



(12)发明专利申请

(10)申请公布号 CN 107341518 A

(43)申请公布日 2017.11.10

(21)申请号 201710553052.5

(22)申请日 2017.07.07

(71)申请人 东华理工大学

地址 330013 江西省南昌市经开区广兰大道418号

(72)发明人 王蕾 何月顺 王坤 蒋年德
钟国韵 蔡友林

(74)专利代理机构 北京国坤专利代理事务所
(普通合伙) 11491

代理人 郭伟红

(51)Int.Cl.

G06K 9/62(2006.01)

G06N 3/08(2006.01)

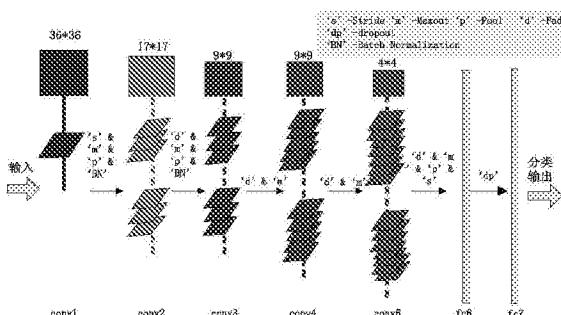
权利要求书4页 说明书16页 附图11页

(54)发明名称

一种基于卷积神经网络的图像分类方法

(57)摘要

本发明公开了一种基于卷积神经网络的图像分类方法，所述方法包括如下步骤：深层卷积神经网络的构建，深层卷积神经网络的改进，深层卷积神经网络的训练及测试，优化网络参数。本发明的图像分类方法围绕卷积神经网络的网络结构和多参数分别进行了改进和优化，能够有效地提高深层卷积神经网络模型的识别率，提高图像分类的准确性。



1. 一种基于卷积神经网络的图像分类方法,其特征在于,所述方法包括如下步骤:深层卷积神经网络的构建,深层卷积神经网络的改进,深层卷积神经网络的训练及测试,优化网络参数;

深层卷积神经网络框架的构建:在ImageNet-2010网络基础上,通过逐层分析网络层的方法设计了一个具有九层网络的深层卷积神经网络,其包括图像输入层、conv1卷积层、conv2卷积层、conv3卷积层、conv4卷积层、conv5卷积层、fc6全链接层、fc7全链接层和输出分类层;

深层卷积神经网络的改进:由随机下采样方法、下采样区域有重叠的采样方式的方法、全链接层使用逐层训练的方法、maxout激活函数的方法四个方面对深层卷积神经网络进行改进;

深层卷积神经网络的训练及测试:深层卷积神经网络的训练流程有前向与反向传播两个步骤,利用逐层训练的方法进行深层卷积神经网络的训练;

优化网络参数:通过对batch、dropout、momentum动量相关网络参数进行优化,以减轻训练网络时出现的过拟合现象。

2. 根据权利要求1所述的一种基于卷积神经网络的图像分类方法,其特征在于,在深层卷积神经网络的构建时,其中对conv1卷积层的设计方法如下:输入原始图像进行卷积处理,经过卷积处理之后得到了一个特征图,再通过maxout激活函数得到特征映射图;其次,再通过池化降采样处理,该池化处理为最大池化处理方法,设置池化层卷积核的大小及滑动步长,经过降采样处理后,输出特征映射图;最后,在输入到conv2卷积层之前做一个Batch Normalization即BN预处理;

其中对conv2卷积层的设计方法如下:以conv1的输出作为conv2的输入,首先经过卷积处理,所以为了防止特征图过快的变小,本层的卷积层增加边缘补偿处理,经过卷积处理之后得到特征图,然后同样采用maxout激活函数得到特征映射图;其次,同样的采用最大池化处理方法,设置池化层卷积核的大小及滑动步长,输出特征映射图;最后,在输入到conv3卷积层之前做BN处理,输出特征图;

其中对conv3卷积层的设计方法如下:将conv2的输出输入到conv3中,首先经过卷积处理,本层的卷积层增加边缘补偿处理,经过卷积处理之后得到了一个特征图,最后,同样采用maxout激活函数得到特征映射图,再将激活函数处理后的输出直接输入到了下一个卷积层;

其中对conv4卷积层的设计方法如下:将conv3的输出输入到conv4中,首先经过卷积处理,本层的卷积层增加边缘补偿处理,经过卷积处理之后得到了一个特征图,最后,同样采用maxout激活函数得到特征映射图,再将激活函数处理后的输出直接输入到了下一个卷积层;

其中对conv4卷积层的设计方法如下:将conv4的输出输入到conv5中,首先经过卷积处理,本层的卷积层增加边缘补偿处理,经过卷积处理之后得到了一个特征图,然后,同样采用maxout激活函数得到特征映射图;最后,采用最大池化处理方法,设置池化层卷积核的大小及滑动步长,输出特征映射图,本层不进行BN预处理;

其中对fc6全链接层的设计方法如下:将conv5的输出输入到fc6全链接层中,输入特征图,设置fc6全链接层的神经元数,然后,采用maxout激活函数,输出神经数,最后,采用

dropout处理方式进行输出；

其中对fc7全链接层的设计方法如下：将fc6的输出输入到fc7全链接层中，该层结构与fc6基本相同，首先，设置fc6全链接层的神经元数，然后采用maxout激活函数，输出神经数，最后，同样采用dropout处理方式进行输出；

其中对输出分类层的设计方法如下：本层分类器选用softmax分类器，根据训练样本的不同来确定神经节点的个数，对于cifar-10图像数据库来说，其节点数设置为10，对于cifar-100图像数据库来说，其神经节点数要设置为100。

3. 根据权利要求1所述的一种基于卷积神经网络的图像分类方法，其特征在于，在深层卷积神经网络的改进时，从四个方面对深层卷积神经网络作出改进，分别是随机下采样方法、下采样区域有重叠的采样方式的方法、全链接层使用逐层训练的方法、maxout激活函数的方法；

(1) 采用随机下采样方法

首先，对图像区域里的元素值进行求和，然后把区域里的每个元素与总和相除以便得出采样区域所对应的概率，接着根据采样区域概率值用0-1分布式采样进行采样并获得采样结果；

(2) 采用下采样区域有重叠的采样方式的方法

在tensorflow里把相应cifar文件中的layer类型配置里面的参数pool:Stochastic改为pool:MAX就可以实现随机下采样的方式。

(3) 采用全连接层使用逐层训练的方法

tensorflow会支持自动编码网络，在全链接层可使用自动编码网络的训练方式进行改进。

(4) 采用maxout激活函数的方法

使用maxout激活函数来代替ImageNet-2010所采用的relu函数，其数学表达式如下：

$$h_i(x) = \max(z_{ij}) \quad j \in [1, k]$$

$$z_{ij} = x^T W_{ij} + b_{ij}, \quad W \in \mathbb{R}^{d \times m \times k}$$

表达式中的d表示输入层的节点个数，k为每一个隐含层节点所对应的k个“隐隐含层”节点，m代表隐含层的节点个数。

4. 根据权利要求1所述的一种基于卷积神经网络的图像分类方法，其特征在于，在进行深层卷积神经网络的训练及测试时，训练阶段的算法流程如下：

输入：训练集

误差阈值 ϵ

最大迭代次数num

输出：网络权重

步骤：

初始化：网络权重： $W \leftarrow N(0, 1)$ ，偏置值： $b \leftarrow \text{const}$

Step 1 Do

Step 2 从训练集中随机选择一个Batch，输入到卷积神经网络中；

Step 3 训练样本前向传播，进行conv卷积计算；

Step 4 进行Max pooling特征提取运算；

Step 5进行Batch Normalization归一化运算；
Step 6输入到下一个卷积层，直至输入到全连接层
Step 7全连接层提取特征，输入到softmax函数进行分类，选取输出概率最大的作为分类结果。

Step 8 if分类误差小于误差阈值or训练次数==最大迭代次数num

Step 9 brea;

Step10 els

Step11计算误差，进行误差反向传播

Step12网络参数更新；

Step13 Until所有的Batch都训练完毕；

其测试阶段的算法流程如下：

输入：测试集

输出：分类结果

步骤：

初始化：网络权重： $W, b \leftarrow$ 训练好的网络值

Step 1 Do

Step 2测试样本前向传播，进行conv卷积计算；

Step 3进行Max pooling特征提取运算；

Step 4进行Batch Normalization归一化运算；

Step 5输入到下一个卷积层，直至输入到全连接层

Step6全连接层提取特征，输入到softmax函数进行分类，选取输出概率最大的作为分类结果。

Step7判断比较标签与分类结果，并统计分类结果；

Step8 Until所有的Batch全部都测试完毕。

5. 根据权利要求1所述的一种基于卷积神经网络的图像分类方法，其特征在于，优化网络参数包括

可视化特征的提取，输入图像历经卷积神经网络卷积层conv1的处理之后，得到输出特征映射，通过输入图像历经卷积神经网络卷积层conv1的处理之后，得到输出特征映射，通过卷积层conv1之后得到可视化图像，卷积层conv1所学习的大量信息体现在输入图形的边缘轮廓信息上；如要展示卷积层conv2的可视化结果，需要把它的高维卷积核拆成低维的卷积核，再进行可视化处理；

还包括，

设定batch的值，采取不同的batch小数据集进行训练，在经历多次试验后可以得到batch的范围，如要停止训练，要当损失函数的值低于预定的阈值，测试五次后取平均值最大程度避免偶然事件突发；

还包括，

dropout的设计，深度学习网络在每次batch样本输入时，权重必须进行保留，对于一个含有N个节点的深层卷积神经网络，当加入dropout处理之后，该网络就可以看成是 2^n 个模型的组合了，深层卷积神经网络加入dropout后，其训练层面和测试层面应做出相应的改

进；

还包括，

数据集扩增，为训练出更好的网络模型，需要充足的训练数据并采用更深层的网络，对于扩充数据集，既简单又有效的方法就是对原始数据进行扩充，通过对原始数据进行改动，可以得到更多的数据。对于图片数据而言，扩充其数据最简单的方法就是调整图片的角度。

