Proposal

Hongxin Kong
Department of Electrical and Computer Engineering
Texas A&M University
College Station, USA
konghongxin911@tamu.edu

Lang Feng
Department of Electrical and Computer Engineering
Texas A&M University
College Station, USA
flwave@tamu.edu

I. Background

Recent years, with high performance hardware and software emerging, machine learning approaches become more practical to implement. A typical area of machine learning, neural network, becomes a popular approach for doing various of machine learning tasks. With the demand of high performance and low overhead neural networks, sometimes software-based neural networks cannot satisfy the requirements, thus, many researches focus on implementing neural networks by hardware [1].

Field-programmable gate array (FPGA) is a prevalent platform for hardware implemetation. Neural networks which are based on FPGA are also proposed by many researchers [2]. Some works like [3, 4] proposed FPGA-based neural network implementations and have advantages over normal software-based implementations. However, there are still many spaces for improving current FPGA-based neural network architectures. We plan to improve the neural network architecture based on the work [3].

II. Design Plan

For many existing hardware-based neural networks, a fixed circuit is used for a specific neural network architectures. This will damage the flexibility of the hardware circuits. For example, in [3], the weights of the neural network is hard-coded and cannot be changed. We plan to implement a general hardware architecture that can be used to do training and inference for any kinds of neural networks. That means, on different hardware platform, the only difference is the size of such architecture, which will affect the training or inference latency. When given the size of such architecture, any kinds of neural network tasks can be finished. Since different neural networks have different structures, weights, etc., a ROM/RAM will be used in our design, which is used by users to store their settings for the neural networks. We plan to implement the design on FPGA.

III. Expected Outcome

Our design is expected to work for any neural network sturctures. Users can input their expected neural network structures by inputing some data with specific format into ROM/RAM. The neural networks can then be trained by the training data, and users can use them later. The expected speed of our design is similar to work [3], but has higher flexibility than it.

References

[1] Y. Liao, “Neural networks in hardware: A survey.”

[2] A. R Omondi and J. C Rajapakse, *FPGA Implementations of Neural Networks*. 2006.

[3] T. V. Huynh, “Deep neural network accelerator based on fpga,” in *2017 4th NAFOSTED Conference on Information and Computer Science*, pp. 254–257, 2017.

[4] S. B. Yun, Y. J. Kim, S. S. Dong, and C. H. Lee, “Hardware implementation of neural network with expansible and reconfigurable architecture,” in *Proceedings of the 9th International Conference on Neural Information Processing, 2002. ICONIP ’02.*, pp. 970–975 vol.2, 2002.