I-5-01 (Invited) Extended Abstracts of the 2020 International Conference on Solid State Devices and Materials, VIRTUAL conference, 2020, pp627-628

Programming Artificial Intelligence: A Reconfigurable AI shield for Embedded Microcontrollers

Tetsuya Asai and Hiroshi Momose

Faculty of Information Science and Technology, Hokkaido University.
Kita 14, Nishi 9, Kita-ku, Sapporo 060-0814, Japan
Phone: +81-11-706-6080 E-mail: {asai, hiroshi.momose}@ist.hokudai.ac.jp

Abstract

Artificial Intelligence (AI) has become the undisputed protagonist of most technological development projects. To develop AI, a learning (training) phase is fundamental, in fact, it is the moment in which the artificial intelligence system learns to repeat precise operations in the presence of certain inputs, just as a student at school learns to give the correct answers to the questions that are asked. The longer and more detailed the training will be, the more situations and requests (inference) that the AI will be able to manage in an appropriate manner. Obviously, all this requires a great computing power that not everyone can access. A possible candidate could be Arduino since many less experienced users are able to manage and program it, but it turns out to be a platform that is not powerful enough to support these operations. For this reason, we have created a field-programmable gateway array shield for artificial intelligence (FPGA2I shield) as an accelerator for Arduino, and will introduce it here. The board is able to guarantee a great computing power in all the phases of programming artificial intelligence and is also totally modifiable and reconfigurable at will.

In the past few years, deep learning has contributed to significant progress and success for various AI applications, including image recognition [2], [3], natural language processing [4], and reinforcement learning [5]. However, such applications have been built mainly within the cloud-based domain accelerated by powerful TPUs [6] and GPUs more than hundreds of watts of power. By contrast, in an edge domain with applications including smartphones, drones, and other smart products, smaller AI accelerators or IPs have recently started to be utilized or embedded as inference engines; *e.g.*, Edge TPU, Jetson Nano, Myriad, A12 Bionic and Kirin970, etc. However, these can be recognized simply as scaled-down versions from the cloud domain to the edge domain, leaving their learning capability on the cloud. As a result, they provide few originalities owing to a lack of learning ability to determine the features and behaviors of humans and their environment on-site, and few capabilities to keep personal privacy. To make the AI in the edge domain more attractive and fruitful, two key issues are considerably important and need to be solved.

First, from the viewpoint of technical issues, low-power and low-resource AI devices utilizing an online learning algorithm are required. To develop an effective technique, various algorithms of bit-width quantization and a sparsification of the weights and neurons have been studied [7], [8], [9], [10], [11], [12], [13]. Among them, ternary quantization [14], [15], [1] has recently become more attractive as a way to provide the best balance between a reduction of power and the resources of the AI engine while maintaining the quality during the inference and training stages. Meanwhile, for training at the edge, several papers have been published [16], [17], [18], [19]. In [16] and [18], a larger bit-width of more than 8 bits has been considered. In addition, the authors of [17] demonstrated a flexible architecture with a bit width ranging from 16 bits down to 1 bit. However, the architecture could not be optimized for a lower bit width. In [14] and [15], the authors described a ternary quantization algorithm using several techniques and presented a state-of-the-art results without any degradation in the recognition accuracy. However, its target is to achieve a higher accuracy, which differs from our approach, which places the priority on both lowest-power and lowest-resource devices.

Second, from the viewpoint of platform development, an AI open-innovation platform is required, upon which anyone can touch, enjoy, build, and use their own AI systems. The objective of the platform is exploring "killer applications" that can inspire activities and stimulate more creativity over various research and development scenes. Several large projects on open-innovation platforms have been promoted using specialized hardware [20], [21], [22] and commonly used software, including Tensorflow, PyNN, and etc., and collaborated with partners in industry, academia, and private individuals (makers). A do-it-yourself AI project (AIY), started in around 2017 with their software frame-work, TensorFlow, connected edge devices to the cloud, and allowed collaboration with "*makers*" who are individuals familiar with hardware prototyping and basic programming, and have the passion and potential to create their own AI systems based on their knowledge and ideas. In addition, an AI democratization movement [23] started in 2016-2017 and studied the effectiveness and importance of the collaboration with the makers who are not familiar with the AI and released the maker kit for them in 2018. Some projects have started using brain-inspired chips [20], [21], [22], collaborating mainly with academic institutions, and have been challenged to explore new AI systems and create new applications. In [20], the authors targeted low-power applications including IoTs using their low-power inference engine, TrueNorth. In [21] and [22], the researchers challenged the new AI exploration using Loihi [22] and its STDP (spike-timing dependent plasticity) learning mechanism.

Motivated by the above observations, we adapted and implemented a backpropagation algorithm to a reconfigurable hardware platform of the FPGA as an AI accelerator. Using this, we built an FPGA AI accelerator shield (FPGA2I shield), which has external memory on the shield for mainly storing the weights of neural networks and an interface compatibility with a micro controller (Arduino). Our system is composed of the FPGA2I shield, micro controller, and various interface shields like sensors and actuators. This system was evaluated, analyzed, and verified by not only us but *makers* through our user-driven open-innovation AI platform with AI software including basic and various example applications. For the details, please visit our website: https://fpga2i.org.

Acknowledgements

This work is based on results obtained from a project commissioned by the New Energy and Industrial Technology Development Organization (NEDO)..

References

[1] T. Kaneko, K. Orimo, I. Hida, T. Asai, et al, "A study on a low power optimization algorithm for an edge-AI device," NOLTA J., Vol. E10N, No. 4, pp. 373-389, 2019.

[2] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg and L. Fei-Fei, "ImageNet Large Scale Visual Recognition Challenge," International Journal of Computer Vision, Vol. 115, No. 3, pp. 211 – 252, Dec. 2015.

