Received: 20 July 2021 | Accepted: 2 September 2021

DOI: 10.1111/exsy.12831

REVIEW

Expert Systems WILEY

High performance accelerators for deep neural networks: A review

Mohd Saqib Akhoon[1] | Shahrel A. Suandi[1] | Abdullah Alshahrani[2] | Abdul-Malik H. Y. Saad[3] | Fahad R. Albogamy[4] | Mohd Zaid Bin Abdullah[1] | Sajad A. Loan[5]

[1]Intelligent Biometric Group, School of Electrical and Electronic Engineering, Universiti Sains, George Town, Malaysia

[2]Department of Computer Science and Artificial Intelligence, College of Computer Science and Engineering, University of Jeddah, Jeddah, Saudi Arabia

[3]Division of Electronics and Computer Engineering, School of Electrical Engineering, Faculty of Engineering, Universiti Teknologi Malaysia, Johor Bahru, Malaysia

[4]Turabah University College, Computer Sciences Program, Taif University, Saudi University, Ta'if, Saudi Arabia

[5]Department of Electronics and Communication, Jamia Millia Islamia, New Delhi, India

Correspondence
Sajad A. Loan, Department of Electronics and Communication, Jamia Millia Islamia, New Delhi 11025, India.
Email: sloan@jmi.ac.in

Abstract

The availability of huge structured and unstructured data, advanced highly dense memory and high performance computing machines have provided a strong push for the development in artificial intelligence (AI) and machine learning (ML) domains. AI and machine learning has rekindled the hope of efficiently solving complex problems which was not possible in the recent past. The generation and availability of big-data is a strong driving force for the development of AI/ML applications, however, several challenges need to be addressed, like processing speed, memory requirement, high bandwidth, low latency memory access, and highly conductive and flexible connections between processing units and memory blocks. The conventional computing platforms are unable to address these issues with machine learning and AI. Deep neural networks (DNNs) are widely employed for machine learning and AI applications, like speech recognition, computer vison, robotics, and so forth, efficiently and accurately. However, accuracy is achieved at the cost of high computational complexity, sacrificing energy efficiency and throughput like performance measuring parameters along with high latency. To address the problems of latency, energy efficiency, complexity, power consumption, and so forth, a lot of state of the art DNN accelerators have been designed and implemented in the form of application specific integrated circuits (ASICs) and field programmable gate arrays (FPGAs). This work provides the state of the art of all these DNN accelerators which have been developed recently. Various DNN architectures, their computing units, emerging technologies used in improving the performance of DNN accelerators will be discussed. Finally, we will try to explore the scope for further improvement in these accelerator designs, various opportunities and challenges for the future research.

KEYWORDS
artificial intelligence, convolutional neural networks, deep neural network, machine learning, accelerators

1 | INTRODUCTION

The theoretical concept of neural network (NN) was independently proposed by two researchers A. Bain and William James (Bain, 1873; James, 1890). As per their independent work, thoughts and activities in a human being are related and are due to interactions among various

neurons within the brain. As per A. Bain, all activities done by humans are due to the firing of certain of neurons and as per W. James the electric current between various neurons results in various activities and memory realization. Alan Turing in 1948 has proposed the concept of neural networks in his paper "*intelligent machinery*" (Turing, 1950) and (McCulloch & Pitts, 1990) gave the well-known McCulloch and Pitts computational model for neural networks in 1943, called as threshold logic model. Artificial neural network (ANN) is a backbone of AI and has been successfully applied in various AI based strategic domains, like robotics, speech and image recognition, adaptive control, autonomous vehicles, and so forth. There was a deep pause in neural network study and development for the last many decades due to the absence of efficient and high performance processors and big data. However, in the recent past huge amounts of data has become available in the form of videos, audios, text, and so forth, along with the development of high performance computing machines due to the revolution in integrated circuit industry. This has rekindled the hope in NN design and development in general and artificial intelligence field in particular. The availability of huge data and high performance computing machines has given birth to Deep Learning in 2006 (Chen et al., 2015; Deng et al., 2013; Krizhevsky et al., 2012a, 2012b; LeCun et al., 2015; Lee et al., 2016; Netzer et al., 2011) and hence the creation of DNN. DNNs are important for many modern AI applications, like speech recognition, image recognition, videos, analysing medical data, robotics, military surveillance, and so forth, with an important feature that DNNs exceeded human accuracy. The various studies have revealed that the convolutional neural networks (CNN) on small data sets have achieved record-breaking accuracy in image recognition domain. Andri et al. (2017) have revealed that the accuracy in image recognition achieved in Street View House Number (SVHN) (Du et al., 2015), Modified National Institute of Standards and Technology (MNIST) and Canadian Institute for Advanced Research (CIFAR-10) data sets are 98.31%, 99.79%, and 96.53%, respectively (Andri et al., 2017; Graham, 2014; Lee et al., 2016; Wan et al., 2013).

DNNs have a special feature of extracting high-level features from raw sensory data, thus resulting in their high performance. However, the high performance and accuracy in DNNs is obtained at the penalty of their diverse shape and sizes and high computational complexity. Further, energy and power efficiency, throughput, latency, requirement of large memory, and so forth are other performance limiting parameters in DNN, which need to be optimized to reap the benefits of DNNs. The slow nature of NN or DNN is due to load/store latency, shuffling of data in/out of the GPU pipeline, the limited width of the pipeline, the unnecessary extra precision NN calculations and the sparsity of the input data and many other features (Esteva et al., 2017; LeCun et al., 2015; Lee et al., 2016; Wan et al., 2013). Now, the question in front of researchers is how to make DNN, training, prediction and testing faster and efficient. Therefore, the need of the hour is to design dedicated high performance DNN accelerators which are capable of achieving high performance and energy efficiency across a wide range of DNN applications and hence enabling AI in real-world applications. The development of fast algorithms, software and hardware optimizations and the use of emerging electronics can improve the performance of DNN accelerators significantly. A lot of research has gone into the development of high performance and energy efficient DNN accelerators.

Andri et al. (2017) have proposed an extremely low power Binary-Weight CNN Accelerator architecture. It is the first optimized, flexible and energy-efficient architecture supporting binary-weight CNNs. A configurable cloud architecture for CNN is proposed by Caulfield et al. (2016). Moini et al. (2017) have proposed an efficient resource limited accelerator design architecture for CNNs in embedded vision applications. It uses the parallelism in CNNs and improves almost all performance measuring parameters. A new sparse CNN inference accelerator architecture was designed and developed by Parashar et al. (2017). The proposed architecture uses PE and multiplier array and exploiting both weight and activation sparsity. A memory-centric accelerator has been designed and implemented on an FPGA with improved performance by Peemen et al. (2013). It addresses the challenging problem of limited memory bandwidth in DNN accelerators. A novel convolutional network accelerator, named as Origami: A 803 GOp/s/W has been designed and implemented by Cavigelli and Benini (2017). They are the first to report the silicon measurements of such an CNN accelerator. Lee, Shao, et al. (2018a) have proposed a novel Stitch-X architecture for a DNN inference accelerator. It stitches together sparse weights and input activations for realizing parallel execution. Shafiee et al. (2016) have designed a novel CNN Accelerator, called as ISAAC. The proposed architecture is pipelined with memristor based cross bars for each NN layer and eDRAM buffers. Chen, Du, et al. (2014a) and Chen et al. (2017) have given two well-known architectures, DianNao and DaDianNao, respectively, for machine learning accelerator designing. The focus of these architectures is on memory reduction. Chen et al. (2017, 2019) have designed and realized Eyeriss and Eyeriss V2 DNN accelerator architectures. Both architectures use parallelism more efficiently and results in high performance. Reagen et al. (2016) have designed and developed a novel, low power DNN accelerator, called as Minerva. It is a multistage, energy efficient architecture. Wu et al. (2019) and Aimar et al. (2019) have designed and developed an energy and power efficient accelerator for sparse compressed CNNs. Accelerators for binarized neural networks (BNNs) have been designed by Zhao et al. (2017) and Guo et al. (2019). For the first time fused layer CNN accelerator have been developed by Alwani et al. (2016). The fusion of multiple CNN layers enables caching of intermediate data generated between the evaluations of adjacent CNN layers. Esmaeilzadeh et al. (2014) and Lee, Shao, et al. (2018a) have designed a new class of programmable accelerators called as the neural processing unit (NPU). Hu et al. (2019) have designed an efficient configurable accelerator for DNN. It is an array based structure with four level processing elements. A lot of accelerators have been designed and developed employing emerging memories, like resistive RAM (ReRAM). These emerging electronics device based memories enables huge storage along with the high speed data access. They can implement in-memory computing, an important feature in these devices. PRIME (Chi et al., 2016), Pipelayer (Song et al., 2017), REGAN (Chen et al., 2018), Atom layer (Qiao et al., 2018), (Ambrogio et al., 2013; Ambrogio et al., 2020); (Chen el al., 2011) all use ReRAM based accelerators. Jouppi et al. (2017) have designed and developed an ASIC, called as Tensor Processing Unit (TPU). The TPUs are used in data centres and accelerate the inferencing part of neural networks. Deep Neural Network accelerator has also been realized with Spintronic Memory concept (Zang et al., 2020).

In rest of the paper, Section 2 discusses various popular DNN models. Section 3 discusses the state of the art of DNN accelerators and Section 4 concludes the paper.

2 | POPULAR DNN MODELS

Deep neural network is a neural network with multiple number of hidden layers between the input and the output layers. A lot of research work has gone into the development of novel, energy efficient and high performance DNN models. These models differ in terms of number, types, shapes of layers and the connection between them. Some of the important well known DNNs are discussed below.

a. AlexNet DNN model (Krizhevsky et al., 2012a, 2012b): This model has been designed by Alex Krizhevsky, Ilya Sutskever and Geoffrey Hinton and is a winner of 2012 the ImageNet challenge. It has five and three, CONV and fully connected layers (FC) respectively. The range of filter count in each CONV layer is 96 to 384 and the range of filter size is from 3 × 3 to 11 × 11 with 3 to 256 channels each. The ReLu activation function is used in each layer, which helps in improving the training performance over tanh and sigmoid functions.
b. Overfeat DNN model (Sermanet et al., 2014): It is similar to AlexNET with three FC layers followed by five CONV layers. The number of filters for various layers in Overfeat gets significantly increased in comparison to the AlexNET. Two different models exist for Overfeat DNN model, with unique features, one is highly accurate and the other one is very fast.
c. VGG-16 DNN model (Simonyan & Zisserman, 2015). This model is one of the preferred CNN architectures in the recent past. It has 16 convolutional layers, with a very uniform architecture that is why it is preferred. It has two models, VGG 16 and VGG 19. VGG 19 has reduced error (0.1% lower top 5 error) than VGG 16 but at the cost of large number of MACs (1.27× more)
d. GoogLeNet model (Szegedy et al., 2015): An important and well known DNN model came into existence in 2014 is GoogleNet, also known as inception. It is also the winner of famous ImageNet challenge contest. This model employs batch normalization and hence improves critical parameters like speed, performance and stability significantly. It has more deeper layer and has 22 convolutional layers, followed by 9 inception layers and 1 fully connected layer.
e. ResNet model (Xie et al., 2016): This model also known as residual net, it is a deep DNN model with 34 layers or more. Its unique feature is that it is the first DNN model exceeding human level accuracy with top 5 error rate. It has a powerful representational ability and can train efficiently hundreds to thousands of layers with good performance.

There are some other DNN models, like, DenseNet, WideNet, ResNeXt and EfficientNET (He et al., 2016; Huang et al., 2017).

3 | DEEP NEURAL NETWORK ACCELERATORS

Various DNN accelerators have been designed and implemented in the last a few years. Herein, we discuss the latest updates in the deep neural network accelerators designed and development in the recent past. Andri et al. (2017) have proposed an Ultra-Low-Power architecture for Binary-Weight CNN Accelerator. The proposed architecture is energy and throughput efficient and gives 5.1 times energy efficiency and 1.3 times better throughput than the basic 1.2 V, 12-bit MAC architecture. Figure 1 shows the architecture of the proposed device with input channel i_n and output channels $o_{k.}$ The replacement of more complex fixed-point MAC units in the proposed architecture with simpler complement and multiplexer operations has made it quite simple and that too without losing the classification accuracy.

Caulfield et al. (2016) have proposed a Cloud-Scale Acceleration Architecture, called as the Configurable Cloud architecture. A layer of FPGAs has been kept between the network switches and the servers in the proposed architecture, resulting in more flexibility and reduced latency. The FPGA used acts as a local and a network accelerator and can communicate directly with other FPGAs in the data centre without any CPU software. The proposed architecture has resulted average round-trip latencies of 3 μs in 24 machines, 9 μs in 1000 machines and 20 μs in 250,000 machines. Figures 2 and 3 show the block diagram of proposed accelerator and the photograph of the board with major components of the proposed accelerator.

Moini et al. (2017) have proposed an efficient architecture for accelerating multiple convolution stages in CNNs, which has been employed in high quality vision systems. The proposed architecture optimizes the parallelism in CNNs and hence reduces the power consumption, delay, bandwidth requirement and hardware resources. The FPGA implementation of proposed architecture has been performed by using ZC706 evaluation board having Xilinx-Zynq-7000 SoC. The proposed architecture runs at a frequency of 150 MHz and achieves 19.2 Giga MAC operations per second and with just less than 10 watts power consumption. Figure 4 shows the proposed architecture, having controller in the PL side, data memory and a group of MAC modules. Figure 5 shows off-chip memory employed for data and coefficient loading. A detailed architecture of PL-side is given in Figure 6, with 32 bit data width and coefficient memory.

