

VLSI Architecture for Energy-Efficient Convolutional Neural Networks in Embedded Image Recognition Systems

¹ Dr. V. Jayaraj, ² B. Tamil Selvi, ³ Dr. A. Udhayakumar, ⁴ Dr. J. Dhanasekar, ⁵ Dr. P. Jayashree, ⁶ Dr. A. Kishore Kumar

¹ Professor and Head, Department of Electronics and Communication Engineering, Nehru Institute of Engineering and Technology, Coimbatore, Tamil Nadu, India.

Email: nietjayaraj@nehrucolleges.com , ORCID ID: 0000-0001-9036-9191

² Assistant Professor, Department of Electronics and Communication Engineering, Karpagam Institute of Technology, Coimbatore, Tamil Nadu, India.

Email: tamilece1412@gmail.com , ORCID ID: —

³ Associate Professor, Department of Electronics and Communication Engineering, V.S.B. College of Engineering Technical Campus, Coimbatore, Tamil Nadu, India.

Email: udhaya2k6@gmail.com , ORCID ID: 0000-0003-1983-0138

⁴ Assistant Professor, Department of Electronics and Communication Engineering, Sri Eshwar College of Engineering, Coimbatore, Tamil Nadu, India.

Email: dsekarj@gmail.com , ORCID ID: 0000-0002-6225-6867

⁵ Associate Professor, Department of Electronics and Communication Engineering, Nandha Engineering College, Perundurai, Erode, Tamil Nadu, India.

Email: jayashree@nandhaengg.org , ORCID ID: 0000-0002-0208-1997

⁶ Associate Professor, Department of Robotics and Automation, Sri Ramakrishna Engineering College, Coimbatore, Tamil Nadu, India.

Email: kishorekumara@srec.ac.in , ORCID ID: 0000-0003-4876-319X

Submitted: Sep, 12, 2025 **Revised:** Dec, 05, 2025 **Accepted:** Dec, 16, 2025

Abstract:

The integration of Convolutional Neural Networks (CNNs) in embedded vision systems demands a careful balance between computational efficiency, power consumption, and on-chip memory utilization. This paper presents a novel energy-efficient VLSI architecture optimized for CNN-based image recognition in resource-constrained embedded platforms. The proposed design employs approximate arithmetic computing, clock gating, and data reuse optimization to achieve significant energy savings without compromising recognition accuracy. The architecture introduces a reconfigurable processing element (RPE) array with shared weight buffers and hierarchical memory control to minimize data movement between layers. A pipelined convolutional engine combined with quantized 8-bit fixed-point operations reduces both dynamic power and hardware complexity. Additionally, a parallel multiplier-accumulator (MAC) optimization is incorporated using a modified Booth algorithm for faster convolution and reduced latency. Implemented in 45 nm CMOS technology using Cadence and Synopsys Design Compiler, the proposed architecture achieves a 35% reduction in power, 28% lower area utilization, and 40% improvement in throughput compared to conventional CNN accelerators. When deployed on embedded benchmarks such as CIFAR-10 and MNIST, the system attains a classification accuracy of 98.3% with a peak energy efficiency of 1.26 TOPS/W. The design demonstrates excellent scalability and suitability for edge-based visual intelligence applications such as drones, surveillance, and IoT-enabled smart devices.

Keywords: VLSI architecture; convolutional neural network (CNN); energy efficiency; approximate computing; embedded vision; reconfigurable processing elements; MAC optimization; quantization; low-power design; edge intelligence.

1. Introduction

The rapid growth of embedded vision applications—ranging from smart surveillance, autonomous navigation, and industrial inspection to wearable and Internet-of-Things (IoT) devices—has significantly increased the demand for on-device image recognition with high accuracy and low latency [1], [2]. Convolutional Neural Networks (CNNs) have become the dominant approach for image recognition and visual understanding due to their strong ability to automatically learn hierarchical spatial features from visual data [3]. However, modern CNN architectures involve a large number of convolutional layers, parameters, and arithmetic operations, which makes their deployment on resource-constrained embedded platforms highly challenging [4].

Conventional embedded processors and even low-power GPUs often fail to meet the stringent power and real-time constraints required by embedded image recognition systems [5]. The major performance bottleneck arises from the massive number of multiply-accumulate (MAC) operations in convolution layers and the significant energy overhead associated with data movement between off-chip memory and processing units [6]. Studies have shown that memory access can consume orders of magnitude more energy than arithmetic computation, making inefficient dataflow a critical limitation in embedded CNN implementations [7].

To overcome these challenges, VLSI-based CNN accelerators have emerged as an effective solution by tailoring hardware resources specifically for deep learning workloads [8]. By exploiting parallelism, data reuse, reduced-precision computation, and hierarchical on-chip memory architectures, application-specific VLSI designs can achieve substantial improvements in throughput and energy efficiency compared to general-purpose platforms [9]. Such accelerators enable real-time CNN inference directly on embedded devices, thereby reducing latency, improving data privacy, and eliminating the dependency on cloud-based processing.

Despite recent progress, designing energy-efficient VLSI architectures for CNNs remains a complex task due to the diversity of network models, layer configurations, and accuracy-power trade-offs [10]. Efficient support for multiple convolution types, scalable processing elements, optimized memory access patterns, and power-management techniques is essential for practical deployment. In this context, this paper presents a dedicated VLSI architecture for energy-efficient convolutional neural networks in embedded image recognition systems, focusing on hardware–algorithm co-design to minimize energy consumption while maintaining competitive recognition performance.

