文章编号: 1001-0920(2018)03-0447-08 DOI: 10.13195/j.kzyjc.2017.0048

基于反卷积特征提取的深度卷积神经网络学习

吕恩辉, 王雪松, 程玉虎†

(中国矿业大学 信息与控制工程学院，江苏 徐州 221116)

摘 要: 在深度卷积神经网络的学习过程中,卷积核的初始值通常是随机赋值的. 另外,基于梯度下降法的网络参数学习法通常会导致梯度弥散现象. 鉴于此,提出一种基于反卷积特征提取的深度卷积神经网络学习方法. 首先,采用无监督两层堆叠反卷积神经网络从原始图像中学习得到特征映射矩阵;然后,将该特征映射矩阵作为深度卷积神经网络的卷积核,对原始图像进行逐层卷积和池化操作;最后,采用附加动量系数的小批次随机梯度下降法对深度卷积网络微调以避免梯度弥散问题. 在MNIST、CIFAR-10和CIFAR-100数据集上的实验结果表明,所提出方法可有效提高图像分类精度.

关键词: 反卷积神经网络；卷积神经网络；卷积核；动量系数；小批次随机梯度下降

中图分类号: TP18 **文献标志码:** A

Deep convolution neural network learning based on deconvolution feature extraction

LV En-hui, WANG Xue-song, CHENG Yu-hu†

(School of Information and Control Engineering，China University of Mining and Technology，Xuzhou 221116，China)

Abstract: During the learning process of the deep convolution neural network(DCNN), the initial values of convolution kernels are usually randomly assigned. In addition, the learning rule of network parameters based on gradient descent usually results in gradient vanishing phenomenon. Aiming at the above problems, a learning method for the DCNN based on deconvolution feature extraction is proposed. Firstly, an unsupervised two-layer stacked deconvolution neural network is used to learn feature mapping matrixes from the original images. Then, the learned feature mapping matrixes are used as the convolution kernels to convolve and pool with the images in a layer-wise manner. Finally, the DCNN is fine-tuned by using the mini-batch stochastic gradient descent method with a momentum coefficient, which can avoid the gradient vanishing problem. Experimental results on MNIST, CIFAR-10 and CIFAR-100 data sets show that, the proposed method can effectively improve the accuracy of image classification.

Keywords: deconvolution neural network；convolution neural network；convolution kernel；momentum coefficient；mini-batch stochastic gradient descent

0 引 言

作为机器学习的一个重要分支,深度学习的实质是通过构建多个隐层的机器学习模型和海量数据的训练来学习更具表达能力的特征,最终提升分类或预测的准确率. 与局部图像特征描述SIFT和HOG相比,深度学习模型学习得到的特征更能刻画数据丰富的内在信息. 作为深度学习重要模型之一的深度卷积神经网络(DCNN)在图像分类[1-2]、目标检索[3-4]、行为识别[5]和其他视觉任务[6-7]等领域中均得到了成功应用.

在对深度卷积神经网络进行训练时,需要事先将卷积核随机初始化为一个很小的、接近于零的值. 随机赋值的初始化方法简单、直接,但是其缺点也是显而易见的,不仅会导致网络的学习速度较慢,而且通常会导致学习过程中陷入局部最优问题[8]. 作为一种典型的无监督层次化图像描述结构,反卷积网络(DeCN)[9]和一些深度学习方法有着紧密的联系. DeCN能够从底层边界到高层目标自动提取丰富的隐式特征,思想上与LeCun等[10]的DCNN十分相似. 但是,DeCN在实际操作中又与DCNN有着许多

收稿日期: 2017-01-13；**修回日期:** 2017-06-12.

基金项目: 国家自然科学基金项目(61472424, 61772532).

责任编委: 吕金虎.

作者简介: 吕恩辉(1989−), 男, 博士生, 从事深度学习的研究；程玉虎(1973−), 男, 教授,博士生导师, 从事机器学习、模式识别与智能系统等研究.

†通讯作者. E-mail: chengyuhu@163.com

不同,它的每层网络信号自上向下,卷积核和特征图作卷积求和后可生成逼近原始图像的输入信号[11]. 对于图像的分析和合成,通过构建层次化反卷积网络可以学习到更为鲁棒的图像特征表示[12]. 利用反卷积网络自动提取图像高层特征,此高层特征通常比原始数据集更能反映样本的本质. 因此,反卷积提取到的特征映射矩阵作为DCNN的卷积核在求其目标函数时,可避免因卷积核过小而导致梯度接近于0,使得隐藏层单元输出均匀分布[13].

DCNN网络的训练算法通常采用基于梯度下降法的逐层训练机制,网络自底向上逐层训练,上一层的输出作为下一层的输入. 这种学习机制的缺点在于第1层后的图像像素被遗弃,因此模型的更高层与输入之间连接得越来越稀疏,造成误差校正信号从顶层往下越来越小,尤其是从远离最优区域开始容易收敛到局部最小值. 另外,在使用反向传播算法传播梯度的时候,随着传播深度的增加,DCNN通常采用的Sigmoid和tanh激活函数梯度的幅度会急剧减小,从而导致参数更新非常缓慢,不能有效学习,出现梯度弥散现象. 为此,Glorot等[14]采用非线性、非饱和型激活函数ReLU,该函数不容易饱和,能够避免梯度弥散现象. 另外,ReLU的函数值和导数值的计算也比较简单,其训练速度是tanh型神经元的数倍,有利于加快模型在GPU中的快速迭代. 但是,当存在一个非常大的梯度在流经某ReLU神经元并对其参数进行更新后,其他数据将对该神经元无法产生激活作用,从而造成该神经元的梯度永远为0. Ioffe等[15]提出了批规范化(BN)方法,通过归一化将每层神经网络输入分布强制拉回到标准正态分布上,从而使得激活函数输入值落入对输入较为敏感的区域,达到避免梯度弥散的目的. Qian等[16]使用小批次随机梯度下降(MB-SGD)法完成对DCNN参数的训练,在计算时间和效果上优于随机梯度下降法. 为此,基于小批次随机梯度下降法,在不计算Hessian矩阵的情况下直接求取误差函数的二次导信息,并在此信息的基础上引入一动量系数来改善MB-SGD中反向传播算法的收敛性,从而进一步避免梯度弥散问题,提高模型泛化能力.

