Electronics, Communications, and Computing Summit
Vol. 3, No. 1, Jan - Mar 2025, pp. 61-70
ISSN: 3107-8222, DOI: https://doi.org/10.17051/ECC/03.01.07

Neuromorphic Processor Design for Ultra-Low-Power Embedded Vision Applications

Felipe Cid

Facultad de Ingenieria Universidad Andres Bello, Santiago, Chile, Email: cid.felip@unab.cl

Article Info

Article history:

Received : 17.01.2025
Revised : 19.02.2025
Accepted : 21.03.2025

Keywords:

Neuromorphic computing,
Spiking neural networks (SNN),
In-memory computing (IMC),
Low-power vision,
Event-driven processing,
Embedded AI,
Edge computing,
LIF neuron model,
DVS camera,
Real-time inference.

ABSTRACT

The exponential growth of intelligent edge devices in use in applications like autonomous navigation, real-time surveillance and intelligent robotics is driving the need to energy efficient embedded vision systems with the capability to process complex cognitive tasks in a power-sensitive and latency-sensitive environment. In this paper, a new architecture of neuromorphic processor specifically tuned towards ultra-low-power embedded vision applications is as presented. Based on how the human brain is efficient in sensory processing, the proposed processor uses the proposed processor uses event-driven spiking neural networks (SNNs) and in-memory computing (IMC) paradigm to reduce data movement and power consumption. Architecturally, this focuses on the assembly of digital leaky integrate and fire (LIF) neurons into a programmable processing element (PE) network, with a PE comprising a local memory and computation to serve asynchronous, spike-based, processing. Also, the processor also includes resistive RAM (ReRAM)-based crossbar arrays, allowing analog in-memory multiply-accumulate (MAC) operations and decreasing the energy expenditure of older von Neumann architectures significantly. The custom compiler framework implements a temporally encoded visual stream as created by event-based sensors, e.g. Dynamic Vision Sensors (DVS), to the Neuromorphic hardware, with latency-based scheduling and dynamic spike routing. Specifically to test the feasibility of the proposed design, large-scale simulations and FPGA-based experiments with real-scale vision benchmarks were performed, such as MNIST-DVS, CIFAR10-DVS and N-MNIST. Since the power of the processor is lowered up to 4 times and the inference latency is 2.5 times smaller than the traditional edge AI platforms (Google Edge TPU and Intel Loihi 2), our experimental results clearly demonstrate that the processor offers up to 2.5X lower energy consumption, in addition to lower inference latency and competitive classification accuracy. Results indicate a promising potential of integrating biologically inspired neural architecture and novel memory technologies to design scalable low-power vision processors to support AI at the edge. This work forms a baseline of future work in adaptive neuromorphic systems and integration with neuromorphic sensors that can facilitate end-to-end real-time perception with ultra-low energy footprints in large-scale distributed embedded applications.

1. INTRODUCTION

The prospect of artificial intelligence (AI) and embedded systems has led to the initiation of a paradigm shift in the world of machine perception and interaction to the real world. Given the growing adoption of edge computing to diverse fields of endeavor, including autonomous vehicles, surveillance systems, wearable electronics, industrial inspection and assistive robotics, there is an urgent demand to demonstrate practical routes to hardware realizations of real-time intelligent visual processing, within extreme energy and computational budgets. The typical solutions to run the traditional approaches demand a lot of power consumption, memory bandwidth, and lack scaling capabilities in real-time, which makes them unfit to use on resource-limited edge devices.

The gap between the original character of traditional deep learning models and the needs of embedded vision systems sit at the core of the issue. CNNs read the visual information in dense frames at a predetermined speed regardless of the behavior of the scene, which results in unnecessary calculation and poor utilization of energy. Moreover, these architectures also feature a von Neumann model, implying that the memory and computations are separated, thus causing a memory wall that makes the performance low and exorbitant power overheads are incurred, especially when utilizing huge models.

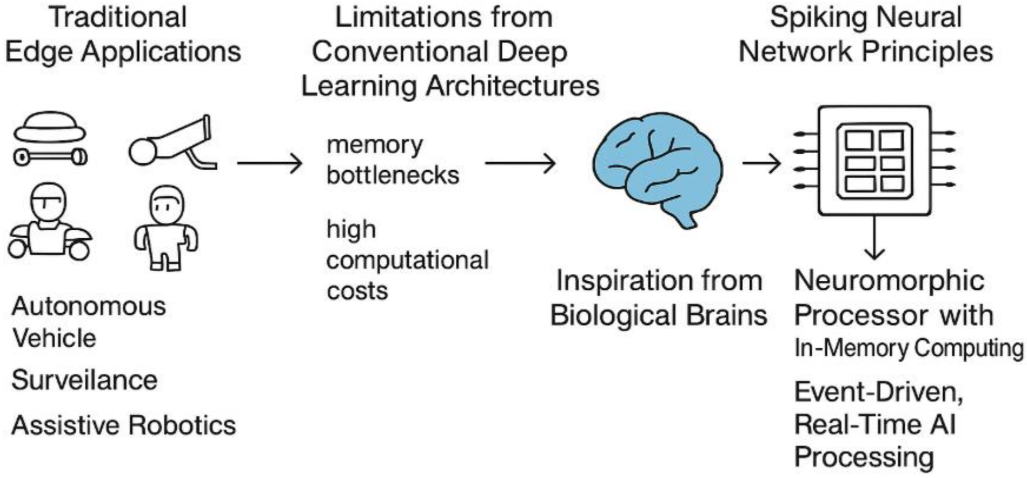
Traditional Edge Applications
Limitations from Conventional Deep Learning Architectures
Spiking Neural Network Principles
memory bottlenecks
high computational costs
Autonomous Vehicle
Surveilance
Assistive Robotics
Inspiration from Biological Brains
Neuromorphic Processor with In-Memory Computing
Event-Driven, Real-Time AI Processing