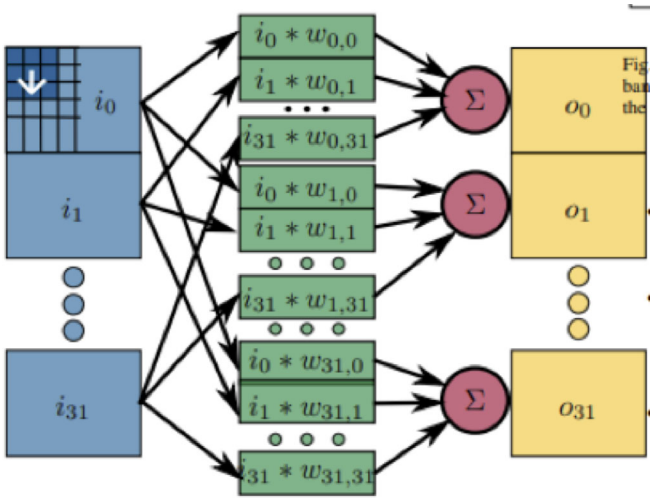

FIGURE 1 A 32 × 32 CNN layer, with n input channels (i_n) and k output channels (o_k) (Andri et al., 2017)

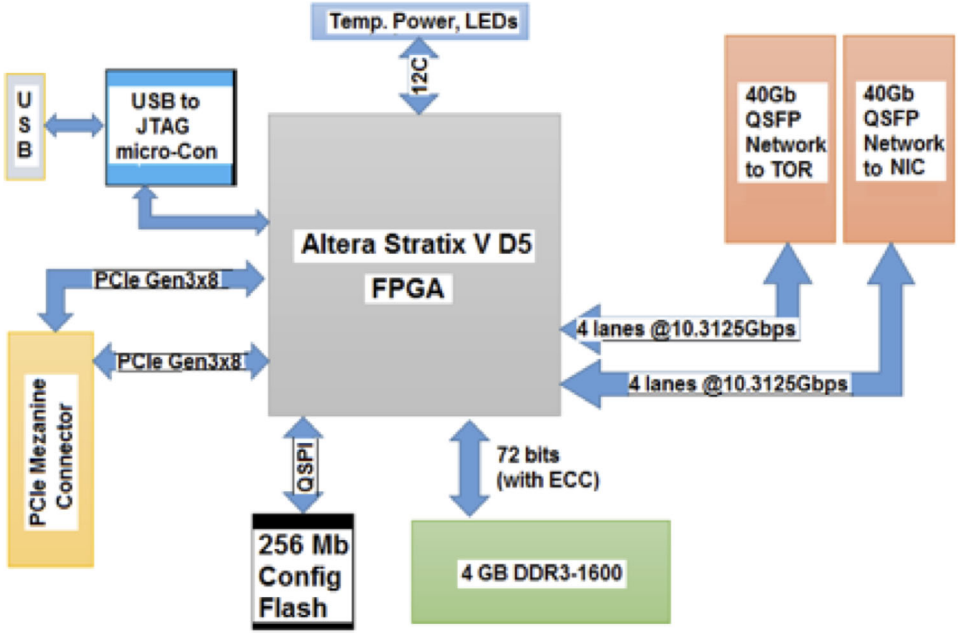

FIGURE 2 Block diagram of the proposed configurable cloud accelerator (Caulfield et al., 2016)

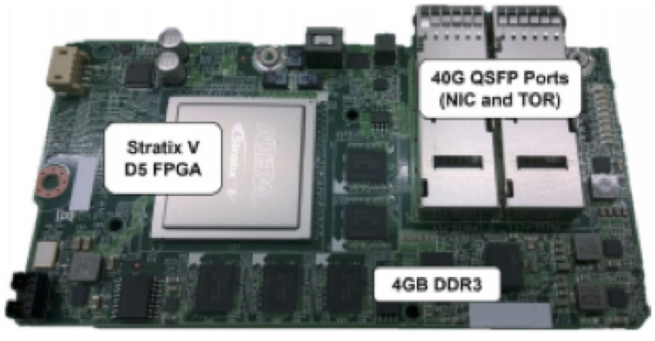

FIGURE 3 Photograph of the manufactured board with the proposed accelerator board (Caulfield et al., 2016)

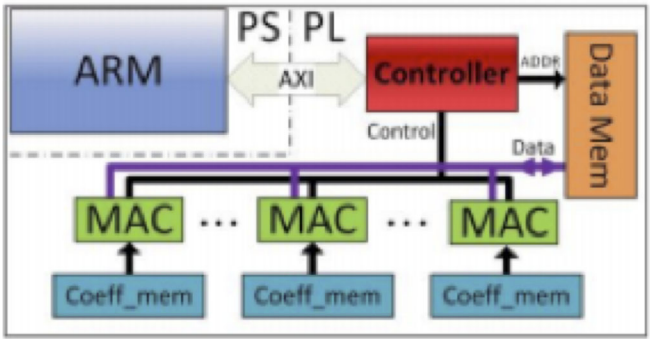

FIGURE 4 Proposed architecture at run-time (Moini et al., 2017)

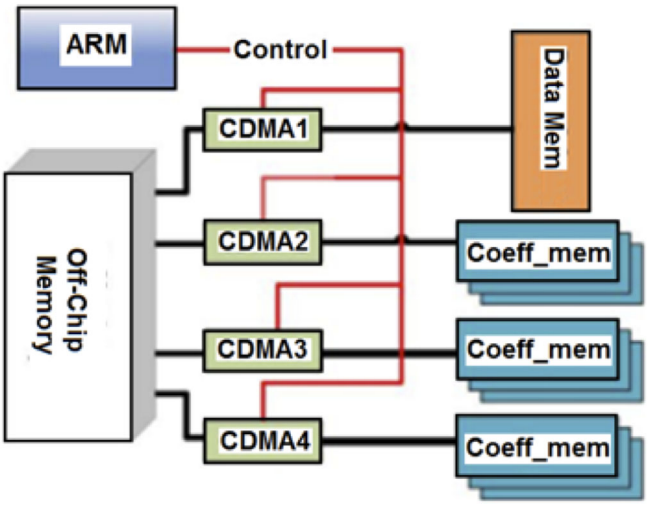

FIGURE 5 Overview of off-chip memory connectivity (Moini et al., 2017)

Parashar et al. (2017) designed a new Sparse-CNN (SCNN) inference accelerator, exploiting both weight and activation sparsity and hence significantly improves both power and performance. A multiplier array based processing element (PE) is an important component of SCNN, accepting the combination of vectors of activations and weights. The results have shown that the proposed SCNN architecture outperforms a dense architecture in overall performance when the activations and the weights are each less than 85% dense. The comparative analysis with the three well known networks (GoogleNet, VGGNet and the AlexNet) have revealed that the SCCN architecture is 2.3× energy efficient and the performance has improved by 2.6× over the dense-architecture. Figure 7 shows the block diagram of the proposed SCNN PE. The main blocks included are a weight buffer (WB), I/O activation RAMs (IA-RAM and OA-RAM), accumulator bank, a crossbar, an array multiplier, and a post-processing unit (PPU).

Peemen et al. (2013) have addressed a challenging problem in CNN accelerators, that is, the limited amount of external memory bandwidth. A memory-centric accelerator has been designed and implemented on an FPGA with improved performance without requiring extra memory bandwidth. A set of specialized memories have been used in the proposed architecture which optimize data movement and data locality scheduling. The FPGA implementation has shown that the proposed architecture is 11× faster than the standard accelerators. Figure 8 shows the block diagram of the proposed Memory-Centric Accelerator for CNNs.

Cavigelli and Benini (2017) have proposed a novel CNN accelerator, Origami: A 803 GOp/s/W. They are the first to report the silicon measurements of such a CNN accelerator. It has been observed from the experimental results that the proposed Origami accelerator out performs the state of the art CNN accelerators in terms of space, power consumption and I/O efficiency. The proposed architecture provides up to 196 GOp/s, with silicon consumption of 3.09 mm^2 in UMC 65 nm technology. Further, a large power efficiency of 803 GOp/s/W is achieved in the proposed architecture. This is the first CNN architecture which is scalable to p/s performance. Figure 9 shows the important functional blocks in the proposed architecture. The Image window data stores the received input image data and provides the sum of product (SOP) unit with image patch which is needed for the computation cycle. The filter bank block which performs the storage of all the weights of filters is read only in the normal operations and its size is quadratically dependent on the number of PEs. Further, the inner product between the image, image path and filter

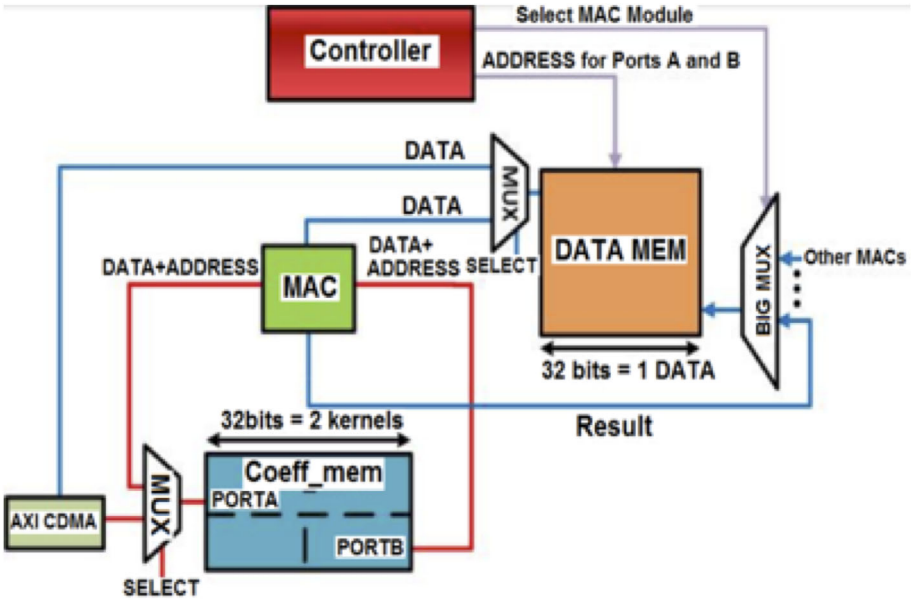

FIGURE 6 Detailed architecture in PL side (Moini et al., 2017)

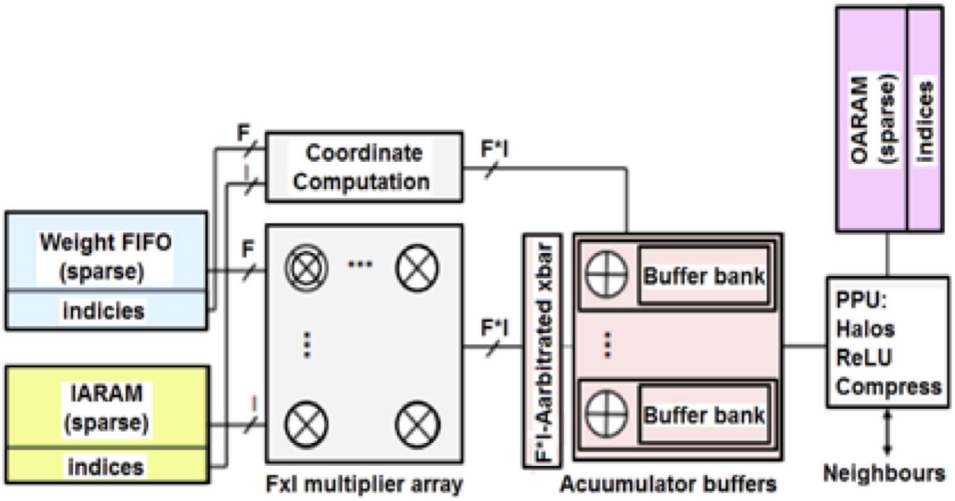

FIGURE 7 Block diagram of SCNN PE with PT-IS-CP-sparse dataflow (Parashar et al., 2017)

kernel is being performed by the dedicated SOP unit. The channel sum unit (Chsum) adds the inner product it receives from the SOP-unit and reduces the amount of data to be transmitted. The Chsum perform the calculation at full precision, resulting in fully truncated results.

Lee et al. (2018a) have proposed a novel architecture for a DNN inference accelerator, called as Stitch-X architecture, as shown in Figure 10. Its name "Stitch" comes from its ability of stitching sparse weights and input activations together to efficient realize the parallel execution. Herein, a new dataflow technique is employed to leverage the temporal and spatial reductions to increase the energy efficiency and to reduce the data flow complexity. To extract the fine grained parallel operations from the irregular data arrays, it employs a dedicated unit, called as parallel discovery unit (PDU), which enhances the performance significantly. The Stitch-X architecture achieves a 3.8× improvement in the speed of operation and 10.3× improvement in the energy-delay-squared-product (ED^2P) in comparison to an efficient, dense DNN accelerator. Shafiee et al. (2016) have designed and developed a high performance and novel CNN accelerator, called as ISAAC. The proposed architecture is pipelined with memristor based cross bars for each NN layer and eDRAM buffers. A new data encoding technique has been used for efficient analogue computations, capable of reducing high overheads of analogue-to-digital conversion (ADC). The proposed ISAAC architecture significantly improves throughput (14.8×), energy (5.5×) and computational density (7.5×) in comparison to the state-of-the-art DaDianNao architecture. The block diagram of the proposed ISAAC architecture is shown in Figure 11. The ISAAC architecture is only used for inference and is not used for in-the-field training.

