + \frac{F_{FLOPS}}{h \times w} \quad (8)$$

由式(8)可见,当 G 增加时,内存使用量也会增加。

3) 尽量减少碎片化的网络结构以增加并行度。在Inception模块和机器学习自动生成的神经网络中,通常会趋向于使用多路网络结构,这样就很容易造成神经网络的碎片化,从而使模型并行度降低,减缓运行速度。

4) 重视元素级操作。激活函数(例如ReLU)和特征图的相加(add)虽然对于浮点运算力的影响很小,但它们对于内存使用量会产生较大的影响。

针对上述4个原则可知ShuffleNet存在以下问题:1)在基本单元中大量使用了 1×1 的组卷积操作;2)在残差网络的瓶颈层中,输入特征和输出特征的通道数不同;3)过量使用组卷积;4)在Shortcut中过量使用元素级操作。ShuffleNet V2是在上述4条原则的基础上对ShuffleNet进行的改进,并引入了通道分离操作。

ShuffleNet V2结构如图7所示。通道分离本质上是将输入特征按通道分成两部分,一部分通道数为 c' ,另一部分为 $c - c'$ 。在图7(a)中左分支等同于恒等映射,对应残差网络中的Shortcut,右分支包含

了3个连续的卷积操作,且满足输入特征和输出特征通道相同的原则。同时,ShuffleNet V2基本单元中的 1×1 卷积不再使用组卷积,而是使用常规卷积,弥补了过度使用组卷积的缺陷。在图7(b)中,首先将左右分支分成两组,两个分支不再使用相加

元素级操作,而是连接在一起,从而满足原则3。然后对其进行通道变换以保证两个分支的信息交流。最后连接和通道变换可以与下一个模块单元的通道分离合成一个元素级运算,减少了元素级操作次数。

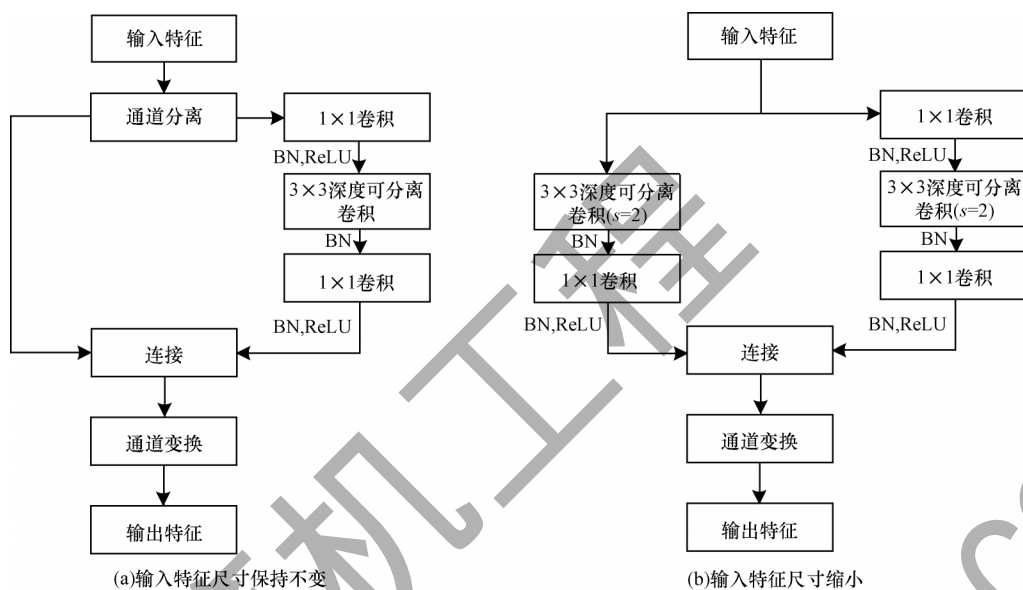


图7 ShuffleNet V2 结构

Fig.7 Structure of ShuffleNet V2

对基于深度可分离卷积的主流轻量化神经网络结构的创新点和优劣势进行分析和对比:

1) MobileNet。创新点和优势为引入深度可分离卷积进行网络结构轻量化设计。劣势为网络结构单一,且过量使用激活函数导致神经元易失活。

2) MobileNet V2。创新点和优势为引入反残差模块。劣势为由于深度可分离卷积中卷积核较小,激活后易为0。

3) ShuffleNet。创新点和优势为引入通道转换。劣势为输入输出特征通道数不同、过量使用组卷积、网络碎片化、元素级操作过多。

4) ShuffleNet V2。创新点和优势为引入通道分离、输入输出特征通道数相等、基础单元中使用常规卷积代替组卷积及使用 concat 代替元素级操作 add。劣势为运行速度和存储空间有待进一步提升。

1.5 Xception

深度可分离卷积可大幅减少计算量,又能保持较高的分类精度,但是存在计算零散的问题。Xception^[29]是谷歌于2017年在 Inception V3^[30]的基础上,基于空间相关性和通道相关性设计的轻量化网络结构。Xception模块如图8所示。

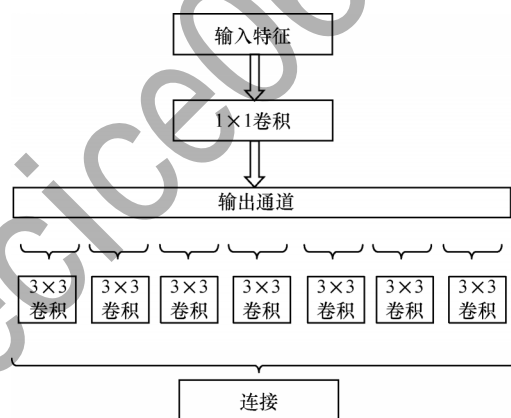


图8 Xception 模块

Fig.8 Xception module

Xception 模块与深度可分离卷积的不同之处在于:1)深度可分离卷积先进行同一平面卷积得到空间相关性,再在不同通道之间进行卷积得到通道相关性,而 Xception 模块采用相反的方法,先得到通道之间的相关性,再学习空间相关性;2)Xception 在空间相关性和通道相关性的学习过程中未使用激活函数,实验证明这一改进有效地加快了收敛速度,提升了网络性能。Xception 网络基于残差网络进行构建,但将其中的卷积层换成了 Xception 模块。如图9所示,Xception 网络被分为

输入流部分、中间流部分和输出流部分^[29],其中,ReLU表示激活函数,SeparableConv表示深度可分离卷积,Maxpool表示最大值池化操作。Xception相比Inception V3提升了网络运算量并降低了参数