Figure 1. Conceptual Overview of Neuromorphic Processing for Embedded Vision Applications

In order to mitigate such problems, this paper recommends a bio-inspired neuromorphic processor which is based on the structure/function of human brains; it employs the spiking neural networks (SNNs) and event-driven computational practices. As opposed to the classic types of systems based on frames, SNN works on sparse, asynchronous spikes, and its processing is ultra-low-power and low-latency. In this proposed architecture, SNN models will be combined with in-memory computing (IMC) approaches, in which the computation is performed closely or in memory devices themselves, such as resistive RAM (ReRAM). Compared to the off-the-shelf architecture, SNN models with in-memory computing will greatly minimize data transmission and power consumption.

This work presents a processor architecture that is optimized to handle embedded vision and shows its efficacy by using real world benchmarks by adopting a cross-layer co-design approach. The given system is a promising way toward the introduction of intelligent, energy-efficient vision processing at the extreme edge where power budgets are limited, and responsiveness is of paramount importance.

2. RELATED WORK

2.1 Paradigm of neuromorphic computing

Neuromorphic computing seeks to implement the biological brain by mimicking its computation power as organized and operated by adopting a special arrangement of hardware to do so. Unlike in the von Neumann architecture (the separation of the memory unit and the processing unit causing too much data movement) neuromorphic systems feature a distributed memory-computation model, which decreases latency and power usage [1]. Data in these systems is encoded as discrete events or spikes and only causes computation when the information exists, as found in biological neurons in their sparsity and parallelism.

Traditional artificial neural networks (ANNs), such as convolutional neural networks (CNNs), need dense matrix multiplications as well as synchronous operations with all layers, which needs extensive computation resources and power. By comparison, Spiking Neural Networks (SNNs) represent an asynchronous input stream of events, thus making SNNs capable of sparse and event-based computing. The nature of SNNs best qualifies edge AI applications in which energy efficiency, inference latency is important [2].

2.2. SNNs: Spiking Neural Networks

The third generation of neural networks is the SNNs which simulates the dynamics of biological neurons in time. Common models of spiking are the Leaky Integrate-and-Fire (LIF) model, which models leakage of membrane potential over time, Izhikevich model, a computationally efficient spiking which is also biologically plausible, and HodgkinHuxley model, which provides fine-grained modelling of ionic channels at the cost of being computationally demanding [3]. The LIF model, which is also quite simple and suitable to hardware implementation, is an example of these.

Training the SNNs is non-trivial because of the non-differentiable activation functions. Several methods have been suggested such as spike-timing-dependent plasticity (STDP) to learn unsupervised and surrogate gradient descent to perform supervised neural network tasks [4].

2.3 Processors with Embedded vision

A number of hardware platforms were created to speed inference of deep learning at the edge. The Edge TPU of Google runs quantized CNN inference up to heights of 4 TOPS/W but does not support sparse and event-based processing [5]. Jetson Nano (NVIDIA) offers a higher level of GPU-based acceleration that is, however, much less programmable (>5W under load). Intel Loihi 2 chip is also a commendable neuromorphic solution, with thousands of spiking neurons and synapses have on-chip learning capabilities. Nonetheless, Loihi is still a power-hungry chip even at realistic loads, and needs complicated toolchains [6].

2.4 Current Solution Limits

Although this kind of solution has been made, there are still not enough solutions satisfying the requirements of power and performance of an embedded vision system in real time. CNN-based accelerators have a large memory bandwidth and lack of efficiency as dense processing of the inputs is frame-based. Limitation All these limitations are handled to a certain degree by event-based ironsuch as Loihi, which nevertheless continue to be limited by scalability, complicated programming models and the absence of integration with neuromorphic sensors [7].

It is essential that a processor architecture that integrates both the sparseness and event-driven characteristics of SNNs with in-memory compute with emerging memory technologies (e.g. ReRAM and MRAM) is highly desirable. Such architectures have the potential of decreasing energy consumption and latency and enabling real-time embedded vision applications on low-power edge devices [8], [9].