[3] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," arXiv preprint, arXiv:1709.01507, 2017.

[4] J. Devlin, M. Chang, K. Lee and L. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," arXiv preprint arXiv:1810.04805v2, 2018.

[5] D. Silver, J. Schrittwieser, K. Simonyan, I. Antonoglou, A. Huangand, D. Hassabis, et al., "Mastering the game of Go without human knowledge," Nature, Vol. 550, 354 – 359, Oct. 2017.

[6] N. Jouppi, C. Young, N. Patil, D. Patterson, G. Agrawal and D. Yoon, et. al., "In-Datacenter Performance Analysis of a Tensor Processing Unit," Proc. the 44th Annual International Symposium on Computer Architecture (ISCA), pp.1 – 12, Jun. 2017.

[7] Song Han, Huizi Mao and William J. Dally, "Deep Compression: Compressing Deep Neural Networks with Pruning, Trained Quantization and Huffman Coding", arXiv preprint arXiv:1510.00149, 2015.

[8] B. Moons, R. Uytterhoeven, W. Dehaene, M. Verhelst, "ENVISION: A 0.26-to-10TOPS/W Subword-Parallel Dynamic-Voltage-Accuracy- Frequency-Scalable Convolutional Neural Network Processor in 28nm FDSOI," Proc. 2017 IEEE International Solid-State Circuits Conference (ISSCC), Session 14.5, p246 – 248, 2017.

[9] M. Courbariaux, I. Hubara, D. Soudry, R. El-Yaniv, Y. Bengio, "Bina- rized Neural Networks: Training Deep Neural Networks with Weights and Activations Constrained to +1 or -1," arXiv preprint, arXiv: 1602.02830, 2016.

[10] M. Rastegari, V. Ordonez, J. Redmon and A. Farhadi, "XNOR-Net: ImageNet Classification Using Binary Convolutional Neural Networks," arXiv preprint, arXiv: 1603.05279, 2016.

[11] D. Miyashita, E. H. Lee and B. Murmann, "Convolutional Neural Networks using Logarithmic Data Representation," arXiv preprint, arXiv: 1603.01025, 2016.

[12] H. Yonekawa and H. Nakahara, "On-chip memory based binarized convolutional deep neural network applying batch normalization free technique on an FPGA," in Proc. IEEE Int. Parallel Distrib. Process. Symp. Workshops (IPDPSW), May 2017, pp. 98 – 105.

[13] S. Zhou, Y. Wu, Z. Ni, X. Zhou, H. Wen and Y. Zou, "Dorefa-net: Training low bitwidth convolutional neural networks with low bitwidth gradients," arXiv preprent, arXiv:1606.06160, 2016

[14] J. Choi, P. Chuang, Z. Wang, S. Venkataramani, V. Srinivasan and K. Gopalakrishnan, "Bridging the Accuracy Gap for 2-bit Quantized Neural Networks (QNN)," arXiv preprint, arXiv:1807.06964v1, 2018.

[15] D. Zhang, J. Yang, D.i Ye and G. Hua, "LQ-Nets: Learned Quantization for Highly Accurate and Compact Deep Neural Networks," arXiv preprent, arXiv:1807.10029v1, 2018.

[16] J. Lee, J. Lee, D. Han, J. Lee, G. Park and H. Yoo, "LNPU: A 25.3TFLOPS/W Sparse Deep-Neural-Network Learning Processor with Fine-Grained Mixed Precision of FP8-FP16," Proc. 2019 IEEE International Solid-State Circuits Conference (ISSCC), Session 7.7, pp.142 – 144, 2019.

[17] B. Fleischer, S. Shukla, M. Ziegler, J. Silberman, J. Ohand, K.Gopalakrishnnan, et. al., "A Scalable Multi-TeraOPS Deep Learning Processor Core for AI Training and Inference," Proc. 2018 Symposium on VLSI Circuits Digest of Technical Papers, C4 – 2, pp. C35 – 36, 2018.

[18] D. Han, J. Lee, J. Lee and H. J. Yoo, "A 1.32 TOPS/W Energy Efficient Deep Neural Network Learning Processor with Direct Feedback Alignment based Heterogeneous Core Architecture," Proc. 2019 Symposium on VLSI Circuits Digest of Technical Papers, C24 – 3, pp. C304 – 305, (2019).

[19] C. Kim, S. Kang, D. Shin, S. Choi, Y. Kim and H. Yoo, "A 2.1TFLOPS/W Mobile Deep RL Accelerator with Transposable PE Array and Experience Compression," Proc. 2019 IEEE International Solid-State Circuits Conference (ISSCC), Session 7.4, pp. 136 –138, 2019.

[20] P. Merolla, J. Arthur, R. Alvarez-Icaza, A. S. Cassidy, J. Sawada, N. Imam, S. K. Esser, M. D. Flickner, D. S. Modha, "A million spiking neuron integrated circuit with a scalable communication network and interface," Science, Vol. 345, No. 6197, pp. 668 – 673, Aug. 2014.

[21] HBP Neuromorphic Computing Platform Guidebook, "Using the SpiNNaker System," Human Brain Project, https://electronicvisions.github.io/hbp-sp9-guidebook/mc/usingspiNNaker.html

[22] M. Davies, N. Srinivasa, T. Lin, G. Chinya, Y. Cao and H. Wang, et. al., "Loihi: A Neuromorphic Manycore Processor with On-Chip Learning," IEEE Micro, Volume 38, No. 1, pp. 82 – 99, Jan. 2018.

[23] V. Dibia, A. Cox and J. Weisz, "Designing for Democratization: Introducing Novices to Artificial Intelligence Via Maker Kits," arXiv preprent, arXiv:1805.10723v3, 2018.