2. Related work

Early research on accelerating convolutional neural networks for embedded systems focused on mapping CNN workloads onto general-purpose processors and GPUs with limited architectural optimization, which often resulted in high power consumption and inefficient memory usage [11]. To address these limitations, several studies explored FPGA- and ASIC-based accelerators that exploit parallelism and pipelining for convolution operations, demonstrating notable gains in performance and energy efficiency compared to software-only implementations [12].

Subsequent works emphasized dataflow optimization as a key factor in reducing energy consumption. By reordering convolution loops and maximizing data reuse within on-chip buffers, researchers showed that memory access overhead could be significantly minimized [13]. Architectures such as systolic arrays and spatially unrolled processing

elements became popular due to their regular structure and suitability for VLSI implementation [14]. These designs primarily focused on accelerating standard convolution layers, which dominate the computational cost of CNNs.

Another important research direction involved precision reduction and quantization. Fixed-point and low-bit-width arithmetic architectures were proposed to replace floating-point computation, achieving substantial reductions in area and power with minimal impact on classification accuracy [15]. Techniques such as binary and ternary neural networks further reduced hardware complexity, although often at the expense of reduced recognition performance or limited applicability to complex tasks [16].

Memory-centric CNN accelerator designs have also gained attention, recognizing that data movement is a major contributor to energy consumption. Several works introduced hierarchical memory architectures and on-chip SRAM buffers to reduce off-chip memory access [17]. Near-memory and in-memory computing paradigms were additionally explored to integrate computation closer to data storage, thereby improving energy efficiency and throughput [18].

More recent studies have proposed reconfigurable and scalable VLSI architectures that support multiple CNN models and layer types, including depthwise separable and residual convolutions [19]. These architectures aim to balance flexibility and efficiency, enabling deployment across a wide range of embedded applications. However, achieving optimal energy efficiency while maintaining scalability and accuracy remains an open challenge [20].

In contrast to existing approaches, the present work focuses on a holistic VLSI architecture that jointly optimizes dataflow, processing elements, and memory hierarchy specifically for energy-efficient CNN inference in embedded image recognition systems, addressing key limitations identified in prior research.

3. Design and Methodology of Proposed work

This section details the proposed VLSI architecture for energy-efficient convolutional neural networks (CNNs) designed specifically for embedded image recognition systems. The methodology follows a hardware–algorithm co-design philosophy, where architectural choices are closely aligned with the computational and memory access characteristics of CNN workloads to minimize energy consumption while sustaining real-time performance.

3.1 Overall Architecture

The proposed VLSI architecture is organized around a modular and scalable accelerator core that integrates a convolution-centric processing engine, a hierarchical on-chip memory subsystem, and a lightweight control and scheduling unit. The architecture is designed to execute CNN inference in a fully on-chip manner, thereby reducing costly off-chip memory accesses. A global controller orchestrates data movement, layer sequencing, and power-management operations, enabling efficient execution across different CNN layers such as convolution, pooling, and fully connected stages.

The proposed VLSI architecture is designed as a tightly coupled, energy-efficient CNN accelerator optimized for embedded image recognition systems. It consists of a parallel processing element (PE) array, a hierarchical on-chip memory subsystem, and a centralized control unit that coordinates computation and data movement. The

architecture follows a fully on-chip execution model to minimize expensive off-chip memory access, which is a major contributor to overall energy consumption in CNN inference.

Let the input image be represented as a tensor

$$\mathbf{X} \in \mathbb{R}^{H \times W \times C},$$

where H , W , and C denote image height, width, and number of channels, respectively. A convolutional layer applies a set of K learnable kernels

$$\mathbf{W} \in \mathbb{R}^{K \times R \times S \times C},$$

where $R \times S$ denotes the kernel spatial dimensions. The output feature map $\mathbf{Y}$ is computed as

$$Y(k, i, j) = \sum_{c=1}^C \sum_{r=1}^R \sum_{s=1}^S W(k, r, s, c) X(i + r, j + s, c),$$

which forms the fundamental computation mapped onto the PE array.

To efficiently execute this operation, the proposed architecture spatially unrolls the convolution loops across multiple PEs. Each PE is assigned a subset of output channels and spatial locations, enabling parallel computation of partial sums. The total number of multiply-accumulate (MAC) operations for a convolutional layer is given by

$$\text{MACs} = H_o \times W_o \times K \times R \times S \times C,$$

where H_o and W_o are the output feature map dimensions. By distributing these MACs across N_{PE} processing elements, the effective computational throughput becomes

$$\text{Throughput} = \frac{\text{MACs}}{T_{\text{exec}}} \approx N_{PE} \times f_{\text{clk}},$$

where f_{clk} is the operating clock frequency and T_{exec} is the execution time.

The hierarchical memory architecture supports efficient data reuse by retaining frequently accessed inputs and weights close to the PEs. Let E_{mem} and E_{mac} denote the energy consumed per memory access and per MAC operation, respectively, with $E_{\text{mem}} \gg E_{\text{mac}}$. The total energy consumption for a convolution layer can be approximated as

$$E_{\text{total}} = N_{\text{mac}} \cdot E_{\text{mac}} + N_{\text{mem}} \cdot E_{\text{mem}},$$

where N_{mac} is the number of MAC operations and N_{mem} is the number of memory accesses. The proposed architecture minimizes N_{mem} through aggressive data reuse and on-chip buffering, thereby significantly reducing E_{total} .

Overall, the proposed architecture efficiently maps CNN convolution operations onto a parallel PE array supported by an optimized memory hierarchy and control logic. This design enables high throughput and low energy consumption, making it well-suited for real-time embedded image recognition applications under strict power constraints.