一种基于卷积神经网络的图像分类方法

技术领域

[0001] 本发明涉及图像分类方法技术领域,特别涉及一种基于卷积神经网络的图像分类方法。

背景技术

[0002] 近几年以来,随着科学计算机网络及人工智能领域的发展,图形图像数据量逐渐增多,于是,如何从大量的自然图像中快速提取到视觉特征已经成了机器智能学习中的热点研究课题,进而对自然图像的分类必然成为获取自然图像信息的研究重点。

[0003] 卷积神经网络是深度学习在图像处理方面的一个重要应用,它相比于其它机器学习算法如SVM等,其优点是能够直接对图像像素进行卷积并提取特征,也能够利用海量的图像数据将网络参数训练充分,以达到更好的分类效果。一般浅层卷积神经网络的层数是小于或等于5层,通过研究发现浅层卷积神经网络的结构相对比较简单而且清晰,它对简单图像数据有着比较好的分类正确率,然而其对较为复杂的图像数据如cifar-10、cifar-100等的分类效果并不是那么理想。

发明内容

[0004] 本发明的主要目的在于提供一种基于卷积神经网络的图像分类方法,旨在解决现有技术对复杂的图像数据的分类效果不理想的问题。

[0005] 为实现上述目的,本发明采取的技术方案为:

[0006] 一种基于卷积神经网络的图像分类方法,所述方法包括如下步骤:深层卷积神经网络的构建,深层卷积神经网络的改进,深层卷积神经网络的训练及测试,优化网络参数;

[0007] 深层卷积神经网络框架的构建:在ImageNet-2010网络基础上,通过逐层分析网络层的方法设计了一个具有九层网络的深层卷积神经网络,其包括图像输入层、conv1卷积层、conv2卷积层、conv3卷积层、conv4卷积层、conv5卷积层、fc6全链接层、fc7全链接层和输出分类层;

[0008] 深层卷积神经网络的改进:由随机下采样方法、下采样区域有重叠的采样方式的方法、全链接层使用逐层训练的方法、maxout激活函数的方法四个方面对深层卷积神经网络进行改进;

[0009] 深层卷积神经网络的训练及测试:深层卷积神经网络的训练流程有前向与反向传播两个步骤,利用逐层训练的方法进行深层卷积神经网络的训练;

[0010] 优化网络参数:通过对batch、dropout、momentum动量相关网络参数进行优化,以减轻训练网络时出现的过拟合现象。

[0011] 进一步的,在深层卷积神经网络的构建时,其中对conv1卷积层的设计方法如下:输入原始图像进行卷积处理,经过卷积处理之后得到了一个特征图,再通过maxout激活函数得到特征映射图;其次,再通过池化降采样处理,该池化处理为最大池化处理方法,设置池化层卷积核的大小及滑动步长,经过降采样处理后,输出特征映射图;最后,在输入到

conv2卷积层之前做一个Batch Normalization即BN预处理；

[0012] 其中对conv2卷积层的设计方法如下：以conv1的输出作为conv2的输入，首先经过卷积处理，所以为了防止特征图过快的变小，本层的卷积层增加边缘补偿处理，经过卷积处理之后得到特征图，然后同样采用maxout激活函数得到特征映射图；其次，同样的采用最大池化处理方法，设置池化层卷积核的大小及滑动步长，输出特征映射图；最后，在输入到conv3卷积层之前做BN处理，输出特征图；

[0013] 其中对conv3卷积层的设计方法如下：将conv2的输出输入到conv3中，首先经过卷积处理，本层的卷积层增加边缘补偿处理，经过卷积处理之后得到了一个特征图，最后，同样采用maxout激活函数得到特征映射图，再将激活函数处理后的输出直接输入到了下一个卷积层；

[0014] 其中对conv4卷积层的设计方法如下：将conv3的输出输入到conv4中，首先经过卷积处理，本层的卷积层增加边缘补偿处理，经过卷积处理之后得到了一个特征图，最后，同样采用maxout激活函数得到特征映射图，再将激活函数处理后的输出直接输入到了下一个卷积层；

[0015] 其中对conv4卷积层的设计方法如下：将conv4的输出输入到conv5中，首先经过卷积处理，本层的卷积层增加边缘补偿处理，经过卷积处理之后得到了一个特征图，然后，同样采用maxout激活函数得到特征映射图；最后，采用最大池化处理方法，设置池化层卷积核的大小及滑动步长，输出特征映射图，本层不进行BN预处理；

[0016] 其中对fc6全链接层的设计方法如下：将conv5的输出输入到fc6全链接层中，输入特征图，设置fc6全链接层的神经元数，然后，采用maxout激活函数，输出神经数，最后，采用dropout处理方式进行输出；

[0017] 其中对fc7全链接层的设计方法如下：将fc6的输出输入到fc7全链接层中，该层结构与fc6基本相同，首先，设置fc6全链接层的神经元数，然后采用maxout激活函数，输出神经数，最后，同样采用dropout处理方式进行输出；

[0018] 其中对输出分类层的设计方法如下：本层分类器选用softmax分类器，根据训练样本的不同来确定神经节点的个数，对于cifar-10图像数据库来说，其节点数设置为10，对于cifar-100图像数据库来说，其神经节点数要设置为100。

[0019] 进一步的，在深层卷积神经网络的改进时，从四个方面对深层卷积神经网络作出改进，分别是随机下采样方法、下采样区域有重叠的采样方式的方法、全链接层使用逐层训练的方法、maxout激活函数的方法；

[0020] (1) 采用随机下采样方法

[0021] 首先，对图像区域里的元素值进行求和，然后把区域里的每个元素与总和相除以便得出采样区域所对应的概率，接着根据采样区域概率值用0-1分布式采样进行采样并获得采样结果；

[0022] (2) 采用下采样区域有重叠的采样方式的方法

[0023] 在tensorflow里把相应cifar文件中的layer类型配置里面的参数pool：Stochastic改为pool：MAX就可以实现随机下采样的方式。

[0024] (3) 采用全连接层使用逐层训练的方法

[0025] tensorflow会支持自动编码网络，在全链接层可使用自动编码网络的训练方式进

行改进。

[0026] (4) 采用maxout激活函数的方法

[0027] 使用maxout激活函数来代替ImageNet-2010所采用的relu函数,其数学表达式如下:

[0028] $h_i(x) = \max(z_{i,j}) , j \in [1, k]$

[0029] $z_{ij} = x^T W_{ij} + b_{ij}, W \in R^{d \times m \times k}$

[0030] 表达式中的d表示输入层的节点个数,k为每一个隐含层节点所对应的k个“隐隐含层”节点,m代表隐含层的节点个数。

[0031] 进一步的,在进行深层卷积神经网络的训练及测试时,训练阶段的算法流程如下:

[0032] 输入:训练集

[0033] 误差阈值 ϵ

[0034] 最大迭代次数num

[0035] 输出:网络权重

[0036] 步骤:

[0037] 初始化:网络权重: $W \leftarrow N(0, 1)$,偏置值: $b \leftarrow \text{const}$

[0038] Step 1 Do

[0039] Step 2 从训练集中随机选择一个Batch,输入到卷积神经网络中;

[0040] Step 3 训练样本前向传播,进行conv卷积计算;

[0041] Step 4 进行Max pooling特征提取运算;

[0042] Step 5 进行Batch Normalization归一化运算;

[0043] Step 6 输入到下一个卷积层,直至输入到全连接层

[0044] Step 7 全连接层提取特征,输入到softmax函数进行分类,选取输出概率最大的作为分类结果。

[0045] Step 8 if分类误差小于误差阈值or训练次数 == 最大迭代次数num

[0046] Step 9 brea;

[0047] Step10 els

[0048] Step11 计算误差,进行误差反向传播

[0049] Step12 网络参数更新;

[0050] Step13 Until所有的Batch都训练完毕;

[0051] 其测试阶段的算法流程如下:

[0052] 输入:测试集

[0053] 输出:分类结果

[0054] 步骤:

[0055] 初始化:网络权重: $W, b \leftarrow$ 训练好的网络值

[0056] Step 1 Do

[0057] Step 2 测试样本前向传播,进行conv卷积计算;

[0058] Step 3 进行Max pooling特征提取运算;

[0059] Step 4 进行Batch Normalization归一化运算;

[0060] Step 5 输入到下一个卷积层,直至输入到全连接层

- [0061] Step6 全连接层提取特征,输入到softmax函数进行分类,选取输出概率最大的作为分类结果。
- [0062] Step7 判断比较标签与分类结果,并统计分类结果;
- [0063] Step8 Until所有的Batch全部都测试完毕。
- [0064] 进一步的,优化网络参数包括
- [0065] 可视化特征的提取,输入图像历经卷积神经网络卷积层conv1的处理之后,得到输出特征映射,通过输入图像历经卷积神经网络卷积层conv1的处理之后,得到输出特征映射,通过卷积层conv1之后得到可视化图像,卷积层conv1所学习的大量信息体现在输入图形的边缘轮廓信息上;如要展示卷积层conv2的可视化结果,需要把它的高维卷积核拆成低维的卷积核,再进行可视化处理;
- [0066] 还包括,
- [0067] 设定batch的值,采取不同的batch小数据集进行训练,在经历多次试验后可以得到batch的范围,如要停止训练,要当损失函数的值低于预定的阈值,测试五次后取平均值最大程度避免偶然事件突发;
- [0068] 还包括,
- [0069] dropout的设计,深度学习网络在每次batch样本输入时,权重必须进行保留,对于一个含有N个节点的深层卷积神经网络,当加入dropout处理之后,该网络就可以看成是 2^n 个模型的组合了,深层卷积神经网络加入dropout后,其训练层面和测试层面应做出相应的改进;
- [0070] 还包括,
- [0071] 数据集扩增,为训练出更好的网络模型,需要充足的训练数据并采用更深层的网络,对于扩充数据集,既简单又有效的方法就是对原始数据进行扩充,通过对原始数据进行改动,可以得到更多的数据。对于图片数据而言,扩充其数据最简单的方法就是调整图片的角度。
- [0072] 本发明具有如下有益效果:
- [0073] 本发明提供了一种基于卷积神经网络的图像分类方法,该方法围绕卷积神经网络的网络结构和多参数分别进行了改进和优化,能够有效地提高深层卷积神经网络模型的识别率,提高图像分类的准确性。

