控制与决策
Control and Decision

基于忆阻器边缘计算的图像分类电路设计

罗佳, 冉欢欢, 何凯霖, 丁晓峰

引用本文:
罗佳,冉欢欢,何凯霖,丁晓峰. 基于忆阻器边缘计算的图像分类电路设计[J]. 控制与决策, 2022, 37(9): 2353−2359.

在线阅读 View online: https://doi.org/10.13195/j.kzyjc.2021.0120

基于忆阻器边缘计算的图像分类电路设计

罗 佳[1], 冉欢欢[2†], 何凯霖[1], 丁晓峰[1]

(1. 四川大学 锦江学院，四川 眉山 620860；2. 西华师范大学 电子信息工程学院，四川 南充 637002)

摘 要: 针对边缘智能设备低功耗、轻算力的要求,采用新型存算一体器件——忆阻器作为基础电路元件,设计低功耗图像识别电路. 该电路采用多个忆阻卷积层和忆阻全连接网络串联的方式,获得较高的识别精度. 为了减小忆阻卷积层计算所需的忆阻交叉阵列的行尺寸与列尺寸的不平衡,同时降低输入电压方向电路的功耗,将输入电压反相器置于忆阻交叉阵列之后. 所设计电路可以将完成忆阻卷积网络运算所需的忆阻交叉阵列的行大小从$2M+1$减少至$M+1$,同时将单个卷积核计算所需的反相器的数量降至1,大幅度降低忆阻卷积网络的体积和功耗. 利用数学近似,将BN层和dropout层计算合并到CNN层中,减小网络层数同时降低电路的功耗. 通过在CIFAR-10数据集上的实验表明,所设计电路可以有效地对图像进行分类,同时具备推理速度快(136 ns)和功耗低的优点(单个神经元功耗小于3.5 uW).

关键词: 边缘计算；忆阻器；卷积；反相器；数据集；神经元

中图分类号: V211.3 **文献标志码:** A

DOI: 10.13195/j.kzyjc.2021.0120

引用格式: 罗佳,冉欢欢,何凯霖,等. 基于忆阻器边缘计算的图像分类电路设计[J]. 控制与决策, 2022, 37(9): 2353-2359.

Design of image classification circuit based on edge computing of memristor

LUO Jia[1], *RAN Huan-huan*[2†], *HE Kai-lin*[1], *DING Xiao-feng*[1]

(1. Jinjiang College，Sichuan University，Meishan 620860，China；2. School of Electronic Information Engineering，West Normal University，Nanchong 637002，China)

Abstract: Aiming at the requirements of low power consumption and light computing power for edge smart devices, this paper uses a new type of integrated storage and computing device-memristor as the basic circuit element, and designs low power consumption and image-specific circuits. The circuit uses a series of multiple memristive convolutional layers and a memristive fully connected network to obtain high recognition accuracy. In order to reduce the imbalance of the row size and column size of the memristive interleaved array required for calculation of the memristive convolutional layer, and at the same time reduce the power consumption of the input voltage direction circuit, the input voltage inverter is placed after the memristive interleaved array. This circuit can reduce the row size of the memristive interleaved array required to complete the memristive convolution network operation from $2M$+1 to M+1, and at the same time reduce the number of inverters required for the calculation of a single convolution core to 1, which greatly reduces the volume and power consumption of the memristive convolutional network. Using mathematical approximation, the calculations of the BN layer and the dropout layer are merged into the CNN layer to reduce the number of network layers and reduce the power consumption of the circuit. Experiments on the CIFAR-10 data set show that the circuit can effectively classify images, while having the advantages of fast inference speed (136 ns) and low power consumption (the power consumption of a single neuron is less than 3.5 uW).

Keywords: edge computing；memristor；convolution；inverter；data set；neuron

0 引 言

近年来,随着人工智能的蓬勃发展,大量卷积神经网络被相继提出,进一步依托于互联网的高速发展,由大量数字处理设备和服务器组成的大型数据计算中心相应产生,为各类模型的提出和运算提供了极佳的实验基础,同时数据计算中心不断扩大,也促进互联网技术随之发展,云计算正式诞生.

云计算拥有大量的应用场景和需求,在商用上,

收稿日期: 2021-01-21；**录用日期:** 2021-07-05.

基金项目: 国家自然科学基金项目(51208434)；四川省科技厅科技计划项目(2021YJ0315).

†通讯作者. E-mail: 364483391@qq.com.

类似于谷歌和亚马逊这样的互联网公司已早早布局[1-2],拥有了持续的用户数据接入. 云计算目前拥有广阔的市场和用户,主要通过网络通信接入海量的用户数据运算后反馈给用户. 显然,整个过程极度依赖网络通信,对通信要求较高,同时由于双向通信会损失一部分时间延时,对应高响应要求的处理事件不友好. 而涉及到用户数据,对于用户的隐私保护和接入管理也同样重要,云计算难以处理以上问题. 在这种情况下,边缘计算进入了大家的视野,将用户数据保留于边缘端,减少数据对于云计算的上传,降低通信要求,在边缘端部署部分算力,满足低功耗要求,并且对用户隐私数据进行实时保护[3].