量。输入流部分通过下采样模块来降低特征图的空间维度;中间流部分通过优化网络特征提取来学习关联关系;输出流部分将特征进行汇总输出,最终由全连接层进行表达。

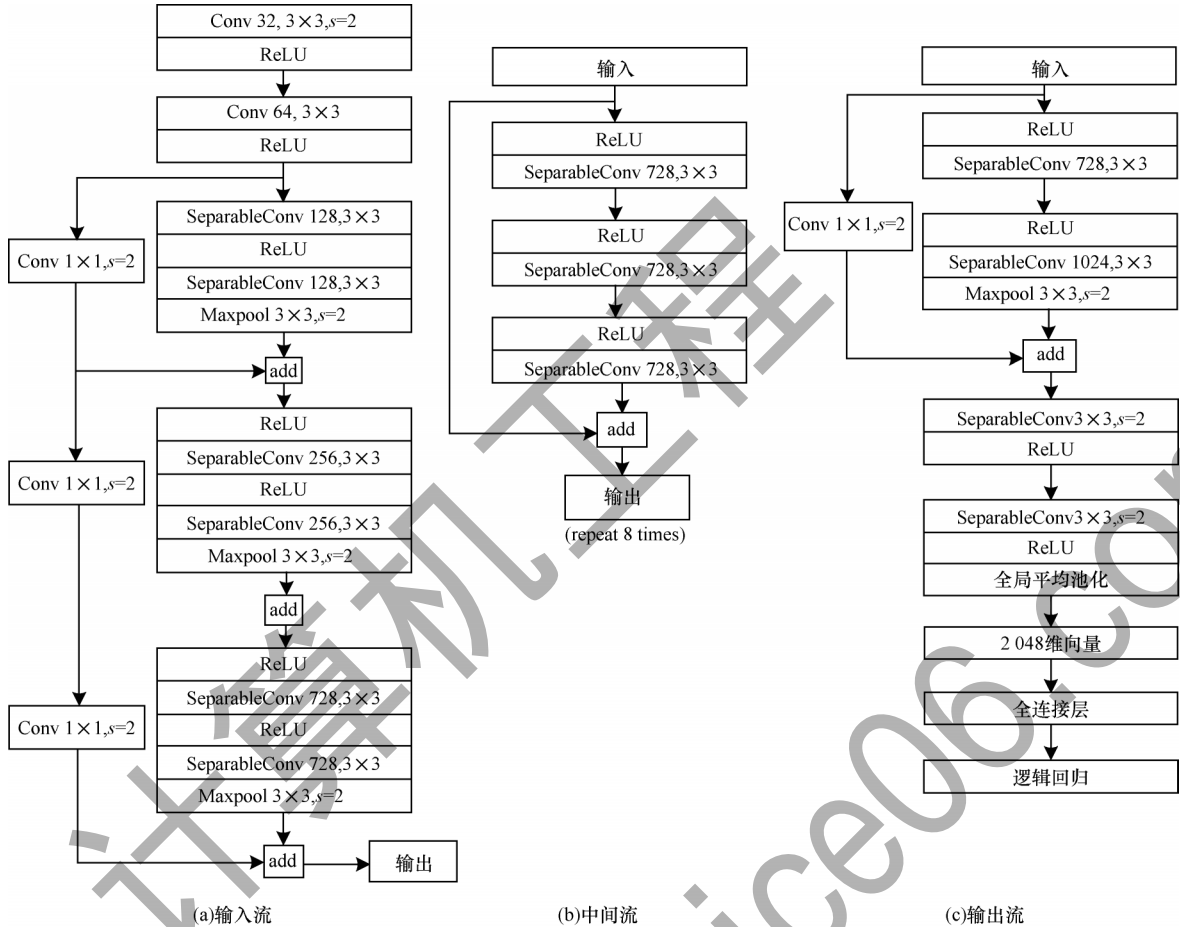


图9 Xception网络结构

Fig.9 Structure of Xception network

1.6 基于Octave卷积的改进基线网络

在现实生活中,图片中不同的信息都以不同的频率传递,主要分为高频信息和低频信息,其中:高频通常用于细节编码,高频信息代表图片中的细节特征;低频通常用于全局编码,低频信息代表图片中的全局特征,即较低空间分辨率下变化较慢的特征。图像低频和高频信息的分离如图10所示。由高频信息和低频信息组成的特征图就是Octave特征图。

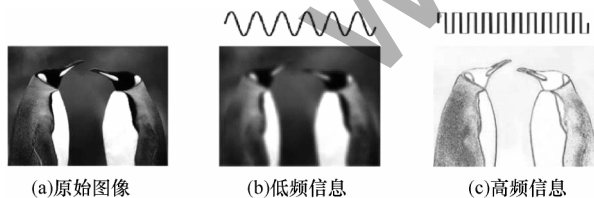


图10 图像低频和高频信息的分离

Fig.10 Separation of low-frequency and high-frequency information in the image

卷积层之间的特征图可以看作是高频信息和低频信息的混合特征图。在传统卷积方式中,无论高频信息还是低频信息都是用同一种方式存储的,这对于其中的低频信息而言就会造成存储冗余并增加计算成本。Octave卷积^[24]是针对这一问题提出的新型卷积方式,将特征图根据不同的频率进行因式分解,对不同频率的信息进行不同的存储和操作,再在不同频率的信息之间进行信息交换。Octave卷积的作用在于将传统的特征存储方式转化成基于低频和高频的轻量化存储方式,具体转变过程如图11所示,其中, α_{in} 代表在卷积操作输入的Octave特征图中高频信息所占的比例, α_{out} 代表卷积操作输出的Octave特征图中高频信息所占的比例。在Octave卷积的输入特征图中,当 $\alpha_{in} = \alpha_{out} = 0$ 时,Octave卷积就等同于常规卷积。当 $\alpha_{in} = 0$ 且 $\alpha_{out} \neq 0$ 时,代表当输入特征图为常规卷积特征图时,将其转化成用于Octave卷积的Octave特征图,通常应用于Octave卷

积的第一层。当 $\alpha_{in} \neq 0$ 且 $\alpha_{out} \neq 0$ 时,代表当输入是 Octave 特征图时进行 Octave 卷积操作,通常应用于 Octave 卷积的中间层。当 $\alpha_{in} \neq 0$ 且 $\alpha_{out} = 0$ 时,代表在获得传统特征图时需进行 Octave 卷积,其作用是将 Octave 特征图经过卷积之后得到传统特征图,通常应用于 Octave 卷积的最后一层。

