ISSN: 2581-3404 (Online) **IF: 5.68 (SJIF)**

IJIRTM, Volume-9, Issue-1, February-2025.

Neuromorphic Computing for AI Workloads: Spiking Neural Networks and Event-Driven Hardware for Ultra-Low-Power Inference

Ananya[1], Tishta[2], Sanju[3], Dev[4], Mamta Kumari[5]

Department of Computer Science and Engineering (AI & ML) [1,2,3,4,5]

Panipat Institute of Engineering and Technology (PIET), Samalkha, Panipat, Haryana, India[1,2,3,4,5]

2822326.cse@pietgroup.co.in[1], 2822343.cse@pietgroup.co.in[2], 2822310.cse@pietgroup.co.in[3], 2822330.cse@pietgroup.co.in[4], mamta.cse-aiml@piet.co.in[5]

Abstract: *Conventional deep learning accelerators built on the von Neumann architecture are approaching fundamental limits in energy efficiency, primarily because of the cost of continuously moving data between separate memory and compute units. Neuromorphic computing offers an alternative computational paradigm inspired by the structure and dynamics of biological nervous systems, in which information is encoded as sparse, asynchronous electrical events called spikes, and computation and memory are co-located within massively parallel arrays of artificial neurons and synapses. This paper presents a comprehensive survey and technical analysis of neuromorphic computing for artificial intelligence workloads, with emphasis on spiking neural network (SNN) models, event-driven hardware architectures, and the software ecosystems used to train and deploy them. We review the mathematical foundations of spiking neuron models, including the leaky integrate-and-fire and Izhikevich formulations, and the principal neural coding schemes used to represent information temporally. We survey representative digital, mixed-signal, and emerging memristive neuromorphic platforms, including Intel Loihi 2, IBM TrueNorth and NorthPole, SpiNNaker2, and BrainChip Akida, comparing their architectural philosophies, reported energy efficiencies, and target workloads. We examine training methodologies spanning unsupervised spike-timing-dependent plasticity, artificial-neural-network-to-SNN conversion, and gradient-based surrogate learning, along with emerging software toolchains and hardware-agnostic intermediate representations. Benchmark data drawn from recent literature demonstrate one to two orders of magnitude improvements in energy-per-inference and latency for event-driven workloads relative to GPU and CPU baselines. Finally, we discuss open challenges including device variability, training instability, immature software ecosystems, and the absence of standardized cross-platform benchmarks, and we outline promising research directions such as photonic neuromorphic computing, three-dimensional monolithic integration, and in-sensor computing that may further extend the energy and latency advantages of brain-inspired hardware for ubiquitous, always-on edge intelligence.*

Keywords: - Spiking Neural Networks, Neuromorphic Computing, Event-Driven Hardware, Edge AI, Low-Power Inference, Loihi, Truenorth, Memristor, Dynamic Vision Sensor, Surrogate Gradient Learning.

Introduction

The past decade of progress in artificial intelligence has been driven overwhelmingly by scaling: larger models, larger datasets, and larger clusters of graphics processing units (GPUs) executing dense matrix multiplications at ever-higher clock rates. This scaling strategy has produced remarkable capabilities, but it has also produced an energy problem. Training and, increasingly, serving large neural networks now consumes electricity at data-center scale, while a growing share of AI inference must instead run on power- and thermally-constrained edge devices

such as wearables, implantable sensors, autonomous drones, and battery-operated industrial monitors. Conventional accelerators, however efficient per operation, are built on the von Neumann architecture, in which a physically separate memory hierarchy must repeatedly transfer weights and activations to arithmetic units. This data movement, not the arithmetic itself, dominates the energy budget of most inference workloads, a phenomenon widely termed the memory wall or von Neumann bottleneck. Biological brains solve an analogous computational problem with several orders of magnitude greater energy efficiency. The human brain performs the equivalent of roughly ten quadrillion synaptic operations per second while consuming approximately twenty watts of power, a feat unmatched by any digital system built to date. Neuroscience research has identified several architectural principles that appear to underlie this efficiency: computation and memory are physically co-located at the synapse; neurons communicate through sparse, asynchronous, binary events called action potentials or spikes rather than continuously valued signals; and processing is inherently event-driven, meaning that energy is expended only when and where information actually changes. Neuromorphic computing is the discipline that attempts to translate these principles into silicon and, increasingly, into other physical substrates, producing hardware whose computational substrate mirrors the sparse, asynchronous, and massively parallel character of biological neural tissue. This paper focuses on two intertwined threads within neuromorphic computing that are directly relevant to practical AI workloads. The first is the spiking neural network (SNN), a class of neural network model in which artificial neurons integrate incoming signals over time and emit discrete spikes only when an internal state variable crosses a threshold. SNNs replace the dense, synchronous tensor operations of conventional artificial neural networks (ANNs) with sparse, temporally extended, event-driven computation, and are the natural algorithmic counterpart to neuromorphic hardware. The second is event-driven hardware, encompassing both neuromorphic processors that natively execute SNNs through asynchronous spike routing, and event-based sensors, most notably dynamic vision sensors (DVS), that report only pixel-level brightness changes rather than full frames. Together, spiking algorithms and event-driven silicon promise a full-stack alternative to the dense, frame-based, synchronous paradigm that dominates today's AI infrastructure, with the explicit goal of achieving ultra-low-power, low-latency inference suitable for always-on edge intelligence. The contribution of this paper is threefold. First, we provide a self-contained technical review of the neuron models, coding schemes, and learning rules that constitute the algorithmic foundation of spiking neural networks. Second, we survey the current landscape of neuromorphic hardware, spanning large-scale digital research chips, commercial edge processors, and emerging memristive and photonic devices, drawing on the most recent publicly reported specifications and benchmark results. Third, we synthesize evidence on the energy and latency advantages of neuromorphic systems relative to conventional accelerators, and we critically examine the practical obstacles, including training difficulty, device non-idealities, and software immaturity, that currently limit widespread adoption. The remainder of the paper is organized as follows. Section II reviews the biological and mathematical foundations of spiking neurons and neural coding. Section III surveys neuromorphic hardware architectures. Section IV discusses event-driven sensing. Section V covers training methodologies for SNNs. Section VI reviews software toolchains. Section VII presents benchmarking and energy-efficiency evidence. Section VIII surveys applications. Section IX discusses open challenges, Section X outlines future research directions, and Section XI concludes the paper.