在海量图像数据分析和处理领域,深度学习网络拥有极佳的效率和性能,可以满足图像识别对效率和精度的要求. 而图像数据的分析和处理的效率及性能极大程度取决于神经网络的构架, Iandola等[4]提出了名为“squeezunet”的小型卷积神经网络架构. Szegedy等[5]提出了更加深入的卷积神经网络架构,称为Inception. Dai等[6]对Szegedy的研究进行了改进. Zeiler等[7]提出了一种可视化卷积神经网络激活函数. Google的研究人员提出了BlazeFace[8],一个超高性能的人脸检测网络,通过BlazeFace,移动平台上可以达到1 ms以内的检测速度,拓宽了很多基于边缘计算的人脸相关联应用的发展空间. Krizhevsky等[9]提出了一种深度卷积神经网络结构AlexNet,这是深度学习的一个重大突破. AlexNet由5个卷积层和3个全连接层组成,使用图形处理单元(GPU)进行卷积运算, ReLU作为激活函数,其不仅在计算方面比Sigmoid更加简单,而且可以克服Sigmoid函数在接近0和1时难以训练的问题,同时使用dropout减少过拟合问题[10].

传统深度学习对于计算能力的需求极大,传统的冯诺依曼计算机体系结构通过运算器和控制器进行数据处理,提取和存储存储器内的数据,数据传输效率较低,而神经态计算计算结构有别于传统的冯诺依曼计算机体系结构,其以生物神经元系统的信息传递为参照,实现神经系统类似的神经元的计算[11]. 作为模拟神经元的基础电路元件所需元件信息的传递功能,系统记忆学习能力和高度灵敏性是目前神经态计算系统面临的主要问题. 近年来,随着元件领域的发展,纳米级原件出现并被广泛应用,比传统CMOS体积更小的纳米硅薄膜晶体管的出现,不仅大大降低了电路的功耗,电路的速度也得到极大提升[12]. 忆阻器的出现为上述问题提供了理想的解决方案,它是具有动态特性的电阻器,根据激励电压其阻值可变[13]. 与传统电路元件相比,忆阻器的记忆性和灵活性使其更适用于神经态计算计算结构,且忆阻器纳米级尺寸和高速的特性非常适合边缘设备的低功耗和速度要求. 因此,忆阻器可以作为神经态计算计算结构的基础电路元件进行神经网络突触功能的记忆储存功能,实现深度学习网络的结合,拥有广阔的应用空间[14]. Bala等[15]提出针对忆阻器深度学习神经网络的高效激活电路,从电路设计方向对忆阻神经网络进行优化. Wen等[16]、Liu等[17]和Chen等[18]针对忆阻器神经网络实际应用场景对网络进行裁剪和优化,通过二值化和稀疏化等方法降低网络对忆阻器性能要求,提高性能. Bao等[19]针对忆阻器模型和组成电路特性,对比传统网络电路,基于电压电流关系仿真验证了忆阻器网络模型的性能和优越性. Wan等[20]通过联系忆阻器在神经网络中的应用,在电路和网络上设计一种改进的忆阻器遗忘模型,通过仿真验证了其有效性,在模拟人脑运算和记忆方面提供了可能性.

通过上述工作,基于忆阻器构建的神经网络电路对于边缘设备计算有着广阔的应用前景. 本文从神经网络各层的运算要求出发,对整体网络构架优化电路,设计出适合各层网络的忆阻器运算电路;依托于忆阻器记忆和计算一体的特性,提高图像识别的效率,降低运算所需功耗.

1 基本忆阻器神经电路

1.1 忆阻器模型

忆阻器的常用模型利用了其电压阈值的特性,只有当输入电压超过忆阻器的阈值电压时忆阻器的阻抗才会发生变化. 输入电压$V(t)$、忆阻器阻抗$G(t)$与当前产生电流$I(t)$有如下关系:

$$V(t)=G(t)I(t). \tag{1}$$

忆阻器阻抗的微分方程可表示为

$$\frac{\mathrm{d}G(t)}{\mathrm{d}t}=-L(v(t))H(G(t),V(t)). \tag{2}$$

其中: $H(G(t),v(t))$为一个窗口函数,将忆阻器的阻抗限制在$[1/G_{\rm on},1/G_{\rm off}]$内,且有

$$H(G(t),v(t))=\theta(-v(t))\theta\Big(\frac{1}{G_{\rm off}}-G(t)\Big)+\theta(v(t))\theta\Big(H(t)-\frac{1}{G_{\rm on}}\Big). \tag{3}$$

$\theta(x)$为一个参数方程,当$x>0$时$\theta(x)=1$,当$x<0$时$\theta(x)=0$. $L(v(t))$阈值电压函数为

$$L(v(t))=\varphi[V(t)-0.5(|v(t)+V(t)|-|v(t)-V(t)|)], \tag{4}$$

当$v(t)>V(t)$时,$T(v(t))>0$,忆阻器阻抗$G(t)$会发生变化,φ为忆阻器阻抗变化率.