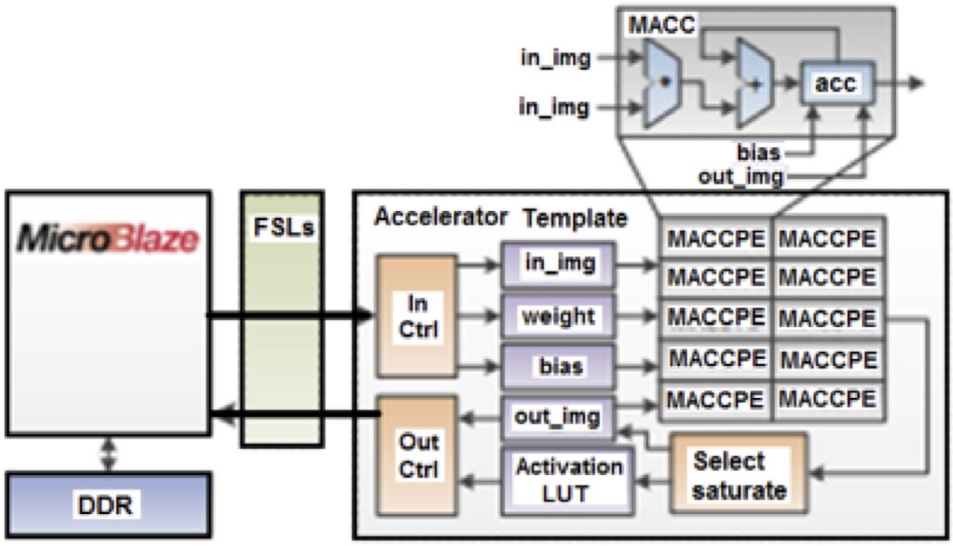

FIGURE 8 CNN accelerator template connected to a host processor for control (Peemen et al., 2013)

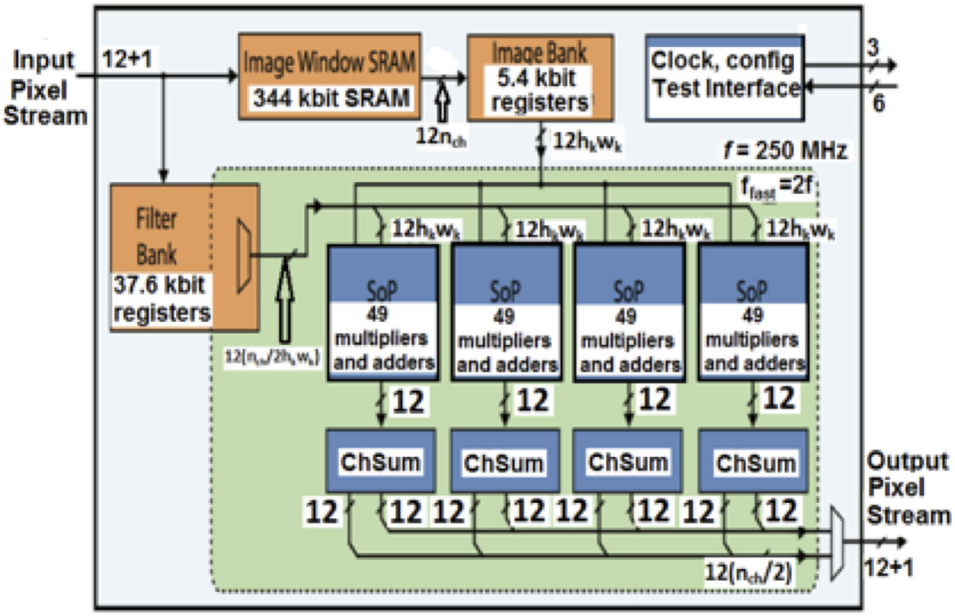

FIGURE 9 Block diagram of the proposed CNN architecture, Origami (Cavigelli & Benini, 2017)

A series of accelerators have been proposed with the name DianNao series. Chen, Du, et al. (2014a) have proposed an accelerator architecture, called as DianNao, for machine learning. The design focuses on optimizing the memory usage, that is, the reduction in the memory transfers. The design is realized at 65 nm technology node and is able to perform 496, 16-bit fixed-point operations (FPO) in parallel every 1.02 ns, that is, approximately 452 giga operations per second (GOP/s). The space requirement is small, just 3.02 mm^2 and consumes 485 mW power. Comparing with ten state of the art accelerators, the proposed one is faster (117.87 times) and more energy-efficient on average (21.08 times) than an 128-bit SIMD core processor clocked at 2GHz. The prominent components of the proposed accelerator include an input buffer (IPB) for input-neurons (NBin), an output buffer (OPB) for output-neurons (NBout), and a third synaptic weights buffer (SWB). All are connected to high performance computational blocks called as neural functional unit (NFU) and the control logic (CP) blocks, as shown in Figure 12a. Chen, Luo, et al. (2014b) have designed an advanced version of DianNao architecture, called as DaDianNao architecture, as shown in Figure 12b. It is a multi-chip hardware system running more efficiently than DianNao architecture. Liu, Chen, et al. (2015b) have presented a machine learning accelerator, called as PuDianNao. The proposed accelerator is supporting seven machine learning techniques. These include DNN, classification tree, k-means, k-NN, naïve bayes, support vector machine and linear regression. The PuDianNao architecture has been implemented in 65 nm TSMC process

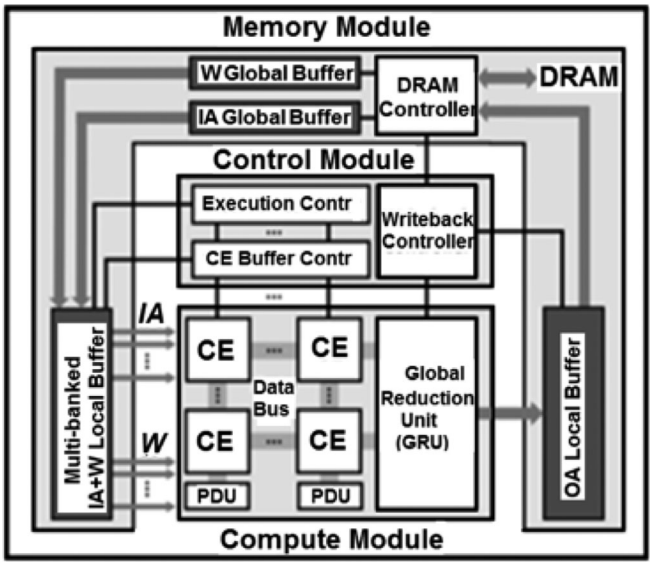

FIGURE 10 Stitch-X block diagram (Lee et al., 2019)

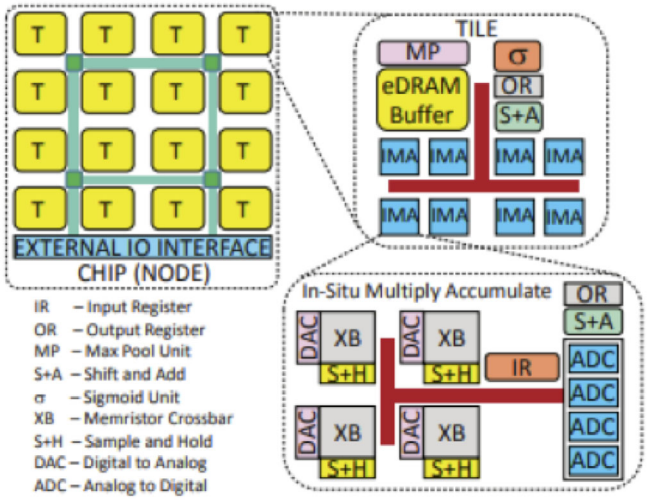

FIGURE 11 Proposed ISAAC architecture (Shafiee et al., 2016)

and it has been found that it is 128.41 times energy efficient and 1.2 times faster than the NVIDIA K20M GPU. Du et al. (2015) have proposed a CNN accelerator, called as ShiDianNao. Again, the focus of this work is to increase the energy efficiency and high performance of a CNN accelerator. Herein, the CNN accelerator is placed in close proximity to a CCD or CMOS sensor thereby eliminating the DRAM accesses for weights. The design has been implemented in 65 nm technology and it has consumed 4.86 mm^2 footprint. This accelerator is 30 times faster than the high performance GPUs with a power consumption of 320 nW.

Chen et al. (2017) have designed and realized a configurable, energy efficient accelerator for deep CNN. The architecture is being called as Eyeriss architecture. This architecture employs 168 PE array and realizes a four-level memory hierarchy. It employs a CNN dataflow, called Row Stationary (RS) algorithm, which is reconfigurable and energy efficient. The proposed Eyeriss is a network-on-chip (NoC) architecture and efficiently employs multicast technique and point-to-point single-cycle data delivery to support the RS dataflow. Eyeriss processes the convolutional layers at 35 frames/s. The block diagram of the Eyeriss architecture is given in Figure 13. Chen et al. (2017, 2019) have developed second version of Eyeriss, Eyeriss V2, which is more flexible, energy efficient and can be efficiently employed for emerging DNNs. Eyeriss v2 has a new dataflow, which enables spatial-tiling of data from all around. It utilizes the parallelism more efficiently and results in high performance. The new data flow is called the Row-Stationary Plus (RS+). The comparative analysis of original Eyeriss with Eyeriss v2 has shown a performance enhancement of 10.4 times to 17.9 times for 256 PEs, 37.7 times to 71.5 times for 1024 PEs, and 448.8 times to 1086.7 times for 16,384 PEs on DNNs with widely varying data reuse.

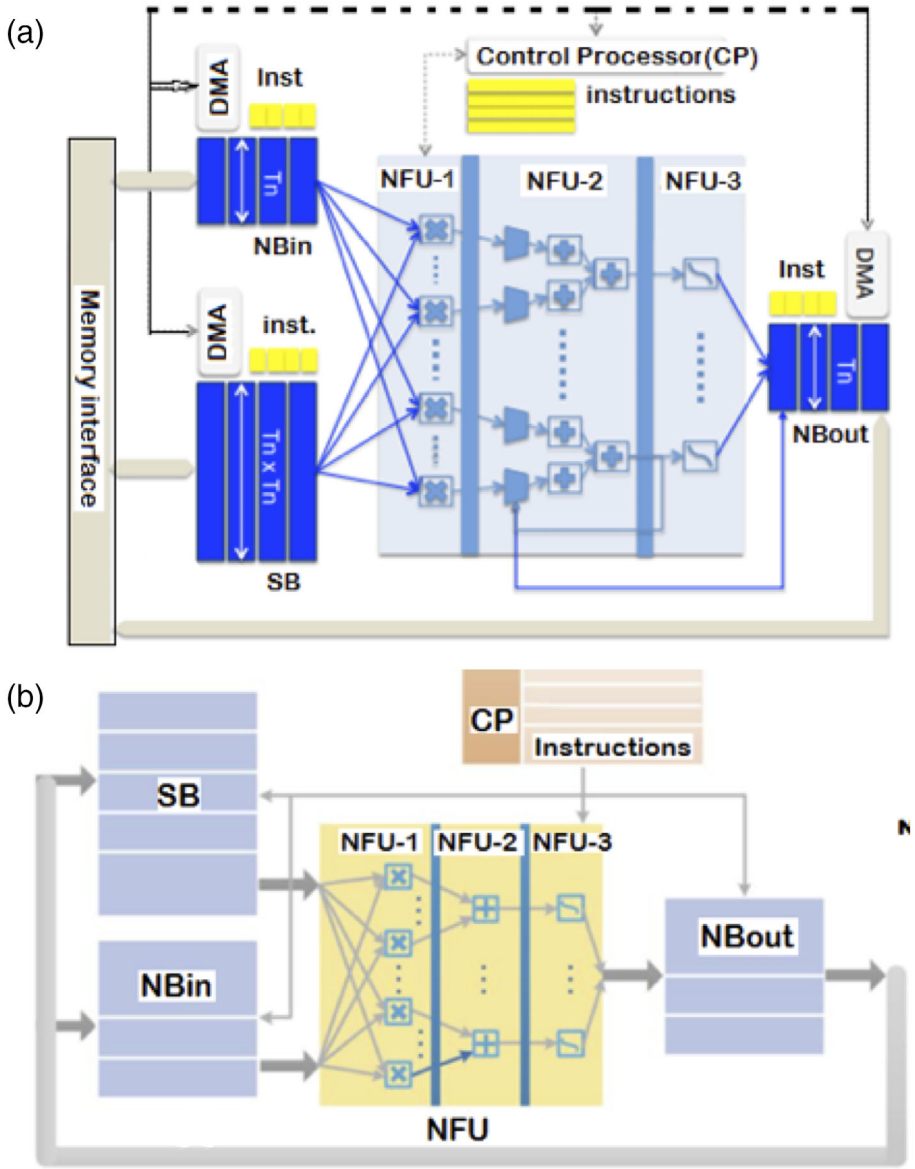

FIGURE 12 Accelerator architectures of (a) DianNao (Chen, Du, et al., 2014a) and (b) DaDianNao (Chen, Luo, et al., 2014b)

Reagen et al. (2016) have designed and developed a novel, low power DNN accelerator, called as Minerva. The block diagram of Minerva is shown in Figure 14. Minerva has five stages, with the first one governing training space exploration and also a network topology and training weights. Stage 2 governs an optimum accelerator implementation. The Stages 3–5 focuses on minimizing power consumption by reducing slack in data type, reducing memory accesses and MAC operations and aggressive SRAM supply voltage.

Wu et al. (2019) have designed and developed an energy efficient and high performance accelerator for sparse compressed CNNs. The unique things about the architecture are highly reduced DRAM accesses and the elimination of zero-operand computation. Figure 15 shows the architecture of the proposed accelerator. The various blocks include 8×8 PE array, 3-level accumulators, 8 Huffman decoders, activation encoder and a pooling module to form the non-zero (NZ) activations. Figure 16 shows the block diagram of processing element (PE), an important block of the architecture. The performance of the entire architecture is substantially dependent on the performance of PE. The simulation results reveal that the proposed devices achieves 1.79 times speed enhancement in comparison to a dense-CNN accelerator. Further, the on-chip memory size, energy consumption, and DRAM access have got significantly reduced by 23.51%, 69.53%, 88.67%, respectively, in comparison to a dense CNN accelerator.