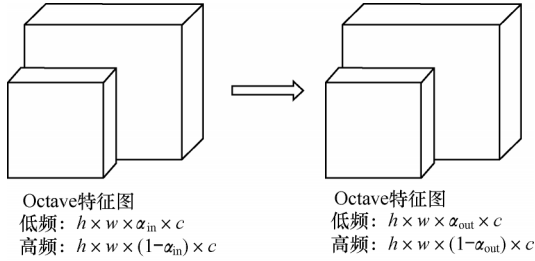


图 11 Octave 卷积前后的数据变化

Fig.11 Data changes before and after Octave convolution

Octave 卷积通常对低频信息和高频信息进行分别存储和处理,如果不能实现不同频率信息之间的信息交换,则非常影响网络性能。因此,在 Octave 卷积中必须同时实现同频率信息的传递和不同频率信息的交换,如图 12 所示。

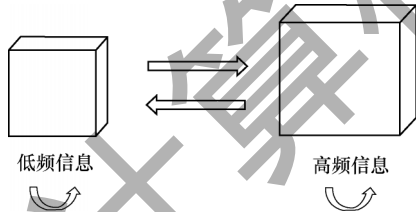


图 12 Octave 卷积中信息的传递和交换

Fig.12 Transfer and exchange of information in Octave convolution

在获得高频信息时,对输入特征图中的高频信息进行常规卷积操作,同时对低频信息进行上采样,将两者结合得到卷积之后的高频信息。在获得低频信息时,对输入特征图中的低频信息进行常规卷积操作,同时对高频信息进行池化,将两者结合得到卷积之后的低频信息,如图 13 所示,其中, X^H, X^L 分别表示输入的高频和低频特征图, Y^H, Y^L 代表输出特征中的高频和低频特征图, $W^{H \rightarrow H}, W^{L \rightarrow H}, W^{H \rightarrow L}, W^{L \rightarrow L}$ 分别表示根据输入的高频和低频特征得到输出高频和低频特征时的权值矩阵, pool 代表池化操作, upsample 代表上采样操作, $h, w, 0.5h, 0.5w$ 分别表示高频和低频信息的高度和宽度, $(1-\alpha_{in}), \alpha_{in}, (1-\alpha_{out}), \alpha_{out}$ 分别表示输入和输出时高频和低频特征的通道数, $Y^{H \rightarrow H}, Y^{L \rightarrow H}$ 分别表示从输入的高维特征图中得到的部分输出的高维特征和从输入的低维特征中得到的部分输出的高维特征, $Y^{H \rightarrow L}, Y^{L \rightarrow L}$ 分别表示从输入的高维特征图中得到的部分输出的低维特征和从输入的低维特征中得到的部分输出的低维特征。

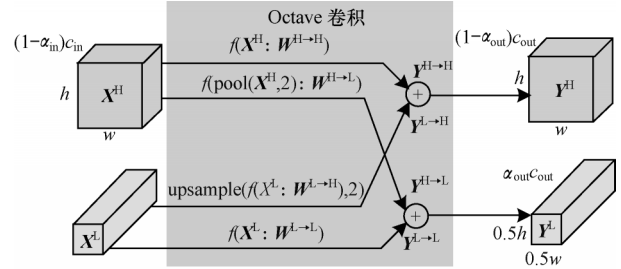


图 13 Octave 卷积操作

Fig.13 Octave convolution operation

根据实验效果, Octave 卷积可用于 ResNet^[13]、GoogLeNet^[31] 等基线网络及 MobileNet、MobileNet V2、ShuffleNet、ShuffleNet V2 等常规轻量化网络进行网络优化,实现轻量化处理。虽然 Octave 卷积对于存储空间优化效果较好,但是在提高运算速度和效率方面有待进一步提升。

1.7 基于 Ghost 特征的 GhostNet

传统深度神经网络的轻量化方法研究主要集中在减少参数量及改进卷积方式。2020 年, HAN 等^[25] 对深度神经网络特征图进行分析,发现常规卷积中特征图的冗余性在神经网络结构中很少被关注,为了从特征图冗余的角度实现网络结构轻量化, GhostNet 应运而生。如图 14 所示,对常规卷积生成的特征图进行可视化,其中同色方框内的特征图非常相似,这说明在训练好的神经网络进行前向传播时,中间过程使用的特征图含有大量相似的冗余特征,这样做的目的是为了使神经网络对输入的特征有更全面的理解。

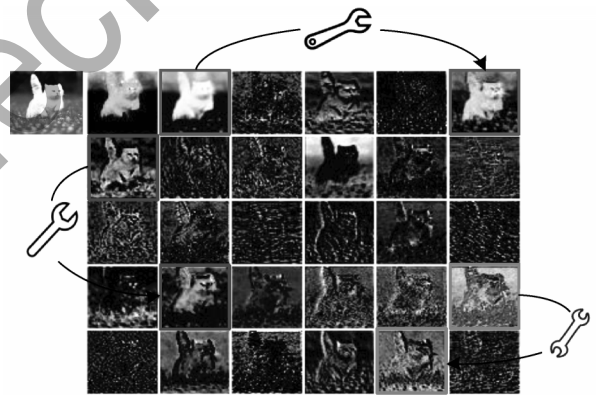


图 14 常规卷积的可视化分析

Fig.14 Visual analysis of conventional convolution

在此基础上, GhostNet 为了使用更低的成本完成更多的特征映射,采用线性变化得到 Ghost 特征。GhostNet 模块^[26]如图 15 所示,其中 Φ_k 表示对初次卷积之后的特征图进行线性变换。首先使用较少的卷积核对输入进行常规卷积,获得通道较少的输出特

征并将其作为内在特征图,然后对内在特征图的每个通道进行线性变换,得到其对应的 Ghost 特征图,最后将内在特征图与 Ghost 特征图进行通道连接,取得最终的 GhostNet 卷积输出特征。

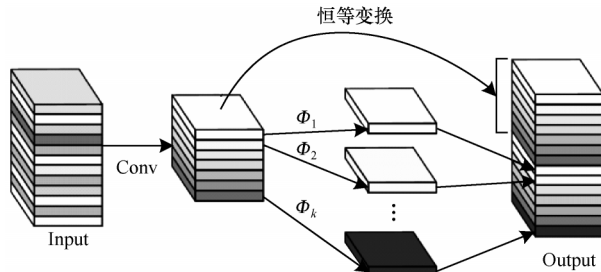


图 15 GhostNet 模块

Fig.15 GhostNet module

GhostNet 使用 GhostNet 模块代替传统 MobileNet 中的 bottleneck 层, GhostNet 整体结构如表 2 所示,其中, SE 代表是否在该操作中使用 SE 模块, 1 表示使用 SE 模块, 0 表示未使用 SE 模块, G-bn 表示使用 GhostNet 模块构建的 bottleneck 结构。GhostNet 模块具有很强的即插即用性,可以用于优化现有深度神经网络结构或者轻量化网络结构,对于神经网络运算速度优化效果较明显,但不能有效降低轻量化过程中的参数量及存储空间。