1.2 忆阻器交叉阵列和记忆卷积层

令V_{in}为外围控制电路的输入(j为忆阻器交叉阵列的字线号数,V_{in}为j维向量). 根据基尔霍夫电流定律,电路中任一个节点上任一时刻,流入节点的电流之和等于流出节点的电流之和,即

$$L_{\text{out}} = V\lambda_{\text{in}}^{\text{T}} \times G. \tag{5}$$

令I为N维向量,G为$M \times N$的电导矩阵. 对于一个$M \times N$的忆阻器交叉阵列,$M \times N$维加法和乘法可同时进行,极大地提高了矩阵运算效率并且降低了功耗. 随着忆阻器交叉阵列规模的增加,计算量需求更大的矩阵和向量也可以得到满足. 记忆卷积层的计算主要是向量与矩阵的点乘运算,FC层可以看作是一种$H=1$、$W=1$的特殊卷积层. 记忆卷积层由多个卷积核组成,输出特征映射由卷积核的卷积结果和输入数据组成. 令第k次记忆卷积层的输入图像为$H\lambda_{\text{in}}^{k} \times W\lambda_{\text{in}}^{k} \times C\lambda_{\text{in}}^{k}$,卷积核为$n \times n$,步长为1,其中$n$为用户决定的参数. 当使用相同的填充模式时,输出特征映射为$H\lambda_{\text{out}}^{k} \times W\lambda_{\text{out}}^{k} \times C\lambda_{\text{out}}^{k}$. (i,j,c)输出特征可表示为

$$y_{i,j,c}^{k} = \sum_{m=-n/2}^{n/2}\sum_{n=-n/2}^{n/2}\sum_{k=0}^{C_{\text{in}}} I_{\text{in}}(i+m, j+n, k)\times W(m,n,k,c). \tag{6}$$

若i和j是不变的,则第c次卷积核W^c和输入数据$I_{\text{in}}^{i,j}$是不变的,可以在忆阻器交叉阵列位线中得到一个输出特征映射的值. 通过图1所示的忆阻器交叉阵列,需要$(2 \times N^2+1) \times C_{\text{in}}$个忆阻器交叉阵列才能在一个卷积计算周期内得到一个$1 \times 1 \times C_{\text{out}}$.

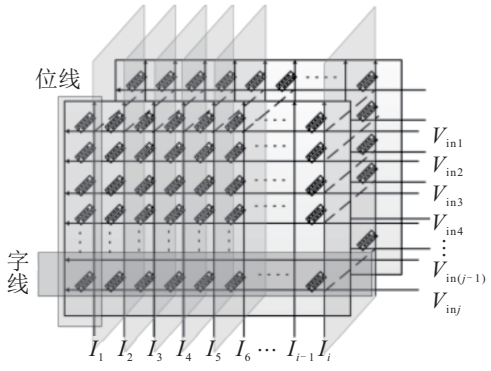


图 1　基于忆阻器crossbar的向量和矩阵点乘运算

1.3 忆阻卷积神经网络上的批量标准化

在深度神经网络中,批量标准化层是一种典型的处理层,主要为标准化每一层的输入. 理想的数据标准化公式如下:

$$\hat{x}_{\text{bn}}^{(k)} = \frac{x_{\text{bn}}^{(k)} - E[x_{\text{bn}}^{(k)}]}{\sqrt{\text{Var}[x_{\text{bn}}^{(k)}]}}. \tag{7}$$

数据标准化将影响下一层网络学习提取的特征,因此,BN层还包括数据重建功能,其输出为

$$\gamma_{\text{bn}}^{(k)} = \beta_{\text{bn}}^{(k)}\hat{x}_{\text{bn}}^{(k)} + \rho_{\text{bn}}^{(k)}, \tag{8}$$

其中$\beta_{\text{bn}}^{(k)}$和$\rho_{\text{bn}}^{(k)}$为本层的学习参数. 在反向传播网络中,BN层的参数包括$\widehat{E[x_{\text{bn}}^{(k)}]}$,$\widehat{D[x_{\text{bn}}^{(k)}]}$,分别是网络训练过程中所有数据的均值和方差的统计值,计算为

$$\widehat{E[x_{\text{bn}}^{(k)}]} = (1-\text{momentum}) \times \widehat{E[x_{\text{bn}}^{(k)}]} + \text{momentum} \times \widehat{E[x_{\text{bn}}^{(k)}]}_t. \tag{9}$$

$$\widehat{D[x_{\text{bn}}^{(k)}]} = (1-\text{momentum}) \times \widehat{D[x_{\text{bn}}^{(k)}]} + \text{momentum} \times \widehat{D[x_{\text{bn}}^{(k)}]}_t. \tag{10}$$