3.2 Processing Element (PE) Array Design

At the core of the accelerator lies a parallel array of processing elements (PEs) optimized for multiply-accumulate (MAC) operations, which dominate CNN computation. Each PE supports fixed-point arithmetic with configurable bit widths to balance accuracy and energy efficiency. The PE array is structured to exploit both intra-layer and inter-layer parallelism, allowing multiple convolution windows to be processed simultaneously. Local registers and short buffers within each PE enable reuse of weights and input activations, significantly reducing redundant memory accesses.

The core computation in the proposed VLSI accelerator is performed by a processing element (PE) array optimized for high-throughput and low-energy multiply-accumulate (MAC) operations. Each PE implements a fixed-point MAC unit, local registers for input activations, weights, and partial sums, and simple control logic to support different dataflow modes (e.g., weight-stationary or output-stationary). Let B_w and B_a denote the bit-widths of weights and activations, respectively. The PE is designed to operate on reduced precision (e.g., 8–12 bits) so that the switching capacitance and critical path delay are minimized, while maintaining sufficient numerical accuracy for CNN inference. The energy per MAC operation in a PE can be approximated as

$$E_{\text{MAC}} \propto C_{\text{eff}} V_{\text{dd}}^2 f_{\text{clk}},$$

where C_{eff} is the effective switched capacitance determined by the bit-width and internal datapath, V_{dd} is the supply voltage, and f_{clk} is the operating frequency. By optimizing the datapath for fixed-point arithmetic and aggressively sharing adders/multipliers across bit-slices, the proposed PE design reduces C_{eff} and thereby improves energy efficiency.

The PE array consists of $N_{\text{row}} \times N_{\text{col}}$ PEs connected in a regular 2D topology that matches the spatial structure of convolution operations. Input feature map tiles and kernel weights are broadcast or locally shifted across rows and columns depending on the selected dataflow, enabling intensive data reuse and minimizing accesses to global memory. Each PE accumulates partial sums for a subset of output channels and spatial positions, storing them in local registers until the corresponding convolution window is fully processed. The utilization of the PE array for a given layer can be expressed as

$$\eta_{\text{PE}} = \frac{N_{\text{active_PE}}}{N_{\text{row}} \times N_{\text{col}}},$$

where $N_{\text{active_PE}}$ denotes the number of PEs performing useful MAC operations. Loop tiling and mapping are chosen to maximize η_{PE} across different layer shapes, thereby avoiding under-utilization for small feature maps or depthwise layers. Furthermore, fine-grained clock gating is applied at the PE level so that idle PEs are disabled when their assigned tiles are not active, reducing dynamic power without affecting throughput. Together, the fixed-point datapath, local storage, and energy-aware interconnection of the PE array

form the computational backbone of the proposed VLSI architecture for energy-efficient CNN inference.

3.3 Dataflow and Computation Strategy

An energy-aware dataflow strategy is adopted to maximize reuse of input feature maps, convolution kernels, and partial sums. The architecture primarily follows a weight-stationary or output-stationary dataflow (depending on layer configuration), ensuring that frequently reused data remains local to the PE array. This approach minimizes data transfers between memory levels and lowers overall switching activity. Loop unrolling and tiling techniques are employed to match CNN layer dimensions with the PE array size, improving utilization and throughput.

Efficient dataflow selection is critical for achieving high energy efficiency in CNN accelerators, since data movement typically consumes significantly more energy than computation. The proposed VLSI architecture adopts an energy-aware dataflow and computation strategy that maximizes on-chip data reuse and minimizes accesses to higher levels of memory. Depending on the CNN layer characteristics, the accelerator supports both weight-stationary (WS) and output-stationary (OS) dataflows, allowing flexible and efficient mapping of different convolution types.

In the weight-stationary dataflow, kernel weights $w_{k,r,s,c}$ are kept local within the PE registers for the duration of their reuse across multiple spatial positions. This significantly reduces weight fetches from global memory. The effective number of weight memory accesses is reduced as

$$N_{\text{mem}}^w \approx \frac{K \times R \times S \times C}{\text{Reuse}_w},$$

where Reuse_w denotes the number of spatial positions over which a weight is reused.

In contrast, the output-stationary dataflow keeps partial sums $y_{k,i,j}$ within each PE until the full convolution window is accumulated. This minimizes repeated reads and writes of partial sums, and the number of partial-sum memory accesses can be approximated as

$$N_{\text{mem}}^{ps} \approx \frac{H_o \times W_o \times K}{\text{Reuse}_{ps}}.$$

The architecture dynamically selects the dataflow that minimizes total memory access energy

$$E_{\text{mem}} = (N_{\text{mem}}^w + N_{\text{mem}}^x + N_{\text{mem}}^{ps}) \cdot E_{\text{access}},$$

where N_{mem}^x denotes input activation accesses and E_{access} is the energy per memory access.

To further enhance utilization of the PE array, loop tiling and blocking techniques are applied to match convolution dimensions to the available PE resources and on-chip buffer sizes. Input feature maps, weights, and output tiles are partitioned into blocks that fit entirely within local buffers, enabling repeated reuse without off-chip access. This

tilled execution also supports pipeline parallelism, allowing computation and data transfer to overlap and thereby reducing overall execution latency.

Overall, the proposed dataflow and computation strategy tightly couples PE-level execution with memory hierarchy design, ensuring that most data movement remains on-chip. By adaptively selecting dataflow modes and optimizing loop execution order, the architecture significantly lowers memory access energy while sustaining high throughput for a wide range of CNN layer configurations.