Aimar et al. (2019) have proposed an energy efficient and flexible CNN accelerator architecture which can be used efficiently in low power/low latency application domains. The proposed architecture expedites computational speed and reduces the memory requirement significantly and has been implemented on an FPGA. The implementation results reveal that the proposed NullHop architecture reaches over 450 GOp/s, with

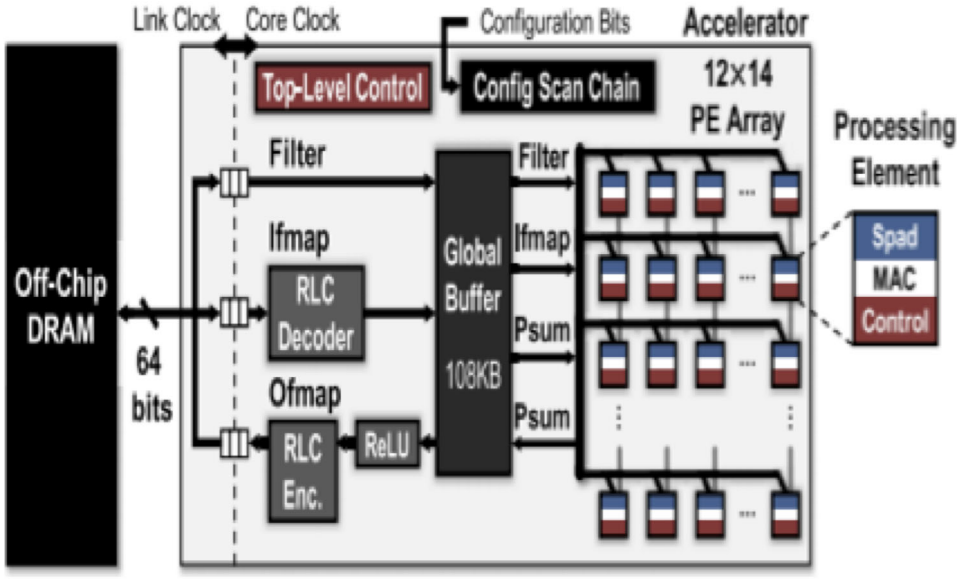

FIGURE 13 Eyeriss architecture (Chen et al., 2017)

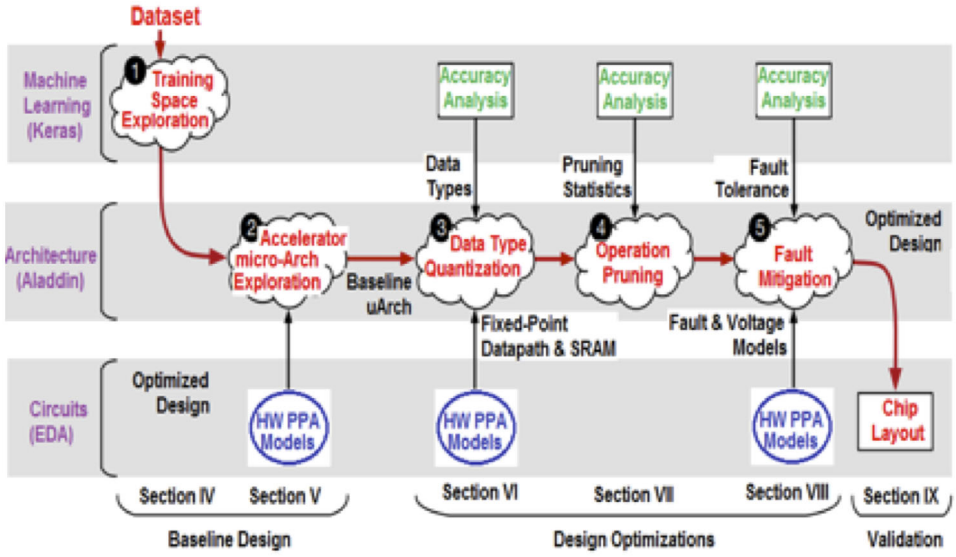

FIGURE 14 Block diagram of Minerva (Reagen et al., 2016)

an efficiency of 368% and maintaining 98% utilization of the MAC units. Further, NullHop is power efficient as over 3 TOp/s/W power efficiency in a core area of 5.8 mm^2 is achieved in the processor. Figure 17 shows the block diagram of the proposed accelerator, with important units like input data processor, Pooling-ReLU encoder and multiplication-accumulator (MAC) units. Figure 18 shows the chip place and route of Nullop architecture.

Zhao et al. (2017) have designed an accelerator for binarized neural networks (BNNs). BNNs are computationally efficient, require less memory and bitwise logic operations are the main computations performed by the architecture. The authors have performed the Verilog modelling of the proposed BNN and have implemented it on an FPGA. It has been found that the proposed BNN architecture outperforms the state-of-the-art FPFA based CNN accelerators in terms of energy efficiency, GOPs and resource requirement. The proposed BNN accelerator and its binary convolution unit are shown in Figure 19.

Guo et al. (2019) have also designed and implemented an accelerator for Binarized Neural Network. The designed BNN is energy efficient and has been efficiently implemented in an FPGA. An algorithmic optimization has been performed to binarize further the first layer and the padding bits of the proposed BNN. The proposed accelerator has been implemented on the Zynq-ZC702 board, and the experimentation has been performed on the SVHN and Cifar10 datasets. The experimental study reveals that the proposed one is outperforming the state-of-the-art BNNs

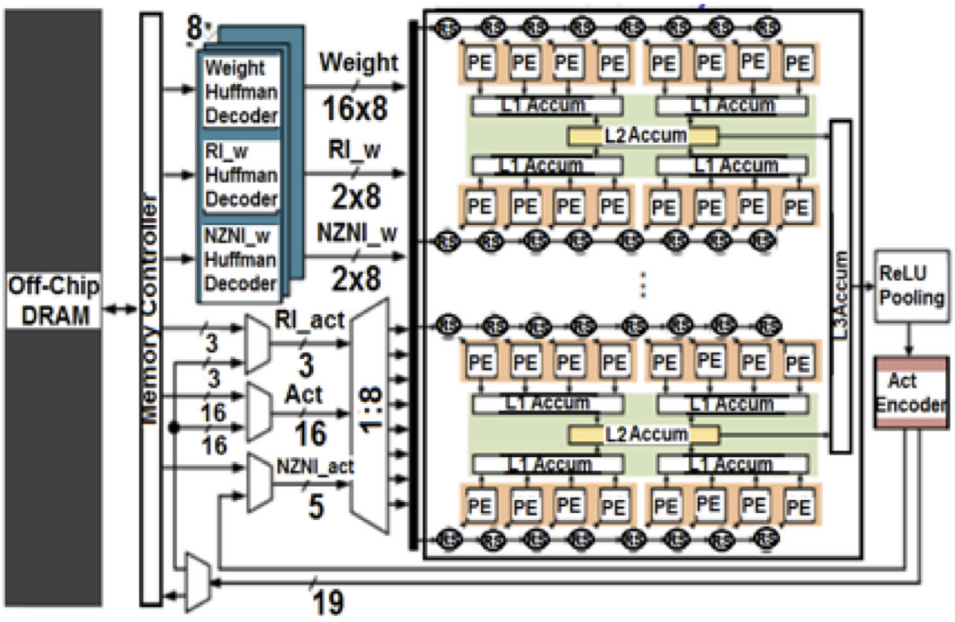

FIGURE 15 Block diagram of the proposed accelerator (Wu et al., 2019)

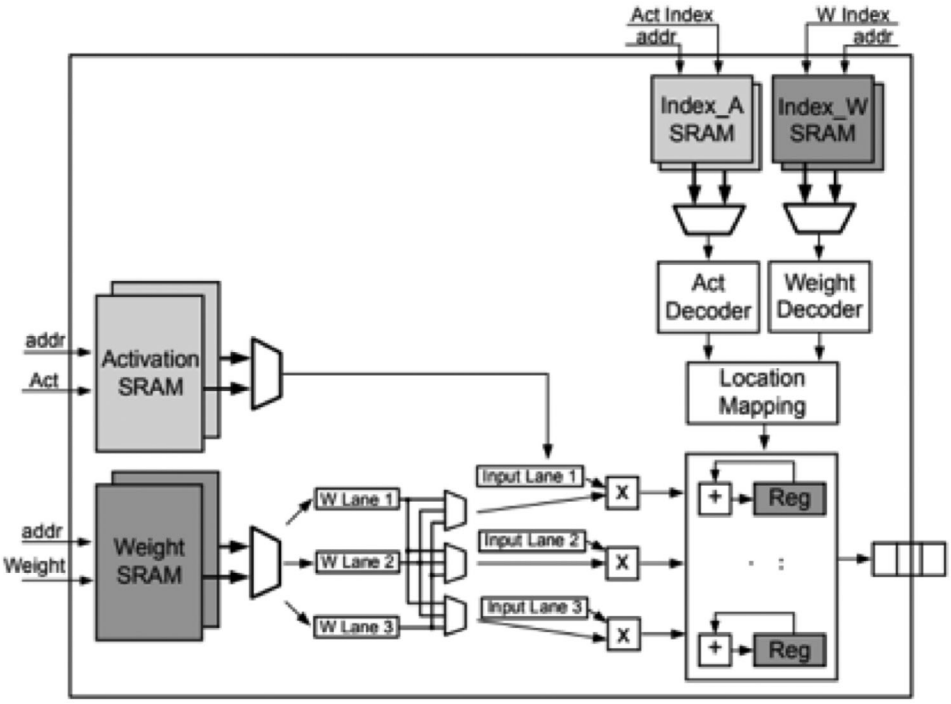

FIGURE 16 Architecture of the processing element (Wu et al., 2019)

architectures in terms of resource-efficiency. Figure 20 shows the block diagram of the proposed BNN accelerator. Its main components include the processing elements (PEs), local memory and the control unit.

Alwani et al. (2016) have designed and implemented fused layer CNN accelerators for the first time. In this design the fusion of CNN layers is performed to catch the intermediate data getting created during the evaluation process of adjacent CNN layers. In this work a fusion of first-five convolutional layers of the VGG-Net-E network has been performed and the fused architecture is compared with the Virtex-7 FPGA implemented state-of-the-art accelerators. It has been observed that with 362 KB of on-chip storage, the proposed fused layer accelerator reduces the off-chip feature-map data transfer significantly, achieving a reduction of 95% data transfer, from 77 MB to 3.6 MB/image. Figure 21 shows the fusion of two convolution layers, referred as Layer 1 and layer 2. Layer 1 is consists of M filters of $3 \times 3 \times N$ weights and Layer 2 is consists of P filters of

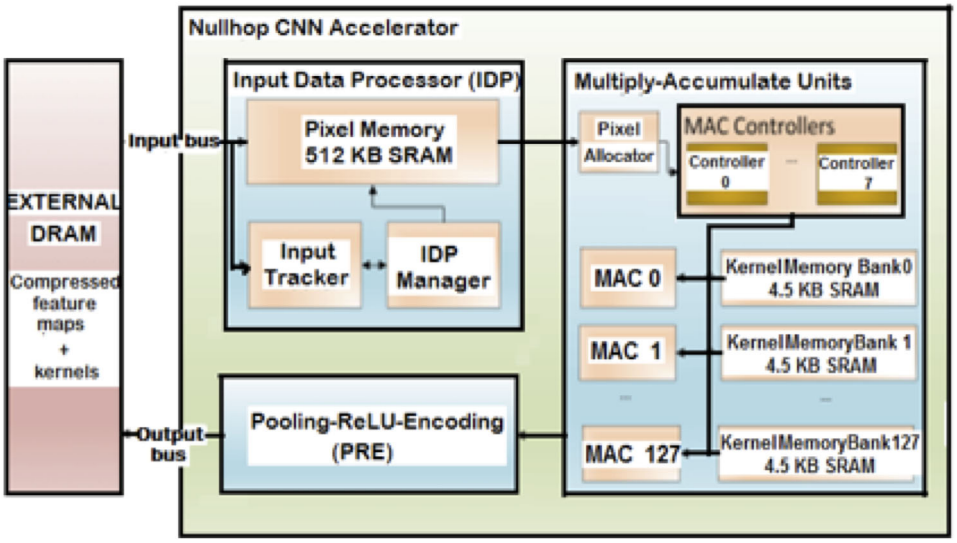

FIGURE 17 Block diagram of the proposed null-hop CNN accelerator (Aimar et al., 2019)

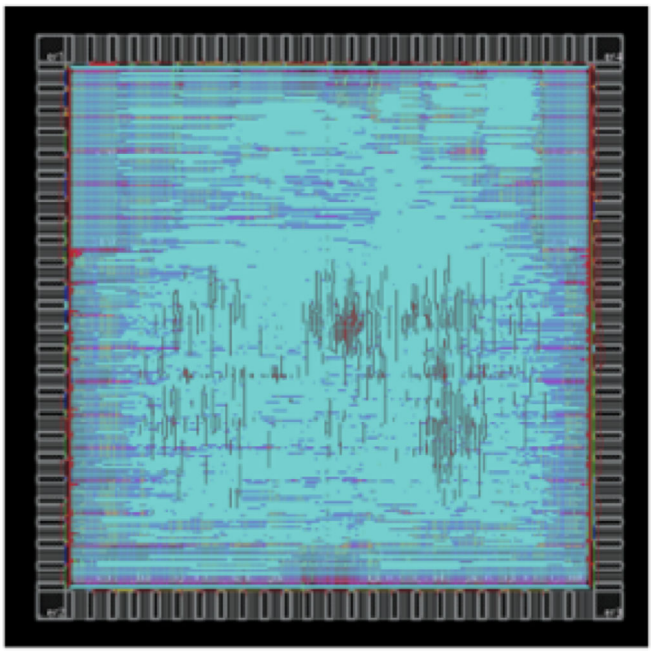

FIGURE 18 Null-hop chip place and route (Aimar et al., 2019)

3 × 3 × M weights. The layer 1 performs operations on 5 × 5× N input values, a tile of its input feature maps and then performs convolution on all M of its filters (each 3 × 3× N) across this tile and hence producing the 3 × 3 × M region. Then, the Layer 2 operates on these 3 × 3 × M values to produce 1 × 1 × P outputs in the output feature maps.

Esmaeilzadeh et al. (2014) have designed a new class of programmable accelerators called as the neural processing unit (NPU). The NPUs are designed in such a way so as to accelerate just a part of a program instead of executing the whole program on a CPU. The NPU has a simple architecture, with eight processing elements. Each PE behaves like a neuron and performs all the operations of a neuron, like accumulation, multiplication, and a sigmoid function. An NPU can reduce dynamic CPU instructions by up to 97% and achieve speed enhancement up to 11.1 times. Figure 22 shows the important blocks of NPU and the architecture of a single PE used in NPUs.

Lee, Kim, et al. (2018b) have designed and fabricated a neural network processor called as unified neural processing unit (UNPU). The proposed architecture supports convolutional layers (CL), fully-connected layers (FCLs) and recurrent layers (RLs), that too with fully-variable weight bit-precision from 1 b to 16 b. A 65 nm CMOS technology has been used to fabricate the proposed UNPU. It occupies 16 mm^2 die area.