当momentum趋近于0,网络训练结束时,$\widehat{E[x_{\text{bn}}^{(k)}]}$和$\widehat{D[x_{\text{bn}}^{(k)}]}$是近似值,因此BN层的输出可以简化为输入$X_{\text{bn}}$的一个线性变换,即

$$Y_{\text{bn}} = M_{\text{bn}}X_{\text{bn}} + B_{\text{bn}}, \tag{11}$$

其中M_{bn}为对角矩阵. 若网络中BN层之后是CNN,显然$X_{\text{bn}}=Y_{\text{cnn}}$在CNN中,则输出$X_{\text{cnn}}$和输出$Y_{\text{cnn}}$的关系可表示为

$$Y_{\text{cnn}} = M_{\text{cnn}}X_{\text{cnn}} + B_{\text{cnn}}. \tag{12}$$

下一个网络中设momentum=0,有

$$Y_{\text{bn}} = M_{\text{bn}}M_{\text{cnn}}X_{\text{cnn}} + B_{\text{bn}}. \tag{13}$$

令$M_{\text{bn}}=M_{\text{bn}}M_{\text{cnn}}$,$B_{\text{bc}}=M_{\text{bn}}B_{\text{cnn}}B_{\text{bn}}$,则有

$$Y_{\text{bc}} = M_{\text{bc}}X_{\text{cnn}} + B_{\text{bc}}. \tag{14}$$

式(7)与(14)意义相同,因此只改变忆阻器交叉阵列的系数,便可使BN层和CNN层共用一个忆阻器交叉阵列. 当一个内核为N的CNN层后连接一个BN层时,具体电路如图2(a)所示,通过共用忆阻器减少运算放大器的使用,降低功耗.

1.4 池化操作

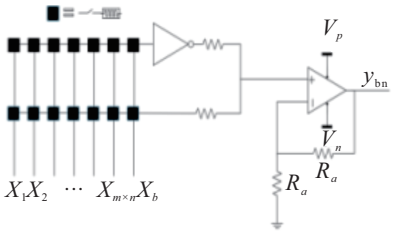
池化层将输入的特征映射分割为多个$N \times N$的单元进行最大和平均值的计算,进一步简化提取特征,因此提取的特征图像尺寸变为原来的$1/N^2$. 最大池化和平均池化是池化层的主要操作方式. 最大池化通过并联的两个nmos二极管选择输入中的最大电压,如图2(b)所示,平均池化一般通过加法电路实现,如图2(c)所示,计算为

$$V_{\text{avg}} = \frac{N \times N}{\sum_{i=1}^{N \times N} V_i}\left(1 + \frac{R_f}{R_a}\right). \tag{15}$$

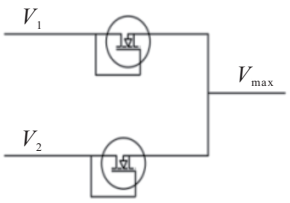
当$R_f \ll R_a$时,表示为

$$V_{\text{avg}} = \frac{N \times N}{\sum_{i=1}^{N \times N} V_i}. \tag{16}$$

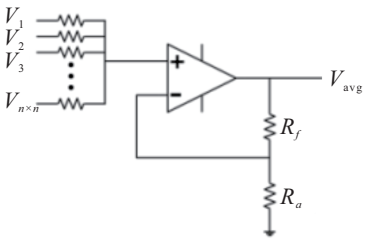
当图2(b)中输入数大于2时,通过多个上述结构组合完成.



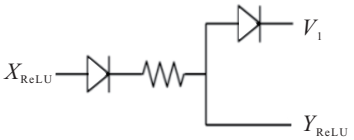
(a) 忆阻卷积神经网络下的BN层



(b) 最大池化电路



(c) 平均池化加法电路



(d) ReLU层电路

图 2 忆阻卷积神经网络下的电路设计

1.5 限流二极管下的ReLU

ReLU函数是人工神经网络中的典型激活函数[21]. 其计算简单,能够有效避免梯度消失和梯度爆炸的特性,在深度神经网络中得到了广泛应用. ReLU函数通过如下方式将线性输入非线性化:

$$F(x) = \max(0, M^{\text{T}}X + B). \tag{17}$$

ReLU的实现相当于单向限制电路,只允许电压大于1的信号通过,因此限流二极管非常适合担当ReLU电路的基础元件. 但ReLU层的输出可能会大于忆阻器的阈值,因此在ReLU后加上限制电路,实现方式如下:

$$F(x) = \min(\max(0, M^{\text{T}}X + B), V_l). \tag{18}$$

本文实现ReLU层选用限流二极管的另一个原因是功耗较低,在此基础上,只需加上1个上限二极管,实现如图2(d)所示. 当导通时电路存在压降,所以理论计算应加上二极管的偏置电压.