本文针对DCNN的卷积核初始化方法,首先提出一种基于反卷积特征提取的卷积核初始化方法,采用一个两层的堆叠反卷积神经网络从原始图像中学习其隐藏的高层特征表示,得到一个特征映射矩阵;然后,将学习得到的特征映射矩阵投入到DCNN模型中,以此作为卷积核从原始图像中提取特征并卷积和池化特征. 在MNIST、CIFAR-10和CIFAR-100数据集上的实验结果表明,所提出方法可有效提高图像分类精度.

1 基于反卷积特征提取的深度卷积神经网络学习

1.1 网络系统结构

基于反卷积特征提取的深度卷积神经网络(DeDCNN)学习系统结构如图1所示,主要由两部分构成,其中cccp表示多层感知器层. 阶段I为基于反卷积网络的特征提取,给定原始图像和随机初始化的特征映射矩阵,利用两层堆叠反卷积网络直接对所有原始图像至隐层特征空间执行稀疏分解映射操作. 同时,解码器将隐层特征反向映射回输入空间,重构出相似于原始图像的输入. 通过上述反卷积网络的训练,可从原始图像中学习到其隐藏的高层特征表示,得到一个更新后的特征映射矩阵. 阶段II为深度卷积神经网络训练,首先将阶段I学习得到的特征映射矩阵作为DCNN的卷积核,对输入图像执行卷积和池化操作;此外,在每个卷积层后均引入一个多层感知器层,实现跨通道的信息交互与整合;最后,采用附加动量系数的小批次随机梯度下降法对DCNN微调以避免梯度弥散问题.

1.2 反卷积特征提取

使用反卷积神经网络对原始输入图像进行无监督特征提取. 反卷积神经网络是一个由反卷积层、反池化层和矫正层交替组成的神经网络,层次化的网络结构有助于提取图像的中层和高层特征. 本文使用两层反卷积层堆叠出层次化网络模型,并利用高效优化技术直接提取特征映射矩阵. 两层堆叠反卷积神经网络作为一个仅有解码器的神经网络,在解码阶段网络的每一层沿着自上而下的方向无监督提取特征,并用卷积核卷积特征重构出相似于原始图像的输入图像. 模型的每层网络仅重构出网络的下一层,重构图像和输入图像的差值信息作为目标函数的误差信息. 网络结构的层与层之间采用全连接的方式进行联通,通过训练调整参数,得到每层权重,可计算出输入信号的几种不同表示,这些表示记为输入的特征表达. 假设网络的输入和输出相同,两层堆叠反卷积网络模型的推理操作过程如图2所示. 映射操作是通过卷积核卷积输入图像$\boldsymbol{x}$将误差信号从第1层映射到第2层,从而得到相应特征图; 重构操作是从特征图$\boldsymbol{z}_2$开始进行卷积直到输入层,其具体计算推导过程如下所述.