表 2 GhostNet 整体结构

Table 2 The overall structure of GhostNet

输入尺寸	操作	扩展系数	c	SE	s
224×224×3	3×3 Conv 2d	—	16	0	2
112×112×16	G-bn	16	16	0	1
112×112×16	G-bn	48	24	0	2
56×56×24	G-bn	72	24	0	1
56×56×24	G-bn	72	40	1	2
28×28×40	G-bn	120	40	1	1
28×28×40	G-bn	240	80	0	2
14×14×80	G-bn	200	80	0	1
14×14×80	G-bn	184	80	0	1
14×14×80	G-bn	184	80	0	1
14×14×80	G-bn	480	112	1	1
14×14×112	G-bn	672	112	1	1
14×14×112	G-bn	672	160	1	2
7×7×160	G-bn	960	160	0	1
7×7×160	G-bn	960	160	1	1
7×7×160	G-bn	960	160	0	1
7×7×160	G-bn	960	160	1	1
7×7×160	G-bn	—	960	0	1
7×7×960	7×7 Avgpool	—	—	0	—
1×1×960	1×1 Conv 2d	—	1 280	0	1
1×1×1 280	Full-connected	—	1 000	0	—

2 基于神经网络结构搜索的轻量化方法

2.1 神经网络结构搜索

人工设计的轻量化深度神经网络虽然得到了广泛应用,但是人工方法需要丰富的神经网络设计经验,以及在设计整体网络的模块和超参数时需要投入大量的人力和时间。随着强化学习的快速发展,基于神经网络结构搜索^[26]的轻量化方法应运而生。MobileNet、MobileNet V2、ShuffleNet、ShuffleNet V2 传统轻量化网络都是将各自的基本单元堆叠成相应的神经网络结构,这种堆叠基本单元的方式用到的超参数是一个有序数列,而循环神经网络(Recurrent Neural Network, RNN)^[32]则是擅长学习这种有序数列。神经网络结构搜索的主要目的是利用强化学习方法,在搜索空间中搜索到最适合的基本单元中的超参数,再将搜索到的基本单元进行堆叠得到神经网络结构搜索的轻量化网络。神经网络结构搜索流程如图 16 所示。

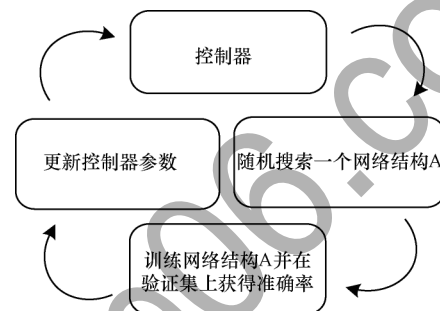


图 16 神经网络结构搜索流程

Fig.16 Procedure of neural network structure search

控制器根据结构搜索形成搜索空间内的子网络结构,将神经网络结构搜索应用于搜索 ResNet、Inception 等具有跳跃连接的 CNN 网络,并将整个控制器分成 N 段,每段控制器的搜索流程如图 17 所示。

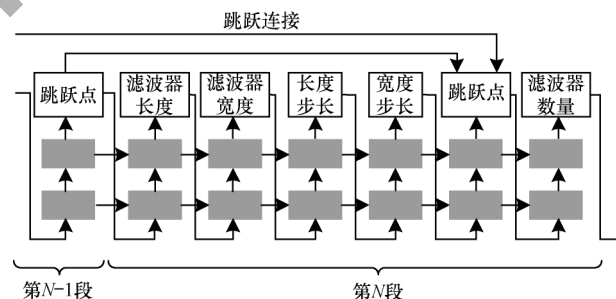


图 17 控制器搜索流程

Fig.17 Procedure of controller search

通过训练控制器能够学习到一层卷积层所需的所有超参数,并且基于 RNN 注意力机制为其添加可以学习的跳跃点。为了判断第 N 段是否与之前的某一段有跳跃连接,神经网络结构搜索添加了 $N-1$ 个

跳跃点,用于学习是否具有跳跃连接。具体地,第 N 段的跳跃点由2个隐节点和1个Sigmoid激活函数确定,第 $N-1$ 段的输出为第 N 段的输入的概率计算公式如下:

$$P(\text{第}N-1\text{段的输出是第}N\text{段的输入}) = \text{Sigmoid}(\mathbf{v}^T \cdot \tanh(W_{N-1} \cdot h_{N-1} + W_N \cdot h_N)) \quad (9)$$

其中: \mathbf{v}^T 、 W_{N-1} 、 W_N 是需要学习的网络中的超参数; h_{N-1} 和 h_N 分别是第 $N-1$ 段和第 N 段的跳跃点的状态。根据准确率来更新控制器的参数形成新的子网络,依次循环,得到子网络在训练完成后在目标验证集上的最高准确度。传统NasNet只会搜索复杂单元,然后对这些单元进行堆叠来构建网络结构,但是是一些与上述单元不同的神经层对于降低网络延迟、提高精度具有重要的作用,因此传统NasNet仅堆叠相同的单元层会忽略神经层的多样性。

2.2 基于神经网络结构搜索的MnasNet

在目前主流的轻量化方法中,基于神经网络结构搜索的自动化网络模型设计的优势非常明显,通过搜索并堆叠小的基本单元来产生移动端模型,但是没有考虑移动设备的约束条件。为了弥补这一不足,谷歌在2019年提出MnasNet^[27]。MnasNet在神经网络结构搜索过程中考虑了模型延迟,能够搜索到一个在模型精度和模型延迟之间取得最优平衡的深度神经模型。传统轻量化网络使用FLOPS间接评价模型延迟,MnasNet选择直接在移动设备上运行模型来得到真实的模型延迟参数,并且为了在较小的搜索空间中获得更高的网络性能,提出分解式层次搜索空间。

在平衡模型精度和模型延迟方面,MnasNet采用多目标优化方法来改进常规方法并将移动设备的真实延迟作为衡量标准。MnasNet设计模型 M ,在 M 中使用 $T_{\text{Delay}}(M)$ 表示网络在移动设备上的真实延迟, $A_{\text{Acc}}(M)$ 表示训练好的网络在目标验证集上的准确度,目标延迟设为 T ,则多目标优化如式(10)所示:

$$\begin{aligned} \max_M A_{\text{Acc}}(M) \\ \text{s.t. } T_{\text{Delay}}(M) \leq T \end{aligned} \quad (10)$$