2 忆阻卷积神经网络的图像分类

2.1 概 述

忆阻卷积神经网络由5个卷积层和2个全连接层组成,其输入图像是尺寸为$32 \times 32 \times 3$的RGB色彩图像. 前4个卷积层每层之间由1个最大池化层连接,第5个卷积层后经过平均池化层输出到全连接层. 全连接层输出后通过递增函数softmax(x)层将分类网络结果x进行标准化映射到0-1. softmax(x)层输出值最大的通道代表输入图像的分类结果,如图3所示.

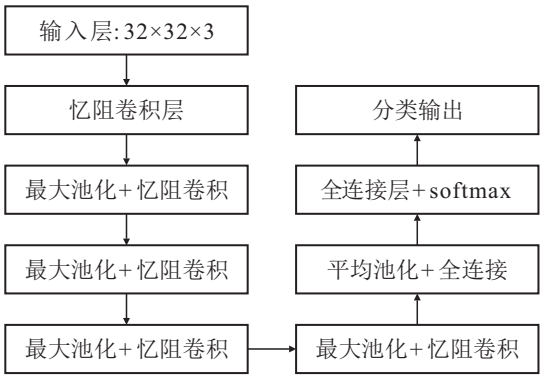


图 3 基于忆阻器神经网络的图像分类流程

2.2 忆阻卷积层

由文献[22],卷积层的参数可以分离为一个正系数矩阵K^+和一个负系数矩阵K^-,当$\sigma_b > 0, x_b > 0$时,有

$$V_{o1} = \sum^{N\times N} x_i\sigma_{i,j}^{+} + x_b\sigma_b, \tag{19}$$

$$V_{o2} = \sum^{N\times N} x_i\sigma_{i,j}^{-}, \tag{20}$$

$$\gamma_j = V_{o1} - V_{o2} = \sum^{N\times N} x_i\sigma_{i,j}^{+} - x_i\sigma_{i,j}^{-} + x_b\sigma_b. \tag{21}$$

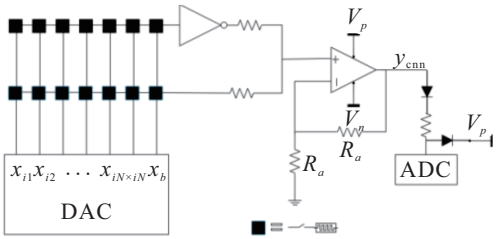
当$\sigma_b < 0, x_b > 0$时,有

$$V_{o1} = \sum^{N\times N} x_i\sigma_{i,j}^{-}, \tag{22}$$

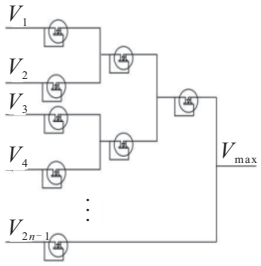
$$V_{o2} = \sum^{N\times N} x_i\sigma_{i,j}^{+} - x_b\sigma_b, \tag{23}$$

$$\gamma_j = V_{o1} - V_{o2} = \sum^{N\times N} x_i\sigma_{i,j}^{+} - x_i\sigma_{i,j}^{-} - x_b\sigma_b. \tag{24}$$

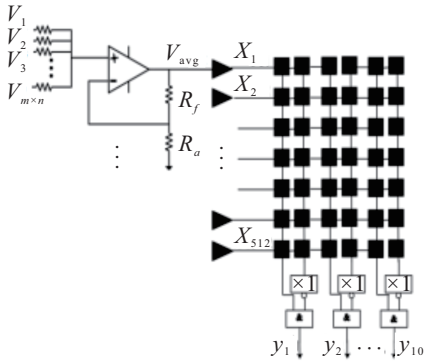
对于图像分类的第1层忆阻卷积层,卷积核的大小为$3 \times 3 \times 3$,输出通道数为32. 为了实现卷积核的计算需求,需要忆阻器交叉阵列的规模为$(3\times3\times3+1)\times32\times2$. 通过DAC将数据转换输入进忆阻器交叉阵列,经过BN层和ReLU处理后,通过ADC转换将数据送出,如图4(a)所示.



(a) BN和ReLU下的卷积层



(b) 卷积层的最大池化输入



(c) 平均池化下的全连接层

图 4　忆阻卷积层设计

2.3　最大池化和忆阻卷积层

在第1个忆阻卷积层输出后,经过最大池化层处理后的数据输入第2个忆阻卷积层,第2次忆阻卷积的卷积核尺寸为$3\times3\times3$,输出通道数为64,为了实现卷积核的计算需求,需要忆阻器交叉阵列的规模为$(3\times3\times32+1)\times64\times2$,如图4(b)所示.

