SCIENCE CHINA
Information Sciences

• LETTER •

January 2022, Vol. 65 119104:1–119104:3
https://doi.org/10.1007/s11432-020-3031-9

Quantized and adaptive memristor based CNN (QA-mCNN) for image processing

Xiaofang HU[1,3], Wenqiang SHI[2], Yue ZHOU[4], Hongan TANG[5] & Shukai DUAN[1*]

[1]*College of Artificial Intelligence, Southwest University, Chongqing* 400715*, China;*
[2]*College of Computer and Information Science, Southwest University, Chongqing* 400715*, China;*
[3]*School of Mathematics and Statistics, Southwest University, Chongqing* 400715*, China;*
[4]*College of Electronics and Information Engineering, Southwest University, Chongqing* 400715*, China;*
[5]*School of Artificial Intelligence, Chongqing University of Technology, Chongqing* 401135*, China*

Received 29 February 2020/Revised 8 May 2020/Accepted 11 August 2020/Published online 20 May 2021

Citation Hu X F, Shi W Q, Zhou Y, et al. Quantized and adaptive memristor based CNN (QA-mCNN) for image processing. Sci China Inf Sci, 2022, 65(1): 119104, https://doi.org/10.1007/s11432-020-3031-9

Dear editor,

The cellular neural network (CNN) is one of the most hardware-implementable neural networks with promising prospects on large-scale and real-time problem solving [1]. However, the lack of adaptive templates and advanced implementation technology has hindered its developments and practical applications. In order to solve the problem that traditional templates of CNN are not adaptive, a series of adaptive templates have been proposed [2, 3]. Nevertheless, these template elements are usually highly non-linear or complicated decimal fractions, which greatly increase CNN's computational and circuit complexity.

Therefore, it is desired to develop a CNN with both better adaptability and circuit realization feasibility for wide practical applications, especially in end-side intelligent image processing. In this study, a soft-hardware co-design along with incremental network quantized (INQ) and adaptive memristor-based CNN (QA-mCNN) is proposed to solve these dilemmas.

In a traditional CNN circuit, the weight and weighting operation are achieved through several amplifiers and multipliers that are usually fixed once the circuit is built and needs complex operations. Moreover, the multiplication processing is always non-linear. Therefore, a proper artificial synapse needs to be developed. Recently, there have been considerable interests in memristor-based synapses [4, 5]. And the synaptic circuit based on memristor crossbar array is adopted to implement template parameters or weights of CNN (mCNN).

Based on the KCL principle, the dynamics of a cell $c(i,j)$ in mCNN (1) and the output function (2) can be written as

$$\frac{\mathrm{d}x_{ij}(t)}{\mathrm{d}t} = -\, m(x_{ij}(t)) + \sum_{c(k,l)} (a_{ij,kl}y_{kl}(t) + b_{ij,kl}u_{kl}(t)) + I, \tag{1}$$

$$y_{ij}(t) = f(x_{ij}) = \frac{1}{2}(|x_{ij}(t) - 1| - |x_{ij}(t) + 1|), \tag{2}$$

where $1 \leqslant i \leqslant M, 1 \leqslant j \leqslant N$, $c(k,l)$ denotes the neighbors of $C(i,j)$ in r-neighborhood and $r = 1$, $a_{ij,kl}$ denotes the element of feedback template A, and $b_{ij,kl}$ denotes the element of control template B. The term $m(x_{ij}(t))$ represents the current through the state memristor:

$$m(x_{ij}(t)) = \frac{v_m}{M(t)} = \frac{x_{ij}(t)}{M(t)}, \tag{3}$$

where $M(t)$ and v_m represent the memristance and voltage of the state memristor, respectively. To facilitate computation and simulations, the flux-controlled memristor model [6] deduced from the typical HP memristor model is employed.

Following the introduction of mCNN, the hardware-friendly adaptive template design scheme can be constructed.

As described in (1), the gain of a cell in mCNN is determined by the templates (A, B) and the bias (I). When $r = 1$, these templates and bias can be abbreviated as the parameters of the optimization algorithm, and the superior adaptive template can be generated. But these template's parameters are all floating-point data that may increase the computational complexity, storage capacity and accuracy requirements. Certainly, these templates are not suitable for the memristor-based hardware implementation because of the limited memristor accuracy. Therefore, proper quantization that can reduce computational complexity and guarantee required accuracy is quite necessary.

