Systolic-CNN: An OpenCL-defined Scalable Run-time-flexible FPGA Accelerator Architecture for Accelerating Convolutional Neural Network Inference in Cloud/Edge Computing

Akshay Dua
Arizona State University
Email: adua5@asu.edu

Yixing Li
Arizona State University
Email: yixingli@asu.edu

Fengbo Ren
Arizona State University
Email: renfengbo@asu.edu

***Abstract*—This paper presents Systolic-CNN, an OpenCL-defined scalable, run-time-flexible FPGA accelerator architecture, optimized for performing the low-latency, energy-efficient inference of various convolutional neural networks (CNNs) in the context of multi-tenancy cloud/edge computing. Systolic-CNN adopts a highly pipelined and parallelized 1-D systolic array architecture, which efficiently explores both spatial and temporal parallelism for accelerating CNN inference on FPGAs. Systolic-CNN is highly scalable and parameterized, which can be easily adapted by users to achieve 100% utilization of the coarse-grained computation resources (i.e., DSP blocks) for a given FPGA. In addition, Systolic-CNN is run-time-flexible, which can be time-shared, in the context of multi-tenancy cloud or edge computing, to accelerate a variety of CNN models at run time without the need of recompiling the FPGA kernel hardware nor reprogramming the FPGA. The experiment results based on an Intel Arria 10 GX FPGA Development board show that Systolic-CNN, when mapped with a single-precision data format, can achieve 100% utilization of the DSP block resource and an average inference latency of 10ms, 84ms, 1615ms, and 990ms per image for accelerating AlexNet, ResNet-50, RetinaNet, and Light-weight RetinaNet, respectively. The peak computational throughput is measured at 80-170 GFLOPS/s across the acceleration of different CNN models. Codes are available at https://github.com/PSCLab-ASU/SystolicCNN.**

There have been many recent work [1], [2] on accelerating convolutional neural network (CNN) inference for computer vision tasks on FPGAs using OpenCL showing promising performance. Nevertheless, these work suffer from two major limitations that make them insufficient for realizing acceleration-as-a-service for multi-tenancy cloud or edge computing: 1) the lack of flexibility for supporting multiple CNN models at run time; and 2) the poor scalability resulting in underutilized FPGA resources and limited computational parallelism. For example, [3] is designed for accelerating the YOLO model [4] exclusively, which cannot be adapted at run time for supporting other CNN models. In addition, the performance results in [1], [2] showing under-utilized DSP block resources reveal scalability issues that lead to reduced performance under-utilizing the device capability.

In this paper, we present Systolic-CNN, an OpenCL-defined scalable, run-time-flexible FPGA accelerator architecture for accelerating CNN inference in cloud/edge computing. Systolic-CNN adopts a highly pipelined and parallelized 1-D systolic array architecture, which efficiently explores both spatial and temporal parallelism for accelerating CNN inference on FPGAs. Systolic-CNN is highly scalable and has three key architectural parameters, based on which we propose a strategy for users to optimally scale the accelerator architecture to fully utilize the external memory bandwidth and available computing resource given an FPGA board. We tested the performance of AlexNet, ResNet-50, RetinaNet, and Light-weight RetinaNet, to show the flexibility of our CNN accelerator. In addition, Systolic-CNN is run-time-flexible, which can be time-shared, in the context of multi-tenancy cloud or edge computing, to accelerate a variety of CNN models at run time without the need of recompiling the FPGA kernel hardware nor reprogramming the FPGA.

The experiment results based on an Intel Arria 10 GX FPGA Development board show that Systolic-CNN, when mapped with a single-precision data format, can achieve 100% utilization of the DSP block resource and an average inference latency of 10ms, 84ms, 1615ms, and 990ms per image for accelerating AlexNet, ResNet-50, RetinaNet, and Light-weight RetinaNet, respectively. The experiment of image classification and object detection tasks are based on the ImageNet (224×224) and COCO dataset (800×800), respectively. The peak computational throughput is measured at 80-170 GFLOPS/s across the acceleration of different CNN models.

References

[1] D. Wang, K. Xu, and D. Jiang, "Pipecnn: An opencl-based open-source fpga accelerator for convolution neural networks," in *2017 International Conference on Field Programmable Technology (ICFPT)*. IEEE, 2017, pp. 279–282.

[2] N. Suda, V. Chandra, G. Dasika, A. Mohanty, Y. Ma, S. Vrudhula, J.-s. Seo, and Y. Cao, "Throughput-optimized opencl-based fpga accelerator for large-scale convolutional neural networks," in *Proceedings of the 2016 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays*, 2016, pp. 16–25.

[3] D. T. Nguyen, T. N. Nguyen, H. Kim, and H.-J. Lee, "A high-throughput and power-efficient fpga implementation of yolo cnn for object detection," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 2019.

[4] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 779–788.

2576-2621/20/$31.00 ©2020 IEEE
DOI 10.1109/FCCM48280.2020.00064