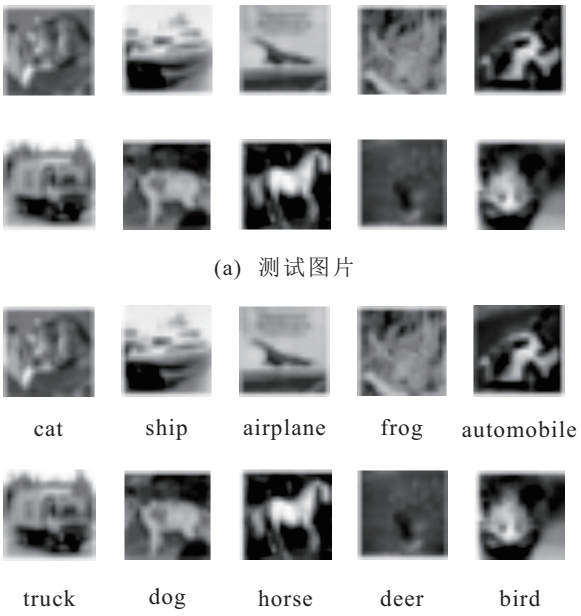
2.4　平均池化和全连接层

在第5个忆阻卷积层输出后,经过平均池化层处理后的数据输入全连接层. 将之前忆阻卷积层计算的$3\times3\times256$特征映射为$1\times1\times512$特征向量,需要平均池化电路实现. 为了提高计算速度,同时铺设多个平均池化电路,降低计算周期数以减少电路搭建,在一个周期内结束运算,提高整体运算速率,如图4(c)所示.

3　实验分析

3.1　精准度分析

目前,国内外学者根据忆阻器的电路特性,建立了大量完善的Matlab的仿真模型,本文仿真模型采用基于Matlab的忆阻器电路模型[23],能够有效模拟忆阻器的电学特性. 首先建立忆阻器的Matlab电路模型,然后对神经网络中的各层进行电路设计,完成忆阻卷积神经网络电路的构建. 采用文献[24]提出的方法对忆阻器网络参数进行编程. 输入尺寸为$32\times32\times3$的RGB图像,输出对该图像10种分类的预测. 采用CIFAR-10数据集,该数据集包含10种分类对象,每种图像数量为6 000张,共60 000张图像数据集. 实验采用每个分类5 000张,共50 000张图像进行训练,最后10 000张图像进行检验测试,准确率大于86%,如图5所示.



(a) 测试图片

(b) 电路预测结果

图 5　基于忆阻神经网络的图像识别分类结果

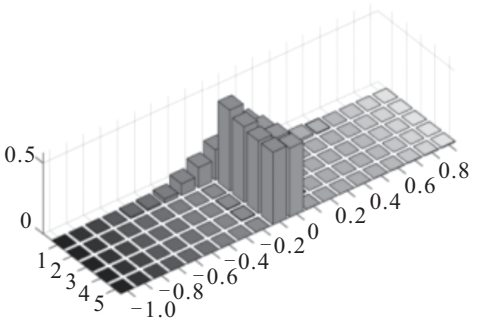
3.2　能耗分析

在忆阻卷积神经网络中有5个忆阻卷积层和2个全连接层. 从忆阻卷积神经网络的图像分类中可以得出卷积层的等效电导率分布,如图6(a)所示,并由等效电导率可以计算出忆阻卷积神经网络的功耗. 功耗主要由两部分组成,一是来源于写入电导率,二是预测过程产生功耗,上述两个过程组成了忆阻卷积神经网络工作的主要流程. 基于忆阻卷积神经网络的图像分类功耗为

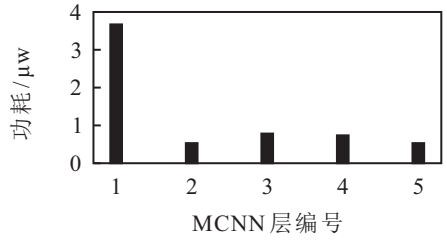
$$P=\frac{W}{t}=\frac{\int_{t_0}^{t_i}U_1^2G(t)\mathrm{d}t+\int_{t_i}^{t_r}U_2^2G\mathrm{d}t}{t_r-t_0}. \tag{25}$$

其中:U为通过单个忆阻器的电压,G为忆阻器的等效电导率,W为忆阻器消耗的功率. 忆阻器的$1/G_{\mathrm{on}}$和$1/G_{\mathrm{off}}$分别为1 kΩ和1 mΩ,因此忆阻器的等效电压电导率分别为10^{-3}和10^{-6}. 为了匹配网络的权重,网络权重和忆阻器电导率关系如下:

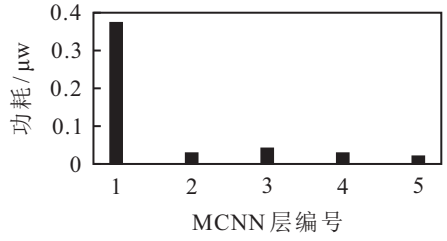
$$W_{ij}=1\,000\,G_{ij}. \tag{26}$$



(a) MCCN中忆阻器交叉阵列等效电导分布



(b) 预测过程功耗



(c) 写入过程功耗

图 6 MCCN中忆阻器能耗分析结果

令$\beta=0.6$,写入电压为$G(t)=6.5$ V,可得出每个忆阻卷积层的功耗. 如图6(b)所示,在导通阶段,忆阻卷积平均功耗为3.5 uW. 在网络的预测过程中,功耗随着输入数据的变化而变化,可以利用写入电压最大时的最大功耗表示,如图6(c)所示. 显然,由以上两个过程组成的忆阻卷积神经网络的主要功耗,远远小于CMOS电路下同样计算过程的功耗[25].