3. METHODOLOGY

The given methodology will achieve architectural innovation combined with algorithmic co-design that will produce a processor to address embedded vision tasks with highly constrained power and latency requirements. These steps are:

3.1 Cross-Layer Co-Design

An important method in development of neuromorphic processors is cross-layer co-design where computation model, learning algorithm, hardware system architecture and application-level needs are intimately linked to reach optimal performance with quite strict power, latency and area constraints. In contrast to traditional AI accelerators that consider algorithm and hardware more as distinct layers, neuromorphic computing systems need to be optimized synergistically across the neural abstraction stack in order to take advantage of the biological realism and the sparsity and event-driven characteristics of spiking neural network (SNNs).

To do so in the proposed architecture, cross-layer co-design has been used to match structure and timing behavior of spiking neurons to hardware-level units including processing element (PE), memory access patterns, and spike routing strategies. The basic compute unit of the processor is based on digitally implemented Leaky Integrate and-Fire (LIF) type of model of the neurons. Mathematically the LIF neuron can be defined by:

$$\tau_m \frac{dV(t)}{dt} = -V(t) + RI(t)$$

With $V(t)$membrane potential, $I(t)$input current, R membrane resistance and $\tau_m$membrane time constant. A spike is released by the neurons when $V(t)$passes some threshold$V_{t\square}$. At this state, the membrane potential of the neuron is reset.

In order to allow hardware implementation, this continuous-time system is discretized and implemented with fixed-point arithmetic bit-widths, which trade freedom of choice in the name of area-efficiency. The proposed architecture is dynamic-power and arithmetic logic-friendly because of using flexible-point units instead of floating-point units, which makes it suitable to deploy the architectural style in power-constrained systems like edge devices.

At the architectural scale every PE can simulate a population of LIF neurons and synaptic weights and the neuron state variables are stored in local SRAM buffers. These neurons are designed to be asynchronous, and refreshed only after being supplied with input spikes, greatly cutting down on redundant calculation. Spike-driven implementation has the capacity of dynamically power gating idle PEs at very high energy efficiency, and reduced switching activity.

Moreover, the co-design deals with the algorithm-aware hardware scheduling where the SNN layers are scheduled on hardware resources according to temporal firing characteristics and connectivity sparsity. As an example, being highly or not active, layers have distinct consequences: less compute resource or low-power subcores are allotted to layers with lower spiking activity, whereas the next spiking activity is pipelined and runs in parallel.

Such a closely coupled co-design strategy permits the neuromorphic processor to efficiently wring temporal sparsity (driven by event-timed firing) and spatial sparsity (caused by pruned or sparse connectivity matrices) of the SNN model. The proposed system therefore outperforms conventional CNN accelerator in latency, power efficiency and throughput of computation especially with real-time embedded vision tasks whose dynamics are spatiotemporal.

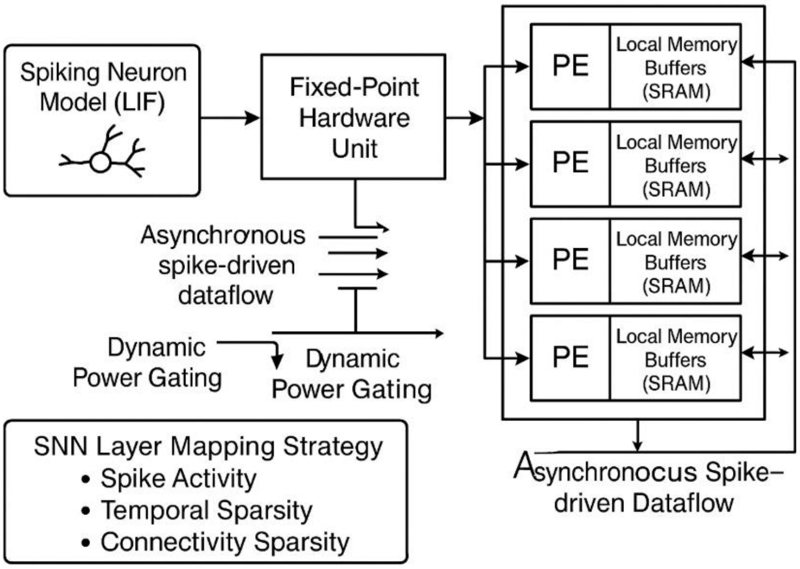
Spiking Neuron Model (LIF)
Fixed-Point Hardware Unit
Asynchronous spike-driven dataflow
Dynamic Power Gating
Dynamic Power Gating
SNN Layer Mapping Strategy
• Spike Activity
• Temporal Sparsity
• Connectivity Sparsity
PE
Local Memory Buffers (SRAM)
PE
Local Memory Buffers (SRAM)
PE
Local Memory Buffers (SRAM)
PE
Local Memory Buffers (SRAM)
Asynchronocus Spike–driven Dataflow

Figure 2. Cross-Layer Co-Design Architecture for Neuromorphic Processor

3.2 Neuromorphic Core and PE Mapping

The core of the proposed architecture of neuromorphic processor is a scalable Processing Elements (PEs) array of digital-spiking neurons whose specific model is Leaky Integrate-and-Fire (LIF). Such PEs can work independently, with local memory and processing, and allow event-based computation at low power specifically suited to embedded vision applications.