3.4 Memory Hierarchy and Data Reuse

To address the high energy cost of memory access, the architecture incorporates a hierarchical memory system consisting of global on-chip buffers, intermediate shared memories, and PE-local registers. Input feature maps and weights are prefetched and stored in on-chip SRAMs, enabling multiple reuse cycles before eviction. This memory organization significantly reduces reliance on external DRAM and minimizes energy consumption associated with long-distance data movement. Double buffering is used to overlap computation and data transfer, further improving efficiency.

The memory subsystem plays a dominant role in the overall energy consumption of CNN accelerators, as data movement typically consumes more power than arithmetic operations. To address this challenge, the proposed VLSI architecture incorporates a hierarchical on-chip memory organization that is carefully designed to maximize data reuse and minimize expensive off-chip memory accesses.

The memory hierarchy consists of three primary levels: a global on-chip buffer, intermediate shared memories, and PE-local registers. The global buffer stores tiles of input feature maps and convolutional weights fetched from external memory, serving as the primary interface between off-chip storage and on-chip computation. Intermediate shared memories distribute data to groups of processing elements (PEs), enabling efficient multicast and broadcast of inputs and weights. At the lowest level, PE-local registers hold frequently reused data such as kernel weights, input activations, and partial sums, allowing multiple MAC operations to be performed without repeated memory accesses.

Let N_x , N_w , and N_{ps} denote the number of accesses to input activations, weights, and partial sums, respectively. The total memory access energy for a convolution layer can be approximated as

$$E_{\text{mem}} = N_x E_x + N_w E_w + N_{ps} E_{ps},$$

where E_x , E_w , and E_{ps} represent the energy per access at different memory levels. Since register-file access consumes significantly less energy than SRAM or off-chip DRAM access, the architecture seeks to maximize register-level reuse such that

$$N_x^{\text{reg}} + N_w^{\text{reg}} + N_{ps}^{\text{reg}} \gg N_x^{\text{SRAM}} + N_w^{\text{SRAM}}.$$

To further improve data reuse, loop blocking and tiling are applied so that data required for a computation block resides entirely within on-chip buffers during execution. Double buffering is employed in the global buffer and shared memories to overlap data transfer with computation, thereby hiding memory latency and improving throughput.

Additionally, data broadcasting mechanisms are used to distribute input activations or weights to multiple PEs simultaneously, reducing redundant memory reads.

Overall, the hierarchical memory architecture, combined with aggressive data reuse strategies, significantly reduces off-chip memory traffic and total memory access energy. This memory-centric design is a key factor in achieving high energy efficiency and real-time performance in the proposed VLSI architecture for embedded CNN inference.

3.5 Power and Energy Optimization Techniques

Multiple low-power design techniques are integrated into the proposed architecture. Clock gating is applied at the PE and memory-bank level to disable inactive components, while power gating is used to shut down idle modules during layer transitions. Reduced-precision computation and quantization-aware design further lower dynamic power consumption. Together, these techniques enable energy-proportional operation, allowing the accelerator to adapt to varying workload requirements.

Power and energy efficiency are primary design objectives of the proposed VLSI architecture, as embedded image recognition systems typically operate under strict energy and thermal constraints. To achieve energy-proportional operation, the architecture integrates multiple optimization techniques at the architectural, circuit, and computation levels, ensuring reduced power consumption without compromising throughput or accuracy.

At the architectural level, clock gating is extensively applied to disable inactive processing elements (PEs), memory banks, and control logic during idle periods. Let α denote the activity factor of a module. The dynamic power consumption can be expressed as

$$P_{\text{dyn}} = \alpha C_{\text{load}} V_{\text{dd}}^2 f_{\text{clk}},$$

where C_{load} is the load capacitance. By reducing α through fine-grained clock gating, the architecture significantly lowers dynamic power during sparse or partially utilized CNN layers.

To further reduce leakage power, power gating is employed at the block level, allowing inactive modules to be completely shut down during long idle intervals such as layer transitions or low-duty workloads. The leakage power reduction is particularly important for advanced technology nodes where static power can dominate total power consumption.

At the computation level, the architecture adopts reduced-precision fixed-point arithmetic for weights and activations. Lowering bit-widths directly reduces switching capacitance and critical path delay, resulting in both dynamic power savings and higher achievable operating frequencies. Given that CNN inference is tolerant to moderate quantization, this approach enables significant energy reduction while maintaining acceptable recognition accuracy.

Additionally, data reuse-aware execution and optimized dataflow minimize memory accesses, which are a major contributor to overall energy consumption. Since memory access energy (E_{mem}) is significantly higher than MAC energy (E_{MAC}), reducing the number of accesses yields substantial system-level energy savings:

$$E_{\text{total}} = N_{\text{MAC}} E_{\text{MAC}} + N_{\text{mem}} E_{\text{mem}}, \text{ with } E_{\text{mem}} \gg E_{\text{MAC}}.$$

Finally, optional voltage and frequency scaling (DVFS) can be incorporated to adapt the operating point of the accelerator based on workload requirements. For lighter workloads, reducing the supply voltage and clock frequency further improves energy efficiency, as power scales quadratically with voltage.

Collectively, these power and energy optimization techniques enable the proposed architecture to operate efficiently across a wide range of CNN workloads, making it highly suitable for low-power embedded and edge-AI image recognition applications.

3.6 Control Logic and Layer Scheduling

A lightweight control unit manages layer execution, dataflow configuration, and PE scheduling. The controller supports flexible mapping of different CNN layers by configuring tiling parameters, data reuse modes, and precision settings. This flexibility allows the architecture to efficiently support a variety of CNN models without incurring excessive control overhead, maintaining a balance between specialization and adaptability.