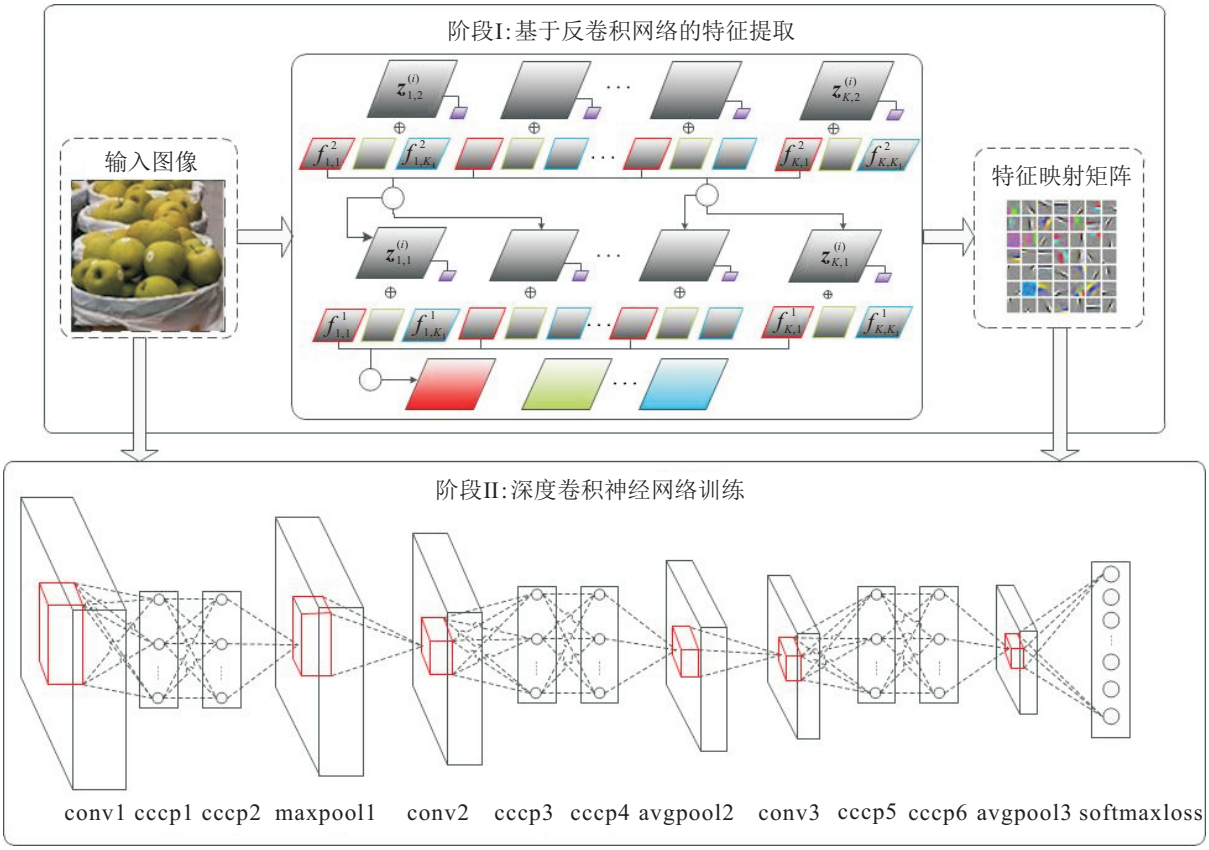

图 1　DeDCNN 系统结构

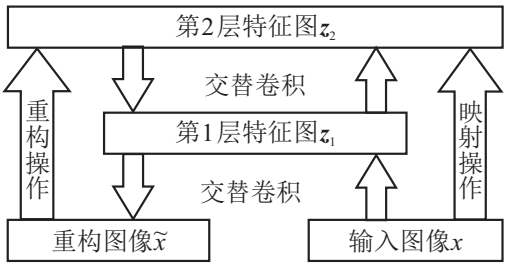

图 2　两层堆叠反卷积网络推理操作过程

假设第i幅原始输入图像为$\boldsymbol{x}^{(i)}$,由K_0个颜色通道构成$\boldsymbol{x}^{(i)}=\{\boldsymbol{x}_1^{(i)},\boldsymbol{x}_2^{(i)},\cdots,\boldsymbol{x}_{K_0}^{(i)}\}$,通过对隐层特征图$\boldsymbol{z}_k^{(i)}$和卷积核$\boldsymbol{f}_{k,c}$作反卷积后求$K$个线性和得到图像相应的第$c$个颜色通道映射,即

$$\sum_{k=1}^{K}\boldsymbol{z}_k^{(i)}\oplus\boldsymbol{f}_{k,c}=\boldsymbol{x}_c^{(i)}. \tag{1}$$

其中:$\boldsymbol{z}_k^{(i)}$为$(N_r+H-1)\times(N_c+H-1)$特征图,每张输入图像都有一个对应的特征图;$\oplus$为反卷积操作符;$\boldsymbol{f}_{k,c}$为$H\times H$大小的卷积核;$\boldsymbol{x}_c^{(i)}$为$N_r\times N_c$的图像. 为满足解的唯一性,在特征图$\boldsymbol{z}_k^{(i)}$中引入正则化稀疏项,单样本目标函数为

$$C_1(\boldsymbol{x}^{(i)})=\frac{\tau}{2}\sum_{c=1}^{K_0}\left\|\sum_{k=1}^{K}\boldsymbol{z}_k^{(i)}\oplus\boldsymbol{f}_{k,c}^1-\boldsymbol{x}_c^{(i)}\right\|_2^2+\sum_{k=1}^{K}|\boldsymbol{z}_k^{(i)}|^p. \tag{2}$$

其中:$\boldsymbol{f}_{k,c}^1$为第1层的卷积核;p为正则化项范数,通常设$p=1$;τ通常取常值1,起到平衡图像$\boldsymbol{x}^{(i)}$和稀疏后特征图$\boldsymbol{z}_k^{(i)}$的作用. 值得注意的是,反卷积网络重构阶段沿自上而下的方向对隐层特征图作卷积合成输入图像,不同于稀疏自编码和深度信念网络的算法原理,它仅通过优化目标函数C_1求得输入图像的隐层特征描述.

假设包含N个样例的样本集为$\boldsymbol{x}=\{\boldsymbol{x}^{(1)},\cdots,\boldsymbol{x}^{(N)}\}$,第2层整体目标函数可通过堆叠单样本目标函数形成,即

$$C_2(\boldsymbol{x})=\frac{\tau}{2}\sum_{i=1}^{N}\sum_{c=1}^{K_0}\left\|\sum_{k=1}^{K}\boldsymbol{g}_{k,c}^2(\boldsymbol{z}_{k,2}^{(i)}\oplus\boldsymbol{f}_{k,c}^2)-\boldsymbol{z}_{k,1}^{(i)}\right\|_2^2+\sum_{i=1}^{N}\sum_{k=1}^{K}|\boldsymbol{z}_{k,2}^{(i)}|^p. \tag{3}$$

其中:$\boldsymbol{z}_{k,1}^{(i)}$和$\boldsymbol{z}_{k,2}^{(i)}$分别为第1层和第2层特征图;$\boldsymbol{g}_{k,c}^2$为固定二进制矩阵的元素,起到连接特征图的作用.

反卷积网络的训练过程可分如成下两个阶段:在第1阶段,给定卷积核$\boldsymbol{f}_{k,c}^2$,求取特征图$\boldsymbol{z}_{k,2}^{(i)}$使得目标函数$C_2(\boldsymbol{x})$最小;在第2阶段,给定特征图$\boldsymbol{z}_{k,2}^{(i)}$,求解卷积核$\boldsymbol{f}_{k,c}^2$使得目标函数$C_2(\boldsymbol{x})$最小. 在具体训练过程中,需要首先引入一个辅助目标函数$\hat{C}_2(\boldsymbol{x})$以避免陷入局部最优[17],然后求解辅助目标函数$\hat{C}_2(\boldsymbol{x})$的最小值,使得辅助变量$\boldsymbol{\xi}_{k,2}^{(i)}$和特征图$\boldsymbol{z}_{k,2}^{(i)}$相逼近. 辅助目标函数$\hat{C}_2(\boldsymbol{x})$定义为

$$\hat{C}_2(\boldsymbol{x})=$$
$$\frac{\tau}{2}\sum_{i=1}^{N}\sum_{c=1}^{K_0}\Big\|\sum_{k=1}^{K}\boldsymbol{g}_{k,c}^2(\boldsymbol{z}_{k,2}^{(i)}\oplus\boldsymbol{f}_{k,c}^2)-\boldsymbol{z}_{k,1}^{(i)}\Big\|_2^2+$$
$$\frac{\beta}{2}\sum_{i=1}^{N}\sum_{k=1}^{K}\|\boldsymbol{z}_{k,2}^{(i)}-\boldsymbol{\xi}_{k,2}^{(i)}\|_2^2+\sum_{i=1}^{N}\sum_{k=1}^{K}|\boldsymbol{\xi}_{k,2}^{(i)}|^p,\quad(4)$$