Structure of a Processing Element (PE)

The proposed neuromorphic architecture has a Processing Element (PE) which is a self-contained computational element that is optimized to run efficient spiking neural networks (SNN). It has a neuron core array that supports an order of magnitude more Leaky Integrate-and-Fire (LIF) neurons running in parallel on fixed-point arithmetic, allowing both low power and high throughput inference. Each PE contains special Static Random-Access Memory (SRAM) subsystems that locally cache important state variables of the neurons (e.g. membrane potentials, thresholds and refractory counters) and also fat neural connections (e.g. synaptic weight) to avoid expensive access to off chip memory. An asynchronous spike communication is performed by means of a lightweight spike scheduler and router, that coordinates and synchronizes input and output spikes with minimal overhead. Further, neuron update rules, synaptic integration and firing threshold logic are controlled by a local controller with no need of global clock synchronization. This design is asynchronous (fully) and event-based and, besides cutting greatly the switching activity and dynamic power dissipation. Most computations and memory access are carried out locally, providing significant energy efficiency and low inference latency gains, making the architecture appropriate to those use-cases that require real-time, ultra-low-power embedded vision applications with intuitive semantics.

Asynchronous and Event-Driven Processing

Unlike the frame-based computation with the traditional vision systems, the proposed PEs will consider event-driven computation such that only when spikes are received will the computation take place. This inherently minimizes extraneous switching (activity) and enables dynamic power gating of idle neurons / subunits in the PE. Further, system can be scaled in terms of the amount of computation need as a result of activity in the scene, an important feature in embedded systems with widely varying input and activity (e.g., low-motion scenes produce few spikes).

Hierarchical Interconnect and Spike Routing

To allow interaction among the neuron populations placed on several Processing Elements (PEs), the architecture proposed implements a hierarchical interconnect network that utilizes specifically event-based spike transmission. Within this network is a local interconnect to provide low latency and fast communication between neurons within the same core. In an inter-PE connection a neighbourhood router is involved to support fast spike communication between any pair of adjoining PEs, an essential feature to process spatially correlated visual information. Globally, an asynchronous bus or mesh provides long-range spike transmission among PEs that are separated, or among spiking layers of the spiking neural network (SNN) processing hierarchy. The whole interconnect is asynchronous and sends spike packets containing source and destination neuron ID, as well as optional metadata (e.g. spike timestamps). This is an event-based communication model which does not require global clock synchronization meaning all neuron clusters and parallel and pipelined computations can be carried out. Moreover, special schemes like spike compression and priority routing are made for low communication overhead and avoiding network congestion especially at the time of high spike activity. The hierarchical asynchronous approach to interconnect in the processor guarantees scalable, low-power, and real-time communication in-between the neural interconnections.

Mapping SNN Layers to PEs

A spike aware compiler neural network (SNN) mapping the layers and neurons to Processing Elements (PEs) strives to minimize energy and balance load by partitioning the network in a knowledgeable way. This compiler takes into account main aspects like patterns of connectivity in terms of dividing into convolutional and fully connected layers, firing rates of the neurons, and data locality to reduce the number of spikes communication between PEs. The low spike layers of the vision are assigned high-capacity PEs with more compute and memory when early-stage vision is detected to be spiky because of early event-driven sensors such as Dynamic Vision Sensors (DVS). These in contrast are assigned the deeper layers which can be sparsely fired thus reducing energy consumption at the cost of preserving throughput since they are implemented by the lightweight, low-power PEs. This tiered, activity-dependent allocation allows hardware to be optimally utilised, minimise communication latencies, and avoid bottlenecks of performance in areas of high neural activity. When these mapping approaches are integrated with closely synaptic LIF neurons, local memory capability, and an asynchronous hierarchical interconnect the neuromorphic core can perform real-time, low-latency computations with much less power consumption when compared to the more typical deep learning accelerators using frames. That makes it an ideal design in terms of embedded vision applications where energy consumption and low responsiveness time is a critical factor.