附图说明

- [0074] 图1为本发明实施例所构建的深层卷积神经网络结构图;
- [0075] 图2为conv1卷积层流程设计图;
- [0076] 图3为conv2卷积层的流程设计图;
- [0077] 图4为conv3卷积层流程设计图;
- [0078] 图5为conv4卷积层流程设计图;
- [0079] 图6为conv5卷积层流程设计图;
- [0080] 图7为fc6全链接层流程设计图;
- [0081] 图8为fc7全链接层流程设计图;
- [0082] 图9为输出分类层流程设计图;

- [0083] 图10为 3×3 大小采样区域图；
- [0084] 图11为采样区域对应元素概率图；
- [0085] 图12为随机采样结果图；
- [0086] 图13为网络过拟合现象图；
- [0087] 图14为输入的原始图像；
- [0088] 图15是由conv1卷积层输出之后的特征映射图；
- [0089] 图16为conv1可视化结果图；
- [0090] 图17为conv2卷积权值的可视化结果图；
- [0091] 图18为conv2的输出特征映射结果图；
- [0092] 图19为conv3输出特征映射结果图；
- [0093] 图20为conv4输出特征映射结果图；
- [0094] 图21为conv5的输出特征映射结果图；
- [0095] 图22为经过pool5池化后的输出特征映射结果图；
- [0096] 图23为fc6全链接层输出结果展示图；
- [0097] 图24为fc6全链接层的输出值直方图结果展示图；
- [0098] 图25为经过fc7全链接层后输出结果展示图；
- [0099] 图26为fc7全链接层的输出值直方图结果展示图；
- [0100] 图27为深层卷积神经网络输出层结果展示图；
- [0101] 图28Dropout过程图；
- [0102] 图29为神经单元加入概率过程图；
- [0103] 图30为测试加入概率p的过程图。

具体实施方式

[0104] 为使本发明实现的技术手段、创作特征、达成目的与功能易于明白了解，下面结合具体实施方式，进一步阐述本发明。

[0105] 本发明实施例提供了一种基于卷积神经网络的图像分类方法，包括如下步骤：深层卷积神经网络的构建，深层卷积神经网络的改进，深层卷积神经网络的训练及测试，优化网络参数；

[0106] 1. 深层卷积神经网络的构建：在ImageNet-2010网络基础上，通过逐层分析网络层的方法设计了一个具有九层网络的深层卷积神经网络，如图1所示，其包括图像输入层、conv1卷积层、conv2卷积层、conv3卷积层、conv4卷积层、conv5卷积层、fc6全链接层、fc7全链接层和输出分类层；

[0107] 从图1可以清晰的看出该深层卷积神经网络的模型结构，其网络参数如下表1所示：

[0108] 表1网络模型参数

[0109]

步骤	参数
输入	原图像大小
	原图像通道数
卷积层 1	GPU 包含卷积核个数
	单个卷积核大小
池化层 1	池化参数大小
卷积层 2	GPU 包含卷积核个数
	单个卷积核大小
池化层 2	池化参数大小
卷积层 3	GPU 包含卷积核个数
	单个卷积核大小
卷积层 4	GPU 包含卷积核个数
	单个卷积核大小
卷积层 5	GPU 包含卷积核个数
	单个卷积核大小
池化层 5	池化参数大小
全连接层 1	神经元个数
全连接层 2	神经元个数
分类层	分类类别数

[0110] 1.1 conv1卷积层设计

[0111] conv1卷积层的流程设计如图2所示。首先,输入原始图像进行卷积处理,其中该图像的大小为36像素*36像素的三通道彩色图,该层的卷积核大小是3*3,卷积核有64个且将卷积核的滑动步长设置为1,经过以上参数的卷积处理之后得到了一个大小为34像素*34像素的特征图,由 $36-3+1=34$ 得到,且其维数为64。然后,再通过maxout激活函数^{[34][51]}得到的特征映射图的大小仍然是34*34,维数是64,maxout激活函数是一种分段的线性函数,它反应的是整体的而不是单一的输入与输出的非线性映关系,其能有效的增强网络的表达能力。其次,再通过池化降采样处理,该池化处理为最大池化处理方法,池化层卷积核的大小设置为2*2,滑动步长设置为2,经过降采样处理后,其输出的特征映射图的大小为17像素*17像素,且维数为64。最后,在输入到conv2卷积层之前做一个Batch Normalization即BN预处理。BN预处理不仅能够有效的解决训练过程中,层与层之间的数据分布发生改变的问题,而且它能随机化样本数据,有效的避免了每批训练时某一个样本总是被选到的概率。

[0112] 1.2 conv2卷积层设计

[0113] conv2卷积层的流程设计如图3所示。conv1的输出就是conv2的输入。首先,也是经过卷积处理。此层的卷积核数为128个,卷积核大小也是3*3,由于本文设计的网络结构为深层卷积神经网络,卷积层比较多,所以为了防止特征图过快的变小^[35],本层的卷积层增加

了边缘补偿处理,补偿值为2个像素。经过以上参数的卷积处理之后得到了一个大小为19像素*19像素的特征图,其维数为128。然后,同样采用maxout激活函数得到的特征映射图的大小是19*19,维数是128。其次,同样的采用最大池化处理方法,池化层卷积核的大小设置为3*3,滑动步长设置为2,其输出的特征映射图的大小为9像素*9像素,且维数为128。最后,在输入到conv3卷积层之前做BN处理,输出的特征图大小为9像素*9像素,维数为128。

[0114] 1.3 conv3卷积层设计

[0115] conv3卷积层的流程设计如图4所示。将conv2的输出输入到conv3中。首先,也是经过卷积处理。此层的卷积核数为256个,卷积核大小也是3*3,其中该层的卷积核数是conv2层卷积核数的两倍,与conv2相同,本层也加了边缘像素补偿,其补偿大小设定为1像素,经过以上参数的卷积处理之后得到了一个大小为9像素*9像素的特征图,其维数为256。可以看出其特征图大小基本保持了不变,然而特征向量增多了,也就是说得到了更多的图像信息^[36]。最后,同样采用maxout激活函数得到的特征映射图的大小是9*9,维数是256。由于要最大程度的保留图像的边缘特征,所以本层没有进行采样处理,将激活函数处理后的输出直接输入到了下一个卷积层。

[0116] 1.4 conv4设计

[0117] conv4卷积层的流程设计如图5所示。将conv3的输出输入到conv4中。首先,也是经过卷积处理。此层的卷积核数与conv3相同也为256个,卷积核大小也是3*3,补偿大小也设定为1像素,同样地,经过以上参数的卷积处理之后得到了一个大小为9像素*9像素的特征图,其维数为256。最后,同样采用maxout激活函数得到的特征映射图的大小是9*9,维数是256。该层设计的处理方式与参数基本与conv3相同。

[0118] 1.5 conv5设计

[0119] conv5卷积层的流程设计如图6所示。conv4的输出就是conv5的输入。首先,也是经过卷积处理。此层的卷积核数为128个,卷积核大小也是3*3,该层也增加了边缘补偿处理^[37],补偿值为2个像素。经过以上参数的卷积处理之后得到了一个大小为9像素*9像素的特征图,其维数为128。然后,同样采用maxout激活函数得到的特征映射图的大小是9*9,维数是128。最后,采用最大池化处理方法,池化层卷积核的大小设置为3*3,滑动步长设置为2,其输出的特征映射图的大小为4像素*4像素,且维数为128。本层没有进行BN预处理。

[0120] 1.6 fc6全链接层设计

[0121] fc6的流程设计如图7所示。conv5的输出就是fc6全链接层的输入,输入特征图的大小为4像素*4像素,维数为128。首先,设置fc6全链接层的神经元数为4096,然后,采用maxout激活函数,输出神经数为4096,最后,采用dropout处理方式,这种处理方式的好处在于既能够有效的防止过拟合,也能够大大降低训练时间。

[0122] 1.7 fc7全链接层设计

[0123] fc7的流程设计如图8所示。fc6的输出就是fc7全链接层的输入,该层结构与fc6基本相同。首先,设置fc6全链接层的神经元数为4096,然后,采用maxout激活函数,输出神经数为4096,最后,同样采用dropout处理方式。

[0124] 1.8输出分类层设计

[0125] 输出分类层流程设计如图9所示。本层分类器选用的是softmax分类器,根据训练样本的不同来确定神经节点的个数。对于cifar-10图像数据库来说,其节点数设置为10,

对于cifar-100图像数据库来说,其神经节点数要设置为100。

[0126] 2卷积层及全连接层的改进

[0127] 由四个方面对深层卷积神经网络做出改进,分别是随机下采样、下采样区域有重叠的采样方式、全连接层使用逐层训练的方法、maxout激活函数。

[0128] (1)采用随机下采样方法

[0129] 如图10所示,这是大小是 3×3 的图像区域,基于此来简要解释采样的工作原理。首先,先对此区域里的元素值进行求和可以得到 $\text{sum} = 0 + 1.1 + 0 + 1.0 + 2.0 + 0.9 + 2.5 + 1.0 + 1.5 = 10$,然后把区域里的每个元素与总和相除以便得出采样区域所对应的概率,接着根据采样区域概率值用0-1分布式采样进行采样并获得采样结果。从图12可以看出所得到的值为2.5。由随机采样这种采样方式可以知道,概率区域如果越大,被采样的概率也就越大。

[0130] ImageNet-2010是采用最大值下采样方式进行。而本节所设计的深度卷积神经网络就采取随机下采样,然后对特征映射中的元素根据它的概率值的大小进行随机选择,如果值越大,该元素被选中的概率也就越大。随机下采样的方式相对于ImageNet-2010的最大值下采样,会更加符合生物神经元的处理方式,避免了最大值百分之百被选中,只不过是被选中的概率会更大一些。

[0131] (2)采用下采样区域有重叠的采样方式的方法

[0132] 有重叠采样方式的采纳,可以更进一步地加强随机下采样中值较大的元素可以被选中的概率会更大。在tensorflow里把相应cifar文件中的layer类型配置里面的参数pool:Stochastic改为pool:MAX就可以实现随机下采样的方式。

[0133] (3)采用全连接层使用逐层训练的方法

[0134] 卷积神经网络初始化常常会要求初始化的数据会服从某一类分布,这样能避免网络对称性的发生,然而这种类型的初始化方式会使得网络参数的初始化起点处于不确定性。逐层训练方法会克服概率不确定性的缺点,这样会使得网络初始化参数处于一个确定性高的起点。tensorflow会支持自动编码网络,所以全连接层可以使用自动编码网络的训练方式。

