DAC YF

FINN-Pruning: Accelerating Sparse Neural Networks Inference using a Data-Flow Architecture on FPGAs

Presenter: Yanyue Xie Advisors: Prof. Xue Lin, Prof. Yanzhi Wang, Prof. Miriam Leeser
Northeastern University

58 DESIGN AUTOMATION CONFERENCE
FROM CHIPS TO SYSTEMS — LEARN TODAY, CREATE TOMORROW

Introduction

Pruning strategies [1, 2] for Deep Neural Networks (DNNs) have been extensively studied during the past few years. However, acceleration of DNN computation on hardware hardly satisfies what we want to achieve. For instance, with the state-of-the-art GPU architecture [3], we could attain up to two times speedup of sparse neural network computation, which is far less than the compression ratio.

To address this problem, we propose a framework called FINN-pruning for generating FPGA-based DNN inference accelerators using block-based column-row (BCR) pruning [6].

The accelerator design is based on FINN's streaming dataflow architecture [4, 5], where all layers are computed on one single FPGA. To take advantage of the matrix-vector multiplication unit on FPGAs, we leverage balanced block-based column-row pruning for more structured computation workloads. In order to save computation time on FPGA, we add the reordering step in FINN-pruning to make the computation in a more structured manner.

Figure 1 shows the block-based column-row pruning strategy. Figure 2 presents the design flow of our framework, from quantization-aware training to final deployment onto FPGAs. Figure 3 demonstrates the balanced BCR pruning for saving computation on the hardware. Finally, Table 1 gives the experimental results on Alveo U280 for two representative DNN models, MobileNetV1 and ResNet50.

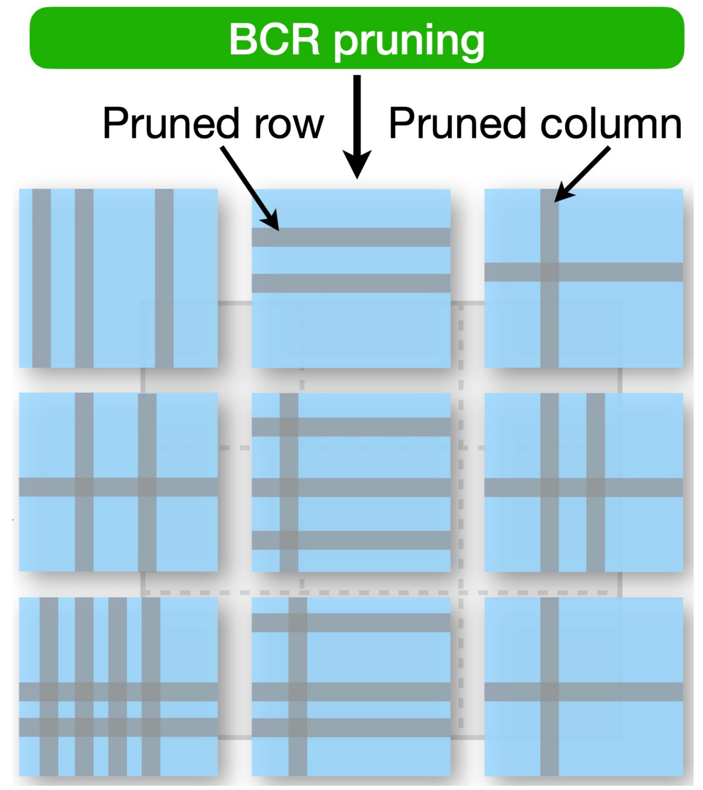
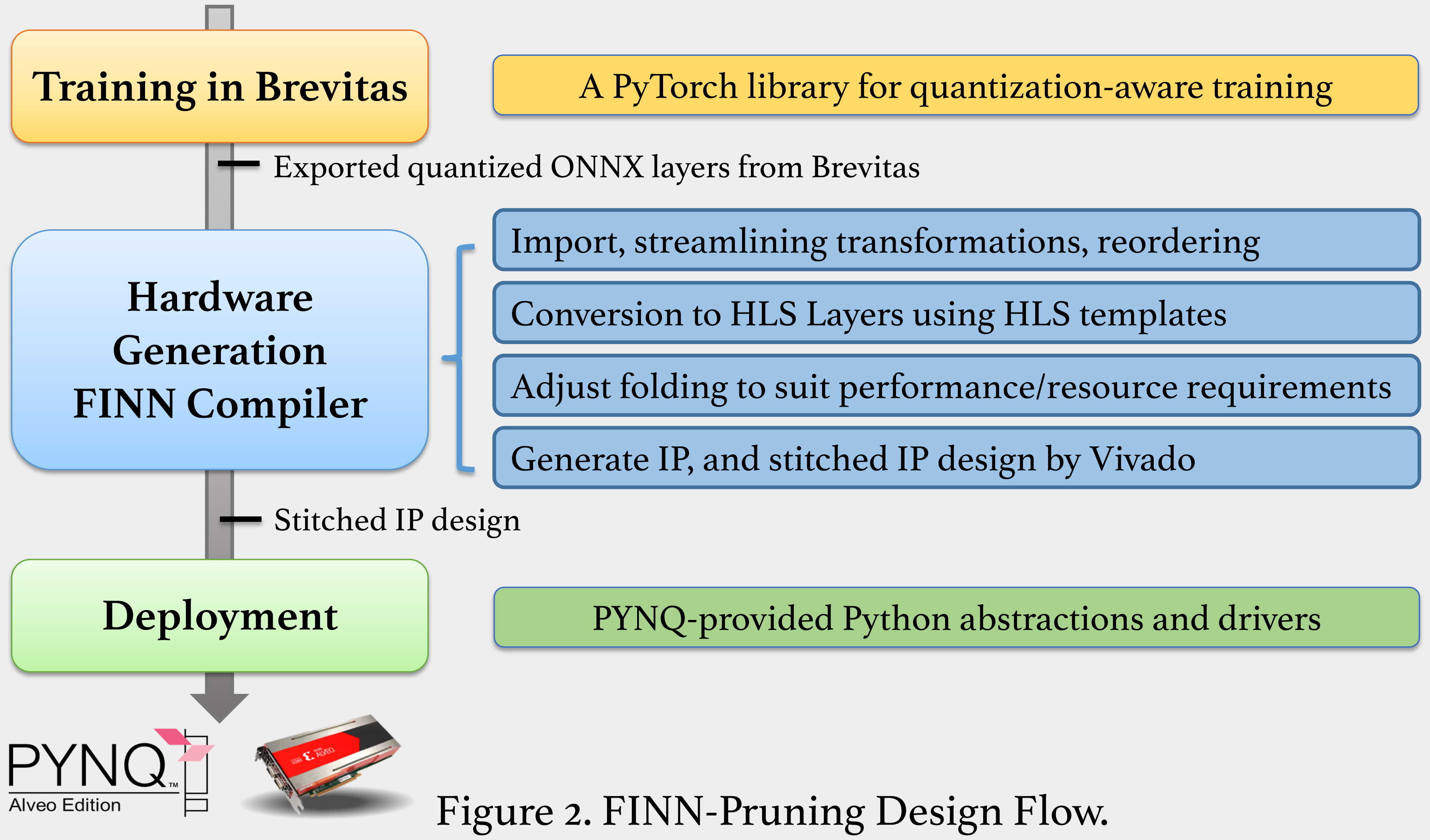