表1为忆阻卷积神经网络所需的忆阻器资源量,由于忆阻器的尺寸一般为方形,为了尽可能发挥忆阻器的优势并且合理利用资源,通常使忆阻器行和列的数目尽可能接近. 由表1可见,需要资源量最大的过程为第4个MCCN层,所以这里设计忆阻器交叉阵列规模为2 305 × 2 305便可以满足网络要求. 忆阻器的响应时间在100 ps以下,若按照每个计算周期100 ps计算,则设计电路所需的推理时间为136.5 ns. 与文献[26]相比,本文用多个最大池化电路在一个时钟周期内完成计算,将最大池化层的计算周期从512个时钟周期降至1个时钟周期,因此推理时间更少,电路速度更快. 在忆阻器的功耗方面,所设计电路的单个忆阻器的功耗略高于文献[26]的数据,但前者只需要5个忆阻阵列,后者需要17个忆阻阵列,在忆阻阵列的数量方面远小于文献[26]的结果. 忆阻阵列的数量直接决定了忆阻神经网络电路的总功耗和面积,忆阻阵列的数量越多,忆阻神经网络电路的体积越大,总功耗越高. 因此本文设计的电路在推理速度、体积和功耗方面与文献[26]相比均有优势.

表 1 图像分类中所需忆阻器交叉阵列规模

层	MCs规模	计算周期
MCNN	28 × 48	32 × 32
MCNN	289 × 128	16 × 16
MCNN	576 × 256	8 × 8
MCNN	1 153 × 512	4 × 4
MCNN	2 305 × 1 024	2 × 2
GAP	0	1
FC	2 × 512	1

4 结 论

本文基于新型计算存储材料忆阻器构建了忆阻神经网络电路用于图像识别,并成功对CIFAR-10图像进行了分类验证. 实验结果表明,忆阻卷积神经网络比传统计算构架以更高的实时性和更低的功耗进行图像分类,通过对各层网络电路的设计和优化,使计算构架更适合于当前的神经网络和所针对的目标分类. 通过优化将卷积网络运算所需的忆阻交叉阵列的行大小从$2M+1$减少至$M+1$,并对忆阻器交叉阵列系数进行更改,在BN层和CNN层共用1个忆阻器交叉阵列,替代运算放大器,同时在ReLU实现电路上通过二极管的限流大大降低功耗. 实际电路的特征与目前Matlab的仿真模型存在一定的差异,在后续实验中将进一步对仿真模型进行优化. 同时将开展忆阻神经网络电路的设计和验证工作,以表明本文仿真电路的有效性.

参考文献(References)

[1] Bahrami M. Cloud computing for emerging mobile cloud apps[C]. The 3rd IEEE International Conference on Mobile Cloud Computing, Services, and Engineering. Piscataway: IEEE, 2015: 4-5.

[2] Mueller H, Gogouvitis S V, Haitof H, et al. Poster abstract: Continuous computing from cloud to edge[C]. IEEE/ACM Symposium on Edge Computing. Piscataway: IEEE, 2016: 97-98.

[3] Mukherjee M, Matam R, Mavromoustakis C X, et al. Intelligent edge computing: Security and privacy challenges[J]. IEEE Communications Magazine, 2020, 58(9): 26-31.

[4] Iandola F N, Han S, Moskewicz M W, et al. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5 MB model size[J/OL]. 2016, arXiv: 1602.07360.

[5] Szegedy C, Liu W, Jia Y Q, et al. Going deeper with convolutions[C]. IEEE Conference on Computer Vision

and Pattern Recognition. Piscataway: IEEE, 2015: 1-9.

[6] Dai J F, Qi H Z, Xiong Y W, et al. Deformable convolutional networks[C]. IEEE International Conference on Computer Vision. Piscataway: IEEE, 2017: 764-773.

[7] Zeiler M D, Fergus R. Visualizing and understanding convolutional networks[M]. Computer Vision — ECCV 2014. Cham: Springer International Publishing, 2014: 818-833.

[8] Bazarevsky V, Kartynnik Y, Vakunov A, et al. BlazeFace: Sub-millisecond neural face detection on mobile GPUs[J/OL]. 2019, arXiv: 1907.05047.

[9] Krizhevsky A, Sutskever I, Hinton G E. Imagenet classification with deep convolutional neural networks[C]. International Conference on Neural Information Processing Systems. Nevada, 2012: 1097-1105.