MnasNet使用自定义加权法来近似帕累托最优解:

$$\max_M A_{\text{Acc}}(M) \times \left[\frac{T_{\text{Delay}}(M)}{T} \right]^\omega \quad (11)$$

$$\omega = \begin{cases} \alpha, & T_{\text{Delay}}(M) \leq T \\ \beta, & \text{其他} \end{cases} \quad (12)$$

其中, α 、 β 为超参数。

针对NasNet中仅搜索复杂单元且重复堆叠相同单元的情况,MnasNet采用分解式层次搜索空间,如图18

所示。将传统卷积神经网络划分为若干块,按照块的顺序逐渐减少输入分辨率,同时增加卷积核数。在分解式层次搜索中将传统网络分成不同的块后,每一个块中都具有相同的层结构,但操作和连接都由各自块的子搜索空间决定。分解式层次搜索空间的优势是可以平衡层的多样性和整个搜索空间的尺寸。

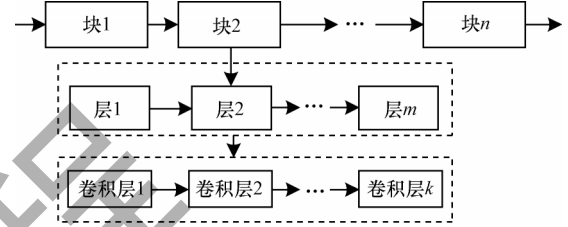


图18 分解式层次搜索空间

Fig.18 Decomposed hierarchical search space

MnasNet的搜索步骤如下:1)基于循环神经网络生成的控制器来搜索并生成深度神经网络;2)在目标数据集上训练控制器生成的网络,并且在验证集上得到模型精度;3)推理得到移动端的真实延迟,根据模型运行之后得到的模型精度和推理延迟,利用近似策略来最大化期望(PPO)奖励并更新控制器,一直循环直至完成所有步骤或者参数收敛。

MnasNet存在以下局限性:1)由于目前使用的MnasNet仍然包含大量人工设计特征,因此不能完全自行设计网络结构,未来神经网络结构搜索以及MnasNet的发展方向需在更广泛的搜索空间内进行,寻找具有高效率且轻量化的网络结构;2)由于目前使用的主流神经网络结构搜索技术多数基于谷歌模型,其过多专注模型准确率而忽略了底层硬件设备的影响,因此在实际应用中还有很大的优化空间。

2.3 基于混合深度可分离卷积的MixNet

在人工设计的轻量化网络和基于神经网络结构搜索的轻量化网络中都广泛使用了常规深度可分离卷积,但由于常规深度可分离卷积使用大小相同的卷积核,忽略了不同大小的卷积核对于卷积效果的影响。谷歌在2019年提出基于混合深度可分离卷积^[33]的MixNet结构,通过将多个尺寸的卷积核混叠到同一层的卷积中替换常规深度可分离卷积。混合深度可分离卷积如图19所示。首先,对输入特征进行分组卷积,并在不同的组使用不同尺寸的卷积核,使其能够捕获不同分辨率的特征模式。其次,混合深度可分离卷积基于神经网络结构搜索的轻量化方法,搜索深度神经网络分组卷积的分组数(1~5),第 i 组的卷积核大小计算公式如下:

$$k_{\text{kernel}}(i) = 2 \times i + 1 \quad (13)$$

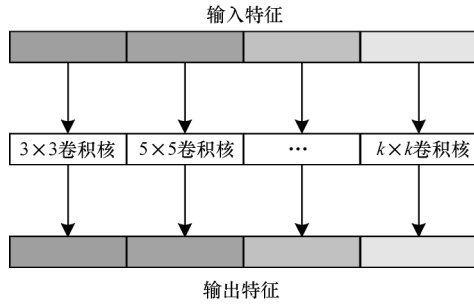


图 19 混合深度可分离卷积

Fig.19 Mix depthwise separable convolution

在通道数的选择上,混合深度可分离卷积考虑以下两种方案:1)使用平均通道数,以32通道4组为例,每组卷积的通道数为8;2)使用组号指数衰减,以32通道4组为例,各组通道数为16、8、4和4。具体卷积方式以输入特征尺寸为 $X^{h \times w \times c}$ 、深度卷积核为 $W^{k \times k \times c \times m}$ 为例,首先将其按通道分成 g 组,则原始输入转化为 $\hat{X}^{h \times w \times c_1}, \hat{X}^{h \times w \times c_2}, \dots, \hat{X}^{h \times w \times c_g}$,其中 $c_1 + c_2 + \dots + c_g$ 对应每组的卷积核尺寸为 $\hat{W}^{h \times k \times c_1 \times m}, \hat{W}^{h \times k \times c_2 \times m}, \dots, \hat{W}^{h \times k \times c_g \times m}$,第 δ 组对应的混合深度可分离卷积如式(14)所示:

$$Y_{x,y,z}^{\delta} = \sum_{-\frac{k_{\delta}}{2} \leq i \leq \frac{k_{\delta}}{2}, -\frac{k_{\delta}}{2} \leq j \leq \frac{k_{\delta}}{2}} \hat{X}_{x+i, y+j, z/m}^{\delta} \cdot \hat{W}_{i,j,z}^{\delta} \quad (14)$$

然后将各组卷积的输出在通道上进行连接,得到:

$$Y = \text{concat}(Y_{x,y,z}^1, Y_{x,y,z}^2, \dots, Y_{x,y,z}^g) \quad (15)$$

MixNet重点研究了卷积核尺寸对网络轻量化效果的影响,基于此提出混合深度可分离卷积,通过在同一卷积层中使用不同大小的卷积核,学习不同分辨率的特征,从而提升网络性能。

3 基于自动模型压缩的轻量化方法

3.1 模型压缩

在人工设计的轻量化神经网络结构部分,使用的轻量化方法多数依赖组卷积、深度可分离卷积等基本单元组成的块,再通过堆叠这些块来构建网络,由此导致的局限性就是其中存在极大的偶然性,有很大概率搜索不到空间的最优解。模型压缩主要分为细粒度修剪和粗粒度修剪两部分,细粒度修剪针对权重中的冗余部分进行修改,粗粒度修剪则是针对通道、行列、块等整个区域按照一定的稀疏率进行压缩。

模型压缩^[12,34]针对常规CNN,通常采用剪枝^[35-36]、权值共享^[15,37]与量化^[38]、哈夫曼编码^[39]这3种方法来减少参数量,达到轻量化目的。剪枝的本质是剪去深度神经网络中不必要的冗余权值和分支,仅保留对于神经网络的目标任务有效用的权值参数。权值共享是使用同一组参数来避免过多参数导致的训练和模型冗余。权值量化旨在用较小的比