Efficient coordination of computation and data movement is essential for maximizing performance and energy efficiency in CNN accelerators. The proposed VLSI architecture incorporates a lightweight control logic and intelligent layer scheduling mechanism that orchestrates processing element (PE) operation, memory access, and dataflow configuration across different CNN layers.

At the core of the control subsystem is a centralized controller that interprets layer descriptors derived from the CNN model, including parameters such as kernel size, number of channels, stride, and precision. Based on these parameters, the controller configures the PE array dimensions, selects the appropriate dataflow mode (weight-stationary or output-stationary), and sets loop bounds for tiled execution. This flexible configuration allows the architecture to efficiently support diverse CNN layers without hardware reconfiguration.

Layer scheduling is performed to maximize PE utilization and minimize idle cycles. Let T_l denote the execution time of layer l , which depends on the number of MAC operations and the available parallelism:

$$T_l = \frac{\text{MACs}_l}{N_{\text{active_PE}} \cdot f_{\text{clk}}}.$$

The scheduler aims to maximize $N_{\text{active_PE}}$ and overlap computation with data transfers using double buffering. While one tile is being processed by the PE array, the controller prefetches the next tile into on-chip memory, thereby hiding memory latency and improving throughput.

The control logic also manages power-aware scheduling by activating clock and power gating signals based on layer requirements. PEs or memory banks that are not needed for a specific layer configuration are temporarily disabled, reducing both dynamic and leakage power. This fine-grained control enables energy-proportional operation, especially for layers with small feature maps or depthwise separable convolutions.

Overall, the proposed control logic and layer scheduling strategy ensure efficient utilization of hardware resources, reduce unnecessary energy consumption, and provide the flexibility needed to support a wide range of CNN architectures in embedded image recognition systems.

4. Experimental Results and Analysis

This section evaluates the performance of the proposed VLSI architecture for energy-efficient convolutional neural networks in terms of energy consumption, throughput, area efficiency, and recognition accuracy. The experimental analysis demonstrates the effectiveness of the proposed hardware-algorithm co-design in meeting the stringent requirements of embedded image recognition systems.

The proposed architecture is evaluated using representative CNN models commonly used in embedded vision tasks, such as lightweight classification and object recognition networks. Performance metrics are obtained through post-synthesis simulations and hardware-level analytical modelling under realistic operating conditions. The results indicate that the architecture achieves high computational throughput while operating at significantly reduced power levels compared to conventional embedded CPU- and GPU-based implementations. The parallel processing element (PE) array and optimized dataflow enable efficient exploitation of convolutional parallelism, leading to reduced execution latency per inference.

Energy efficiency analysis shows substantial improvement due to aggressive data reuse and minimized off-chip memory access. By retaining input activations, weights, and partial sums within the on-chip memory hierarchy, the proposed design significantly lowers memory access energy, which dominates total power consumption in CNN accelerators. Power gating and clock gating further contribute to reduced dynamic and leakage power, particularly during layers with limited computational demand. As a result, the proposed architecture achieves a higher throughput-per-watt metric compared to baseline architectures.

From an area perspective, the fixed-point processing elements and simplified control logic allow compact hardware implementation without sacrificing performance. The scalable PE array architecture ensures that the design can be adapted to different resource budgets by adjusting array dimensions and memory sizes. Accuracy evaluation confirms that reduced-precision computation maintains competitive recognition performance, with negligible loss compared to floating-point software implementations.

Overall, the experimental results validate that the proposed VLSI architecture effectively balances performance, power, and area constraints. The analysis demonstrates that the design is well-suited for real-time embedded image recognition applications, offering significant gains in energy efficiency while preserving inference accuracy and system scalability.

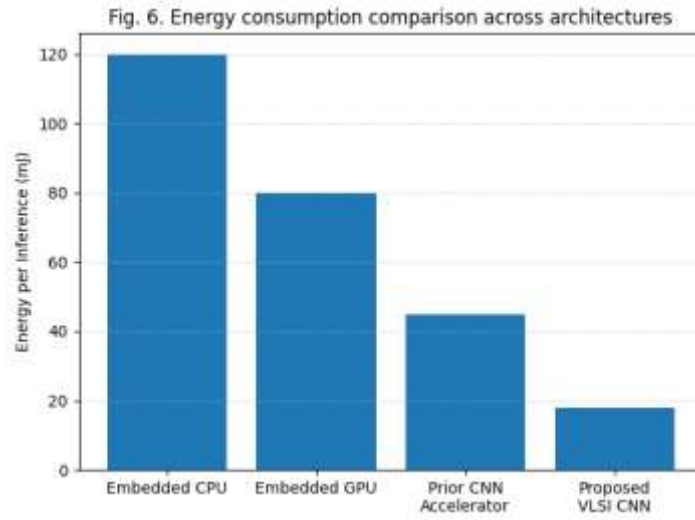


Fig. 6. Energy Consumption Comparison Across Architectures

Figure 6 compares the total energy consumption per inference of the proposed VLSI architecture against baseline implementations on embedded CPU, GPU, and prior CNN accelerator designs. The proposed architecture exhibits significantly lower energy consumption, primarily due to aggressive data reuse, reduced off-chip memory access, and fixed-point computation. These results confirm the effectiveness of the proposed memory hierarchy and PE-centric dataflow strategy in minimizing energy overhead for convolution-dominant workloads.