[0135] (4)采用maxout激活函数的方法

[0136] sigmoid函数或者tanh函数是传统的卷积神经网络所采用的神经元激活函数。我们使用maxout激活函数来代替ImageNet-2010所采用的relu函数,其数学表达式如下:

[0137] $h_i(x) = \max(z_{ij}), j \in [1, k] \quad (1)$

[0138] $z_{ij} = x^T W_{ij} + b_{ij}, W \in R^{d \times m \times k} \quad (2)$

[0139] 表达式中的d表示输入层的节点个数,k为每一个隐含层节点所对应的k个“隐隐含层”节点,m代表隐含层的节点个数.maxout激活函数是让这k个“隐隐含层”节点中输出值最大的那个值为每个隐含节点的值。利用maxout函数强大的拟合能力,会拟合出我们所需要的任意凸函数。

[0140] maxout函数与relu函数的实现原理十分类似,可以弥补目前版本tensorflow没有实现maxout函数的缺陷。先把cifar文件中relu层名称和类型更换为maxout的名称和类型。再把在tensorflow源代码\src\cifar-tensorflow\layers目录下的relu_layer.py重新命名为maxout_layer.py,并把文件里面的内容做出对应修改。

[0141] 3深层卷积神经网络的训练及测试算法

[0142] 深层卷积神经网络的训练流程有前向与反向传播两个步骤。全链接层卷积神经网络有别于传统的卷积神经网络是它利用逐层训练的方法。

[0143] 训练阶段的算法流程如下所示：

[0144] 输入：训练集

[0145] 误差阈值 ϵ

[0146] 最大迭代次数num

[0147] 输出：网络权重

[0148] 步骤：

[0149] 初始化：网络权重： $W \leftarrow N(0, 1)$ ，偏置值： $b \leftarrow \text{const}$

[0150] Step 1. Do

[0151] Step 2 从训练集中随机选择一个Batch，输入到卷积神经网络中；

[0152] Step 3 训练样本前向传播，进行conv卷积计算；

[0153] Step 4 进行Max pooling特征提取运算；

[0154] Step 5 进行Batch Normalization归一化运算；

[0155] Step 6 输入到下一个卷积层，直至输入到全连接层

[0156] Step 7 全连接层提取特征，输入到softmax函数进行分类，选取输出概率最大的

[0157] 作为分类结果。

[0158] Step 8 if 分类误差小于误差阈值 or 训练次数 == 最大迭代次数num

[0159] Step 9 break；

[0160] Step 10 else；

[0161] Step 11 计算误差，进行误差反向传播；

[0162] Step 12 网络参数更新；

[0163] Step 13 Until所有的Batch都训练完毕

[0164] 其测试阶段的算法流程如下：

[0165] 输入：测试集

[0166] 输出：分类结果

[0167] 步骤：

[0168] 初始化：网络权重： $W, b \leftarrow$ 训练好的网络值

[0169] Step 1 Do

[0170] Step 2 测试样本前向传播，进行conv卷积计算；

[0171] Step 3 进行Max pooling特征提取运算；

[0172] Step 4 进行Batch Normalization归一化运算；

[0173] Step 5 输入到下一个卷积层，直至输入到全连接层；

[0174] Step 6 全连接层提取特征，输入到softmax函数进行分类，选取输出概率最大的作为分类结果。

[0175] Step 7 判断比较标签与分类结果，并统计分类结果；

[0176] Step 8 Until所有的Batch全部都测试完毕。

[0177] 4优化网络参数

[0178] 如果遇到训练样本数量较少、神经网络的模型还复杂的情况，在训练网络时可能会出现过拟合现象。此时网络过度拟合训练集，却对训练集以外的数据不能很好地拟合。表现为随着训练迭代次数的不断增多，分类网络在训练集上的测试分类错误逐渐减小，然而测试集上的测试分类错误逐渐增大。网络过拟合现象可由图13描述。

[0179] 实际深度网络的过拟合的现象都普遍存在。所以研究减轻训练网络时出现过拟合现象十分必要，也是提高调试网络性能非常重要的步骤。

[0180] 4.1 特征可视化

[0181] 提取出特征的好坏会直接影响后续图像分类的准确性，所以这是数字图像处理研究中的重点。为了使网络能自动提取有利于分类任务的特征，卷积神经网络把与图像通过卷积之后所提取到的特征进行前向传播，并且把网络输出值与数据标签的差值通过反向传播，然后调整网络参数。前面已经提到卷积神经网络本身具有的黑盒子性质，这会增加优化网络参数的难度。但是倘若可以把卷积神经网络在各个网络层中学习到的特征以图像形式展现出来¹，这会在帮助参与人员更方便快捷地优化网络参数。所以之后在本发明设计的深层卷积神经网络的基础之上，将卷积神经网络学习到的信息用可视化的图像表示出来，然后根据此结果分析规律并进行总结。输入的原始图像如图14所示。

[0182] 输入图像历经卷积神经网络卷积层conv1的处理之后，得到图15所示的输出特征映射。通过卷积层conv1之后的可视化图像如图16所展示，可以清楚看到卷积层conv1所学习的大量信息体现在输入图形的边缘轮廓信息上。从图15可以看出，每个小图像块都是输入图像猫的轮廓，是从视觉的不同角度来观察的。像图中所示的第一和二列从左侧观察输入图像的概率较高，三和四列从正前方观察的概率较高，五和六列从右侧观察的概率较高。通过这些结论，可以预测到，如果经过大量训练，图中所示猫的边缘轮廓信息会被卷积层conv1的卷积核所学习。通过图16更进一步说明图像边缘提取理论适用于底层卷积层的学习方式。也就是说，基于输入图像的轮廓边缘信息的提取原理，不同方向的边缘轮廓信息可由不同角度的卷积算子提取。垂直方向的轮廓信息可被垂直梯度算子所提取，水平方向也是如此。既然已经知道卷积层conv1的卷积核会主要学习图像物体边缘信息，那么在设计卷积层conv1时要注意：

[0183] (1) 丰富卷积核的数量。因为相邻输入图像的观察角度只有细微差异，而距离远一些的观察角度则差异性提高。卷积核数量代表从不同角度去看物体，进一步表示，卷积核数量越多，所提取的特征信息也会随之增加，十分有助于分类结果。

[0184] (2) 卷积核的数量不是越多越好。如果数量超越上限，会产生冗余，此时不同的卷积核可能会提取同一个角度上的特征信息。而且卷积核数量与深度网络的训练速度是一对矛盾需要合理设计。

[0185] 高维的卷积核不能直接可视化，如果要展示卷积层conv2的可视化结果，需要把它的高维卷积核拆成低维的卷积核，再进行可视化处理。图17就表示的是卷积层conv2学到的卷积权值。图的每一行从外形、角度等方面都很接近，而且十分像波纹线条。当观察不同的行，会发现图像相差较大。每个高维卷积核的低维子卷积核所学习的信息十分相似，同时不同高维卷积核之间的不同大于卷积核内部子卷积核间的不同。

[0186] 经conv2层的卷积处理后输出的特征映射被展示在图18中。可以明显看出大部分的图像都是输入图像猫的轮廓，而且相比于conv1层输出的特征映射可能经过图像增强处

理过程。综上,高维卷积层除有提取输入数据特征的能力,还兼备一定的过滤噪声的能力和增强特征信息的本领。所以不难得出增加一些卷积层的数目可以提高所提取的特征信息质量。

[0187] 图19和图20分别为经由conv3和conv4进行卷积操作之后输出的特征映射结果。很明显地,conv3卷积操作之后的输出特征映射中每一个小图片亮点数多且位置零散,而conv4卷积操作之后的输出特征映射中每一个小图片亮点数少但位置集中,输入图像的主要特征信息得到增强。

[0188] conv5经过卷积操作之后输出的特征映射结果展示在图21中。可以与conv4卷积层的输出特征映射结果相互比较不难发现,每一个小图片亮度再次增强,说明主要特征信息得到增强,噪声得到抑制。

[0189] 经过pool5池化层后的输出特征映射结果被显示在图22中,conv5经过卷积操作之后主要特征信息被保留,再经过对特征映射进行下采样,可以看出输出特征映射的像素数目变得更少,所需训练的参数也随之减小,训练率大大提高。

[0190] 图23和图24分别表示fc6全链接层神经节点输出值和全链接层神经节点输出值为正的梯度直方图。从这两个图中可以看出输出特征比较稀疏,因为经过fc6全链接层后输出值都集中在零附近的部分。

[0191] 图25和图26分别表示经过fc7全链接层后的输出值和输出值直方图。比较图23和24可得,fc7全链接层神经节点的输出值更加聚集在零值附近,也就是说fc7全链接层只有少部分神经元被激活,而大部分的神经节点被抑制。结合上述的可视化图像可知在卷积神经网络中网络层越高,它的输出特征向量越稀疏。

[0192] 深层卷积神经网络输出层的结果展示在图27,卷积神经网络的输出层能依据输出值中的最大神经节点序号来判断和预测输入图像的标签值。其次,用所得到的标签值与输入图像的标签值做匹配处理,若相等则证明正确,不等则错误。从图中可以看出当在第四个神经节点时,卷积神经网络的输出值是最大的,而在其余的部分输入值近乎于零。本文输入图像是标签号为3的猫,图像标签号的范围是0~9,输入图像在节点中处于第四,也就是说输入图像的标签值等于输出值,因而网络能正确分类。

[0193] 4.2设定batch的值

[0194] 一般通常依据损失函数的下限值或者最大的迭代次数来决定在何时对网络进行停止训练。在传统的网络训练中,采用随机顺序把每次迭代后的训练样本分为若干大小一样的batch小数据集,再按顺序输入网络开始训练,当所有的小数据集都训练完毕,开始下一个轮回的迭代。但本文采取不同的batch小数据集进行训练,在经历多次试验后可以得到batch的范围。如果想要停止训练,必须要当损失函数的值低于预定的阈值。测试五次后取平均值最大程度避免偶然事件突发。

[0195] 表2采用不同batch值时图像分类网络的性能对比

[0196]

		分类错误率 (%)				
batch 值	50	100	200	300	400	
cifar-10	25.64	15.23	14.98	14.87	14.94	
cifar-100	57.10	43.81	41.45	40.23	40.24	