特值来表示权值,以达到减少存储量的目的。哈夫曼编码^[40-41]是将两个权值最低的节点作为左右子树形成新的节点,再选取两个权值最低的节点作为左右子树形成新的节点,以此类推,达到根据使用频率来最大化节省存储空间的目的。

3.2 自动机器学习

为减少在传统机器学习中特征提取、模型选择、参数调试等方面的人工干预^[42-44],YAO等^[45]提出自动机器学习(AutoML)技术。AutoML的通用计算公式如下:

$$\begin{aligned} & \max(\text{配置学习目标的性能}) \\ & \text{s.t. 无人工干预及有限的计算资源} \end{aligned} \quad (16)$$

自动特征工程的目的是自动发掘并构造相关特征来优化模型性能,包括自动选择最优参数、自动选择最优方法。除此之外,还包含特征选择、特征降维^[46-48]、特征生成^[49-51]、特征编码^[52-54]等特定的特征增强方法。这些方法在自动机器学习领域有很大的发展空间。

3.3 基于AutoML的模型压缩

为了避免在模型压缩过程中过分依赖人工设计的启发式策略和基于规则的策略,西安交通大学与Google于2018年提出基于自动机器学习的模型压缩(AutoML for Model Compression, AMC)^[55]方法。AMC方法的性能明显优于基于规则的压缩策略,压缩模型能在保证准确性的同时大幅减少人工成本。由于压缩模型精度受各层稀疏性的影响,因此需要更细粒度的搜索空间。自动机器学习在模型压缩方面具有很大优势,采用强化学习中的深度确定性策略梯度法(DDPG)产生连续空间上的压缩率,通过大量学习达到提升网络精度和运行速度的目的。

图20给出了AMC引擎示意图。首先,使用一个预训练好的基线网络,代理部分从第 t 层中接收嵌入,输出稀疏率并根据稀疏率对 t 层进行模型压缩。然后,环境部分移动到第 $t+1$ 层进行操作,在完成对所有层的操作后评估整个网络的准确率。最后,将包含准确率和浮点运算量(或者参数量)的奖励反馈给代理部分。

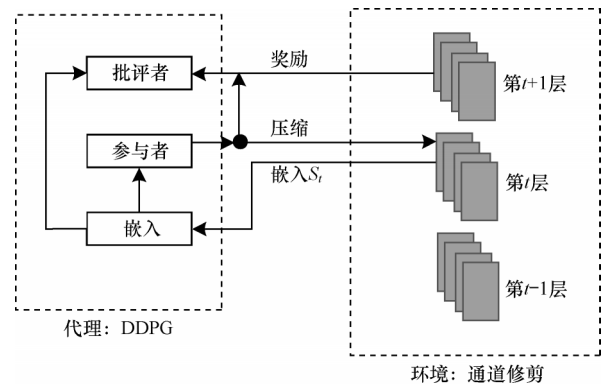


图 20 AMC引擎

Fig.20 AMC engine

在通道修剪环境中的每一层均使用11个特征来描述嵌入 S_t ,如式(17)所示:

$$S_t = \{t, n, c, h, w, s, k, \text{FLOPS}[t], \text{reduced}, \text{rest}, \alpha_{t-1}\} \quad (17)$$

其中:卷积核维度为 $k \times k \times c \times n$;输入数据的维度为 $h \times w \times c$; t 为层数; $\text{FLOPS}[t]$ 为第 t 层的浮点运算量; reduced 为前一层中减少的浮点运算总量; rest 是后面所有层中剩余的浮点运算量; α_{t-1} 为上一层的压缩率。在代理部分,使用深度确定性策略梯度法来连续控制压缩率。Block-QNN^[56]应用Bellman方程^[57]的变体形式,在Block-QNN^[56]之后,DDPG在探索过程中的转换公式过渡为 $(s_t; \alpha_t, R; s_{t+1})$,其中 R 为网络压缩后的奖励,在更新过程中减去基线奖励 r 以减少梯度估计方差^[58-59],如式(18)、式(19)所示:

$$L = \frac{1}{N} \sum_i (y_i - Q(s_i, \alpha_i | \theta^Q))^2 \quad (18)$$

$$y_i = r_i - r + \gamma \cdot Q(s_{i+1}, \mu(s_{i+1}) | \theta^Q) \quad (19)$$

其中: γ 为折扣因子。

根据经验得出,由于AMC误差(E)与 $\log_a F_{\text{FLOPS}}$ 或 $\log_a P_{\text{parameter}}$ ^[60]成反比,因此设计以下奖励公式:

$$r_{\text{FLOPS}} = -E \cdot \log_a F_{\text{FLOPS}} \quad (20)$$

$$r_{\text{parameter}} = -E \cdot \log_a P_{\text{parameter}} \quad (21)$$

AMC利用增强学习方法自动搜索设计空间,对于模型压缩的质量相较人工压缩有了质的飞跃。除此之外,AMC包含两种不同的奖励方案,在实现模型压缩的同时,又能保证模型精度,在ImageNet、MobileNet、MobileNet V2、ResNet和VGG上都展现出了优异的性能。

4 总结和展望

目前,智能移动设备趋向于边缘化、轻量化发展趋势。如何在尽可能保持神经网络模型精度的前提下,最大程度地降低模型延迟和存储空间是目前研究的热点问题。但是现有性能较好的人工设计的轻量化方法不仅需要耗费大量的人力资源,而且需要具备丰富的深度学习经验,才能使得延迟、运算速度、存储空间等神经网络模型的各项性能指标都达到要求。基于强化学习的神经网络结构搜索方法是目前主流的轻量化方法,但是大量基于神经网络结构搜索的轻量化方法仅专注于提高神经网络模型的准确率,却忽视了底层硬件设备的限制,这样得到的高效模型由于对硬件要求较高,通常难以在移动智能端进行部署应用。而基于神经网络结构搜索的轻量化方法通过强化学习控制器在搜索空间内搜索生成网络,无需耗费大量的人力资源,这是其得以快速发展的重要原因。

在人工设计的轻量化方法中,目前效果显著的方法多数集中于提高神经网络模型的运行速度或者降低存储空间,如何同时对其进行优化是下一步需要研究的重要方向。此外,基于神经网络结构搜索的轻量化方法的神经网络模型精度和轻量化效果也有待提升,可以通过设计更加合理的搜索空间来找到更合适的网络或者增加机器自动化学习在整个神经网络中所占的比重,而在基于自动模型压缩的轻量化方法中,如何进一步实现网络搜索与压缩的自动化也是亟待解决的难点之一。

