2016 IEEE 24th Annual International Symposium on Field-Programmable Custom Computing Machines

FPGA-Based Reduction Techniques for Efficient Deep Neural Network Deployment

Adam Page and Tinoosh Mohsenin
Department of Computer Science & Electrical Engineering
University of Maryland, Baltimore County
apage2@umbc.edu | tinoosh@umbc.edu

I. Introduction

Deep neural networks (DNN) have regained considerable attention in recent years with the ability to outperform previous state-of-the-art solutions while requiring minimal preprocessing and domain knowledge. In particular, convolutional neural networks have been shown to dominate on several computer vision benchmarks such as ImageNet. While there has been much improvement in training networks using powerful GPUs, the ability to deploy them in resource-constrained settings is still not feasible [1]. The focus of this work looks to reduce complexity and to efficiently deploy deep networks in an embedded FPGA-based setting with strict power and area budget. We first look to reduce the inherent complexity of a network by applying both fixed-point quantization and low-rank weight approximation. We then propose a custom FPGA-based framework to efficiently deploy trained networks.

Network	MNIST	CIFAR-10	SVHN
Baseline	99.50%	89.05%	97.71%
Reduced	99.35%	89.01%	97.51%
% Difference	↓ 0.15%	↓ 0.04%	↓ 0.20%

TABLE I: Comparison of baseline and reduced network's test accuracy for each of the 3 targeted datasets.

II. Reduction Techniques

1) Fixed-Point Quantization: Instead of using high-precision floating-point representations of convolutional weights, we look to utilize only low precision, fixed-point representations. Utilizing lower-precision weights with stochastic rounding has previously been shown to improve classification accuracy by acting as a form of regularization. We look to further exploit this to significantly reduce power and area.

2) Singular Value Decomposition: Given a fully-connected layer with N neurons and M inputs, the weight matrix $W_{M,N}$ can be factorized as $U_{M,N}\Sigma_{N,N}V^T_{N,N}$, where U and V are unitary matrices and Σ is a rectangular diagonal matrix containing singular values. If the weight matrix is sparse, then we can keep only the K largest singular values. When $K \ll N$, this can be approximated as $U_{M,K}\left(\Sigma_{K,K}V^T_{N,K}\right) = U_{M,K}V'_{K,N}$.

3) Reduction Evaluation: The two reduction techniques were applied to three popular computer vision datasets: MNIST, CIFAR, and SVHN. The first technique imposed fixed-point representations on the initial convolutional layers. Numerous experiments were performed for various fixed-point formats of the weights in which the integer was fixed at 3-bits and fraction varied from 2-bits to 14-bits. Singular value decomposition and truncation of the fully-connected layers was then performed using several different compression amounts. In the end, it was determined that using 3.6 fixed-point format with 75% compression provided the optimal reduction for all three datasets with minimal impact on accuracy (see Fig. I).

III. FPGA Hardware Implementation Framework

The framework is implemented using a combination of Xilinx Vivado High-Level Synthesis (HLS) and a developed set of C++ templates. To translate a network, a *network packetizer* script is used to partition the network into multiple pipelined blocks. Using HLS and C++ templates, IP cores are generated for each of the subnetwork's block which can then be chained together in HDL. The framework further enables specifying network partitions, level of parallelism, and weight format. The baseline and reduced networks for the three datasets were then translated using this framework.

Attribute	MNIST		CIFAR-10		SVHN	
Topology	Baseline	Reduced	Baseline	Reduced	Baseline	Reduced
Accuracy	99.50%	99.35%	89.05%	89.01%	97.71%	97.51%
# Partitions	2	2	4	4	4	4
# Workers	20, 2	16, 1	20,20,20,3	16,16,16,1	10,10,10,3	6,6,6,1
Slices	13,078 (41%)	**8,523 (26%)**	30,159 (22%)	**14,984 (11%)**	22,282 (17%)	**11,744 (8%)**
Memory (BRAM)	41 (54%)	41 (54%)	302 (83%)	264 (72%)	155 (43%)	132 (36%)
Dyn. Power (mW)	206	**125**	531	**304**	298	**211**
Static Power (mW)	62	61	149	145	140	138
Performance (img/sec)	15.1	15.6	18.1	16.9	17.5	16.7
Efficiency (img/sec/W)	56	**83**	27	**38**	40	**48**
Eff. Improvement	**48%**		**41%**		**20%**	

TABLE II: Comparison of baseline and reduced network implementations for each targeted dataset when implemented on Artix-7 FPGA.

IV. Conclusion

The benefits of deep networks have yet to be fully exploited in embedded, resource-bound settings that have strict power and area budgets. In this work, we first demonstrated the ability to reduce the inherent complexity using both singular value decomposition on dense layers and using limited precision fixed-point representations of convolutional weights. Second, a custom FPGA-based framework was proposed to efficiently and uniquely deploy a pre-trained neural network. When running on an Xilinx Artix-7 FPGA, experimental results demonstrated the ability to achieve a classification throughput of 16 images/sec and consume less than 700 mW at 200 MHz when applied to three popular computer vision datasets. In addition, the reduced networks are able to, on average, reduce power and area utilization by 37% and 44%, respectively, while only incurring less than 0.20% decrease in accuracy.

References

[1] A. Page, C. Shea, and T. Mohsenin, "Wearable seizure detection using convolutional neural networks with transfer learning," in *Circuits and Systems (ISCAS), 2016 IEEE International Symposium on*, 2016.

978-1-5090-2356-1/16 $31.00 © 2016 IEEE
DOI 10.1109/FCCM.2016.58