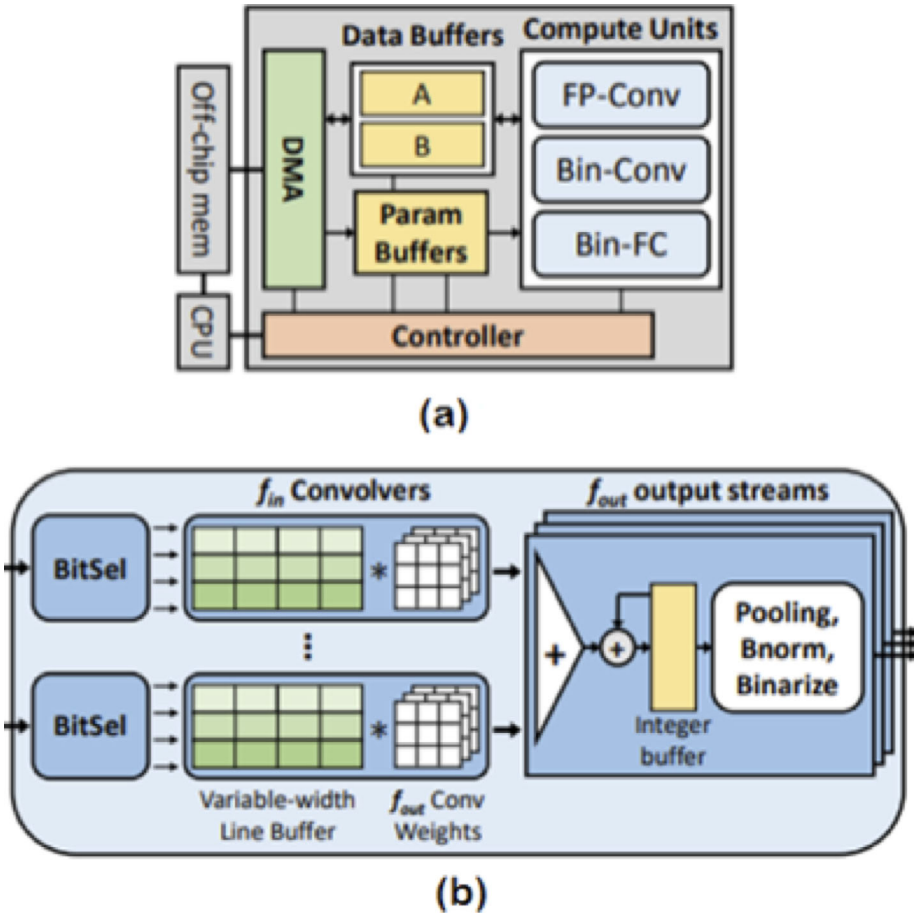

FIGURE 19 (a) System level block diagram of the proposed BNN accelerator; (b) architecture of the binary convolution unit of BNN accelerator (Zhao et al., 2017)

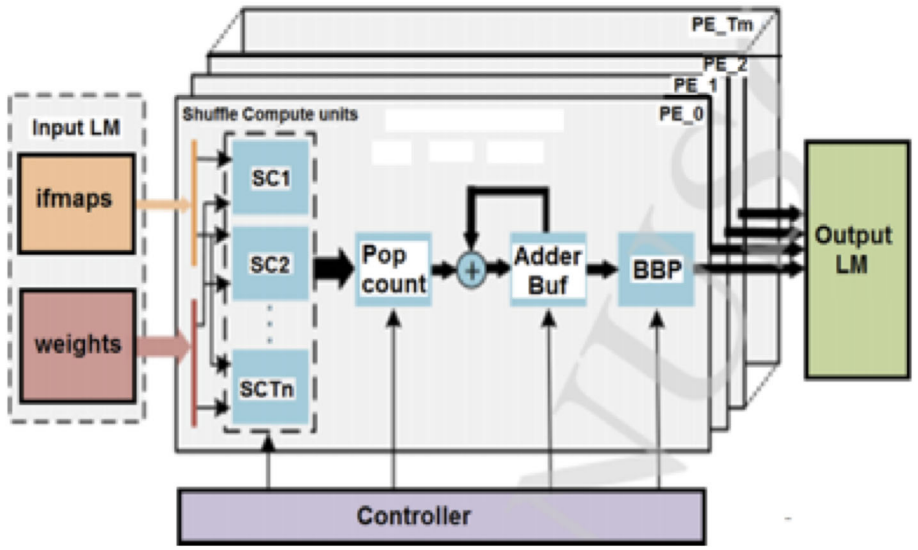

FIGURE 20 Block diagram of the proposed BNN accelerator (Guo et al., 2019)

Hu et al. (2019) have designed an efficient configurable accelerator for DNN. An array based structure is proposed with four level processing elements. The proposed architecture realizes highly parallel convolution operation calculations, employs hybrid stationary (HS) storage pattern and takes full advantage of off-chip memory bandwidth and on-chip memory footprint. The proposed architecture performs 113 G-ops/s at

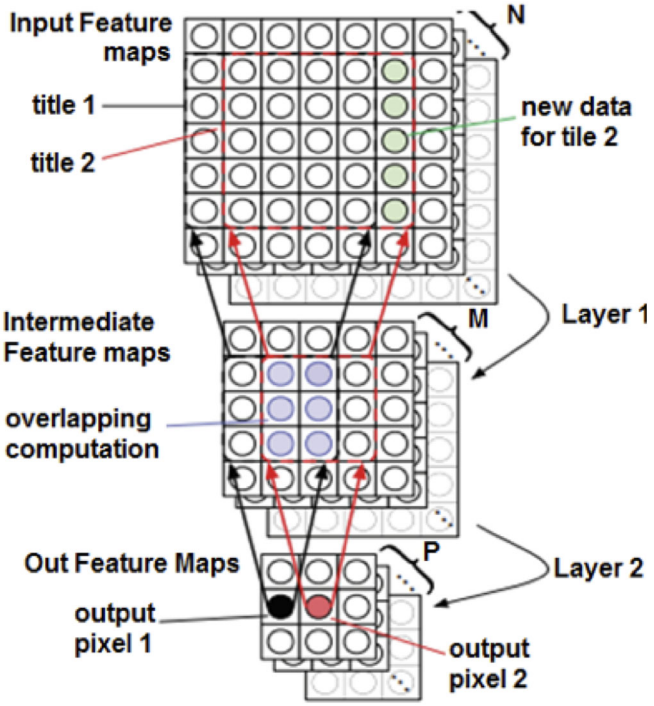

FIGURE 21 Example of fusing two convolutional layers (Alwani et al., 2016)

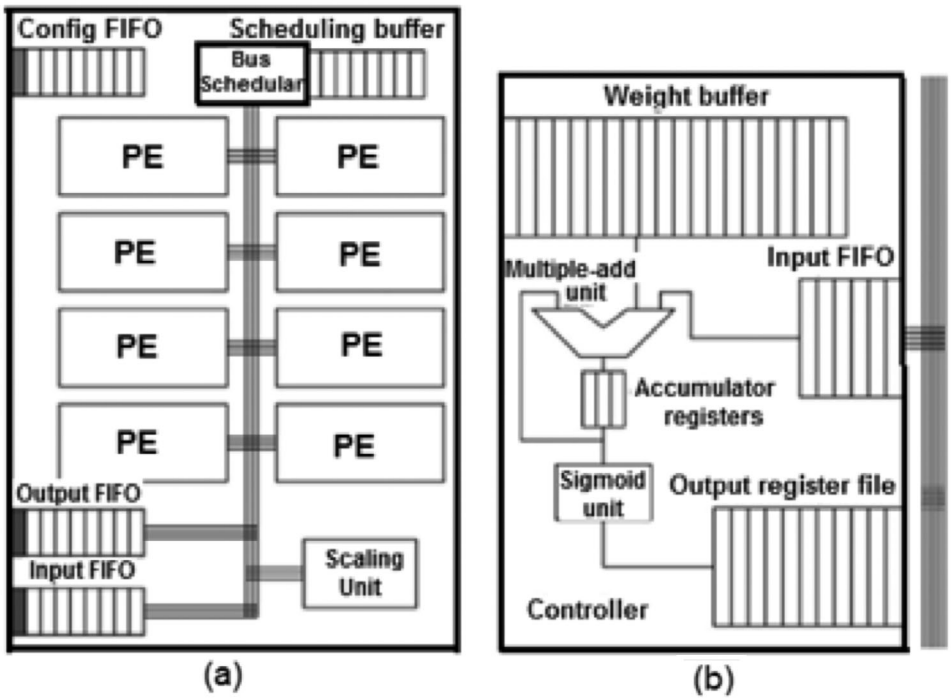

FIGURE 22 (a) The block diagram of NPU with eight-PE NPU; (b) single PE architecture (Esmaeilzadeh et al., 2014)

100 MHz. The FPGA implementation on ZYNQ-7 ZC706 board reveals that the proposed architecture consumes 784 DSP48 modules and 211.5 Block RAM modules. Figure 23 shows the block diagram of the accelerator with MAC1, MAC2, MAC3 and MAC4. It has been observed that the DCNN accelerator consumes 5.905 watts of power, attains 100.7 G-OPS/S average performance on AlexNet and 113.28 G-OPS/S on VGG-16, with an operating frequency of 100 MHz.

X. Liu, Mao, et al. (2015a) have proposed a reconfigurable neuromorphic computing accelerator architecture for neural networks, called as RENO.

The RENO architecture employs resistive RAM (ReRAM) based processing element design and each PE contains four ReRAM-crossbars. The ReRAM crossbar is a basic computational block and performs matrix–vector multiplications. In the RENO architecture, dedicated routers are employed to realize data transfer between the PEs, with digital I/O data and the intermediate data is analogue in nature. Figure 24 is the architecture of the proposed RENO accelerator. It is an on-chip design, hence limited applications are supported. It processes small datasets, like UCI ML (2020) repository (Sze et al., 2020; http://archive.ics.uci.edu/ml/) and the tailored MNIST database (LeCun, Y et al. 2019).

LeCun Y et al. (2019) have designed and developed a custom ASIC processor, called as Tensor Processing Unit (TPU). The TPUs are used in data centres and accelerate the inferencing part of neural networks. The important part of the TPU is a ∼65 K, 8-bit MAC matrix multiplication unit, achieving a maximum throughput of 92 TO/s (TOPS) and a huge software-managed on-chip memory (28 MiB). The TPU is about 15 to 30 times faster than its contemporary processors, CPU or GPU, with TOPS/Watt about 30 to 80 times higher. The use of GPU's GDDR5 memory in the TPU would almost triple the TOPS and raises TOPS/Watt to nearly 200× the CPU and 70 times the GPU. Figure 25 shows the block diagram of the proposed TPU. Its main processing units are the yellow Matrix-Multiply unit, Weight FIFO as weight fetcher, unified buffer for local storage and Accumulators for output. Further, its activation unit performs the nonlinear functions on the data in accumulators. After TPU development, Google announced a new TPU, a cloud TPU. It is also known as Tensor Processor Unit 2 (TPU2) (https://events.google.com/io2018/). TPU2 is designed to manage training and inference processes in the data centre. In 2018, Google announced a new TPU, known as Tensor Processor Unit 3 (TPU3), Google I/O'18 (2019) (https://cloud.withgoogle.com/next18/sf/) (Google I/O'17 (2019), Google I/O'18 (2019)). Apart from

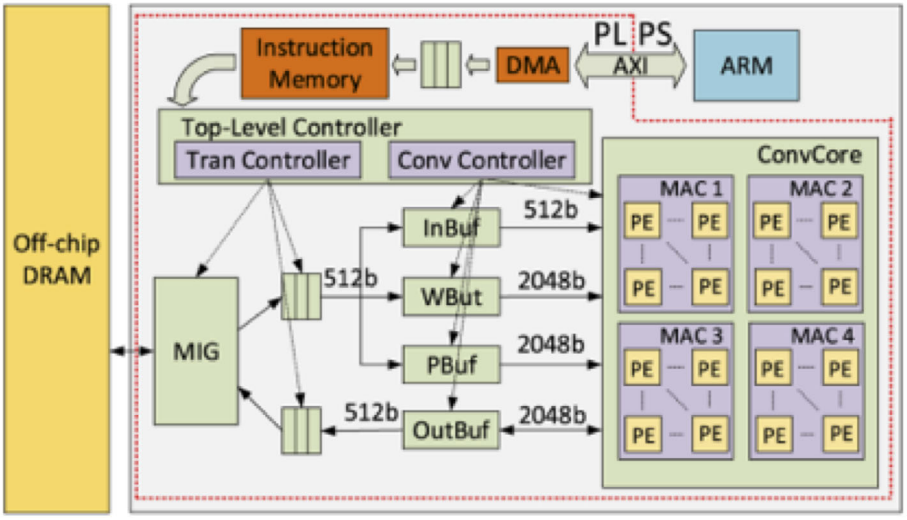

FIGURE 23 DCNN accelerator with 4 MACs (Hu et al., 2019)

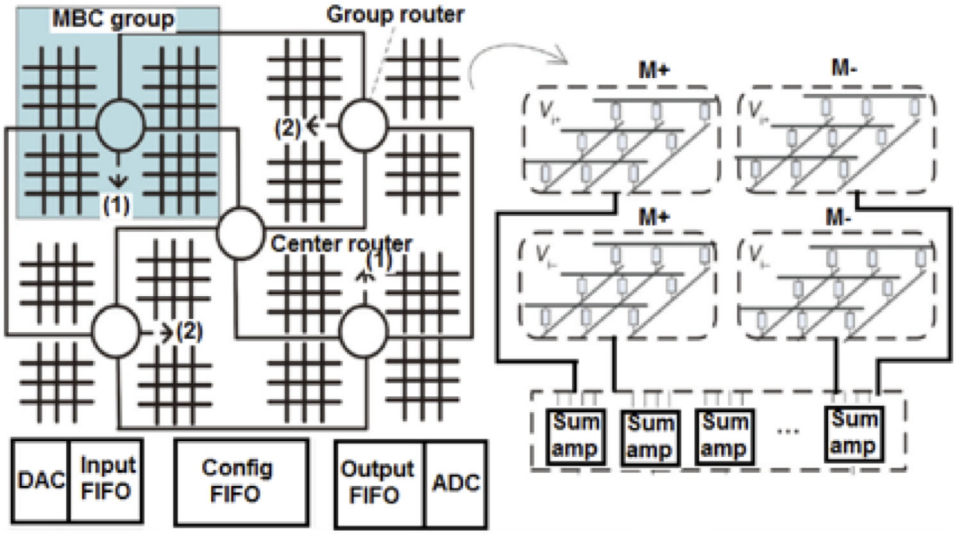

FIGURE 24 RENO architecture (Liu, Mao, et al., 2015a)