[0197] 当batch的值不同,卷积神经网络的测试错误率被列在表2中。可以看出,当的batch的值从50变化到100,cifar-10和cifar-100的值急剧下降,也就是测试错误率降低。当batch的值从100增加到200,cifar-10和cifar-100的值也有所下降,但是幅度没有batch的值从50变化到100的大。当batch的值从200增加到300,cifar-10和cifar-100的值也有所下降但下降幅度更小。甚至当batch的值batch的值,cifar-10和cifar-100的值几乎没有变化。从表中可以看出batch值越小,损失函数波动的范围就越大,同时卷积神经网络损失函数收敛的下降速度会越来越快,但网络的分类错误会升高,则证明网络没有全面优化。

[0198] 训练样本若采用batch的采样方式有两个优势:一是将多个训练样本结合成一个batch小数据集再进行训练,可以有效地抑制训练样本的噪声。二是可以充分利用计算机的资源,在现在所具有的高性能CPU处理速度下,批处理的训练速度可以远大于串行处理速度,进而提升了训练网络的效率。但值得注意的是,batch值有一个上界,随着值的增大,训练样本的噪声抑制能力不会再提高,此时还会导致占用计算机更多的内存空间,训练网络的效率不升反降。4.3dropout的设计

[0199] 通常情况训练样本较少时,会产生模型的过拟合现象,此时我们在训练模型时候,可以采用dropout方法,这是一种类似于ensemble相互结合的方法。

[0200] 一般大规模的神经网络都会有两个致命的弱点:第一种就是耗时,第二种就是容易过拟合。一个过拟合的深度学习模型,基本上就没多大用途了,而为了解决过拟合的问题,人们一般都会采用ensemble组合方法^[54],即训练多个深度模型来进行组合,这种方式虽然在一定程度上解决了过拟合问题,在时间成为了其新的又一大问题,不仅在训练上耗费了很多时间,而且测试上有会耗费大量的时间。总之,在大规模的深度网络学习模型上,此种方法是行不通的。

[0201] 借鉴于ensemble组合方法的出现,我们可以在一个模型上进行拆分,让模型“瘦”下来。如图28所示:

[0202] Dropout过程的原理是:深度学习网络在每次batch样本输入时,会随机地会让一些隐含层节点暂时丧失作用,忽略它们的权重,但是这些权重必须进行保留,网络只是暂时忽略它们是结构的一个部分。因为下一次的batch样本输入时,那些在上一次不起作用的节点在这一次可能就随机起作用了。由图28可以看出,对于一个含有N个节点的深层卷积神经网络,当加入dropout处理之后,该网络就可以看成是 2^n 个模型的组合了,由于此时网络的训练参数数目是不变的,所以这就很好的解决了耗时的问题。

[0203] 深层卷积神经网络加入dropout后,其训练层面和测试层面就应做出相应的改进。下面就这两个层面分别做出介绍:

[0204] (1) 训练层面

[0205] 由于是随机的忽略隐含层的节点,所以无可避免地要在训练网络的每个神经单元

加入一道概率流程。如图29所示：

[0206] 左图为标准神经单元图，也就是没加入dropout的神经网络单元图，其对应公式如下：

$$[0207] z_i^{(l+1)} = w_i^{(l+1)} y_i^{(l)} + b_i^{(l+1)} \quad (3)$$

$$[0208] y_i^{(l+1)} = f(z_i^{(l+1)}) \quad (4)$$

[0209] 右图为加入了dropout的神经网络单元图，其对应公式如下：

$$[0210] R_j^{(l)} \sim Bernoulli(p) \quad (5)$$

$$[0211] \bar{y}^{(l)} = r^{(l)} * y^{(l)} \quad (6)$$

$$[0212] z_i^{(l+1)} = w_i^{(l+1)} \bar{y}^{(l)} + b_i^{(l+1)} \quad (7)$$

$$[0213] y_i^{(l+1)} = f(z_i^{(l+1)}) \quad (8)$$

[0214] (2) 测试层面

[0215] 在预测试部分，将每一个神经单元的参数都要乘以概率P如图30所示：

[0216] 本优化设计中的dropout的节点比例为50%，从理论上来讲其效果是最好的，最直观的解释就是比率为0.5时dropout随机生成的网络结构是最多的。那是因为在对每次的输入网络的样本做权值更新的时候，不能完全肯定每2个隐含节点都会同时出现，因为它们是以确定概率随机出现。这种更新方式会避免一些特征只能在它的特定特征下才有效果，使之不会依赖于有固定关系的隐含节点的共同作用。此外，dropout也是模型平均的一个类型，虽然每次输入的batch样本会对应不同的网络结构，不同的样本会对应不同的模型，但隐含节点的权值可以被它们共享。如表3所示，当增加了dropout之后，cifar-10和cifar-100会分别下降0.14%和0.42%，也就是分类错误率会降低。

[0217] 表3增加dropout前后图像分类效果对比

[0218]

	分类错误率		加入 dropout 深层卷积神经网络
	(%)	神经网络	
cifar-10	14.94		14.80
cifar-100	40.24		39.82

[0219] 4.4 momentum动量值的设置

[0220] 通常情况下在神经网络的训练中需要将损失函数降到最低，也就是最小化损失函数，一般情况下采用梯度下降算法来为最小损失函数增加相关的优化操作，现在基本上也都是采用随机梯度下降法SGD(stochastic gradient descent)，其实，它是梯度下降方法与batch处理的结合，该方法的原理就是把一个训练数据集分配成n个包含m个样本的batch小数据集。在每次训练的时候不是训练整个训练集，而是一个batch的数据集。随机梯度下降法SGD的优势体现在当有很多训练数据的时候，可以减轻机器压力并且整个数据集更新可以更快地收敛且不在时间上显示。并且，当类似的样本多次出现时，其收敛的更快。

[0221] 尽管SGD方法较梯度下降法有上述优势，然而，更新方向会完全依赖当下的batch

数据集,导致更新的不稳定性大。本论文解决这一问的方法就是引入momentum¹。Momentum可以模拟物体运动时候产生的惯性,也就是动量,换句话说就是在每次更新时,利用当前的batch的梯度在之前的方向上做微调来确定最终的更新方向。因此可以使网络学习速度加快并保证稳定性,拜托局部最优的状况,也就说为全局最优增加了可能性。Momentum动量值表示的是在多大程度上保留原来的更新方向,这个值在0到1之间。表4测试统计了不同momentum值所对应的不同分类错误率。通过实验对比发现,在开始训练时,由于梯度可能会很大,所以momentum初始值一般选0.5,当梯度小时,momentum改为0.9时,网络参数得到了很好的优化,图像分类错误率也最小。

[0222] 表4深层卷积神经网络使用不同momentum值的性能对比

[0223]

分类错误率 (%)					
momentum 值	0.5	0.6	0.8	0.9	1.0
cifar-10	15.12	15.03	14.98	14.86	15.14
cifar-100	41.10	41.81	41.45	40.23	41.24

[0224] 4.5数据集扩增

[0225] 在大数据时代的今天,有的时候并不是说有了好的学习算法就能训练出好的神经网络模型,更多的时候还是因为更多的海量数据,特别是在深度学习方法中,为了训练出更好的网络模型,就需要充足的训练数据并采用更深层的网络。于是,获得更多的数据就显得尤为重要了,但是,如果人为的收集更多的数据就意味着需要耗费更多的人力和物力。从人工标注就可以看出这个工作的效率是极其低下的。

[0226] 对于扩充数据集,既简单又有效的方法就是对原始数据进行扩充,通过对原始数据进行改动,可以得到更多的数据。对于图片数据而言,扩充其数据最简单的方法就是调整图片的角度,本论文为扩充数据集所做的工作就是将图片镜像顺时针旋转45度,用实验来说明训练数据集的数量对图像分类网络性能的影响。表5记录了深层卷积神经网络对扩充了的cifar-10和cifar-100数据集的分类测试结果。从表5可以看出,用扩展了的cifar-10和cifar-100数据集训练深层卷积神经网络,其分类错误率分别下降了0.11%和0.32%,说明了该方法能够改善深层卷积神经网络。

[0227] 表5数据集的数量对图像分类效果的对比

[0228]

分类错误率 (%)	原始数据集	扩展数据集
cifar-10	14.86	14.75
cifar-100	40.23	39.91

[0229] 5实验结果与分析

[0230] 本论文结合上述五种优化网络参数的方法,通过调试各方法的参数,使用不同的数据集来对深层卷积神经网络进行训练测试,并与其它网络算法模型进行图像分类性能对比如下。

[0231] 5.1数据库cifar-10实验结果及分析

[0232] Cifar-10是由Hinton的两个学生Alex Krizhevsky和Ilya Sutskever收集的一个用于普适物体识别的数据库,该数据库包含10类图像,共有60000张32*32的彩色图像,其中有50000张训练图像和10000张测试图像。数据库分为5个训练块和1个测试块,每个块包含10000张图像,其测试块的10000张图像都是随机地从每类中选择1000张图像,其每个训练块的10000张图像也是随机地从每类中选择图像,但并不是均等的选择,一些训练块中可能出现某一类图像比另一类图像多的情况。同已经成熟的人脸识别相比^[60],cifar-10这种普适物体识别更具有挑战性,因为其数据中含有大量的特征、噪声以及识别物体的物理特征比例不一,所以该数据库较适合深度复杂的神经网络模型的学习。

[0233] 表6 cifar-10在不同网络上的分类错误率

[0234]

算法模型	分类错误率 (%) / 训练时间 (min)
SVM ^[32]	25.57
Boosting	24.38
LR	24.01
ImageNet-2010	14.30 / 51
深层卷积神经网络	14.75 / 34