[10] Srivastava N, Hinton G, Krizhevsky A, et al. Dropout: A simple way to prevent neural networks from overfitting[J]. Journal of Machine Learning Research, 2014, 15: 1929-1958.

[11] 王洋昊, 刘昌, 黄如, 等. 神经形态器件研究进展与未来趋势[J]. 科学通报, 2020, 65(10): 904-915.
(Wang Y H, Liu C, Huang R, et al. Progresses and outlook in neuromorphic devices[J]. Chinese Science Bulletin, 2020, 65(10): 904-915.)

[12] 杨辉, 段书凯, 董哲康, 等. 基于忆阻器-CMOS的通用逻辑电路及其应用[J]. 中国科学: 信息科学, 2020, 50(2): 289-302.
(Yang H, Duan S K, Dong Z K, et al. A memristor-CMOS-based general-logic circuit and its applications[J]. Scientia Sinica: Informationis, 2020, 50(2): 289-302.)

[13] Chua L O, Kang S M. Memristive devices and systems[J]. Proceedings of the IEEE, 1976, 64(2): 209-223.

[14] 王春华, 蔺海荣, 孙晶如, 等. 基于忆阻器的混沌、存储器及神经网络电路研究进展[J]. 电子与信息学报, 2020, 42(4): 795-810.
(Wang C H, Lin H R, Sun J R, et al. Research progress on chaos, memory and neural network circuits based on memristor[J]. Journal of Electronics & Information Technology, 2020, 42(4): 795-810.)

[15] Bala A, Yang X H, Adeyemo A, et al. A memristive activation circuit for deep learning neural networks[C]. The 8th International Symposium on Embedded Computing and System Design. Piscataway: IEEE, 2018: 1-5.

[16] Wen S P, Wei H Q, Yan Z, et al. Memristor-based design of sparse compact convolutional neural network[J]. IEEE Transactions on Network Science and Engineering, 2020, 7(3): 1431-1440.

[17] Liu H J, Sun S Y, Liu J J, et al. Binary memristive synapse based vector neural network architecture and its application[J]. IEEE Transactions on Circuits and Systems II: Express Briefs, 2021, 68(2): 772-776.

[18] Chen J D, Wu Y C, Yang Y, et al. An efficient memristor-based circuit implementation of squeeze-and-excitation fully convolutional neural networks[J]. IEEE Transactions on Neural Networks and Learning Systems, 2021, DOI: 10.1109/TNNLS.2020.3044047.

[19] Bao G, Zhang Y D, Zeng Z G. Memory analysis for memristors and memristive recurrent neural networks[J]. IEEE/CAA Journal of Automatica Sinica, 2020, 7(1): 96-105.

[20] Wan G L, Wang L M, Zou H Y, et al. A new model of associative memory neural network based on an improved memristor[C]. The 39th Chinese Control Conference. Shenyang, 2020: 7589-7994.

[21] 范丽丽, 赵宏伟, 赵浩宇, 等. 基于深度卷积神经网络的目标检测研究综述[J]. 光学精密工程, 2020, 28(5): 1152-1164.
(Fan L L, Zhao H W, Zhao H Y, et al. Survey of target detection based on deep convolutional neural networks[J]. Optics and Precision Engineering, 2020, 28(5): 1152-1164.)

[22] Hassan A M, Li H H, Chen Y R. Hardware implementation of echo state networks using memristor double crossbar arrays[C]. International Joint Conference on Neural Networks. Piscataway: IEEE, 2017: 2171-2177.

[23] Jo S H, Chang T, Ebong I, et al. Nanoscale memristor device as synapse in neuromorphic systems[J]. Nano Letters, 2010, 10(4): 1297-1301.

[24] Yakopcic C, Alom M Z, Taha T M. Extremely parallel memristor crossbar architecture for convolutional neural network implementation[C]. International Joint Conference on Neural Networks. Piscataway: IEEE, 2017: 1696-1703.

[25] Kim S, Choi B, Lim M, et al. Pattern recognition using carbon nanotube synaptic transistors with an adjustable weight update protocol[J]. ACS Nano, 2017, 11(3): 2814-2822.

[26] Ran H H, Wen S P, Wang S Q, et al. Memristor-based edge computing of ShuffleNetV2 for image classification[J]. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2021, 40(8): 1701-1710.

作者简介

罗佳(1981–), 女, 教授, 从事计算机图像识别等研究, E-mail: luojia_dou@163.com;

冉欢欢(1987–), 女, 副教授, 博士, 从事边缘智能的研究, E-mail: 364483391@qq.com;

何凯霖(1982–), 男, 副教授, 从事图像处理的研究, E-mail: 20788470@qq.com;

丁晓峰(1985–), 男, 副教授, 从事图像处理的研究, E-mail: 419416215@qq.com.

(责任编辑: 魏 冰)