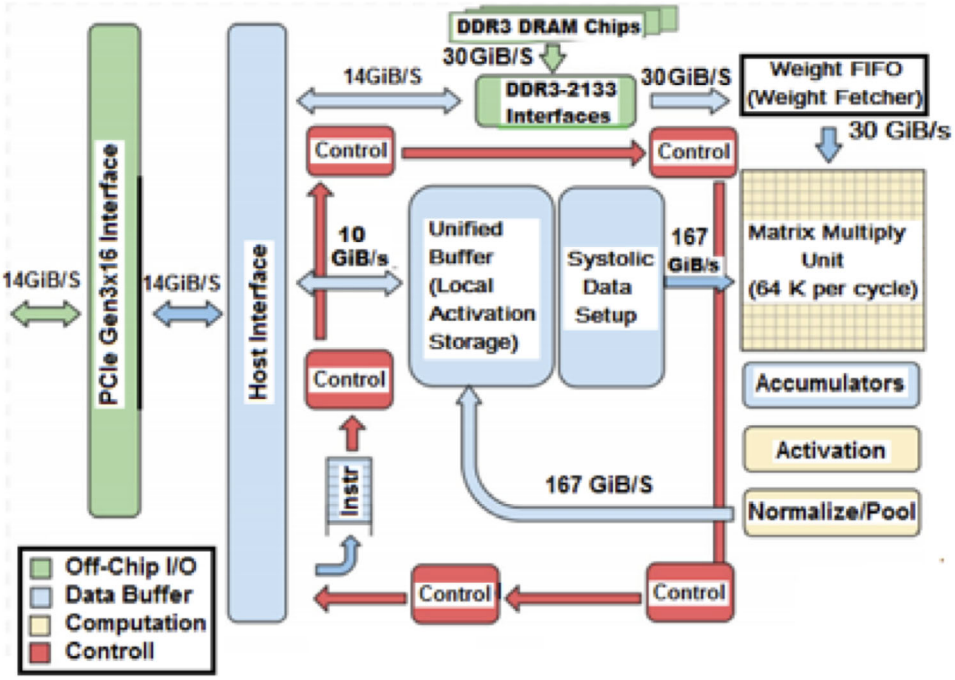

FIGURE 25 TPU block diagram (LeCun Y et al., 2019)

working at the architecture level to improve DNN accelerator performance, the state of the art device level technologies have been used to improve DNN accelerators. The concept of memristor based emerging memories like resistive RAM (ReRAM) (Bojnordi et al., 2016; Chen et al., 2018; Chi et al., 2016; Dongale et al., 2015; Qiao et al., 2018; Song et al., 2017) and hybrid memory cube (HMC) (Pawlowski, 2011) technology have been used to improve DNN accelerators. These memories are able to perform processing in memory (PIM) functions which greatly reduce the data movement between CPU and the off-chip memory. P. Chi et al. (2016) have designed and developed PRIME, a new DNN architecture. This architecture is novel and computationally efficient and is employing memristor (resistive RAM) based main memory. PIM is considered a potential solution to the "memory wall" challenge of future computing system, as PIM structures keep additional computational logic in or near memory (Wulf & McKe, 1995). In PRIME architecture, a segment of ReRAM crossbar arrays can realize accelerators for DNN applications. The combination of PIM architecture and the efficient ReRAM for DNN computation results in a significant improvement in performance enhancement in PRIME in comparison to the state of the art CNN architectures. The experimental results show that PRIME architecture achieves performance enhancement by $\sim$2360$\times$ and the energy consumption gets reduced by $\sim$895$\times$. Figure 26 shows the block diagram of the PRIME architecture. On the left is the bank structure, with blue lines representing the conventional data flow from the normal memory and the red lines represent computation. Right side shows various functional blocks in the PRIME architecture (Wan et al., 2013; Wong et al., 2012; Wu et al., 2019; Xue et al., 2019).

Song et al. (2017) have also used emerging memory (resistive RAM) in proposing a novel deep learning accelerator, called as Pipelayer architecture. This architecture has addressed the issues of PRIME (Chi et al., 2016) and ISAAC (Shafiee et al., 2016) architectures, like inefficient data organization and kernel mapping in PRIME and pipeline bubbles in ISAAC. The PipeLayer is a ReRAM-based processing in memory accelerator for CNNs, helping in both training and testing in CNNs. Experimentally, it has been observed that the proposed accelerator has better speed (42.45$\times$) compared with the conventional GPU platform on average and is energy efficient with the average energy saving of $\sim$7.17$\times$ in comparison to conventional GPU. Chen et al. (2018) have designed and proposed a pipelined resistive RAM based accelerator for generative adversarial networks (GAN). GAN architectures provide high performance, however, at the cost of complexity.

The GAN accelerator architecture employs ReRAM emerging memory and has processing in memory feature. A significant reduction in off-chip memory access and a significant throughput enhancement are achieved by optimizing the pipelining computation in the accelerator. Two prominent techniques, computation sharing and spatial parallelism, have been used to enhance the training efficiency of GAN architecture significantly. The experimental results reveal that the proposed ReGAN can result in 240 times speed enhancement, along with 94 times energy saving in comparison to the GPU platform. Qiao et al. (2018) have proposed a novel architecture of a CNN accelerator, called as Atom-layer architecture. It is again a ReRAM based architecture with atomic layer computation, supports efficiently both CNN training and inference. The atomic layer computation processes one network layer each time and hence addresses the latency, pipeline bubble and on-chip buffer issues. Experimentally, it has been found that the Atom-layer is power efficient than ISSAC inference by 1.1 times and Pipelayer in training by 1.6 times. It reduces footprint by 15 times. Gao et al. (2018) have developed a power efficient recurrent NN accelerator, named as DeltaRNN (DRNN) architecture. An

important observation about RNN is that it can reduce memory access computation and is highly accurate. The DRNN have been implemented on a Xilinx Zynq-7100 FPGA. The implementation report shows that the proposed DRNN architecture achieves 1.2 TOp/s throughput and a power efficiency of 164 GOp/s/W. A 5.7× speedup is achieved in comparison to the conventional RNN. Song et al. (2019) have proposed a novel architecture, HyPar, invoking hybrid parallelism for deep learning (DL) accelerator array designing. HyPaR architecture divides various tensors, like kernel, gradient, error and feature map tensors for DNN accelerators. The performance of the HyPar architecture has been compared with ten earlier designed accelerator architectures, classic Lenet to modern VGGs. It has been observed that HyPar architecture performs better due to the hybrid parallelism than either by data parallelism or model parallelism in earlier accelerators. The implementation has shown a performance and efficiency gain of 3.39× and of 1.51× respectively compared to the data parallelism on average. Further, it has been found that the HyPaR outperforms up to 2.40 times the "one weird trick". Recently, DNN accelerators have been designed and developed using spintronic memories and analogue memory devices (Hu et al., 2012; Hu et al., 2016; Zang et al., 2020). Wang et al. (2018) have designed and developed an efficient

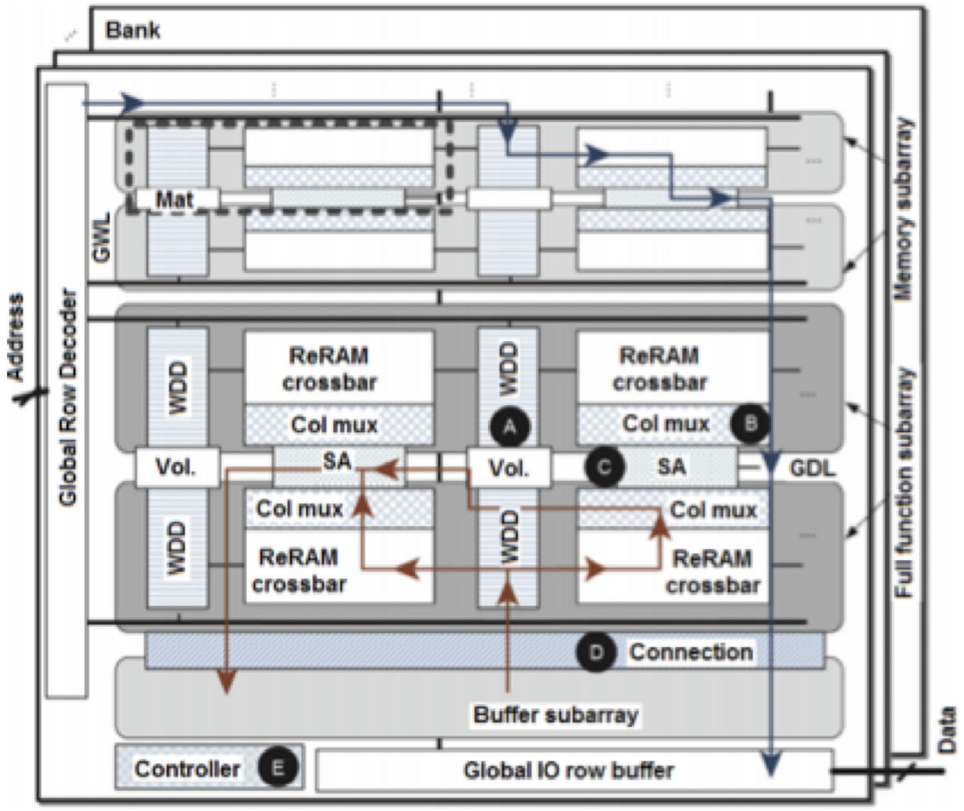

FIGURE 26 The proposed PRIME architecture (Chi et al., 2016)

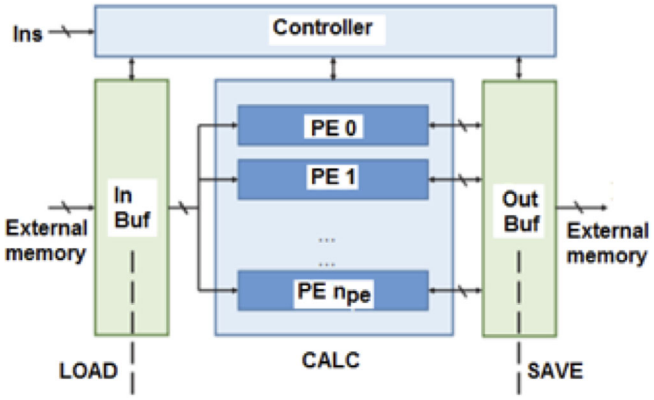

FIGURE 27 Proposed angel-eye architecture (K. Guo et al., 2016, 2018)

sparse NN computational architecture employing ReRAM, called as SNrram. The computational efficiency is being enhanced here by exploiting the sparsity in both weight and activation. SNrram utilizes resources efficiently by storing and organizing non trivial weights and eliminates zero-value multiplications for better resource utilization. The experimental results show that the proposed SNrram architecture saves RRAM resources by ∼70%, increases speed by ∼2.5 times and reduces the power consumption by ∼36%, in comparison to the state-of-the-art RRAM-based NN accelerator. K. Guo et al. (2016, 2018) have proposed an energy and computationally efficient CNN architecture, called as Angel-Eye. Angel-Eye

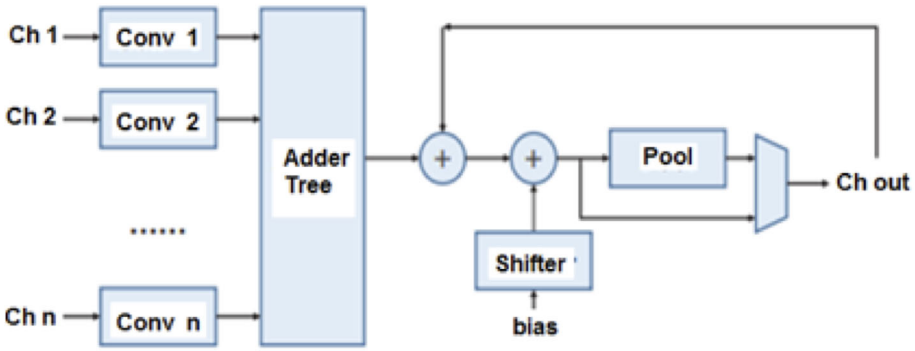

FIGURE 28 Structure of a single processing element (PE) (K. Guo et al., 2016, 2018)

TABLE 1 Comparative analysis of various DNN accelerators

Accelerator	Approach	Energy efficiency	Throughput	Footprint	Power consumption	Technology node	Speed up
YodaNN1 (Andri et al., 2017)	Optimization of binary weights in CNNs.	5.1× as compared to 12 bit MAC units operating at 1.2v	1.5 TOP/s at 1.2v	1.99 mm^2	895 μW	65 nm at 0.6v	NA
DianNao (Chen, Du, et al., 2014a)	Memory optimization.	21.08× energy efficient than 2 GHz SIMD processor.	452 GOP/s	3.02mm^2	NA	28 nm	117.87× faster than 2 GHz SIMD processor
DaDianNAO (Chen, Luo, et al., 2014b)	Custom multichip architecture for CNNs and DNNs.	150.31× energy efficient than a single GPU.		0.78mm^2	NA	28 nm	460.6× faster than GPU
ShiDianNao (Du et al., 2015)	Employing weight sharing property for the state of art visual recognition problems.	4688.13× more energy efficient than DianNao.	194 GOP/s	4.86mm^2	320 mW	65 nm	60× faster than DianNao
PuDianNao (Liu, Chen, et al., 2015b)	Seven machine learning techniques used; k-means, k-nearest neighbours, naive bayes, SVM, linear regression, classification tree.	Reduced energy consumption by 128.4× than NVIDIA K20M GPU	1056 GOP/s	3.51 mm^2	596 mW	65 nm	120× faster than NVIDIA K20M GPU
Origami (Cavigelli & Benini, 2017)	Improves the external memory bottleneck of previous architectures.	803 GOP/s/W (power efficiency)	196 GOP/s	3.09mm^2	93 mW	65 nm	NA
Sparse CNN (Parashar et al., 2017)	Improves performance by exploiting zero valued weights	2.3× over a dense CNN accelerator	2.7× over a dense CNN accelerator	7.4 mm^2	NA	16 nm	NA
ISAAC (Shafiee et al., 2016)	In situ processing approach using memristor crossbar array	5.5× more energy efficient than DianNao	1707 GOP/s mm^2 with respect to DianNao		65.8 W (due to high computational density)	32 nm	Computational density 7.5× than DIANnao
NullHop (Aimar et al., 2019)	Exploits the sparsity of neuron activity in CNNS to accelerate computation and reduce memory requirements	3 Top/s/W (power efficiency)	450 GOP/s	5.8mm^2	NA	28 nm	NA

is a high performance, flexible and programmable CNN accelerator architecture with some unique features like data quantization strategy to compress the data bit-width in CNN. The FPGA implementation of the Angel Eye architecture has been performed on Zynq XC7Z045 platform. It is has been observed that the Angel-Eye architecture is power efficient (5 times) and faster (6 times) than the state of the art CNN accelerators. Figure 27 is the block diagram of the proposed Angel-Eye architecture, with four important parts, processing element array, external memory, on-chip buffer and controller. Processing element is an important part and is shown in Figure 28. The PE realizes the convolution operation in CNN and implements three parallelisms; (a) Kernel level parallelism. (b) Input channel parallelism. (c) Output channel parallelism.