其中β是一连续变量,从一个极小的初始值缓慢增加,直到特征图$\boldsymbol{z}_{k,2}^{(i)}$与辅助变量$\boldsymbol{\xi}_{k,2}^{(i)}$最终逼近. 在求取特征图$\boldsymbol{z}_{k,2}^{(i)}$阶段,假设辅助变量$\boldsymbol{\xi}_{k,2}^{(i)}$给定,关于目标函数$\hat{C}_2(\boldsymbol{x})$对特征图$\boldsymbol{z}_{k,2}^{(i)}$的导数为

$$\frac{\partial\hat{C}_2(\boldsymbol{x})}{\partial\boldsymbol{z}_{k,2}^{(i)}}=\tau\sum_{c=1}^{K_0}(\boldsymbol{F}_{k,c}^2)^{\mathrm{T}}\Big(\sum_{k=1}^{K}\boldsymbol{F}_{k,c}^2\boldsymbol{z}_{k,2}^{(i)}-\boldsymbol{z}_{k,1}^{(i)}\Big)+$$
$$\beta(\boldsymbol{z}_{k,2}^{(i)}-\boldsymbol{\xi}_{k,2}^{(i)}).\quad(5)$$

其中:如果$\boldsymbol{g}_{k,c}^2=1$,则$\boldsymbol{F}_{k,c}^2$等同于用$\boldsymbol{f}_{k,c}^2$卷积的稀疏卷积矩阵;如果$\boldsymbol{g}_{k,c}^2=0$,则$\boldsymbol{F}_{k,c}^2$为零矩阵.

特征图$\boldsymbol{z}_{k,2}^{(i)}$的更新过程如下:首先给定$i$,令$\partial\hat{C}_2(\boldsymbol{x})/\partial\boldsymbol{z}_{k,2}^{(i)}=0$,求解$\boldsymbol{z}_{k,2}^{(i)}$即是求解下述$K(N_r+H-1)(N_c+H-1)$维线性系统:

$$\boldsymbol{A}\begin{bmatrix}\boldsymbol{z}_{1,2}^{(i)}\\ \vdots\\ \boldsymbol{z}_{K,2}^{(i)}\end{bmatrix}=\begin{bmatrix}\sum\limits_{c=1}^{K_0}(\boldsymbol{F}_{1,c}^2)^{\mathrm{T}}\boldsymbol{z}_{c,1}^{(i)}+\dfrac{\beta}{\tau}\boldsymbol{\xi}_{1,2}^{(i)}\\ \vdots\\ \sum\limits_{c=1}^{K_0}(\boldsymbol{F}_{K,c}^2)^{\mathrm{T}}\boldsymbol{z}_{c,1}^{(i)}+\dfrac{\beta}{\tau}\boldsymbol{\xi}_{K,2}^{(i)}\end{bmatrix},\quad(6)$$

其中

$$\boldsymbol{A}=$$
$$\begin{bmatrix}\sum\limits_{c=1}^{K_0}(\boldsymbol{F}_{1,c}^2)^{\mathrm{T}}\boldsymbol{F}_{1,c}^2+\dfrac{\beta}{\tau}\boldsymbol{I} & \cdots & \sum\limits_{c=1}^{K_0}\boldsymbol{F}_{1,c}^2(\boldsymbol{F}_{K,c}^2)^{\mathrm{T}}\\ \vdots & \ddots & \vdots\\ \sum\limits_{c=1}^{K_0}(\boldsymbol{F}_{K,c}^2)^{\mathrm{T}}\boldsymbol{F}_{1,c}^2 & \cdots & \sum\limits_{c=1}^{K_0}(\boldsymbol{F}_{K,c}^2)^{\mathrm{T}}\boldsymbol{F}_{1,c}^2+\dfrac{\beta}{\tau}\boldsymbol{I}\end{bmatrix};$$

然后采用共轭梯度下降法即可求得式(6)的最优解.

在求取辅助变量$\boldsymbol{\xi}_{k,2}^{(i)}$阶段,假设特征图$\boldsymbol{z}_{k,2}^{(i)}$固定,$\boldsymbol{\xi}_{k,2}^{(i)}$的最优问题可转化为对特征图求解1D最优问题. 若$p=1$,则辅助变量$\boldsymbol{\xi}_{k,2}^{(i)}$的求解表达式为

$$\boldsymbol{\xi}_{k,2}^{(i)}=\max\Big(|\boldsymbol{z}_{k,2}^{(i)}|-\frac{1}{\beta},0\Big)\frac{\boldsymbol{z}_{k,2}^{(i)}}{|\boldsymbol{z}_{k,2}^{(i)}|}.\quad(7)$$

根据梯度下降法对卷积核$\boldsymbol{f}_{k,c}^2$进行更新,有

$$\frac{\partial\hat{C}_2(\boldsymbol{x})}{\partial\boldsymbol{f}_{k,c}^2}=$$
$$\tau\sum_{i=1}^{N}\sum_{c=1}^{K_0}(\boldsymbol{Z}_{k,2}^{(i)})^{\mathrm{T}}\Big(\sum_{k=1}^{K}g_{k,c}^2\boldsymbol{Z}_{k,2}^{(i)}\boldsymbol{f}_{k,c}^2-\boldsymbol{z}_{c,1}^{(i)}\Big),\quad(8)$$

其中$\boldsymbol{Z}$为相似于$\boldsymbol{F}$的卷积矩阵.