In [7], a quantization scheme was proposed, where the 32-bit floating-point parameters can be quantized into l-bit (2^l) integers and multiplication is transferred to binary shift operation. In this paper, one memristor synapse can represent a multiple bit on account of the memristor is a multi-level memory. In addition, the multiplication operation can be carried out in a more effective way, that is, Ohm's law without the needs of switching circuits and binary displace-

* Corresponding author (email: duansk@swu.edu.cn)

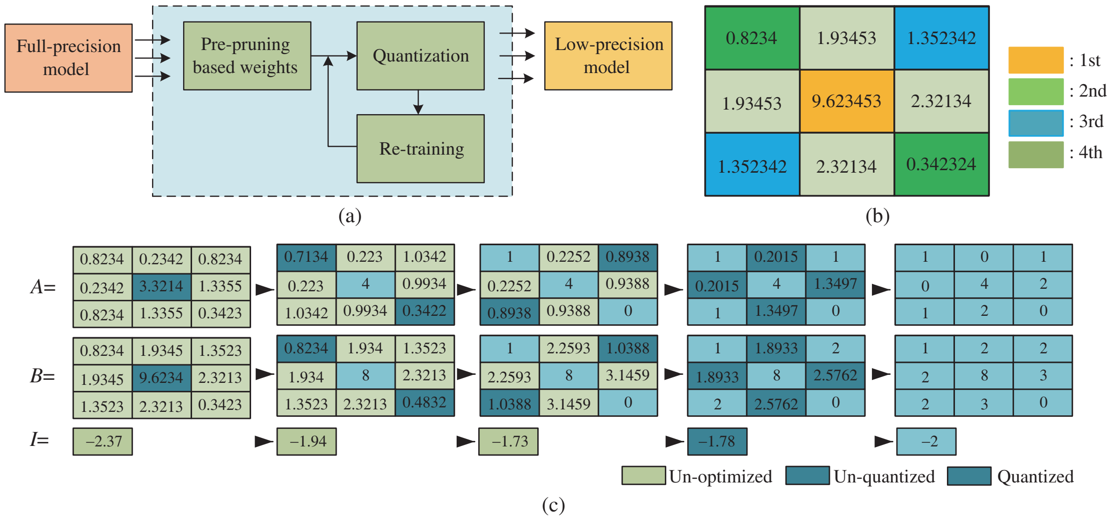

Figure 1 (Color online) (a) Framework of the proposed INQ, (b) quantization order determined by pre-pruning strategy, (c) quantization process.

ment operations, saving resource consumption and improving real-time performance.

The original logic shift can be omitted and the quantization formula can be got:

$$\bar{z} = \begin{cases} \mathrm{sgn}(z) \times 2^b, & \mathrm{abs}(z) \geqslant 2^b, \\ \mathrm{sgn}(z) \times \mathrm{abs}(\lceil z \rceil), & b < z < a, \\ 0, & \mathrm{abs}(z) < 2^{a-1}, \end{cases} \tag{4}$$

where z and $\bar{z}$ denote the generated template parameters from mPSO algorithm and the corresponding quantized template parameters, respectively. $\mathrm{abs}(\lceil z \rceil)$ represents the absolute value after rounding up z. The non-zero elements (z) are constrained in the range of $[2^a, 2^b]$ and $[-2^b, -2^a]$, where b and a are upper/lower boundaries in the processes of quantization. And $b - a$ denotes the expected bit-width of the quantization.

The pure quantization process may, however, cause large accuracy loss and even lead to distortion of template properties. As shown in Figure 1(a), in order to reduce the accuracy loss and save the computation complexity. The concept of INQ for mCNN is adopted. This scheme uses pre-pruning strategy to partition weights in pre-trained templates. To be specific, the parameters with bigger values mean more important and will be quantized prior [8], while the parameters with smaller values will remain to be retrained. In Figure 1(b), an edge extraction template can be divided into 4 parts (quantization steps).

Subsequently, the process of INQ is executed as Figure 1(c). The yellow rectangles represent the weights to be retrained, the dark blue rectangles represent the weights to be quantized, and the light blue rectangles denote quantized weights. The first to the fifth column respectively denote the full-precision model of templates, the results after one-step, the second and the third quantization, and the final quantized result which can be called the low-precision model, also.