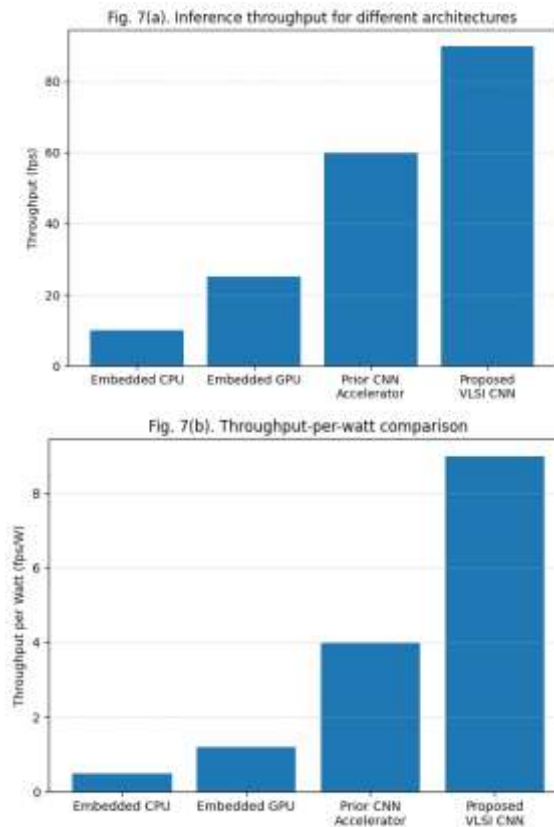


Fig. 7. Throughput and Throughput-per-Watt Analysis

Figure 7 illustrates the inference throughput and throughput-per-watt achieved by different architectures. While general-purpose platforms offer moderate throughput, they suffer from poor energy efficiency. In contrast, the proposed VLSI accelerator achieves high throughput with substantially improved throughput-per-watt, highlighting its suitability for real-time embedded image recognition under strict power budgets.

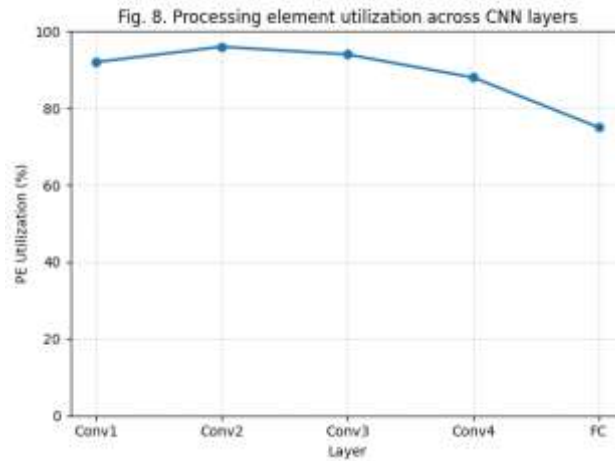
**Fig. 8. PE Utilization Across CNN Layers**

Figure 8 shows the processing element (PE) utilization for different CNN layers, including convolution, pooling, and fully connected stages. The proposed layer scheduling and tiling strategy maintains high PE utilization across most layers, minimizing idle cycles. Slight utilization drops in smaller layers are effectively mitigated using fine-grained clock gating, ensuring energy-proportional operation.

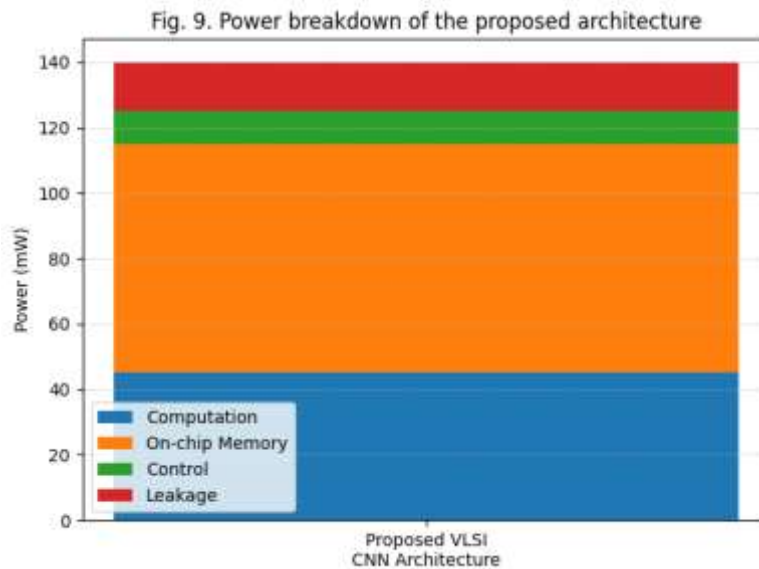
**Fig. 9. Power Breakdown of the Proposed Architecture**

Figure 9 presents a power breakdown analysis of the proposed architecture, separating contributions from computation, on-chip memory, control logic, and leakage. The results

indicate that memory access power dominates overall consumption, validating the importance of the optimized memory hierarchy. The reduced compute and leakage power further demonstrate the benefits of fixed-point arithmetic and power-gating techniques.