1.3 深度卷积神经网络训练

DCNN包括卷积层、池化层和多层感知器层,输出采用softmaxloss分类器进行多任务分类. 卷积层中某一特征图依据权值共享策略由相同卷积核卷积生成,以降低模型复杂度,减少训练参数的个数. 经池化层对卷积层特征进行非线性下采样,过滤掉相邻相似特征,降低计算复杂度并增强局部特征的不变性. 最后使用softmaxloss分类器对学习到的深层次特征建立一个多任务分类器. 与传统DCNN相比,此处使用的多层感知器层实际上相当于作了两个1×1的卷积,只改变了卷积核大小,不会改变特征图大小. 在传统卷积层后接入两个1×1卷积可实现跨通道的信息交互与整合,进一步增强网络泛化能力[18]. 参照目前主流DCNN训练技巧,采用非线性激活函数ReLU替代Sigmoid函数,增强了特征稀疏性和线性可分性.

对包含N个样例的训练样本集$\{(\boldsymbol{x}^{(1)},\boldsymbol{y}^{(1)}),\cdots,(\boldsymbol{x}^{(N)},\boldsymbol{y}^{(N)})\}$,卷积层将输入数据或者前一层特征图与多组卷积核进行卷积运算. 由于图像为RGB三通道图像,输入数据$\boldsymbol{x}^{(i)}$与卷积核均为三维结构,将三维的卷积核分别与3个输入进行卷积,再将这3个输出的对应位置相加,特征图计算如下:

$$\boldsymbol{x}_{j'}^{l}=f\Big(\sum_{j\in M^l}\boldsymbol{x}_j^{l-1}*\boldsymbol{f}_{jj'}^{l}\Big).\quad(9)$$

其中:l为该卷积层所在的层数,$\boldsymbol{x}_{j'}^{l}$为第l层第j'个输出特征图,$\boldsymbol{f}_{jj'}^{l}$为连接第$l-1$层第j个特征图与第l层第j'个特征图的卷积核,M^l为第$l-1$层的特征图个数,$f(\cdot)$为非线性激活函数ReLU,$*$为卷积运算操作符. 由于输入数据经过卷积操作后其分布发生变化,为解决数据分布发生变化引起的内部协变量偏移问题,引入BN对数据分布加以处理. 经过BN处理后,数据相当于PCA降维,即降低了特征间的相关性,数据均值、标准差归一化使得每一维特征均值为0,标准差为1. 将BN置于网络激活函数前卷积操作之后,因此前向传导卷积计算式(9)变换为

$$\boldsymbol{x}_{j'}^{l}=f\Big(\mathrm{BN}\Big(\sum_{j\in M^l}\boldsymbol{x}_j^{l-1}*\boldsymbol{f}_{jj'}^{l}\Big)\Big).\quad(10)$$

池化层对上一卷积层的特征图进行下采样,得到与输入特征映射一一对应的维度更小的输出特征映射,即

$$\boldsymbol{x}_{j'}^{l}=f(\beta_{j'}^{l}\mathrm{down}(\boldsymbol{x}_{j'}^{l-1})).\quad(11)$$

其中:down$(\cdot)$为下采样函数,β为下采样系数. 同样,池化层后使用BN对特征图进行处理,将BN置于网络激活函数前和池化操作之后. 因此,前向传导池化

计算式(11)变换为

$$\boldsymbol{x}_{j'}^{l}=f(\mathrm{BN}(\beta_{j'}^{l}\mathrm{down}(\boldsymbol{x}_{j'}^{l-1}))). \tag{12}$$

DCNN模型训练采用反向传播逐层训练机制,需要训练的参数包括卷积核$\boldsymbol{f}$. 令$\hat{y}_n^{(i)}$表示第i个样本对应标签的第m维,$y_m^{(i)}$表示第i个样本对应输出的m维,则平方误差代价函数为

$$J(\boldsymbol{f})=\frac{1}{2}\sum_{i=1}^{N}\sum_{m=1}^{M}\left(\hat{y}_m^{(i)}-y_m^{(i)}\right)^2, \tag{13}$$

其中M表示类别总数. 采用MB-SGD对卷积核进行更新,有

$$\boldsymbol{f}_{jj'}^{l}(t)=\boldsymbol{f}_{jj'}^{l}(t-1)-\frac{1}{N}\eta\sum_{i=1}^{N}\frac{\partial J}{\partial \boldsymbol{f}_{jj'}^{l}(t-1)}. \tag{14}$$

其中:t为当前时刻,η为学习率. 在使用MB-SGD训练网络的过程中,当梯度保持方向变化时,误差表面沿不同方向有着不同的曲率,容易造成表面上的点随着梯度的连续下降从一边振荡到另一边,从而使得梯度无法到达最小值. 为此,考虑在MB-SGD中不仅保留上一时刻的梯度矢量信息,而且保留在上一时刻网络参数更新值情况下求取得到的误差函数二次导信息. 此二次导信息不仅估计了代价函数曲面在某点处的梯度(一阶信息),还估计了曲面曲率(二阶信息). 计算出曲率后,即可估计代价函数最小值的近似位置. 求取二次导信息后的卷积核更新公式为

$$\Delta\boldsymbol{f}_{jj'}^{l}(t-1)=\frac{\nabla\boldsymbol{f}_{jj'}^{l}(t-1)}{\nabla\boldsymbol{f}_{jj'}^{l}(t-2)-\nabla\boldsymbol{f}_{jj'}^{l}(t-1)}\Delta\boldsymbol{f}_{jj'}^{l}(t-2), \tag{15}$$

