| Metrics | [3] | [5] | [16] | **This work** | |
|---|---|---|---|---|---|
| Platform | Virtex-5 | Zynq | Virtex-7 | **Zynq** | Jetson TK1 |
| Precision (bit) | 48 Fixed | 16 Fixed | 322 Float | **16 Fixed** | 32 Float |
| Clock (MHz) | 120 | 150 | 100 | **100** | 2320 |
| Network | N/A | N/A | AlexNet | **Inception** | Inception |
| Network Complexity (GOP) | 0.52 | 0.552 | 1.33 | **15.92** | 55.30 |
| Network Performance (GOP/s) | 16 | 25.15 | 61.62 | **1592** | 55.30 |
| Total Power (W) | 14 | 8 | 18.61 | **2.07** | 12.08 |
| Network Energy Efficiency (GOP/s/W) | 1.14 | 2.90 | 3.31 | **5.70** | 4.58 |

**Table 2: Comparison of the proposed accelerator with related works when evaluated on a convolutional network. For this work, measurements are taken when running on Zedboard containing three SCALENet accelerators without sparsification.**

large increase in efficiency with the accelerator. The SCALENet accelerator on average can improve efficiency over its host processor by 3x whereas the K1 GPU can increase efficiency by 4.5x over its ARM CPUs. Figure 9 similarly compares the efficiency of the FPGA and GPU platforms when targeting the ImageNet networks. The Zedboard with 3 SCALENet accelerators can achieve an average increase in efficiency of 9x whereas the TK1 accelerated with the GPU can increase efficiency on average by 5x. In Table 2 the last row outlines the power efficiency savings for using the SCALENet architecture, combined with the scheduler, compared to just the ARM processor running torch compiled against the OpenBlast libraries. The execution time improved by 99.7% while the power only improved by 74.10%.

## 6 COMPARISON WITH EXISTING WORK

Table 2 compares the proposed SCALENet with existing FPGA-based CNN accelerators. Deployed on Zedboard platform with Cortex-A9 running at 667.67 MHz and FPGA running at 100 MHz, the accelerator achieves an energy efficiency of 5.70 GOP/J with total system power of 2.07 W and throughput of 15.92 GOP/s. SCALENet delivers higher energy efficiency for a given network complexity than the previous best accelerators [5] and [16] while targeting a much lower power utilization. Additional gains in efficiency are achieved by exploiting the ability to perform fused convolution, batch normalization, and ReLU.

## 7 CONCLUSION

In this work, we proposed contributions in two enterprises for deploying convolutional and fully connected neural networks in resource-bound, real-time embedded systems. In the first contribution, we evaluated eight image processing focused networks trained on CIFAR-10 and ImageNet datasets for computation, memory, and accuracy and we assess the use of a modified network for time series EEG seizure detection. In the second contribution, we proposed and evaluated SCALENet: a SCalable Low power Accelerator for real-time deep neural Networks. The accelerator enables deploying a variety of networks through coarse and fine grain configuration as well as acceleration of FC and CNN layers and implementation flexibility for hardware only or hardware and software designs. The proposed accelerator was evaluated in real-time on different FPGA platforms using eight image processing networks and one custom time series network for biomedical seizure detection in EEGs and demonstrated to outperform all other COTS platforms. When accelerating the highest complexity network evaluated, Inception with 11.78 GOP, on the Zedboard platform with a dual-core ARM Cortex-A9 running at 667.67 MHz and FPGA running at 100 MHz, the accelerator achieves an energy efficiency of 5.70 GOP/s/W with a total system power of 2.07 W and throughput of 15.92 GOP/s. SCALENet produces higher energy efficiency than the prior best accelerators as well as Jetson TK1 while targeting a power profile that is more than 4x lower than the Jetson TK1. In the case of time series seizure detection, our SCALENet architecture shows an execution time and power improvement of 99.75% and 97.9% respectively compared to software-only implementations.

## REFERENCES


[1] T. Abtahi, A. Kulkarni, and T. Mohsenin. 2017. Accelerating convolutional neural network with FFT on tiny cores. In *2017 IEEE International Symposium on Circuits and Systems (ISCAS)*. 1–4. https://doi.org/10.1109/ISCAS.2017.8050588

[2] T. Abtahi, C. Shea, A. Kulkarni, and T. Mohsenin. 2018. Accelerating Convolutional Neural Network with FFT on Embedded Hardware. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* (2018).

[3] U. Rajendra Acharya et al. 2017. *Computers in Biology and Medicine* (2017). https://doi.org/10.1016/j.compbiomed.2017.09.017

[4] Y. Chen et al. 2016. 14.5 Eyeriss: An energy-efficient reconfigurable accelerator for deep convolutional neural networks. In *ISSCC*.

[5] Vinayak Gokhale et al. 2014. A 240 G-ops/s mobile coprocessor for deep neural networks. In *CVPRW*.

[6] A. L. Goldberger, L. A. N. Amaral, L. Glass, J. M. Hausdorff, P. Ch. Ivanov, R. G. Mark, J. E. Mietus, G. B. Moody, C.-K. Peng, and H. E. Stanley. 2000 (June 13). PhysioBank, PhysioToolkit, and PhysioNet: Components of a New Research Resource for Complex Physiologic Signals. *Circulation* 101, 23 (2000 (June 13)), e215–e220. Circulation Electronic Pages: http://circ.ahajournals.org/content/101/23/e215.full PMID:1085218; doi: 10.1161/01.CIR.101.23.e215.

[7] Forrest N. Iandola et al. 2016. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <1MB model size. *CoRR* (2016).

[8] Adam Page, Ali Jafari, Colin Shea, and Tinoosh Mohsenin. 2017. SPARCNet: A Hardware Accelerator for Efficient Deployment of Sparse Convolutional Networks. *J. Emerg. Technol. Comput. Syst.* 13, 3, Article 31 (May 2017), 32 pages. https://doi.org/10.1145/3005448

[9] A. Page, C. Shea, and T. Mohsenin. 2016. Wearable Seizure Detection using Convolutional Neural Networks with Transfer Learning. In *ISCAS*.

[10] Mohammad Samragh et al. 2017. Customizing Neural Networks for Efficient FPGA Implementation. In *FCCM*. IEEE.

[11] Jaehyeong Sim et al. 2016. 14.6 A 1.42 TOPS/W deep convolutional neural network recognition processor for intelligent IoE systems. In *ISSCC*. IEEE.

[12] Karen Simonyan et al. 2014. Very Deep Convolutional Networks for Large-Scale Image Recognition. *CoRR* (2014).

[13] L. Song et al. 2016. C-Brain: A deep learning accelerator that tames the diversity of CNNs through adaptive data-level parallelization. In *DAC*.

[14] Abdulhamit Subasi et al. [n. d.]. Classification of EEG signals using neural network and logistic regression. *Computer Methods and Programs in Biomedicine* ([n. d.]).

[15] Xilinx. 2011. Power Methodology Guide. (2011).

[16] Chen Zhang et al. 2015. Optimizing FPGA-based Accelerator Design for Deep Convolutional Neural Networks. In *FPGA (FPGA '15)*. ACM.