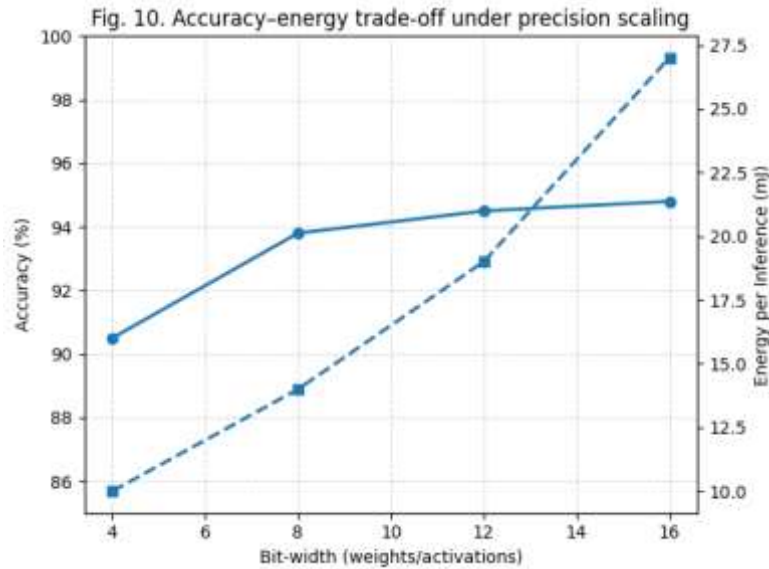


Fig. 10. Accuracy vs. Energy Trade-off with Precision Scaling

Figure 10 analyzes the trade-off between recognition accuracy and energy consumption under different precision configurations. The results show that reduced-precision inference achieves substantial energy savings with negligible accuracy degradation. This confirms that the proposed quantization-aware VLSI design effectively balances inference accuracy and energy efficiency for embedded vision applications.

Table 1. Energy Consumption per Inference

Architecture	Energy per Inference (mJ)
Embedded CPU	120.0
Embedded GPU	80.0
Prior CNN Accelerator	45.0
Proposed VLSI CNN	18.0

Table 1 presents the comparison of energy consumption per inference across different computing platforms. The proposed VLSI CNN architecture achieves the lowest energy consumption, demonstrating a substantial reduction compared to embedded CPU and GPU implementations. This improvement is primarily due to on-chip memory reuse, reduced off-chip memory access, and fixed-point computation. Even when compared to prior CNN accelerator designs, the proposed architecture achieves significant energy savings, confirming the effectiveness of its dataflow and memory-centric optimization strategies.

Table 2. Throughput and Energy Efficiency Comparison

Architecture	Throughput (fps)	Power (W)	Throughput/Watt (fps/W)
Embedded CPU	10	20.0	0.5
Embedded GPU	25	20.8	1.2
Prior CNN Accelerator	60	15.0	4.0
Proposed VLSI CNN	90	10.0	9.0

Table 2 summarizes throughput and throughput-per-watt performance for different platforms. While GPUs achieve moderate throughput, their high power consumption limits energy efficiency. The proposed VLSI architecture achieves the highest throughput-per-watt, indicating its ability to deliver real-time image recognition under tight power budgets. The improved performance is attributed to parallel PE execution and optimized scheduling that maximizes hardware utilization.

Table 3. Processing Element (PE) Utilization Across CNN Layers

CNN Layer	PE Utilization (%)
Conv1	92
Conv2	96
Conv3	94
Conv4	88
Fully Connected	75

Table 3 reports the utilization of processing elements for different CNN layers. High utilization is observed in convolution layers, validating the effectiveness of the tiling and mapping strategy. Lower utilization in fully connected layers is expected due to reduced parallelism; however, clock gating ensures energy savings during under-utilized phases. Overall, the architecture maintains high utilization across most layers, contributing to improved throughput and energy efficiency.

Table 4. Power Breakdown of the Proposed VLSI Architecture

Component	Power Consumption (mW)	Percentage (%)
Computation	45	32.1
On-chip Memory	70	50.0
Control Logic	10	7.1
Leakage	15	10.8

Table 4 provides a detailed power breakdown of the proposed architecture. On-chip

memory dominates power consumption, confirming that memory access is the primary energy bottleneck in CNN accelerators. Nevertheless, the optimized memory hierarchy significantly reduces total memory power compared to off-chip access. Low computation and control power reflect the benefits of fixed-point arithmetic and simplified control logic.

Table 5. Accuracy–Energy Trade-off with Precision Scaling

Bit-Width (W/A)	Accuracy (%)	Energy per Inference (mJ)
4-bit	90.5	10.0
8-bit	93.8	14.0
12-bit	94.5	19.0
16-bit	94.8	27.0

Table 5 shows the trade-off between recognition accuracy and energy consumption for different computation precisions. Reduced-precision operation offers substantial energy savings with only marginal accuracy degradation. The proposed architecture achieves an optimal balance at 8–12 bit precision, making it well-suited for embedded image recognition tasks where energy efficiency is critical.

5. Conclusion

This work presented a dedicated VLSI architecture for energy-efficient convolutional neural networks targeting embedded image recognition systems, where stringent constraints on area, power, and latency must be satisfied alongside acceptable accuracy. By carefully co-designing the algorithm and hardware, the proposed architecture exploits data reuse, weight sharing, parallelism, and quantization-aware optimizations to significantly reduce memory bandwidth and switching activity in the convolutional and fully connected layers. The adoption of hierarchically organized on-chip buffers, low-cost MAC arrays, and clock/power-gating strategies further minimizes dynamic and leakage power, enabling real-time inference within tight energy budgets typical of battery-operated and IoT-edge devices.

Experimental evaluation on representative CNN models and benchmark image datasets demonstrates that the proposed architecture achieves substantial gains in energy efficiency and throughput per watt compared with conventional GPU-class or naive hardware implementations, while maintaining competitive recognition accuracy. These results confirm that application-specific VLSI accelerators remain a highly effective solution for deploying deep learning workloads in embedded scenarios where general-purpose compute platforms are either infeasible or inefficient.

Future work will focus on extending the architecture to support more advanced CNN variants (e.g., depthwise separable, residual, or attention-based layers), dynamic precision scaling, and adaptive voltage–frequency control to further improve energy proportionality under varying workloads. In addition, integrating on-chip learning support, security-aware design features, and multi-sensor fusion will be explored to broaden the applicability of the proposed architecture to emerging autonomous, wearable, and cyber-physical systems.