[0235] 表6为本论文设计的深层卷积神经网络、几种传统浅层卷积神经网络以及目前性能较好的复杂神经网络ImageNet-2010分别在cifar-10数据库上的测试结果。本论文设计的深层卷积神经网络在训练时间34min内,cifar-10上的实验测试错误率为14.75%,低于表格中那几种浅层的卷积神经网络的错误率。也就是说深层卷积神经网络复杂图像数据的分类效果比浅层卷积神经网络对复杂图像数据的分类效果要好。深层卷积神经网络与ImageNet-2010网络相比,其分类错误几乎相近,只相差了0.45%。尽管,本文的深层卷积神经网络与ImageNet-2010网络的网络层数相同,但是本文的深层卷积神经网络的卷积核数远小于ImageNet-2010网络。从理论上来讲,卷积核数越多,学到的特征也就越多,网络的分类效果也更佳。如果只关注这一个层次,深层卷积神经网络的分类效果确实稍逊于ImageNet-2010网络,但实验结果却表明两种分类效果很接近,这是由于我们的深层卷积神经网络在网络优化过程中采取了激活函数Maxout和逐层训练的方法有效降低卷积核数少所引入的缺陷。而且随着深层卷积神经网络卷积核数的减少,训练时间也随之大幅减少。所以综合考虑,深层卷积神经网络在cifar-10图像库上取得了很理想的分类效果。

[0236] 5.2数据库cifar-100的实验结果及其分析

[0237] 相似于cifar-10的数据库,cifar-100数据库也是一个用于普适物体识别的数据集,只不过它比cifar-10更复杂,它包含了100类图像,其中每类图像有100张测试图像和500张训练图像。表7为本发明设计的深层卷积神经网络以及目前性能较好的复杂神经网络ImageNet-2010和Conv-KN分别在cifar-100数据库上的测试结果。

[0238] 表7 cifar-100在不同网络上的分类错误率

[0239]

算法模型	分类错误率(%) / 训练时间 (min)
Conv-KN ^[53]	43.46
ImageNet-2010 ^[52]	39.76 / 68
深层卷积神经网络	39.91 / 40

[0240] 由表7可知深层卷积神经网络在cifar-100上的分类错误为39.91%，比Conv-KN网络的分类效果要好很多，其下降了3.55%，与ImageNet-2010网络的分类错误率相近，但训练时间比ImageNet-2010网络短，这是因为深层卷积神经网络的卷积核数比深层卷积神经网络的卷积核数少。所以综合深层卷积神经网络在cifar-10和cifar-100上的分类错误率及训练时间，本论文设计的深层卷积神经网络取得了很好的分类效果。

[0241] 以上显示和描述了本发明的基本原理和主要特征和本发明的优点。本行业的技术人员应该了解，本发明不受上述实施例的限制，上述实施例和说明书中描述的只是说明本发明的原理，在不脱离本发明精神和范围的前提下，本发明还会有各种变化和改进，这些变化和改进都落入要求保护的本发明范围内。本发明要求保护范围由所附的权利要求书及其等效物界定。