Figure 1. Block-based column-row pruning.

Methods


Training in Brevitas — A PyTorch library for quantization-aware training

Exported quantized ONNX layers from Brevitas

Hardware Generation FINN Compiler
- Import, streamlining transformations, reordering
- Conversion to HLS Layers using HLS templates
- Adjust folding to suit performance/resource requirements
- Generate IP, and stitched IP design by Vivado

Stitched IP design

Deployment — PYNQ-provided Python abstractions and drivers

Figure 2. FINN-Pruning Design Flow.

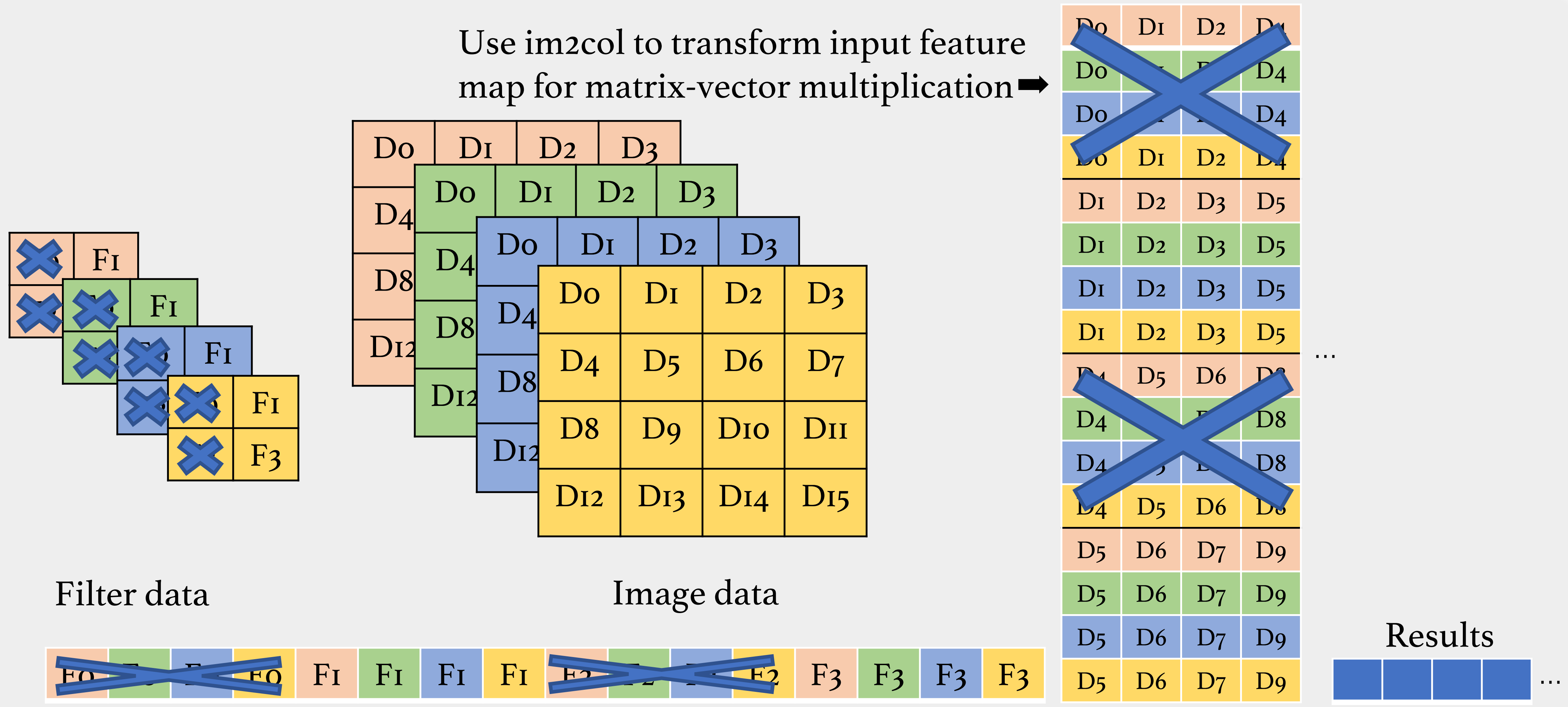

Use im2col to transform input feature map for matrix-vector multiplication ➡

Figure 3. Pruning saves computation on hardware after reordering.

Experimental Results

	FPGA	Quantization*	Fclk	Top-1 Accuracy	Performance
MobileNetV1	Alveo U280	4W4A	144MHz	70.4%	925img/sec
ResNet50		1W2A	135MHz	65.0%	703img/sec

*Quantization type xWyA means that the model employs quantization of x-bit for weights and y-bit for activations.

Table 1. Experimental results on Alveo U280.

References

[1]. Song Han, Huizi Mao, and William J. Dally. "Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding." In *International Conference on Learning Representation (ICLR)*, 2016.

[2]. Tianyun Zhang, Shaokai Ye, Kaiqi Zhang, Jian Tang, Wujie Wen, Makan Fardad, and Yanzhi Wang. "A systematic dnn weight pruning framework using alternating direction method of multipliers." In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 184-199. 2018.

[3]. Nvidia A100 Tensor Core GPU Architecture Whitepa-per. https://images.nvidia.com/aem-dam/en-zz/Solutions/data-center/nvidia-ampere-architecture-whitepaper.pdf . Last accessed Dec. 2, 2021.

[4]. Yaman Umuroglu, Nicholas J. Fraser, Giulio Gambardella, Michaela Blott, Philip Leong, Magnus Jahre, and Kees Vissers. "FINN: A framework for fast, scalable binarized neural network inference." In *Proceedings of the 2017 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays (FPGA)*, pp. 65-74. 2017.

[5]. Michaela Blott, Thomas B. Preußer, Nicholas J. Fraser, Giulio Gambardella, Kenneth O'brien, Yaman Umuroglu, Miriam Leeser, and Kees Vissers. "FINN-R: An end-to-end deep-learning framework for fast exploration of quantized neural networks." *ACM Transactions on Reconfigurable Technology and Systems (TRETS)* 11, no. 3 (2018): 1-23.

[6]. Wei Niu, Zhengang Li, Xiaolong Ma, Peiyan Dong, Gang Zhou, Xuehai Qian, Xue Lin, Yanzhi Wang, and Bin Ren. "GRIM: A General, Real-Time Deep Learning Inference Framework for Mobile Devices based on Fine-Grained Structured Weight Sparsity." *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2021).

Northeastern University