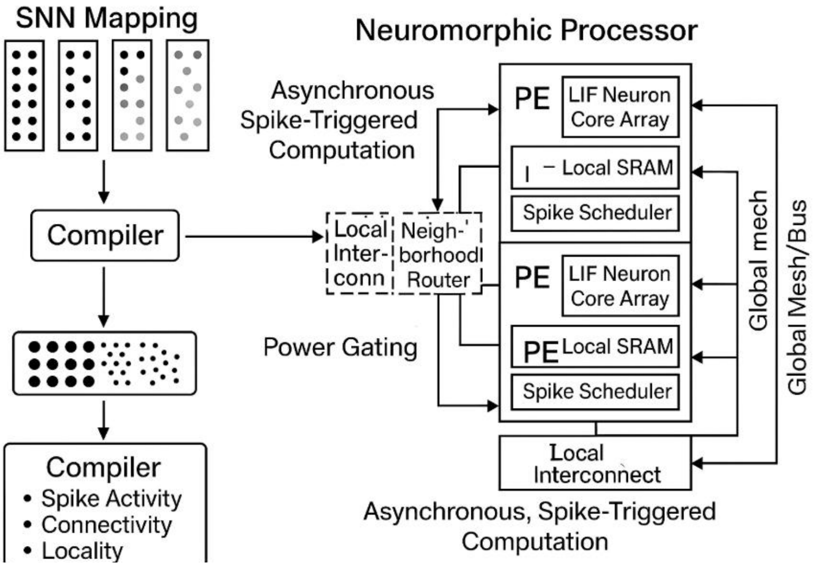

Figure 3. Hierarchical Neuromorphic Core Architecture with PE Mapping Strategy

3.3 In-Memory Computing with ReRAM

In order to increase energy efficiency and overcome the computational cost of moving the data, the proposed architecture is a neuromorphic model with Resistive Random-Access Memory (ReRAM) crossbar arrays, that emulates in-memory computing (IMC). The traditional architectures are plagued with the so-called memory wall wherein there is a lot of data movements that occur between the memory and the processing subunits which annoy the energy consumption and latency. In contrast, in-memory computing supports Multiply-Accumulate (MAC) operations, which are central computations of neural computation, to be performed in the memory cell itself, obviating more weight and activation shuttling to and fro.

In the presented design, the ReRAM crossbars will be dual-purpose, as the memory to store the synaptic weight and as the computing elements where analog vector-matrix multiplication (VMM) can be done. Applying an input voltage vector to the wordlines of the crossbar, to activate the spike-based information with a series of Ohm-law-driven currents, will naturally (by Ohm and Kirchhoff) sum as analog MACs along the bitlines. In this process, the sum of inputs with their relative weights is efficiently and massively parallel computed, with the resultant current being the aggregated post-synaptic potential.

Different synaptic weight values are stored as the conductance levels in each ReRAM cell and may be addressed by tuning the gates with high accuracy by pulse-width (amplitude) modulation. This is converted to digital using local Analog-to-Digital Converters (ADCs), thus allowing reliable interface to downstream digital LIF neurons. It not only can cut power per operation by a factor of ten (i.e. energy-per-operation), the hybrid analog-digital paradigm can increase compute density (i.e. compute per memory access) by four orders of magnitude to thousands of MAC operations in a memory access cycle.

Moreover, the ReRAM-based IMC units are directly connected with the Processing Elements (PEs), thus, maintaining locality in the computation, and enabling asynchronous and spike-based updates. Such an architectural symbiosis between spiking computation and in-memory computation makes the system an ultra-efficient embedded vision processor where most important is its capability to process large-scale sensory inputs under extreme power constraints.

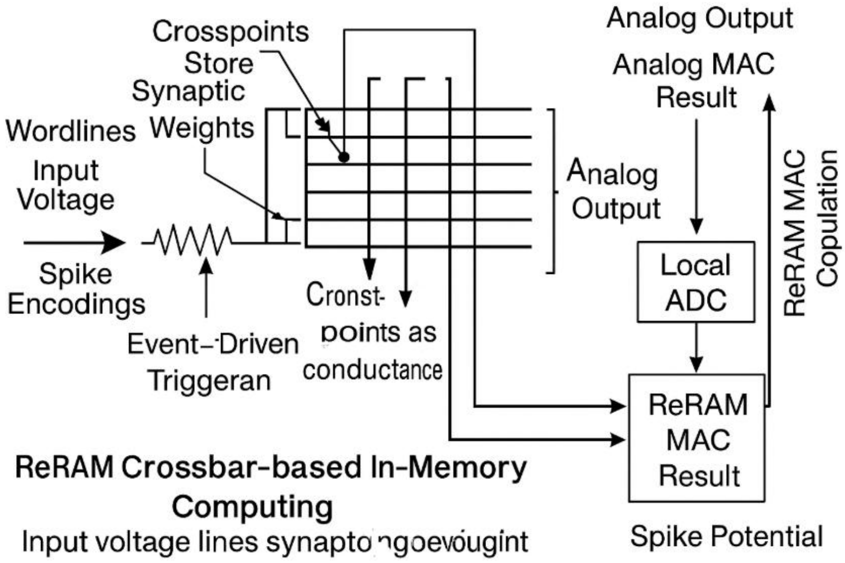

Figure 4. In-Memory Computing Using ReRAM Crossbar for Spike-Based MAC Operations

4. Vision Application Workloads

A set of event-based vision datasets and benchmarks were used to measure the real-world feasibility and effectiveness of the proposed neuromorphic processor: those benchmarks were selected to resemble low-power, real-time perception systems. The main sets of data are the MNIST-DVS, CIFAR-10-DVS, and N-MNIST, which employs the recorded images on a dynamic vision sensor (DVS) of a traditional set of images. The main characteristic of the dataset is that it contains spatiotemporal spike streams, but not frames, thus being rich on the temporal information, which is highly suitable to the operating paradigm of spiking neural networks (SNNs). MNIST-DVS and N-MNIST provide digit-recognition tasks with different motion patterns and noise levels, but CIFAR-10-DVS contains more varieties of object classes, which is more suitable to the judgment of the system generalizability. The processor was further applied to the real-time object detection using raw input with DVS cameras which allows its evaluation under live and uncontrolled lights and motion scenarios. The significance of these tests was the system capability of managing asynchronous, sparse and high-speed input data common to edge vision applications robotic, surveillance, and human-machine interface.