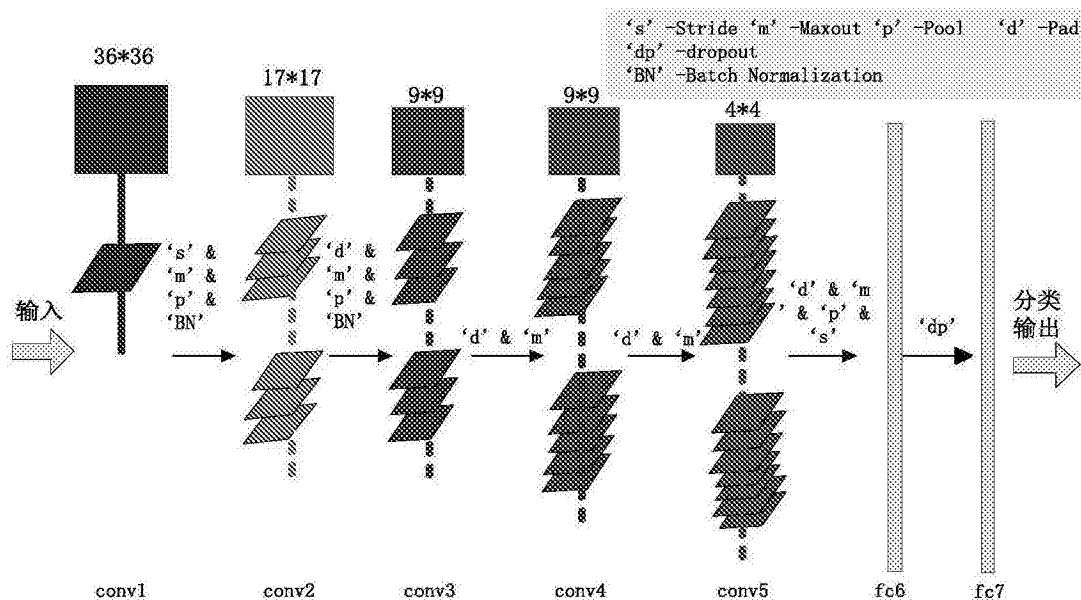


图1

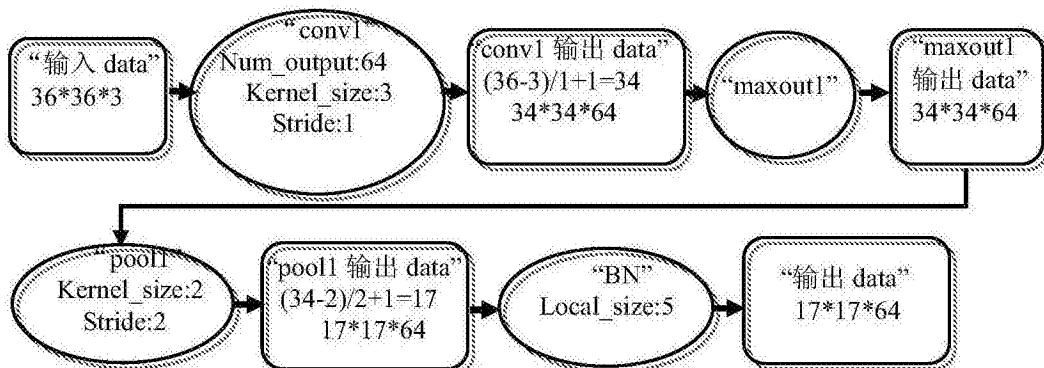


图2

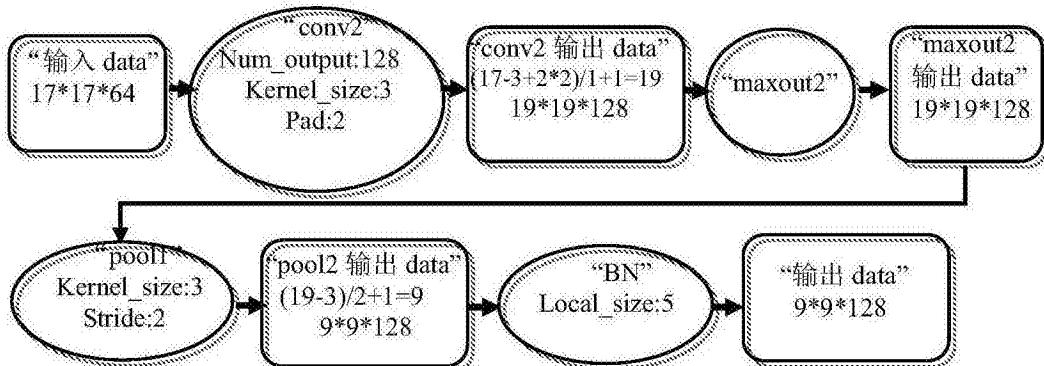


图3

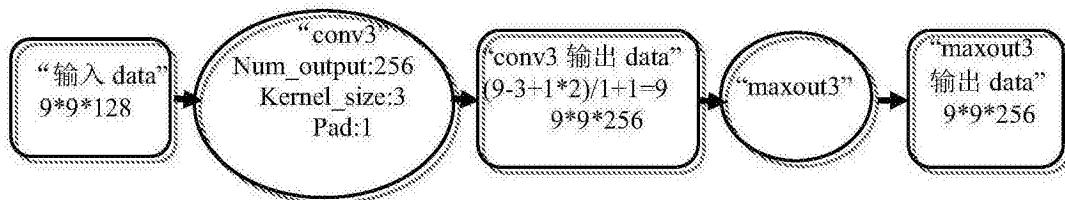


图4

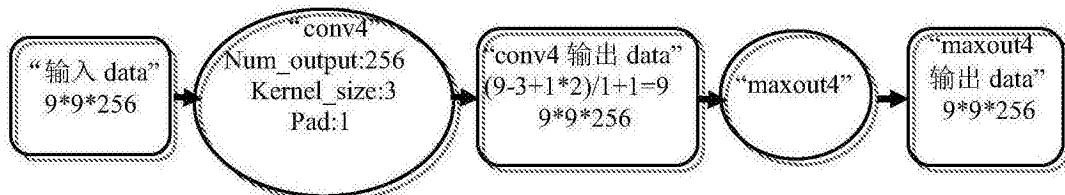


图5

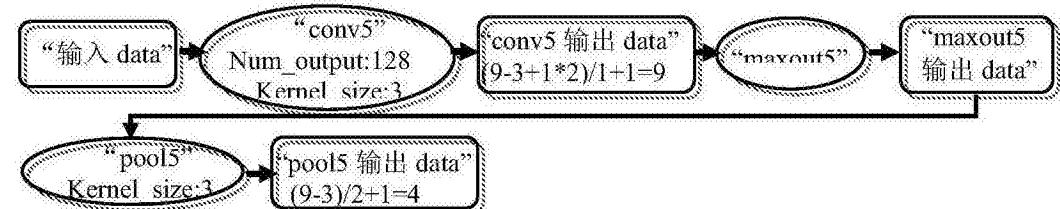


图6

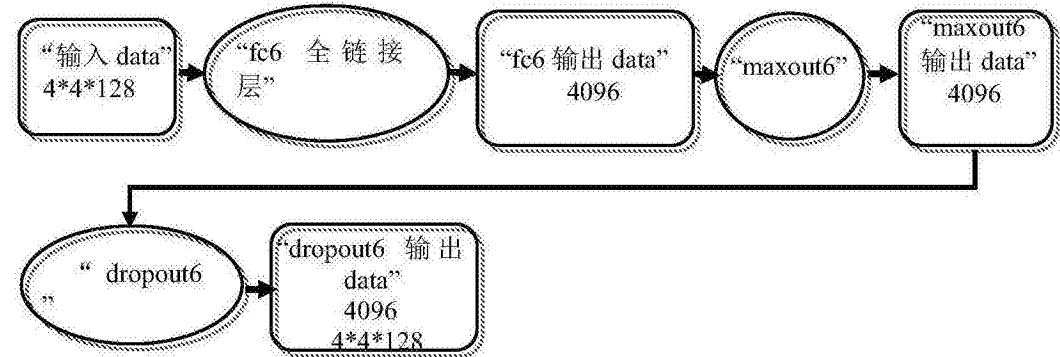


图7

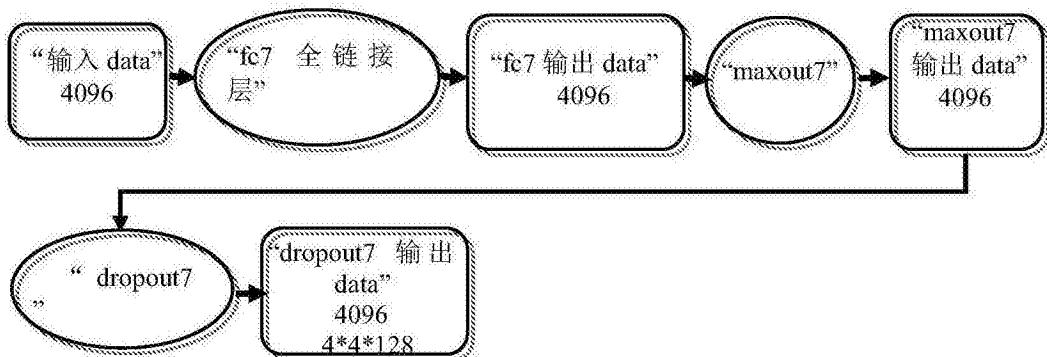


图8



图9

0	1.1	0
1.0	2.0	0.9
2.5	1.0	1.5

图10

0	0.11	0
0.1	0.2	0.09
0.25	0.1	0.15

图11

0	0	0
0	1	0
1	0	0

图12

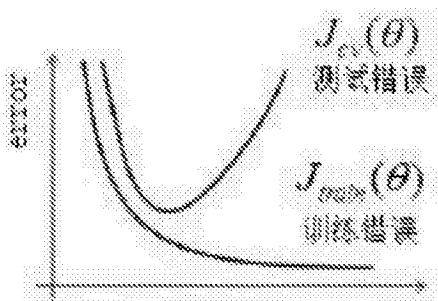


图13



图14

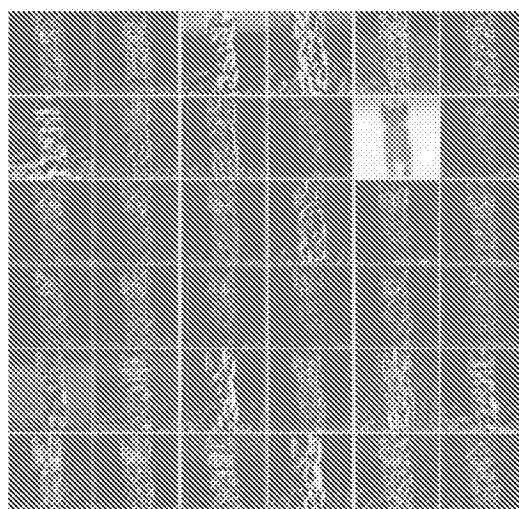


图15

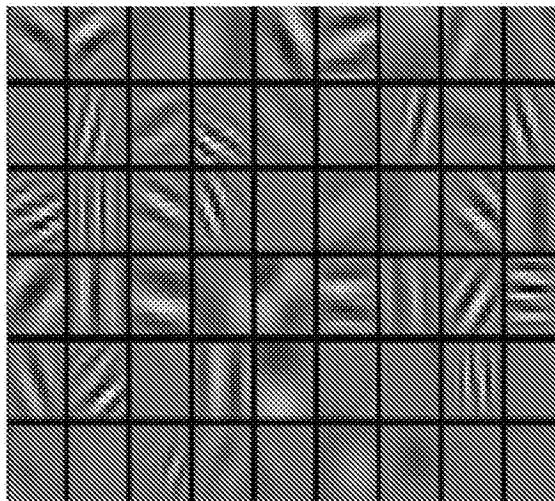


图16