Table 1 gives the comparative analysis of various performance measuring parameters of some well-known DNN accelerators. It clearly shows the approach used in the design of each DNN accelerator, power consumption, throughput, technology node and the area requirement.

4 | CONCLUSION

In this work, we present the state of the art of deep neural network accelerators employed in various artificial intelligence and machine learning applications. The availability of large data in the form of videos, audios, text, medical reports, military surveillance data and the recent development in high-end high performance processors have provided a strong motivation for the development in AI and machine learning domains. However, things are not easy as AI and ML faces several challenges like processing speed, huge memory requirement, large bandwidth requirement, slow memory access, and requirement of highly conductive and flexible interconnections between processing units and the memory blocks. So a lot of work has been done to improve the performance measuring parameters of deep neural network accelerators. Work has been done at various abstraction levels from algorithmic to device levels to address these issues in DNN accelerators. Various DNN accelerator architectures, their computing units and various emerging technologies have been discussed. However, there is a significant scope for the further improvement in these accelerator designs. Further, some future research problems can be (a) designing and developing DNN models when small number of training data points are available, (b) developing models to apply unsupervised, semi-supervised DNN models for complex applications, (c) design and development of DNN based mobile communication chips, and (d) performing stability analysis of DNN based AI and ML applications.

ACKNOWLEDGEMENT

M. Saqib Akhoon would like to acknowledge the support from Graduate Assistance Fellowship from the Universiti Sains Malaysia (USM), Malaysia.

CONFLICT OF INTEREST

The authors declare there is no conflict of interest.

DATA AVAILABILITY STATEMENT

The data that support the findings of this study are available from the corresponding author upon reasonable request.

ORCID

Sajad A. Loan https://orcid.org/0000-0002-3936-3947

REFERENCES

Andri, R., Cavigelli, L., Rossi, D., & Benini, L. (2017). Yoda NN: An architecture for ultra-low power binary-weight CNN acceleration. *IEEE Transactions on Computer Aided Design of Integrated Systems*, *37*(1), 48–60.

Aimar, A., Mostafa, H., Calabrese, E., Rios-Navarro, A., Tapiador-Morales, R., Lungu, I. A., Milde, M. B., Corradi, F., Linares-Barranco, A., Liu, S. C., & Delbruck, T. (2019). NullHop: A flexible convolutional neural network accelerator based on sparse representations of feature maps. *IEEE Transactions on Neural Networks and Learning Systems*, *30*(3), 644–656. https://doi.org/10.1109/TNNLS.2018.2852335

Alwani, M., Chen, H., Ferdman, M., & Milder, P. (2016). Fused-layer CNN accelerators. In *Proceedings of MICRO* (pp. 1–12). IEEE.

Ambrogio, S., Balatti, S., Cubeta, A., Calderoni, A., Ramaswamy, N., & Ielmini, D. (2013). Understanding switching variability and random telegraph noise in resistive RAM. In *Proceedings of the 2013 IEEE International Electron Devices Meeting*; Washington, DC, USA (pp. 31.5.1–31.5.4).

Ambrogio, S., Narayanan, P., Tsai, H. & Mackin, C. (2020). Accelerating deep neural networks with analog memory devices. In 2nd IEEE International Conference on Artificial Intelligence Circuits and Systems (AICAS). IEEE.

Bain, A. (1873). *Mind and body: The theories of their relation*. Henry s. King and Company.

Bojnordi, M. N., & Ipek, E. (2016). Memristive Boltzmann machine: A hardware accelerator for combinatorial optimization and deep learning. In *Proceedings of the 2016 IEEE International Symposium on High Performance Computer Architecture*; Barcelona, Spain (pp. 1–13). IEEE.

Chen, C., Seff, A., Kornhauser, A., & Xiao, J. (2015). Deepdriving: Learning affordance for direct perception in autonomous driving. In *Proceedings of the 2nd international conference on application of intelligent systems in Proc. ICCV* (pp. 2722–2730).

Caulfield, A. M., Chung, E. S., Putnam, A., Angepat, H., Fowers, J., Haselman, M., Heil, S., Humphrey, M., Kaur, P., Kim, J.-Y., Lo, D., Massengill, T., Ovtcharov, K., Papamichael, M., Woods, L., Lanka, S., Chiou, D., & Burger, D. (2016). A cloud-scale acceleration architecture, Microsoft Corporation. In *49th Annual IEEE/ACM International Symposium on Microarchitecture (MICRO)* (pp. 1–13). IEEE.

Cavigelli, L., & Benini, L. (2017). Origami: A 803-GOp/s/W convolutional network accelerator. *IEEE Transactions on Circuits and Systems for Video Technology*, *27*(11), 2461–2475. https://doi.org/10.1109/TCSVT.2016.2592330

Chen, T., Du, Z., Sun, N., Wang, J., Wu, C., Chen, Y., & Temam, O. (2014a). Dian Nao: A small-footprint high-throughput accelerator for ubiquitous machine-learning. In *Proceedings of the 19th International Conference on Architectural Support for Programming Languages and Operating Systems; 2014 March 1–5;* Salt Lake City, UT, USA (pp. 269–284).

Chen, Y., Luo, T., Liu, S., Zhang, S., He, L., Wang, J., Li, L., Chen, T., Xu, Z., Sun, N., & Temam, O. (2014b). DaDianNao: A machine-learning supercomputer. In *Proceedings of the 47th Annual IEEE/ACM International Symposium on Microarchitecture; 2014 Dec 13–17*; Cambridge, UK (p. 609–622).

Chen, Y.-H., Krishna, T., Emer, J. S., & Sze, V. (2017). Eyeriss: An energy-efficient reconfigurable accelerator for deep convolutional neural networks. *IEEE Journal of Solid-State Circuits*, *52*(1), 127–138.

Chen, Y.-H., Yang, T.-J., Emer, J., & Sze, V. (2019). Eyeriss v2: A flexible accelerator for emerging deep neural networks on mobile devices. *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, *9*(2), 292–308.

Chi, P., Li, S., Xu, C., Zhang, T., Zhao, J., Liu, Y., Wang, Y., & Xie, Y. (2016). PRIME: a novel processing-in memory architecture for neural network. Computation in ReRAM-based main memory. *SIGARCH Computer Architecture News*, *44*(3), 27–39.

Chen, F., Song, L., & Chen, Y. (2018). ReGAN: A pipelined ReRAM-based accelerator for generative adversarial networks. In *2018 23rd Asia and South Pacific Design Automation Conference (ASP-DAC)*, Jeju, Republic of Korea, (pp. 178–183). IEEE. https://doi.org/10.1109/ASPDAC.2018.8297302

Chen, A., & Lin, M. (2011). Variability of resistive switching memories and its impact on crossbar array performance. In *Proceedings of the 2011 International Reliability Physics Symposium*; Monterey, CA, USA (pp. MY.7.1–MY.7.4). IEEE. https://doi.org/10.1109/IRPS.2011.5784590

Deng, L., Li, J., Huang, J.-T., Yao, K., Yu, D., Seide, F., Seltzer, M., Zweig, G., He, X., Williams, J., Gong, Y., & Acero, A. (2013). Recent advances in deep learning for speech research at Microsoft. In *Proceedings of ICASSP* (pp. 8604–8608). IEEE.

Dongale, T. D., Patil, K. P., Mullani, S. B., More, K. V., Delekar, S. D., Patil, P. S., Gaikwad, P. K., & Kamat, R. K. (2015). Investigation of process parameter variation in the memristor based resistive random access memory (RRAM): Effect of device size variations. *Materials Science in Semiconductor Processing*, *35*, 174–180.

Du, Z., Fasthuber, R., Chen, T., Ienne, P., Li, L., Luo, T., Feng, X., Chen, Y., & Temam, O. (2015). Shi Dian Nao: Shifting vision processing closer to the sensor. In *ACM/IEEE 42nd Annual International Symposium on Computer Architecture (ISCA)*; Portland, OR, USA (pp. 92–104). ACM/IEEE. https://doi.org/10.1145/2749469.2750389

Esteva, A., Kuprel, B., Novoa, A., Ko, J., Swetter, S. M., Blau, H. M., & Thrun, S. (2017). Dermatologist-level classification of skin cancer with deep neural networks. *Nature*, *542*(7639), 115–118.

Esmaeilzadeh, H., Sampson, A., Ceze, L., & Burger, D. (2014). Neural acceleration for general purpose approximate programs. *Communications of the ACM*, *58*(1), 105–115.

Graham, B. (2014). Fractional Max-Pooling. *ArXiv*, *1412*, 607.

Guo, P., Ma, H., Chen, R., & Wang, D. (2019). A high-efficiency FPGA-based accelerator for binarized neural networks. *Journal of Circuits, Systems and Computers*, *28*(1), 1–21. https://doi.org/10.1142/S0218126619400048

Google I/O'17 (2019). [Internet]. Google. https://events.google.com/io2017/.

Google I/O'18. (2019). [Internet]. Google. https://events.google.com/io2018/.

Google Cloud Next'18. (2019). [Internet]. Google. https://cloud.withgoogle.com/next18/sf/.

Gao, C., Neil, D., Ceolini, E., Liu, S. C., & Delbruck, T. (2018). Delta RNN: A power-efficient recurrent neural network accelerator. In *Proceedings of the 2018 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays*; Monterey, CA, USA (pp. 21–30). Proceedings of ACM.

Guo, K., Sui, L., Qiu, J., Yu, J., Wang, J., Yao, S., Han, S., Wang, Y., & Yang, H. (2018). Angel-eye: A complete design flow for mapping CNN onto embedded FPGA. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, *37*(1), 35–47. https://doi.org/10.1109/TCAD.2017.2705069

Guo, K., Sui, L., Qiu, J., Yao, S., Han, S., Wang, Y., & Yang, H. (2016). Angel-eye: A complete design flow for mapping CNN onto customized hardware. In *2016 IEEE Computer Society Annual Symposium on VLSI (ISVLSI)* (pp. 24–29). IEEE. https://doi.org/10.1109/ISVLSI.2016.129

Hu, X., Zeng, Y., Li, Z., Zheng, X., Cai, S., & Xiong, X. (2019). A resources-efficient configurable accelerator for deep convolutional neural networks. *IEEE Access*, *7*.72113–72124.

Hu, M., Li, H., Wu, Q., & Rose, G. S. (2012). Hardware realization of BSB recall function using memristor crossbar arrays. In *Proceedings of the 49th Annual Design Automation Conference; 2012 Jun 3–7*; (pp. 498–503). San Francisco, CA, USA.

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (pp. 770–778). IEEE. https://doi.org/10.1109/CVPR.2016.90

Huang, G., Liu, Z., Van Der Maaten, L., & Weinberger, K. Q. (2017). Densely connected convolutional networks. In *Conference on Computer Vision and Pattern Recognition (CVPR)* (237). IEEE Explore. https://doi.org/10.1109/cvpr.2017.243.33

Hu, M., Strachan, J. P., Li, Z., Grafals, E. M., Davila, N., Graves, C., Lam, S., Ge, N., Yang, J. J., & Stanley Williams, R. (2016). Dot-product engine for neuromorphic computing: Programming 1T1M crossbar to accelerate matrix–vector multiplication. In *2016 53nd ACM/EDAC/IEEE Design Automation Conference (DAC)*; Austin, TX, USA (pp. 1–6). https://doi.org/10.1145/2897937.2898010

James, W. (1890). *The principles of psychology* (Vol. 1). Henry Holt and Co.

Jouppi, N. P., Young, C., Patil, N., Patterson, D., Agrawal, G., Bajwa, R., Bates, S., Bhatia, S., Boden, N., Al Borchers, R. B., Cantin, P.-l., Chao, C., Clark, C., Coriell, J., Daley, M., Dau, M., Dean, J., Gelb, B., Ghaemmaghami, T. V., … Yoon, D. H. (2017). Indatacenter performance analysis of a tensor processing unit. In *Proceedings of 2017 ACM/IEEE 44th Annual International Symposium on Computer Architecture*; Toronto, ON, Canada (pp. 1–12).

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012a). *Image Net classification with deep convolutional neural networks* (pp. 1097–1105). NIPS.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012b). Image Net classification with deep convolutional neural networks. In *Conference on Neural Information Processing Systems (Neur IPS)*. https://doi.org/10.1145/3065386

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, *521*(7553), 436–444.