In addition, to further improve the accuracy and robustness of the proposed scheme, the non-linear template can be utilized as a compensation strategy. By considering the difficulty in hardware implementation of the non-linear functions, this paper uses multilayer linear templates to implement the non-linear template based on the Turing complete of the CNN [9]. Supposing there are S layers of QA-mCNN templates, S can be controlled as a hyper-parameter to prevent over-fitting and improve real-time performance. If N denotes the iteration numbers of quantization per linear template, the framework of multilayer linear templates QA-mCNN can be illustrated as

$$\begin{aligned} & y_{ij(n)} = Y(u_{ij(n)}, y_{ij(n-1)}, A_{ij(n)}, B_{ij(n)}, I_{ij(n)}), 1 \leqslant n \leqslant S, \\ & \text{s.t.} \ \ A_{ij(n)} \in \hat{A}_{ij}, \ B_{ij(n)} \in \hat{B}_{ij} \in B_{ij}, \ I_{ij(n)} \in \hat{I}_{ij}, \end{aligned} \tag{5}$$

where Y represents the non-linear output function of mCNN in (1), $A_{ij(n)}$, $B_{ij(n)}$, and $I_{ij(n)}$ represent the feedback template, control template and current bias I in the n-th layer, respectively. $\hat{A}_{ij}$ and $\hat{B}_{ij}$ are two non-linear templates together with bias $\hat{I}_{ij}$. It should be noted that in the proposed QA-mCNN, each of the used template is a quantized linear template that functionally belongs to the non-linear templates $\hat{A}_{ij(n)}$ and $\hat{B}_{ij(n)}$.

Conclusion. In summary, by using the hardware-friendly template design scheme INQ, low-precision still fairly accuracy quantized adaptive templates can be obtained. Then along with the realization scheme of the non-linear template with multilayer linear templates, QA-mCNN can integrate with the merits of low complexity, strong adaptability, high robustness and excellent performance.

Acknowledgements The work was supported by National Natural Science Foundation of China (Grant Nos. 61976246, 61601376), Natural Science Foundation of Chongqing (Grant No. cstc2020jcyj-msxm0016), National Key R&D Program of China (Grant No. 2018YFB1306604), China Postdoctoral Science Foundation Special Funded (Grant No. 2018T110937), Innovation Support Program for Chongqing Overseas Returnees (Grant No. cx2019126), and Zhejiang Provincial Key Lab of Equipment Electronics (Grant No. 2019E10009).

Supporting information Appendixes A–D. The supporting information is available online at info.scichina.com and link.springer.com. The supporting materials are published as submitted, without typesetting or editing. The responsibility for scientific accuracy and content remains entirely with the authors.

References

1 Leo M, Sousa L. A programmable cellular neural network circuit. In: Proceedings of the 17th Symposium on Inte-

grated Circuits and Systems Design, 2004. 186–191
2 Dong Z K, Yang T, Wang L D, et al. Color image edge extraction using memristor-based CNN. Sci Sin Inform, 2017, 7: 863–877
3 Wang J, Long F, Ou W H. CNN-based color image encryption algorithm using DNA sequence operations. In: Proceedings of International Conference on Security, Pattern Analysis, and Cybernetics, Shenzhen, 2017. 730–736
4 Yang C J, Adhikari S P, Kim H. Excitatory and inhibitory actions of a memristor bridge synapse. Sci China Inf Sci, 2018, 61: 060427
5 Hu X F, Feng G, Duan S K, et al. A memristive multilayer cellular neural network with applications to image processing. IEEE Trans Neural Netw Learn Syst, 2017, 28: 1889–1901
6 Wang L, Drakakis E, Duan S K, et al. Memristor model and its application for Chaos generation. Int J Bifurcation Chaos, 2012, 22: 1250205
7 Liu Z, Zhuo C, Xu X. Efficient segmentation method using quantised and non-linear CeNN for breast tumour classification. Electron Lett, 2018, 54: 737–738
8 Han S, Pool J, Tran J, et al. Learning both weights and connections for efficient neural network. Adv Neural Inf Process Syst, 2015, 28: 1135–1143
9 Horváth A, Hillmer M, Lou Q, et al. Cellular neural network friendly convolutional neural networks — CNNs with CNNs. In: Proceedings of Design, Automation & Test in Europe Conference & Exhibition (DATE), Lausanne, 2017. 145–150