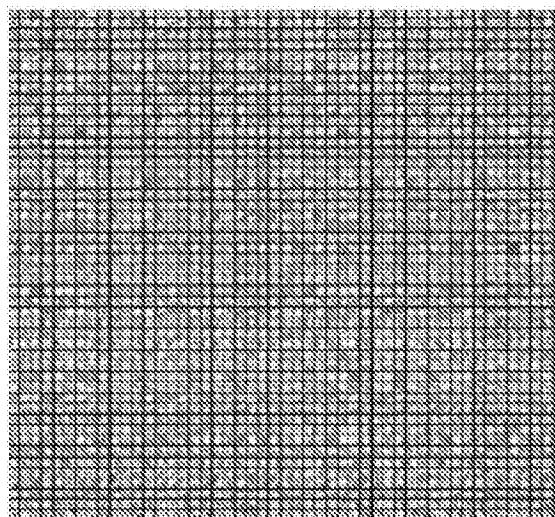


图17

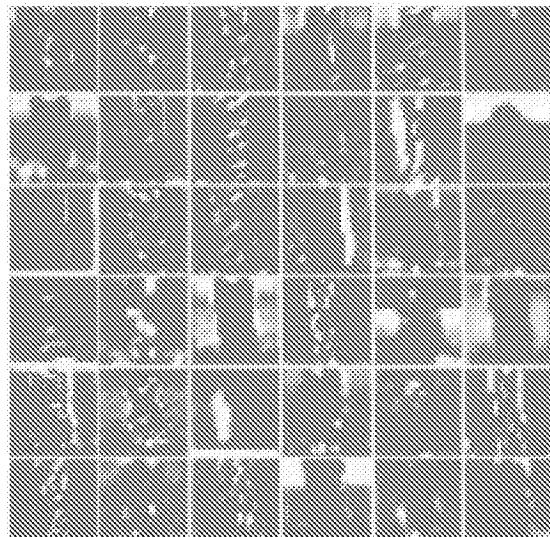


图18

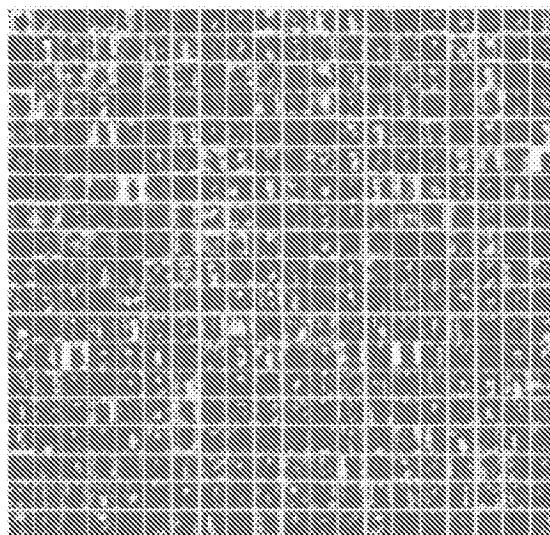


图19

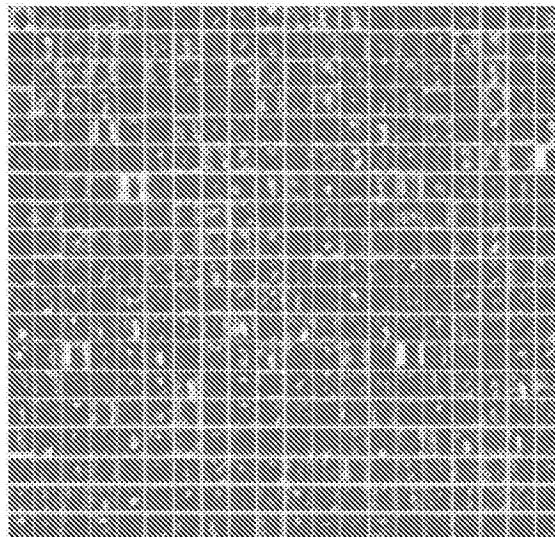


图20

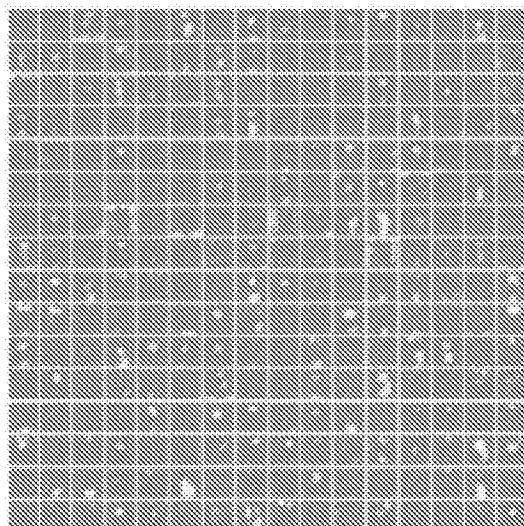


图21

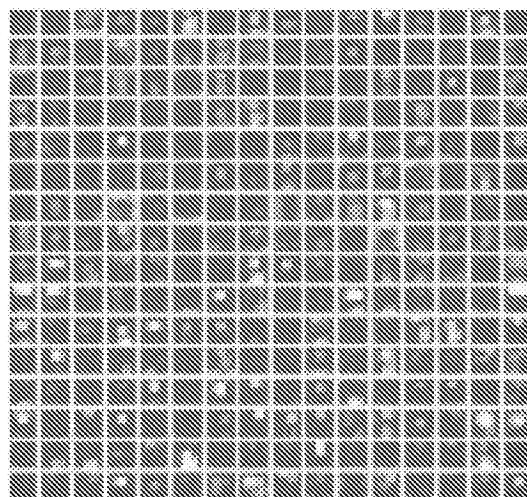


图22

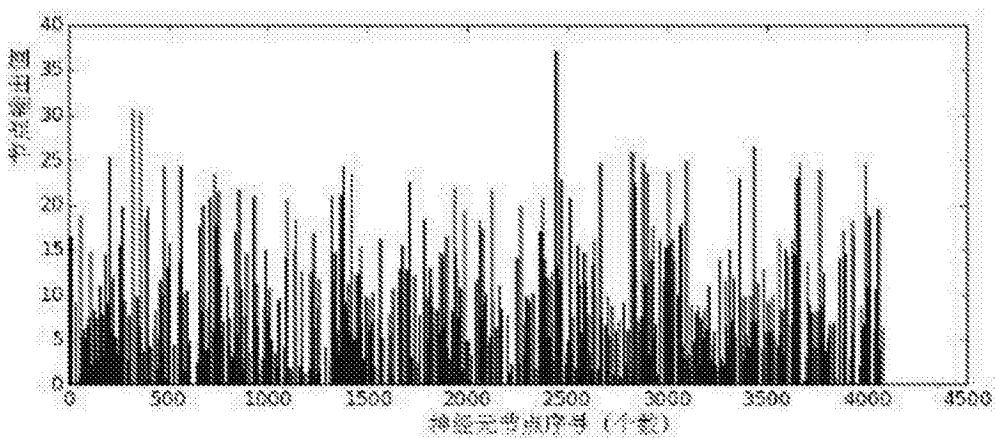


图23

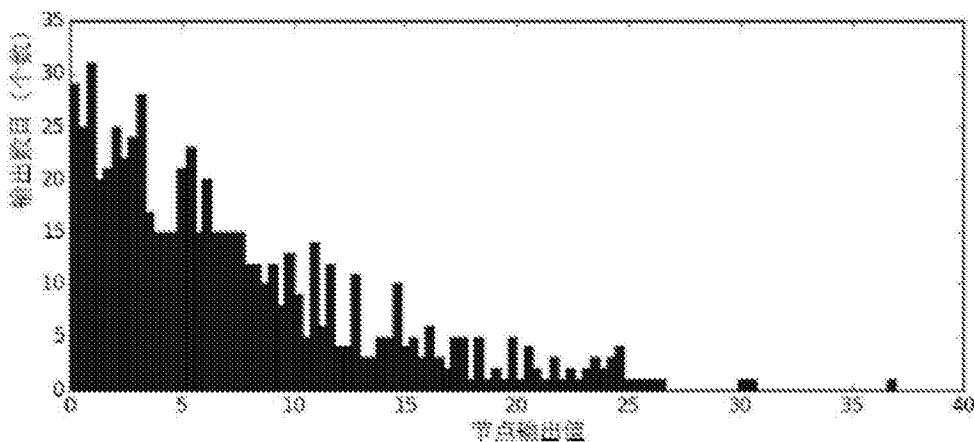


图24

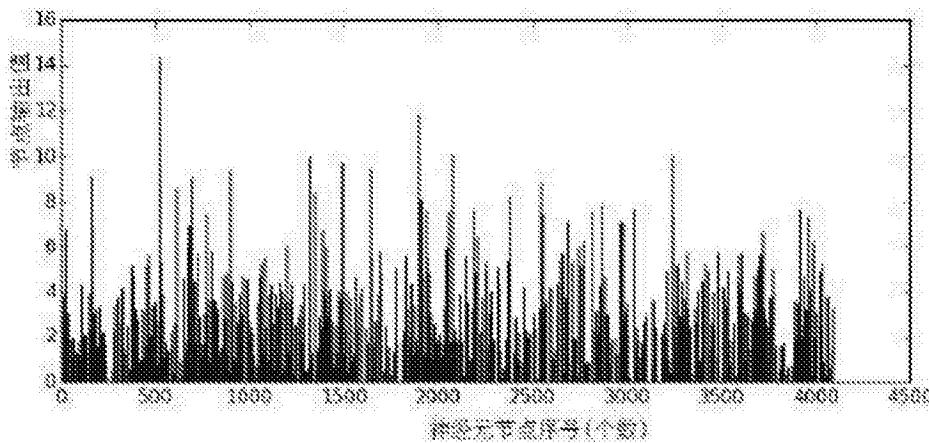


图25

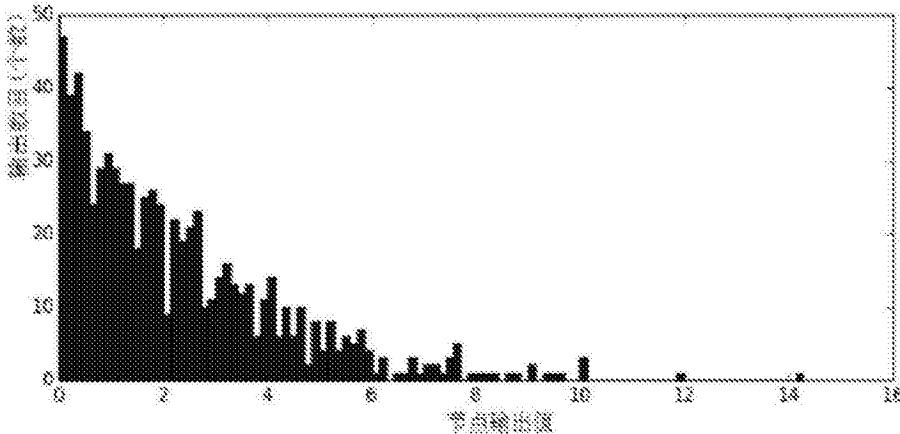


图26

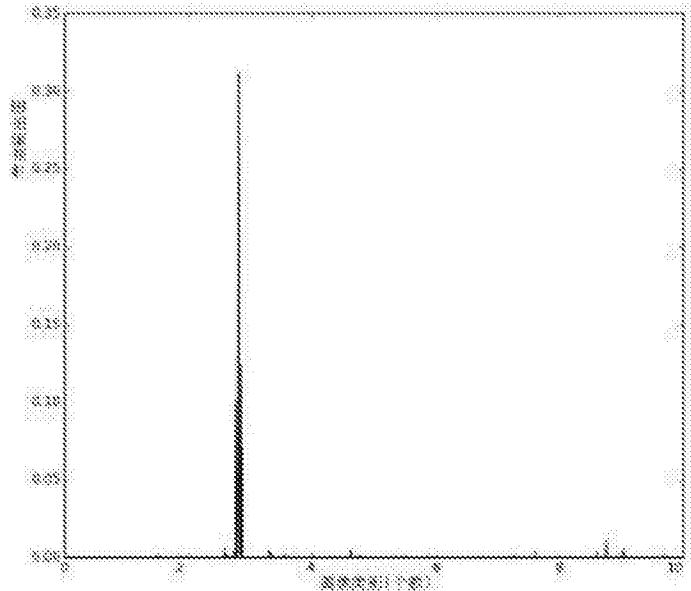


图27

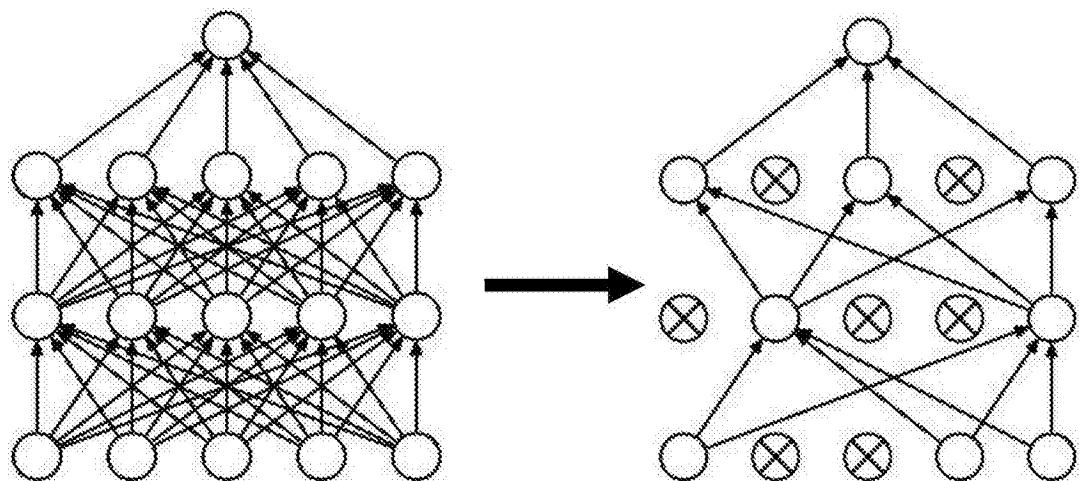


图28

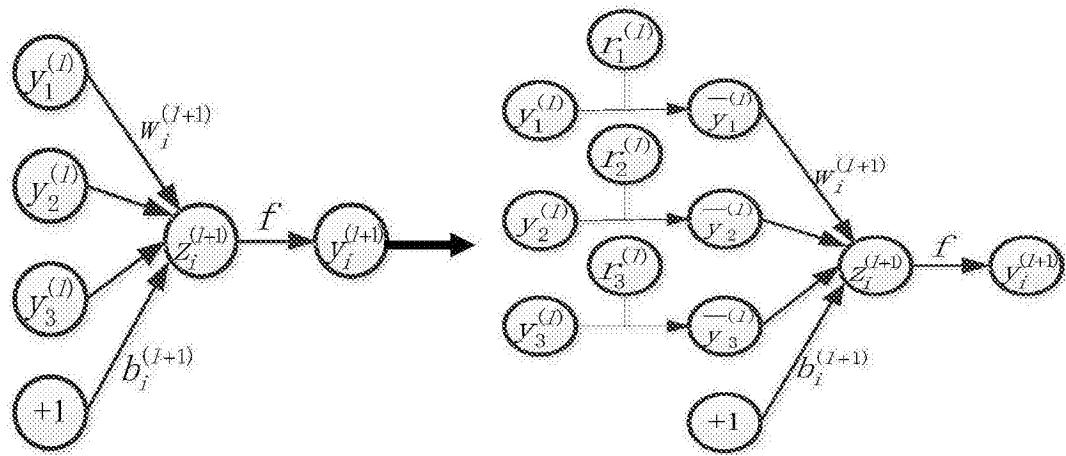


图29

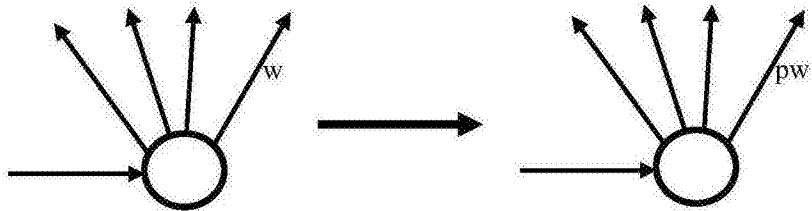


图30