其中$\nabla\boldsymbol{f}_{jj'}^{l}(t-1)$为$t-1$时刻的梯度函数. 由QuickProp[19]理论可知,若卷积核更新公式中步长增长过快,则易导致收敛过程发散. 为此,引入一动量系数μ来克服上述缺陷,即如果

$$\Delta\boldsymbol{f}_{jj'}^{l}(t-1)>\mu\Delta\boldsymbol{f}_{jj'}^{l}(t-2), \tag{16}$$

则

$$\Delta\boldsymbol{f}_{jj}^{l}(t-1)=\mu\Delta\boldsymbol{f}_{jj'}^{l}(t-2). \tag{17}$$

根据上述分析,式(14)变换为

$$\boldsymbol{f}_{jj'}^{l}(t)=\boldsymbol{f}_{jj'}^{l}(t-1)-\frac{1}{N}\eta\sum_{i=1}^{N}\mu\frac{\partial J}{\partial \boldsymbol{f}_{jj'}^{l}(t-2)}. \tag{18}$$

1.4　计算复杂度分析

评价算法时间性能的主要指标是算法的时间复杂度. 设样本个数为N,采用直接计算法对DeDCNN进行计算复杂度分析,主要包括两部分: 1)两层堆叠反卷积网络对输入图像提取特征映射矩阵的计算复杂度为$O(N^2)\times T_1$,其中T_1为反卷积网络的学习迭代次数; 2)由文献[20]可知,深度卷积神经网络的计算复杂度为$O(N^3)\times T_2$,其中T_2为卷积网络的学习迭代次数. 因此,DeDCNN的整体计算复杂度为$O(N^2)\times T_1+O(N^3)\times T_2$.

2　实验结果及分析

2.1　实验数据集

采用MNIST、CIFAR-10和CIFAR-100数据集进行相关实验. MNIST包含手写数字0～9,每类数字均有6 000幅训练图像和1 000幅测试图像,共有70 000幅尺寸为28×28的单通道图像. CIFAR-10包含10类自然图像,每类自然图像均有5 000幅训练图像和1 000幅测试图像,共有60 000幅尺寸为32×32的RGB三通道图像. CIFAR-100类似于CIFAR-10有着同样的尺寸和格式,但CIFAR-100中包含了100类自然图像,每类图像均有500幅训练图像和100幅测试图像,共有60 000幅尺寸为32×32的RGB三通道图像,这使得分类任务难度大幅增加.

2.2　网络参数分析

使用MatConvNet工具箱对基于反卷积特征提取的深度卷积网络模型进行构建,在单块GPU型号为TeslaK40c上运行. 为提高GPU运算效率,采用CUDA和CUDNN对其架构进行优化. 对于附加动量系数的MB-SGD而言,动量系数μ的取值直接影响了代价函数的最小值位置,进而影响图像分类精度. 为此,讨论μ取值对图像分类精度影响. 表1给出了MNIST数据集在迭代45次和不同μ取值下的分类错误率. 由表1可见: 1)随着μ值增大,分类错误率逐渐降低; 2)$\mu=0.9$为临界值,错误率最低,超过此临界值后,错误率增大并进入欠拟合状态,因此后续实验中μ设置为0.9.

表1　不同μ取值情况下的分类错误率

	$\mu=0$	$\mu=0.1$	$\mu=0.2$	$\mu=0.3$	$\mu=0.4$	$\mu=0.5$	$\mu=0.6$
分类错误率/%	0.81	0.76	0.75	0.72	0.68	0.64	0.59
	$\mu=0.7$	$\mu=0.8$	$\mu=0.9$	$\mu=1$	$\mu=2$	$\mu=3$	$\mu=4$
分类错误率/%	0.48	0.45	0.44	11.94	90	90	90

2.3 对比实验

2.3.1 MNIST数据集

相较于CIFAR-10和CIFAR-100, MNIST数据集上的分类任务较为简单. 因此, 网络模型结构中avgpool3被取代, cccp6直接作为softmaxloss分类器的输入层. 小批次训练网络时, 批次大小应能够被训练集和测试集样本个数整除. 为此, 批次大小设置为100, 即训练100幅图像网络进行一次卷积核调整更新. 为避免过拟合, 在网络训练过程中, maxpool1和avgpool2后加入概率为0.5的dropout, 测试过程中则不采用. 权重衰减参数为0.000 1, 学习率为0.02, 迭代200次. 具体网络参数配置如表2所示.

表 2 网络参数配置(MNIST数据集)

名 称	输 入			卷积核个数	卷积核大小	步长	边界扩展	输 出		
	宽度	高度	深度					宽度	高度	深度
input								28	28	1
conv1	28	28	1	192	5	1	0	24	24	192
cccp1	28	28	192	160	1	1	0	24	24	160
cccp2	24	24	160	96	1	1	0	24	24	96
maxpool1	24	24	96		2	2	0	12	12	96
conv2	12	12	96	192	5	1	0	8	8	192
cccp3	8	8	192	192	1	1	0	8	8	192
cccp4	8	8	192	192	1	1	0	8	8	192
avgpool2	8	8	192		2	2	0	4	4	192
conv3	4	4	192	192	4	1	0	1	1	192
cccp5	1	1	192	192	1	1	0	1	1	192
cccp6	1	1	192	10	1	1	0	1	1	10
softmaxloss	1	1	10					1	1	10

在测试模型性能过程中并未对数据集内的训练数据进行扩展, 5种对比方法分别为微网(NIN)[18]、Maxout网络[21]、回归卷积神经网络RCNN[22]、深度监督网络(DSN)[23]. MNIST数据集上的分类错误率比较结果如表3所示. 可以看出: 在所有方法中, DeDCNN在测试样本上的分类错误率最低.