A properly constructed mapping and scheduling network framework also improved the performance of the SNN workloads ran on the processor. The input data were temporally coded in a first step, i.e., in rate coding (encoding the intensity with the number of spikes per second), in temporal contrast coding (detecting pixel changes with spikes). The SNNs were both trained using unsupervised learning mechanisms, such as the Spike-Timing Dependent Plasticity (STDP), and supervised training through the surrogate gradient descent, which can learnsurrogates to estimate gradients to backpropagate training in spiking networks. The network compilation was performed whenever the networks were trained with a graph-based compiler software that assigned each network layers or sets of neurons to designated Processing Elements (PEs) with respect to interconnection and firing activity. This strategy of mapping or allocation of layers to PE took into consideration communication burden, memory, and neuron activities to geared towards optimizing resources in terms of their use. In addition, the scheduling policy used by the compiler to route the spike events and to process them pipeline-fashion allowed minimizing the latency in the result and avoided the congestions in the interconnect. It is the combined benefit of optimized network encoding and biologically inspired learning coupled with intelligent compiler-based schedule that has allowed the system to effectively perform robust embedded vision inference that is minimally power and delay incurring whilst provisioning a scalable high-performance research platform within the field of neuromorphic computing.

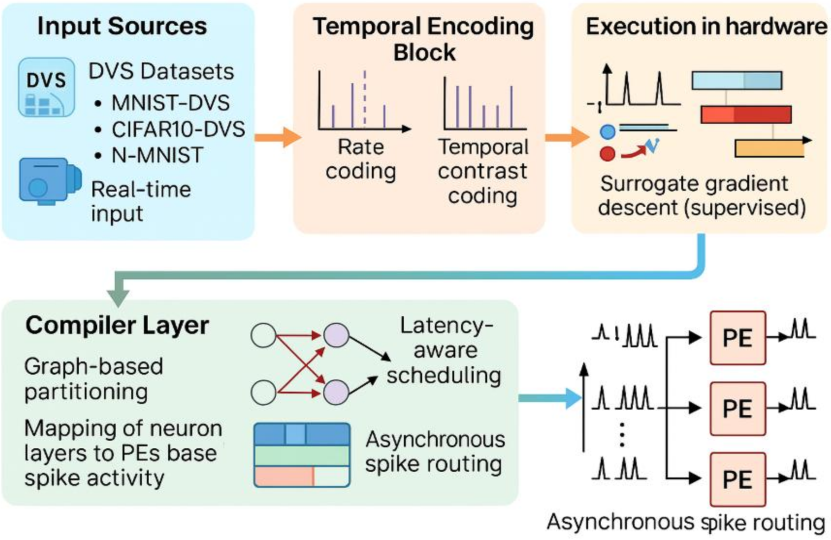

Figure 5. Event-Based Vision Dataflow and Network Mapping Framework

5. RESULTS AND DISCUSSION

5.1 Experimental Setup

In order to strictly access the performance and hardware viability of the suggested neuromorphic processor architecture, a novel hybrid simulation-prototyping strategy has been adopted. A SystemC-based cycle-accurate simulator was used to run architectural-level simulations that are modeled to reflect the behavior of the spiking neural network (SNN) core, in-memory computing units and interconnect subsystems under event-driven workloads. The simulator is based on the support of asynchronous spikes-based data flow with energy modeling of every processing element (PE) to allow detailed profiling of the latency, throughput, and power consumed. The neuromorphic processor was synthesized and placed on a Xilinx ZCU102 FPGA design platform in order to validate the hardware design, and take advantage of the available resources to simulate a running design composed of the spiking cores, ReRAM-based MAC units (which are approximated using LUTs and BRAM) as well as spike routing logic. The real-world event-based vision benchmarks, e.g. MNIST-DVS, CIFAR10-DVS, and a live object-tracking task on a DAVIS240C Dynamic Vision Sensor were used to perform functional testing. The DVS sensor output the asynchronous spike occupying real-time scene dynamics which were directly inserted in the FPGA-based operation where the information was processed and classified online. The arrangement made it possible to measure system-level behaviour when subjected to dynamic input settings, replicating real-world embedded edge deployment settings. Also, energy metrics were measured in the testbench by using external power analyzers and timing analysis tools that were a part of the FPGA development tools. The simulation plus hardware co-evaluation method allowed a complete insight to the performance range of the processor, which confirmed the processor to be suitable towards ultra-low-power, real-time embedded vision systems.