5 结束语

本文研究深度神经网络的轻量化网络结构设计方法,对人工设计的轻量化方法、基于神经网络结构搜索的轻量化方法和基于自动模型压缩的轻量化方法进行创新点与优劣势对比,并指出深度神经网络的轻量化网络结构设计方法的评价指标已从单一的浮点运算量发展到如今包含模型延迟、存储空间等的综合评价指标,而研究人员应根据不同的应用场景合理选取轻量化评价标准和轻量化结构设计方法。后续可将神经网络结构搜索技术应用到轻量化网络模型搜索中,通过算法自动搜索合适的轻量化网络模型,进一步提升神经网络运算速度。

参考文献

- [1] WANG S C. Artificial neural network[M]. Berlin, Germany: Springer, 2003.
- [2] DOERSCH C, GUPTA A, EFROS A A. Unsupervised visual representation learning by context prediction[C]//Proceedings of IEEE International Conference on Computer Vision. Washington D. C., USA: IEEE Press, 2015: 1422-1430.
- [3] GIDARIS S, SINGH P, KOMODAKIS N. Unsupervised representation learning by predicting image rotations[EB/OL]. [2021-01-12]. <http://arxiv.org/abs/1803.07728>.
- [4] ZHOU Z H. A brief introduction to weakly supervised learning[J]. National Science Review, 2018, 5(1): 44-53.
- [5] RABINOVICH E, SZNAJDER B, SPECTOR A, et al. Learning concept abstractness using weak supervision[EB/OL]. [2021-01-12]. <https://arxiv.org/pdf/1809.01285.pdf>.
- [6] ARACHIE C, HUANG B. Adversarial label learning[C]//Proceedings of AAAI Conference on Artificial Intelligence. Palo Alto, SUA: AAAI Press, 2019: 3183-3190.
- [7] MUHAMMAD U R, YANG Y, HOSPEDALES T M, et al. Goal-driven sequential data abstraction[C]//Proceedings of IEEE/CVF International Conference on Computer Vision. Washington D. C., USA: IEEE Press, 2019: 71-80.

- [8] KRIZHEVSKY A, SUTSKEVER I, HINTON G. ImageNet classification with deep convolutional neural networks[C]// Proceedings of the 25th International Conference on Neural Information Processing Systems. Lake Tahoe, USA; Associates Inc., 2012; 1097-1105.
- [9] HAN K, GUO J, ZHANG C, et al. Attribute-aware attention model for fine-grained representation learning [C]// Proceedings of the 26th ACM International Conference on Multimedia. New York, USA; ACM Press, 2018; 2040-2048.
- [10] REN S, HE K, GIRSHICK R, et al. Faster R-CNN: towards real-time object detection with region proposal networks[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2015, 39(6): 1137-1149.
- [11] LIN T Y, GOYAL P, GIRSHICK R, et al. Focal loss for dense object detection[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2020, 42(2): 318-327.
- [12] CHEN L C, PAPANDREOU G, KOKKINOS I, et al. Semantic image segmentation with deep convolutional nets and fully connected CRFs[J]. Computer Science, 2014(4): 357-361.
- [13] HE K, ZHANG X, REN S, et al. Deep residual learning for image recognition[C]//Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. Washington D. C., USA; IEEE Press, 2016; 770-778.
- [14] LECUN Y, DENKER J S, SOLLIA S A, et al. Optimal brain damage[C]//Proceedings of Advances in Neural Information Processing Systems. New York, USA; ACM Press, 1990; 598-605.
- [15] HAN S, MAO H, DALLY W J. Deep compression: compressing deep neural networks with pruning, trained quantization and huffman coding[EB/OL]. [2021-01-12]. https://www.researchgate.net/publication/319770334_Deep_Compression_Compressing_Deep_Neural_Networks_with_Pruning_Trained_Quantization_and_Huffman_Coding.
- [16] CHENG Y, WANG D, ZHOU P, et al. A survey of model compression and acceleration for deep neural networks[EB/OL]. [2021-01-12]. <https://arxiv.org/abs/1710.09282>.
- [17] DENG L, LI G, HAN S, et al. Model compression and hardware acceleration for neural networks: a comprehensive survey[J]. Proceedings of the IEEE, 2020, 108(4): 485-532.
- [18] KRIZHEVSKY A, SUTSKEVER I, HINTON G E. ImageNet classification with deep convolutional neural networks[C]//Proceedings of Advances in Neural Information Processing Systems. New York, USA; ACM Press, 2012; 1097-1105.
- [19] IANDOLA F N, HAN S, MOSKEWICZ M W, et al. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5 MB model size[EB/OL]. [2021-01-12]. <https://arxiv.org/abs/1602.07360>.
- [20] HOWARD A G, ZHU M L, CHEN B, et al. MobileNet: efficient convolutional neural networks for mobile vision applications[EB/OL]. [2021-01-12]. <https://arxiv.org/abs/1704.04861>.
- [21] SANDLER M, HOWARD A, ZHU M, et al. MobileNet V2: inverted residuals and linear bottlenecks[C]//Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. Washington D. C., USA; IEEE Press, 2018; 4510-4520.
- [22] ZHANG X, ZHOU X, LIN M, et al. ShuffleNet: an extremely efficient convolutional neural network for mobile devices[C]//Proceedings of IEEE conference on computer vision and pattern recognition. Washington D. C., USA; IEEE Press, 2018; 6848-6856.
- [23] MA N, ZHANG X, ZHENG H T, et al. ShuffleNet V2: Practical guidelines for efficient CNN architecture design [C]//Proceedings of European Conference on Computer Vision. Berlin, Germany; Springer, 2018; 116-131.
- [24] CHEN Y, FAN H, XU B, et al. Drop an octave: reducing spatial redundancy in convolutional neural networks with octave convolution [C]//Proceedings of IEEE/CVF International Conference on Computer Vision. Washington D. C., USA; IEEE Press, 2019; 3435-3444.
- [25] HAN K, WANG Y, TIAN Q, et al. GhostNet: more features from cheap operations [C]//Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition. Washington D. C., USA; IEEE Press, 2020; 1580-1589.
- [26] ZOPH B, VASUDEVAN V, SHLENS J, et al. Learning transferable architectures for scalable image recognition [C]//Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. Washington D. C., USA; IEEE Press, 2018; 8697-8710.
- [27] TAN M, CHEN B, PANG R, et al. MnasNet: Platform-aware neural architecture search for mobile[C]//Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition. Washington D. C., USA; IEEE Press, 2019; 2820-2828.
- [28] SIMONYAN K, ZISSERMAN A. Very deep convolutional networks for large-scale image recognition[EB/OL]. [2021-01-12]. <https://arxiv.org/pdf/1409.1556.pdf>.
- [29] CHOLLET F. Xception: deep learning with depthwise separable convolutions [C]//Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. Washington D. C., USA; IEEE Press, 2017; 1251-1258.
- [30] SZEGEDY C, VANHOUCKE V, IOFFE S, et al. Rethinking the Inception architecture for computer vision[C]//Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. Washington D. C., USA; IEEE Press, 2016; 2818-2826.
- [31] SZEGEDY C, LIU W, JIA Y, et al. Going deeper with convolutions [C]//Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. Washington D. C., USA; IEEE Press, 2015; 1-9.
- [32] ZAREMBA W, SUTSKEVER I, VINYALS O. Recurrent neural network regularization [EB/OL]. [2021-01-12]. https://www.researchgate.net/publication/265469170_Recurrent_Neural_Network_Regularization.
- [33] TAN M X, LE Q V. MixConv: mixed depthwise convolutional kernels[EB/OL]. [2021-01-12]. <https://arxiv.org/abs/1907.09595v3>.