Lee, C.-Y., Gallagher, P. W., & Tu, Z. (2016). Generalizing pooling functions in convolutional neural networks: Mixed, gated, and tree. *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics (AISTATS) 2016, Cadiz*, Spain. JMLR: W&CP, Vol. 51, 1509.08985.

Lee, C., Shao, Y. S., Zhang, J.-F., Parashar, A., Emer, J., Keckler, S. W., & Zhang, Z. (2018a). Stitch-x: An accelerator architecture for exploiting unstructured sparsity in deep neural networks. In *Proc. SysML Conference*.

Lee, J., Kim, C., Kang, S., Shin, D., Kim, S., Yoo, H-J (2018b) *UNPU: A 50.6TOPS/W unified deep neural network accelerator with 1b-to-16b fully-variable weight bit-precision*. IEEE.

Liu, X., Mao, M., Liu, B., Li, H., Chen, Y., Li, B., Wang, Y., Jiang, H., Barnell, M., Wu, Q., & Yang, J. (2015a). RENO: A high-efficient reconfigurable neuromorphic computing accelerator design. In *Proceedings of 2015 52nd ACM/EDAC/IEEE Design Automation Conference*; San Francisco, CA, USA (pp. 1–6). IEEE.

LeCun, Y., Cortes, C., Burges, CJC. *The MNIST database [Internet]*. 2019. http://yann.lecun.com/exdb/mnist/.

Liu, D., Chen, T., Liu, S., Zhou, J., Zhou, S., Teman, O., Feng, X., Zhou, X., & Chen, Y. (2015b). *PuDianNao: A polyvalent machine learning accelerator*. ACM SIGARCH Computer Architecture News.

McCulloch, S. W., & Pitts, W. (1990). A logical calculus of ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics*, *52*(1 and 2), 99–115. https://doi.org/10.1007/BF02478259

Moini, S., Alizadeh, B., Emad, M., & Ebrahimpour, R. (2017). A resource-limited hardware accelerator for convolutional neural networks in embedded vision applications. *IEEE Transactions on Circuits and Systems II: Express Briefs*, *64*(10), 1217–1221.

Netzer, Y., Wang, T., Coates, A., Bissacco, A., Bo Wu, Andrew Y. Ng (2011) Reading digits in natural images with unsupervised feature. *NIPS Workshop on Deep Learning and Unsupervised Feature Learning 2011*.

Pawlowski, J. T. (2011). Hybrid memory cube (HMC). In *Proceedings of the 2011 IEEE Hot Chips 23 Symposium* Stanford, CA, USA.IEEE.

Parashar, A., Rhu, M., Mukkara, A., Puglielli, A., Venkatesan, R., Khailany, B., Emer, J., Keckler, S. W., & Dally, W. J. (2017). SCNN: An accelerator for compressed-sparse convolutional neural networks. In *Proceedings of 44th IEEE/ACM International Symposium on Computer Architecture (ISCA-44), 2017* (pp. 27–40). IEEE.

Peemen, M. C. J., Setio, A. A. A., Mesman, B., & Corporaal, H. (2013). Memory-centric accelerator Design for Convolutional Neural Networks. In *Proceedings of the 2013 IEEE 31th International Conference on Computer Design (ICCD)*; Asheville, North Carolina (pp. 13–19). Institute of Electrical and Electronics Engineers. https://doi.org/10.1109/ICCD.2013.6657019

Qiao, X., Cao, X., Yang, H., Song, L., & Li, H. (2018). Atomlayer: A universal ReRAM-based CNN accelerator with atomic layer computation. In *Proceedings of the 55th Annual Design Automation Conference*. IEEE.

Reagen, B., Whatmough, P., Adolf, R., Rama, S., Lee, H., Lee, S. K., Hernández-Lobato, J. M., Wei, G.-Y., & Brooks, D. (2016). Minerva: Enabling low-power, highly-accurate deep neural network accelerators. In *2016 ACM/IEEE 43rd Annual International Symposium on Computer Architecture*. ACM/IEEE.

Shafiee, A., Nag, A., Muralimanohar, N., Balasubramonian, R., Strachan, J. P., Hu, M., Williams, R. S., & Srikumar, V. (2016). ISAAC: A convolutional neural network accelerator with in-situ analog arithmetic in crossbars. In *2016 ACM/IEEE 43rd Annual International Symposium on Computer Architecture*. IEEE.

Song, L,. Qian, X, . Li, H,. Chen, Y., (2017) Pipelayer: A pipelined ReRAM-based accelerator for deep learning. In: *Proceedings of the 2017 IEEE International Symposium on High Performance Computer Architecture*; IEEE.

Song, L., Mao, J., Zhuo, Y., Qian, X., Li, H., & Chen, Y. (2019). HyPar: Towards hybrid parallelism for deep learning accelerator array. In *Proceedings of the 2019 IEEE International Symposium on High Performance Computer Architecture*; (pp. 56–68). IEEE.

Sermanet, P., Eigen, D., Zhang, X., Mathieu, M., Fergus, R., & Le Cun, Y. (2014). Over feat: Integrated recognition, localization and detection using convolutional networks. In *International Conference on Learning Representations (ICLR)* (29). NY Scholars.

Simonyan, K., & Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations (ICLR)* (28,29,33,37,187,231,263). Computer Science.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., & Rabinovich, V. (2015). Going deeper with convolutions. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. 28,29,31,33,37,231. IEEE. https://doi.org/10.1109/cvpr.2015.7298594

Sze, V., Chen, Y.-H., Yang, T.-J., & Emer, S.-J. (2020). *Efficient processing of deep neural networks*. Synthesis Lectures on Computer Architecture.

Turing, A. M. (1950). Computing machinery and intelligence. *Mind*, *49*, 433–460.

UCI Machine Learning. (2020). http://archive.ics.uci.edu/ml/.

Wu, I., Huang, P., Lo, C., & Hwang, W. (2019). An energy-efficient accelerator with relative-indexing memory for sparse compressed convolutional neural network. In *2019 IEEE International Conference on Artificial Intelligence Circuits and Systems (AICAS)*; Hsinchu, Taiwan (pp. 42–45). IEEE. https://doi.org/10.1109/AICAS.2019.8771600

Wan, L., Matthew Zeiler, Sixin Zhang, Yann Le Cun, Rob Fergus, (2013) "Regularization of neural networks using drop connect," in *Proceedings of the 30th International Conference on Machine Learning (ICML-13). JMLR Workshop and Conference Proceedings*, vol. 28, no. 3. (pp. 1058–1066). JMLR.

Wong, H. P., Lee, H., Yu, S., Chen, Y., Wu, Y., Chen, P., Lee, B., Chen, F. T., & Tsai, M.-J. (2012). Metal-oxide RRAM. *Proceedings of IEEE*, *100*(6), 1951–1970.

Wang, P., Ji, Y., Hong, C., Lyu, Y., Wang, D., & Xie, Y.(2018). An Efficient Sparse Neural Network Computation Architecture Based on Resistive Random-Access Memory. *2018 55th ACM/ESDA/IEEE Design Automation Conference (DAC)*, 1–6. https://doi.org/10.1109/DAC.2018.8465793

Wulf, W. A., & McKe, S. A. (1995). Hitting the memory wall: Implications of the obvious. *ACM SIGARCH Computer Architecture News*, *23*(1), 20–24.

Xue, C. X., Chen, W. H., Liu, J. S., Li, J. F., Lin, W. Y., Lin, W. E., Wang, J.-H., Wei, W.-C., Chang, T.-W., Chang, T.-C., Huang, T.-Y., Kao, H.-Y., Wei, S.-Y., Chiu, Y.-C., Lee, C.-Y., Lo, C.-C., King, Y.-C., Lin, C.-J., Liu, R.-S., … Chang, M.-F. (2019, 2019). A 1Mb multibit ReRAM computing-in-memory macro with 14.6 ns parallel MAC computing time for CNN-based AI edge processors. In *Proceedings of the 2019 IEEE International Solid-State Circuits Conference*; San Francisco, CA, USA (pp. 388–390). IEEE.

Xie, S., Girshick, R., Dollar, P., Tu, Z., & He, K. (2016). Aggregated residual transformations for deep neural networks. *arXiv preprint arXiv*, 1611.05431v1.

Zang, H., Kang, W., Zhang, K., & Zhao, W. (2020). Deep neural network accelerator with spintronic memory. In *GLSVLSI '20: Proceedings of the 2020 on Great Lakes Symposium on VLSI*.

Zhao, R., Song, W., Zhang, W., Xing, T., Lin, J.-H., Srivastava, M., Gupta, R., & Zhang, Z. (2017). Accelerating binarized convolutional neural networks with software-programmable FPGAs. In *Proceedings of the 2017 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays* (pp. 15–24). ACM.

AUTHOR BIOGRAPHIES

Mohd Saqib Akhoon was born In Baramulla in the state of Jammu and Kashmir, India. He received his BTech in Electronics and Communication Engineering from the Islamic University of Science and Technology Awantipora Kashmir in 2016 and MTech in Electronics and Communication Engineering from Alfalah University Delhi/Haryana in 2019. He is currently pursuing his PhD in Universiti Sains Malaysia (USM). His research interests include Artificial intelligence, Neural Networks, computer vision, VLSI design, computer architecture, Nanoelectronics and Devices

Shahrel Azmin Suandi received his BE in Electronic Engineering, ME, and DE degrees in Information Science from Kyushu Institute of Technology, Fukuoka, Japan, in 1995, 2003, and 2006, respectively. He is currently a professor and also the Deputy Dean of Research, Innovation and Industry-Community Engagement at School of Electrical and Electronic Engineering, Universiti Sains Malaysia, Engineering Campus, Penang, Malaysia. Before joining academia, he was with the industries; Sony Video (M) Sdn. Bhd. And Technology Park Malaysia Corporation Sdn. Bhd., for almost 6 years as an engineer. His current research interests are face-based biometrics, real-time object detection and tracking, optimization and pattern classification using deep learning. He has served as a reviewer for several international conferences and journals, including IET Biometrics, IET Computer Vision, Multimedia Tools and Applications, Neural Computing and Applications, Journal of Electronic Imaging, IEEE Transactions on Information Forensics and Security, IEEE Access and others.

Abdullah Alshahrani holds his Bachelor degree in Computer Science from King Khalid University, in 2007, and received Master degree of Computer Science from School of Engineering & Mathematical Sciences, La Trobe University, Australia in 2010 and his PhD in Computer Science from The Catholic University of America, Washington DC, in 2018. He is currently assistant professor in the Faculty of Computing and Information Technology at the University of Jeddah.

Abdul-Malik H. Y. Saad was born in Jeddah, Saudi Arabia, in 1983. He received the BE degree with the first rank in computer engineering from Hodeidah University, Hodeidah, Yemen, in 2006, and the MSc degree in Electronic Systems Design Engineering from Universiti Sains Malaysia, in 2014. He continued his PhD study in the digital systems field at USM and received the degree in 2018. He is currently a Senior lecturer at the School of Electrical Engineering, Faculty of Engineering, Universiti Teknologi Malaysia, Johor, Malaysia. His research interest includes Digital and Embedded Systems Design, Image Processing, and AI.

Fahad R. Albogamy received the BSc degree in Information Systems with honour from King Saud University in 2003. He received the MSc and PhD degrees in Computer Sciences with distinction from Manchester University, UK in 2010 and 2017, respectively. He was the first dean of Applied Computer Sciences College at King Saud University. He is currently an Assistant Professor of Computer Sciences at Taif University, Saudi Arabia. He works as a consultant for academic affairs at the University Vice Presidency for Academic Affairs and Development. His research interests include Artificial Intelligence, Big Data, Machine learning, NLP and Digital Image and Signal Processing.

Mohd Zaid Bin Abdulla graduated from Universiti Sains Malaysia (USM) with a BAppSc degree in Electronic in 1986 before joining Hitachi Semiconductor (Malaysia) as a Test Engineer. In 1989, he commenced an MSc in Instrument Design and Application at University of Manchester Institute of Science and Technology, UK. He remained in Manchester, carrying out research in Electrical Impedance Tomography at the same university, and received his PhD degree in 1993. In the same year he joined USM as a lecturer and remained with this university as an academic and researcher till today. His principle research area is in Instrumentation and Sensing which covers topics such ultra-wide band imaging, computer vision applications and microwave tomography. He has published more than 140 research articles in international journals as IEEE, IET, IOP, and so forth. One of his papers was awarded The Senior Moulton medal for the best article published by the Institute of Chemical Engineering in 2002.

Sajad A. Loan, IETE Fellow, is currently working as a Professor in the Department of Electronics and Communication Engineering, Jamia Millia Islamia (Central University) New Delhi, India. He received the BE in Electronics and communication Engineering from the National Institute of Technology (REC) at Srinagar, Kashmir, MTech from A.M.U Aligarh and the PhD from Indian Institute of Technology (IIT), Kanpur India in 2010. He has authored and co-authored more than 140 publications in reputed SCI listed journals and conferences; received ten best paper awards in various international conferences around the globe and has granted/filed ten patents. He has supervised/supervising twelve PhDs and more than 20 MTech scholars till date. He has completed many national and international projects, including one multi-million NPST project from Saudi Arabia. Dr. Sajad is a Visiting Professor to the King Saud University under prestigious VPP program. He has been awarded Indo-Canadian Shastri Fellowship by the Govt. of Canada and India and is a Shastri Fellow. Dr. Sajad is a Visiting Professor to the University of Waterloo, Canada, as he has been awarded Visiting Associate Professorship by the University of Waterloo in 2017 and Visiting Professorship in 2018. His current research interests include VLSI Design, Processor Designing, Nanoelectronics, GaN based devices, energy harvesting, low power processor designing, Computer Architecture, Artificial Intelligence and Robotic Vision.

How to cite this article: Akhoon, M. S., Suandi, S. A., Alshahrani, A., Saad, A-M. H. Y., Albogamy, F. R., Abdullah, M. Z. B., & Loan, S. A. (2022). High performance accelerators for deep neural networks: A review. *Expert Systems*, *39*(1), e12831. https://doi.org/10.1111/exsy.12831