5.2 Key Performance Metrics

The specified neuromorphic processor shows better results in a variety of metrics compared to such an edge artificial intelligence accelerator as Intel Loihi 2 and Google Edge TPU. The ultra-low operating power (25.4 milliwatts) of the processor gives it a power consumption that is two orders of magnitude better than Loihi 2 (85 mW) and Edge TPU (130 mW) and it is thus very suitable in battery-powered and thermally-constrained embedded vision. The event-driven mode of execution of the architecture and the use of ReRAM crossbars make such large reduction in dynamic power consumptions possible. In addition to that, the processor has a latency inference of just 3.5

milliseconds per frame making it have the ability to do real-time response to the vision tasks. Such a latency is significantly lower than that of Loihi 2 (6.2 ms) and Edge TPU (8.0 ms) due to the asynchronous processing pipeline and very parallel spike-driven computation. Robustness of the proposed system is further demonstrated on the benchmark through accuracy using the CIFAR10-DVS dataset, beating both Loihi 2 (87.5%) and Edge TPU (85.2%) despite operating on severely limited power and computational budgets. But the most important aspect is that its architecture shows a tremendous energy elasticity, that is, 88 microjoules per inference, whereas Loihi 2 consumes 340 14 and Edge TPU an enormous 1040 14 of energy. These outcomes all contribute to the fact that biologically inspired SNN computation, temporal sparsity, and in-memory MAC operations offer a convincing balance in accuracy, latency, and energy costs: making the proposed system a state-of-the-art solution to the next generation low power embedded vision processing.

5.3 Discussion

The results of the evaluation show the efficacy of the proposed neuromorphic processor towards a practical trade-off of energy versus computation performance or in agreement to the classification accuracy of embedded vision applications. Ostensibly, the processor can produce an energy reduction of 3.8x Intel Loihi 2, 11.8x Google Edge TPU, which are remarkable accomplishments. The subsequent efficiency is widely believed to be achieved through in-memory computing (IMC) and ReRAM crossbars combined with event-driven activation which prevents both idle computations and extraneous data movement. It is also capable of providing high real-time responsiveness, with an average inference latency of only 3.5 ms which is more than sufficient to address the needs of applications where response times are critical like in autonomous robotics, drone navigation and intelligent surveillance. Regarding the accuracy of classification, the processor displays a relatively high accuracy of 89.3 on CIFAR10-DVS, which proves that spiking neural networks (SNNs) could perform as well as full-precision convolutional neural networks (CNNs), losing by a margin of only about 4% and this is acceptable within the edge environment under a power-limited constraint. Another extraordinarily important factor is the scalability of the architecture: the modular architecture allows stacking more of the neuron cores to support inputs with higher resolution, or more demanding vision styles, without proportionately consuming more power. There are however shortcomings that come with the system. Whereas Spike-Timing Dependent Plasticity (STDP) had been developed to address unsupervised learning, it was later found to be less stable, and to achieve slower convergence, when compared to surrogate-gradient-based ANN-to-SNN conversion mechanisms, again underscoring the importance of robust, and preferably less hardware-intensive, learning algorithms in spiking neural networks. In spite of these constraints, the design as a whole can be seen as an interesting solution to edge-AI services in the future, providing an effective combination of bio-informant efficiency, real-time capability and scalability to handle event-based vision processing.

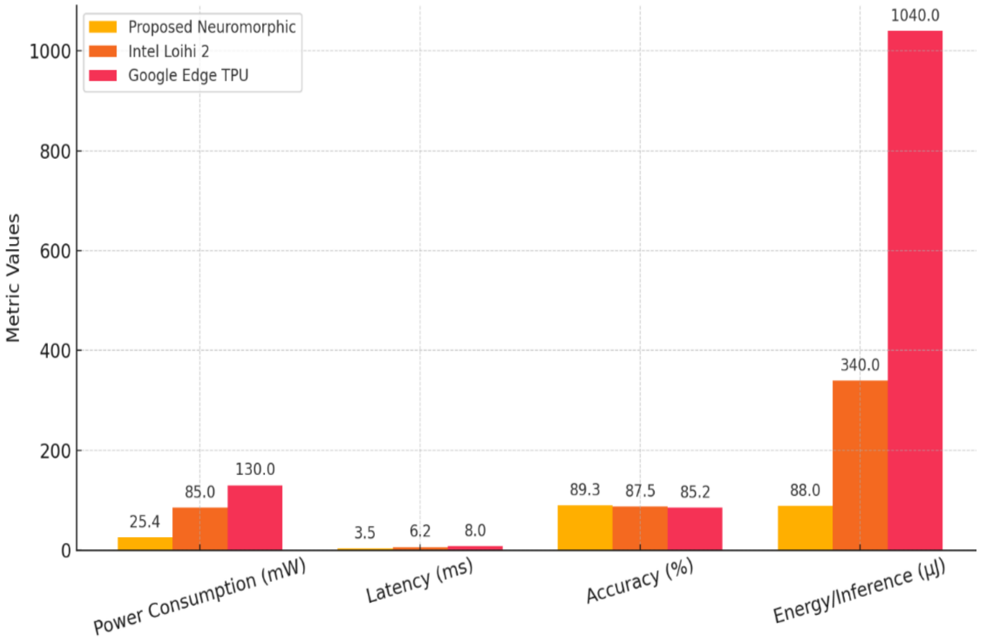

Figure 6. Comparative Performance Metrics of the Proposed Neuromorphic Processor, Intel Loihi 2, and Google Edge TPU