表 3 分类错误率比较(MNIST数据集)

方 法	分类错误率/%
NIN+Dropout[18]	0.47
Maxout Network+Dropout(k=2)[21]	0.45
RCNN-32+Dropout[22]	0.42
DSN+Dropout[23]	0.39
RCNN-96+Dropout[22]	0.31
DeDCNN+Dropout	0.31

考察附加动量系数对分类错误率的影响, 实验对比结果如图3所示. 经过约200次学习迭代后, 基于附加动量系数MB-SGD的DeDCNN分类错误率为0.31%, 而基于MB-SGD的DeDCNN获得了0.55 %的分类错误率. 因此, 采用附加动量系数的小批次随机梯度下降法可减少参数更新变化, 进一步避免梯度弥散问题.

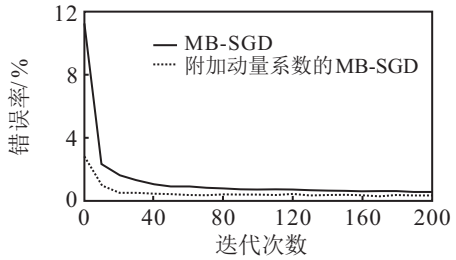

图 3 动量系数在MNIST中测试对比结果

2.3.2 CIFAR-10数据集

针对CIFAR-10数据集, 首先采用全局对比归一化和ZCA白化对数据集作预处理. 小批次训练网络时, 批次大小设置为100. 为避免过拟合, 在网络训练过程中, maxpool1和avgpool2后加入概率为0.5的dropout, 测试过程中则不采用. 学习率以启发式和退火方式设定, 具体为: 当分类准确率不再增加时, 当前学习率设置为前一时刻学习率的1/10. 经过4次退火步幅, 最终学习率设定为初始学习率的1/10 000. 权重衰减参数为0.000 5, 迭代240次. 具体网络参数配置如表4所示.

数据进行扩展, 比较结果如表5所示. 可以看出: 1) 针对未扩展CIFAR-10数据集, 在所有方法中DeDCNN的分类错误率最低; 2)DeDCNN在未扩展数据集上得到的分类错误率甚至比Maxout网络和NIN在扩展数据集上的要低; 3) 扩展数据集上的分类错误率要低于未扩展数据集, 如RCNN在未扩展和扩展数据集上分别获得了8.69 %和7.09 %的分类错误率. 因此, 可以推断: DeDCNN在扩展CIFAR-10数据集上的错误率会比未扩展的要低.

2.3.3 CIFAR-100数据集

对于CIFAR-100, 同样采用全局对比归一化和ZCA白化对数据集作预处理. 对于CIFAR-100未作超参微调, 而是采用与CIFAR-10相同的参数设置, 唯一不同的是网络结构中参数配置, 在最后的cccp6层输出100个特征图. 具体网络参数配置如表6所示.

表 4　网络参数配置(CIFAR-10数据集)

名称	输入			卷积核个数	卷积核大小	步长	边界扩展	输出		
	宽度	高度	深度					宽度	高度	深度
input								32	32	3
conv1	32	32	3	192	5	1	2	32	32	192
cccp1	32	32	192	160	1	1	0	32	32	160
cccp2	32	32	160	96	1	1	0	32	32	96
maxpool1	32	32	96		2	2	0	16	16	96
conv2	16	16	96	192	5	1	2	16	16	192
cccp3	16	16	192	192	1	1	0	16	16	192
cccp4	16	16	192	192	1	1	0	16	16	192
avgpool2	16	16	192		2	2	0	8	8	192
conv3	8	8	192	192	5	1	0	4	4	192
cccp5	4	4	192	192	1	1	0	4	4	192
cccp6	4	4	192	10	1	1	0	4	4	10
avgpool3	4	4	10		4	1	0	1	1	10
softmaxloss	1	1	10					1	1	10

表 5　分类错误率比较(CIFAR-10数据集)

方　法	分类错误率/%
Maxout Network(k=2)+Dropout[21]	11.68
NIN+Dropout[18]	10.41
DSN+Dropout[23]	9.69
RCNN-160+Dropout[22]	8.69
Maxout Network(k=2)+Dropout+Data Augmentation[21]	9.38
NIN+Dropout+Data Augmentation[18]	8.81
DSN+Dropout+Data Augmentation[23]	8.22
RCNN-160+Dropout+Data Augmentation[22]	7.09
DeDCNN+Dropout	8.34

在测试模型性能过程中并未对数据集内的训练数据进行扩展,比较结果如表7所示. 可以看出: 1) 由于CIFAR-100分类任务难度增加,在此数据上获得的分类错误率均较高,但DeDCNN仍取得了31.36 % 的错误率,优于所有对比方法; 2) 由于分类错误率是在未扩展数据下获得,通过扩展数据集中的训练数据可进一步降低分类错误率.

表 6　网络参数配置(CIFAR-100数据集)

名称	输入			卷积核个数	卷积核大小	步长	边界扩展	输出		
	宽度	高度	深度					宽度	高度	深度
input								32	32	3
conv1	32	32	3	192	5	1	2	32	32	192
cccp1	32	32	192	160	1	1	0	32	32	160
cccp2	32	32	160	96	1	1	0	32	32	96
maxpool1	32	32	96		3	2	0	15	15	96
conv2	15	15	96	192	5	1	2	15	15	192
cccp3	15	15	192	192	1	1	0	15	15	192
cccp4	15	15	192	192	1	1	0	15	15	192
avgpool2	15	15	192		3	2	0	7	7	192
conv3	7	7	192	192	3	1	1	7	7	192
cccp5	7	7	192	192	1	1	0	7	7	192
cccp6	7	7	192	100	1	1	0	7	7	100
avgpool3	7	7	100		7	1	0	1	1	100
softmaxloss	1	1	100					1	1	100

表 7　分类错误率比较(CIFAR-100数据集)

方　法	分类错误率/%
Maxout Network(k = 2)+Dropout[21]	38.57
NIN+Dropout[18]	35.68
DSN+Dropout[23]	34.57
RCNN-96+Dropout[22]	34.18
RCNN-128+Dropout[22]	32.59
RCNN-160+Dropout[22]	31.75
DeDCNN+Dropout	31.36