- [34] 李志军,杨楚哲,刘丹,等. 基于深度卷积神经网络的信息流增强图像压缩方法[J]. 吉林大学学报(工学版),2020,50(5):1788-1795.
LI Z J, YANG C X, LIU D, et al. Deep convolutional networks based image compression with enhancement of information flow [J]. Journal of Jilin University (Engineering and Technology Edition), 2020, 50(5): 1788-1795. (in Chinese)
- [35] SUN Y, WANG X, TANG X. Sparsifying neural network connections for face recognition[C]//Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. Washington D. C., USA: IEEE Press, 2016: 4856-4864.
- [36] LUO J H, WU J X. An entropy-based pruning method for CNN compression[EB/OL]. [2021-01-12]. <http://arxiv.org/abs/1706.05791>.
- [37] 胡黄水,赵思远,刘清雪,等. 基于动量因子优化学习率的BP神经网络PID参数整定算法[J]. 吉林大学学报(理学版),2020,58(6):1415-1420.
HU H S, ZHAO S Y, LIU Q X, et al. BP neural network PID parameter tuning algorithm based on momentum factor optimized learning rate [J]. Journal of Jilin University (Science Edition), 2020, 58(6): 1415-1420. (in Chinese)
- [38] HAN S, POOL J, TRAN J, et al. Learning both weights and connections for efficient neural networks[EB/OL]. [2021-01-12]. https://www.researchgate.net/publication/277959043_Learning_both_Weights_and_Connections_for_Efficient_Neural_Networks.
- [39] SAU B B, BALASUBRAMANIAN V N. Deep model compression: distilling knowledge from noisy teachers[EB/OL]. [2021-01-12]. <https://arxiv.org/pdf/1610.09650.pdf>.
- [40] WEN W, XU C, WU C, et al. Coordinating filters for faster deep neural networks[C]//Proceedings of IEEE International Conference on Computer Vision. Washington D. C., USA: IEEE Press, 2017: 658-666.
- [41] HINTON G, VINYALS O, DEAN J. Distilling the knowledge in a neural network[EB/OL]. [2021-01-12]. <http://arxiv.org/pdf/1503.02531>.
- [42] MITCHELL T M. Machine learning[M]. [S. l.]: McGraw-Hill Press, 2003.
- [43] SILVER D, HUANG A, MADDISON C J, et al. Mastering the game of go with deep neural networks and tree search [J]. Nature, 2016, 529(7587): 484-489.
- [44] XIONG W, DROPO J, HUANG X, et al. Toward human parity in conversational speech recognition[J]. IEEE/ACM Transactions on Audio, Speech, and Language Processing, 2017, 25(12): 2410-2423.
- [45] YAO Q M, WANG M S, HUGO J E, et al. Taking human out of learning applications: a survey on automated machine learning[EB/OL]. [2021-01-12]. <http://export.arxiv.org/abs/1810.13306>.
- [46] LIII K P. On lines and planes of closest fit to systems of points in space[J]. The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science, 1901, 2(11): 559-572.
- [47] FISHER R A. The use of multiple measurements in taxonomic problems[J]. Annals of Eugenics, 1936, 7(2): 179-188.
- [48] VINCENT P, LAROCHELLE H, BENGIO Y, et al. Extracting and composing robust features with denoising autoencoders [C]//Proceedings of the 25th International Conference on Machine Learning. Helsinki, Finland: [s. n.], 2008: 1096-1103.
- [49] KATZ G, SHIN E C R, SONG D. ExploreKit: automatic feature generation and selection [C]//Proceedings of the 16th International Conference on Data Mining. Washington D. C., USA: IEEE Press, 2016: 979-984.
- [50] KANTER J M, VEERAMACHANENI K. Deep feature synthesis: towards automating data science endeavors [C]//Proceedings of IEEE International Conference on Data Science and Advanced Analytics. Washington D. C., USA: IEEE Press, 2015: 1-10.
- [51] SMITH M G, BULL L. Genetic programming with a genetic algorithm for feature construction and selection[J]. Genetic Programming and Evolvable Machines, 2005, 6(3): 265-281.
- [52] ELAD M, AHARON M. Image denoising via sparse and redundant representations over learned dictionaries [J]. IEEE Transactions on Image Processing, 2006, 15(12): 3736-3745.
- [53] ZEILER M D, KRISHNAN D, TAYLOR G W, et al. Deconvolutional networks[C]//Proceedings of 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. Washington D. C., USA: IEEE Press, 2010: 2528-2535.
- [54] YU K, ZHANG T, GONG Y. Nonlinear learning using local coordinate coding[EB/OL]. [2021-01-12]. <https://www.mlpac.org/papers/lcc.pdf>.
- [55] HE Y, LIN J, LIU Z, et al. AMC: AutoML for model compression and acceleration on mobile devices [C]//Proceedings of European Conference on Computer Vision. Berlin, Germany: Springer, 2018: 784-800.
- [56] ZHONG Z, YAN J J, LIU C L. Practical network blocks design with Q-learning[EB/OL]. [2021-01-12]. <http://arxiv.org/pdf/1708.05552>.
- [57] WATKINS C J C H. Learning from delayed rewards[J]. Robotics and Autonomous Systems, 1995, 15(4): 233-235.
- [58] ZOPH B, LE Q V. Neural architecture search with reinforcement learning[EB/OL]. [2021-01-12]. https://www.researchgate.net/publication/309738632_Neural_Architecture_Search_with_Reinforcement_Learning.
- [59] CAI H, CHEN T Y, ZHANG W N, et al. Reinforcement learning for architecture search by network transformation [EB/OL]. [2021-01-12]. <http://arxiv.org/pdf/1707.04873>.
- [60] CANZIANI A, PASZKE A, CULURCIELLO E. An analysis of deep neural network models for practical applications[EB/OL]. [2021-01-12]. <https://arxiv.org/pdf/1605.07678.pdf>.