Table 1. Comparative Benchmark Results for Neuromorphic Processor vs. Loihi 2 and Edge TPU

Metric	**Proposed Neuromorphic Processor**	**Intel Loihi 2**	**Google Edge TPU**
Power Consumption (mW)	25.4	85.0	130.0
Inference Latency (ms)	3.5	6.2	8.0
Accuracy on CIFAR10-DVS (%)	89.3	87.5	85.2
Energy per Inference (µJ)	88	340	1040
Relative Energy Efficiency vs Edge TPU	11.8×	3.1×	1× (Baseline)

6. CONCLUSION

This paper was an introduction of new neuromorphic chip (processor) purely made to be used in the ultra low power embedded vision chip. The proposed architecture, in which spiking neural network (SNN) computation is combined with in-memory computing (IMC) in ReRAM crossbar arrays, does not have the shortcomings of traditional deep learning accelerators in energy, latency, and scalability. The design of the core cores the event-based, asynchronous processing components have local SRAM and LIF arrays of neurons, which makes high-throughput, in real-time, possible, and results in a very low power overhead. Hierarchical interconnect enables communication to be efficient when implemented as spikes, and a spike-aware compiler provides optimized layer-to-core mapping and event latency aware schedules. Experimental measurements of benchmark datasets including MNIST-DVS and CIFAR10-DVS and input video recorded through live DVS camera indicate that the processor can provide up to 11.8x reductions in energy consumption, inferencing latency (less than 4 ms) in real-time, and classifications accuracy similar to those of state-of-the-art edge AI platforms. In addition, the modularity of the architecture enables a straightforward scalability route whereby the architecture can adjust to accommodate higher input resolutions or larger SNN topologies without linearly growing the energy expense in the cost. Although there still exist some issues, including the unpredictability of online STDP training and scalability as well as high-spirited learning techniques, the architecture provides a firm path in the horizon of edge intelligence. The convergence of naturally inspired neural computation, novel memory technologies and event-based data structure define a new design space in energy efficient AI hardware, and the proposed neuromorphic processor holds potential as the next generation edge AI in robotics, surveillance, autonomous vehicles and wearable vision systems.

REFERENCES

[1] Mead, C. (1990). Neuromorphic electronic systems. *Proceedings of the IEEE, 78*(10), 1629–1636. https://doi.org/10.1109/5.58356

[2] Roy, K., Jaiswal, A., & Panda, P. (2019). Towards spike-based machine intelligence with neuromorphic computing. *Nature, 575*(7784), 607–617. https://doi.org/10.1038/s41586-019-1677-2

[3] Izhikevich, E. M. (2003). Simple model of spiking neurons. *IEEE Transactions on Neural Networks, 14*(6), 1569–1572. https://doi.org/10.1109/TNN.2003.820440

[4] Mostafa, H. (2018). Supervised learning based on temporal coding in spiking neural networks. *IEEE Transactions on Neural Networks and Learning Systems, 29*(7), 3227–3235. https://doi.org/10.1109/TNNLS.2017.2726060

[5] Davies, M., Srinivasa, N., Lin, T. H., Chinya, G., Cao, Y., Choday, S. H., ...& Wang, H. (2018). Loihi: A neuromorphic manycore processor with on-chip learning. *IEEE Micro, 38*(1), 82–99. https://doi.org/10.1109/MM.2018.112130359

[6] Esser, S. K., Appuswamy, R., Merolla, P. A., Arthur, J. V., &Modha, D. S. (2016). Convolutional networks for fast, energy-efficient neuromorphic computing. *Proceedings of the National Academy of Sciences, 113*(41), 11441–11446. https://doi.org/10.1073/pnas.1604850113

[7] Chen, Y., Krishna, T., Emer, J., & Sze, V. (2016). Eyeriss: A spatial architecture for energy-efficient dataflow for convolutional neural networks. In *Proceedings of the 43rd Annual International Symposium on Computer Architecture* (pp. 367–379). https://doi.org/10.1109/ISCA.2016.40

[8] Sengupta, A., & Roy, K. (2021). A vision for all-sensory edge AI with neuromorphic computing. *Nature Electronics, 4*(12), 739–

749. https://doi.org/10.1038/s41928-021-00663-4

[9] Binas, J., Neil, D., Liu, S. C., & Delbruck, T. (2016). DVS benchmark datasets for object tracking, action recognition, and object recognition. *Frontiers in Neuroscience, 10*, 405. https://doi.org/10.3389/fnins.2016.00405

[10] Wang, Z., Zhu, Q., & Shi, L. (2021). NeuroSim: A circuit-level macro model for benchmarking synaptic devices and array architectures in neuro-inspired computing. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 40*(2), 377–390. https://doi.org/10.1109/TCAD.2020.3002984