3　结　论

深度学习是目前机器学习领域最流行的研究方向之一,能够学习到数据的深层高级特征,具有比浅层算法更强大的非线性表示能力. DCNN通过对局部区域进行多种滤波方式来学习强鲁棒性的特征,从而能够较大程度地提高分类精度. 本文针对DCNN训练过程中的卷积核随机初始化和梯度下降学习法可能导致的陷入局部最优和梯度弥散问题,提出一种基于反卷积特征提取的深度卷积网络学习模型. 主

要思路为:采用反卷积网络自动学习图像的特征映射矩阵,并将其作为卷积核初始值参与DCNN模型的训练; 另外,采用附加动量系数的小批次随机梯度下降法对深度卷积网络微调以避免梯度弥散问题. 多个图像数据集上的实验结果表明,DeDCNN具有较高的分类精度.

参考文献(References)

[1] LeCun Y. Learning invariant feature hierarchies[J]. Lecture Notes in Computer Science, 2012, 7583(1): 496-505.

[2] Oquab M, Bottou L, Laptev I, et al. Learning and transferring mid-level image representations using convolutional neural networks[C]. Proc of the IEEE Conf on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2014: 1717-1724.

[3] Girshick R, Donahue J, Darrell T, et al. Rich feature hierarchies for accurate object detection and semantic segmentation[C]. Proc of the IEEE Conf on Computer Vision and Pattern Recognition. Piscataway: IEEE Computer Society, 2014: 580-587.

[4] Kabani A W, El-Sakka M R. Object detection and localization using deep convolutional networks with softmax activation and multi-class log loss[J]. Lecture Notes in Computer Science, 2016, 9730(1): 358-366.

[5] Karpathy A, Toderici G, Shetty S, et al. Large-scale video classification with convolutional neural networks[C]. Proc of the IEEE Conf on Computer Vision and Pattern Recognition. Piscataway: IEEE Computer Society, 2014: 1725-1732.

[6] Pfister T, Simonyan K, Charles J, et al. Deep convolutional neural networks for efficient pose estimation in gesture videos[J]. Lecture Notes in Computer Science, 2015, 9003(1): 538-552.

[7] Razavian A S, Azizpour H, Sullivan J, et al. CNN features off-the-shelf: An astounding baseline for recognition[C]. Proc of the IEEE Conf on Computer Vision and Pattern Recognition Workshops. Piscataway: IEEE Computer Society, 2014: 512-519.

[8] Masci J, Meier U, Ciresan D, et al. Stacked convolutional auto-encoders for hierarchical feature extraction[J]. Lecture Notes in Computer Science, 2011, 6791(1): 52-59.

[9] Zeiler M D, Fergus R. Visualizing and understanding convolutional networks[J]. Lecture Notes in Computer Science. Springer: Springer-Verlag, 2014, 8689(1): 818-833.

[10] LeCun Y, Boser B, Denker J S, et al. Backpropagation applied to handwritten zip code recognition[J]. Neural Computation, 1989, 1(4): 541-551.

[11] Zeiler M D, Krishnan D, Taylor G W, et al. Deconvolutional networks[C]. Proc of the IEEE Computer Society Conf on Computer Vision and Pattern Recognition. Piscataway: IEEE Computer Society, 2010: 2528-2535.

[12] Liu J, Liu B Y, Lu H Q. Detection guided deconvolutional network for hierarchical feature learning[J]. Pattern Recognition, 2015, 48(8): 2645-2655.

[13] He K M, Zhang X Y, Ren S Q, et al. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification[C]. Proc of the IEEE Int Conf on Computer Vision. Piscataway: IEEE, 2015: 1026-1034.

[14] Glorot X, Bordes A, Bengio Y. Deep sparse rectifier neural networks[C]. Proc of the 14th Int Conf on Artificial Intelligence and Statistics. Piscataway: IEEE Computer Society, 2011: 315-323.

[15] Ioffe S, Szegedy C. Batch normalization: Accelerating deep network training by reducing internal covariate shift[C]. Proc of the 32nd Int Conf on Machine Learning. Lille: International Machine Learning Society, 2015: 448-456.

[16] Qian Q, Jin R, Yi J F, et al. Efficient distance metric learning by adaptive sampling and mini-batch stochastic gradient descent[J]. Machine Learning, 2014, 99(3): 353-372.

[17] Wang Y, Yang J, Yin W, et al. A new alternating minimization algorithm for total variation image reconstruction[J]. Siam J on Imaging Sciences, 2008, 1(3): 248-272.

[18] Chen Y, Yang X N, Zhong B N, et al. Network in network based weakly supervised learning for visual tracking[J]. J of Visual Communication and Image Representation, 2016, 3(C): 3-13.

[19] Fahlman S E. Faster-learning variations on back-propagation: An empirical study[Z]. Burlington: Connectionist Models Summer School Morgan Kaufmann, 1988.

[20] Zhao W Z, Du S H. Spectral-spatial feature extraction for hyperspectral image classification: Adimension reduction and deep learning approach[J]. IEEE Trans on Geoscience and Remote Sensing, 2016, 54(8): 4544-4554.

[21] Goodfellow I J, Warde-Farley D, Mirza M, et al. Maxout networks[C]. Proc of the 30th Int Conf on Machine Learning. Princeton: Int Machine Learning Society, 2013: 2356-2364.

[22] Liang M, Hu X L. Recurrent convolutional neural network for object recognition[C]. Proc of the IEEE Computer Society Conf on Computer Vision and Pattern Recognition. Piscataway: IEEE Computer Society, 2015: 3367-3375.

[23] Lee C Y, Xie S N, Gallagher P W, et al. Deeply supervised nets[C]. Proc of the 18th Int Conf on Artificial Intelligence and Statistics. Brookline: Microtome Publishing, 2015: 562-570.

（责任编辑：郑晓蕾）