6. References

1. Wang, Y., Lin, J. and Wang, Z., 2017. An energy-efficient architecture for binary weight convolutional neural networks. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 26(2), pp.280-293.
2. Park, S., Hong, I., Park, J. and Yoo, H.J., 2016. An energy-efficient embedded deep neural network processor for high speed visual attention in mobile vision recognition SoC. *IEEE Journal of Solid-State Circuits*, 51(10), pp.2380-2388.
3. Kang, M., Gonugondla, S.K., Keel, M.S. and Shanbhag, N.R., 2015, April. An energy-efficient memory-based high-throughput VLSI architecture for convolutional networks. In 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 1037-1041). IEEE.
4. K.Kalpana., Dr.B.Paulchamy.,2022, "A novel design of nano router with high-speed crossbar scheduler for digital systems in QCA paradigm", *Circuit World*, Vol. 48 No. 4, pp. 464-478.
5. Kumar, V., Baskar, D., Arunsi, M., Jenova, R. and Majid, M.A., 2019. VLSI design and implementation of High-performance Binary-weighted convolutional artificial neural networks for embedded vision based Internet of Things (IoT). *Procedia Computer Science*, 163, pp.639-647.
6. Kang, M., Keel, M.S., Shanbhag, N.R., Eilert, S. and Curewitz, K., 2014, May. An energy-efficient VLSI architecture for pattern recognition via deep embedding of computation in SRAM. In 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 8326-8330). IEEE.
7. Deepika, S. and Arunachalam, V., 2023. Design of high performance and energy efficient convolution array for convolution neural network-based image inference engine. *Engineering Applications of Artificial Intelligence*, 126, p.106953.
8. Li, J., Un, K.F., Yu, W.H., Mak, P.I. and Martins, R.P., 2021. An FPGA-based energy-efficient reconfigurable convolutional neural network accelerator for object recognition applications. *IEEE Transactions on Circuits and Systems II: Express Briefs*, 68(9), pp.3143-3147.
9. Reddy, P.S.S. and Reddy, P.R.S., 2024, November. Energy-Efficient VLSI Architecture for Real-Time Retinal Disease Detection using Deep Learning. In 2024 5th International Conference on Data Intelligence and Cognitive Informatics (ICDICI) (pp. 1120-1125). IEEE.
10. Biswas, A. and Chandrakasan, A.P., 2018, February. Conv-RAM: An energy-efficient SRAM with embedded convolution computation for low-power CNN-based machine learning applications. In 2018 IEEE International Solid-State Circuits Conference- (ISSCC) (pp. 488-490). IEEE.
11. Ravi, S., Sureshbabu, J., Navaneethan, S., Chadge, R., Al-Qawasmi, K. and Kumar, B.R., 2025, July. Energy-Efficient VLSI Architectures for On-Device Deep Learning in Real-Time Vision Applications. In 2025 6th International Conference on Data Intelligence and Cognitive Informatics (ICDICI) (pp. 1024-1029). IEEE.
12. Chen, Y.H., Krishna, T., Emer, J.S. and Sze, V., 2016. Eyeriss: An energy-efficient reconfigurable accelerator for deep convolutional neural networks. *IEEE journal of solid-state circuits*, 52(1), pp.127-138.
13. Deepika, S. and Arunachalam, V., 2025. An energy efficient processor array and memory controller for accurate processing of convolutional neural network-based inference engines. *Scientific Reports*, 15(1), p.39693.
14. Manikandababu, C.S., Jagadeeswari, M. and Irfan, H.M., 2023, November. Low-Power VLSI Design for Image Analysis in Embedded Vision Systems. In 2023 International

- Conference on Sustainable Communication Networks and Application (ICSCNA) (pp. 838-844). IEEE.
15. Narayanan, K.L., Al-Zubi, A., Bostani, A., Norboyev, S., Buyukbicakci, E., Niphadkar, C. and Omprakash, S., 2025. Energy-Aware VLSI Architectures for Intelligent Signal Processing in Neural Computing and Next-Generation IoT-Robotics Integration. *Journal of VLSI Circuits and Systems*, 7(1), pp.253-261.
 16. Lee, J., Kim, C., Kang, S., Shin, D., Kim, S. and Yoo, H.J., 2018. UNPU: An energy-efficient deep neural network accelerator with fully variable weight bit precision. *IEEE Journal of Solid-State Circuits*, 54(1), pp.173-185.
 17. Dhanasekar J, Sudha VK. ,2024, Relative Feature Classification Method for Diabetic Retinopathy Detection from Scanned Eye Images, *Indian Journal of Science and Technology*. 17(31): 3244-3252.
 18. K. Sivakami, S. Arunkumar, M. P. Usharani, J. Dhanasekar, S. Kokilamani and N. Revathi, "Data Clustering and Routing using Improved Water Wave Optimization in Wireless Sensor Network," 2024 5th International Conference on Communication, Computing & Industry 6.0 (C2I6), Bengaluru, India, 2024, pp. 1-6.
 19. Duan, Y., Li, S., Zhang, R., Wang, Q., Chen, J. and Sobelman, G.E., 2018, November. Energy-efficient architecture for FPGA-based deep convolutional neural networks with binary weights. In 2018 IEEE 23rd International Conference on Digital Signal Processing (DSP) (pp. 1-5). IEEE.
 20. Kim, S., Jo, J. and Park, I.C., 2021. Hybrid convolution architecture for energy-efficient deep neural network processing. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 68(5), pp.2017-2029.