Spiking Neurons and Neural Coding

A. From Biological Neurons to Computational Models

Biological neurons integrate synaptic currents arriving from thousands of presynaptic partners at their dendrites, and the resulting membrane potential evolves according to the passive and active electrical properties of the cell. When the membrane potential exceeds a threshold, the neuron generates a rapid, stereotyped voltage excursion, the action potential or spike, which propagates along the axon and triggers neurotransmitter release at downstream

synapses. Because the shape of the action potential is largely invariant, information is believed to be carried primarily by the timing and rate of spikes rather than by their amplitude, a property that computational neuroscientists have long argued makes spike trains an efficient, low-bandwidth communication code.

The Hodgkin–Huxley model, developed from experiments on the squid giant axon, describes spike generation through a system of coupled nonlinear differential equations governing sodium and potassium ion channel conductances. While biophysically detailed and highly accurate, the Hodgkin–Huxley formulation is computationally expensive, requiring numerical integration of four coupled state variables per neuron, which makes it impractical for large-scale simulation or hardware implementation. Consequently, neuromorphic engineering has converged on simplified phenomenological models that reproduce the spiking behavior of biological neurons at a fraction of the computational cost.

B. The Leaky Integrate-and-Fire Model

The leaky integrate-and-fire (LIF) neuron is the workhorse model of both computational neuroscience and neuromorphic hardware. The membrane potential V(t) of a LIF neuron evolves according to a single first-order linear differential equation, $\tau \cdot dV/dt = -(V - V\_rest) + R \cdot I(t)$, where τ is the membrane time constant, V_rest is the resting potential, R is the membrane resistance, and I(t) is the input current contributed by incoming spikes weighted by synaptic efficacy. When V(t) reaches a firing threshold V_th, the neuron emits a spike, the membrane potential is reset (either to V_rest or by subtracting the threshold), and the neuron typically enters a brief refractory period during which it cannot fire again. The LIF model captures the essential integrate-and-fire dynamic of biological neurons, including temporal summation of inputs and a leak term that causes the membrane potential to decay toward rest in the absence of input, while remaining simple enough to be simulated with a single multiply-accumulate-and-compare operation per timestep, or implemented directly in digital or analog circuitry.

Variants of the LIF model extend its expressiveness while preserving computational tractability. The adaptive LIF neuron introduces a slow adaptation current that increases the effective threshold after each spike, reproducing spike-frequency adaptation observed in cortical neurons and improving performance on temporal classification tasks. The Izhikevich model instead uses a two-dimensional system of equations, one governing membrane potential and one governing a slower recovery variable, and can reproduce a remarkably wide range of biologically observed firing patterns, including bursting, chattering, and regular spiking, by adjusting only four parameters, at a computational cost only marginally higher than LIF. Neuromorphic chips such as Intel Loihi 2 expose microcode-programmable neuron models that allow designers to select among LIF-family dynamics, graded (multi-bit) spike outputs, and custom dendritic compartments, trading biological fidelity against implementation cost on a per-application basis.

C. Synaptic Plasticity and Network Topology

Beyond the dynamics of individual neurons, the connectivity pattern and adaptive properties of synapses shape the computational capability of a spiking network. Feedforward architectures, analogous to convolutional or fully connected layers in conventional deep learning, remain the most common topology in neuromorphic classification tasks because they map straightforwardly onto existing training pipelines. Recurrent spiking architectures, in which neurons within a layer or across layers form feedback loops, are of particular interest for temporal processing tasks because recurrent connectivity allows the network to maintain an internal, dynamically evolving state that persists beyond the duration of any single input spike, a property exploited in reservoir-computing approaches such as liquid state machines, where a large, fixed, randomly connected recurrent reservoir of spiking neurons projects a low-dimensional input into a high-dimensional, temporally rich feature space that a simple linear readout can then classify. Synaptic delays, the finite propagation time between a presynaptic spike and its effect on a postsynaptic neuron, are a further architectural degree of freedom largely absent from conventional deep learning but native to spiking systems; learnable or heterogeneous synaptic delays have been shown to

improve temporal pattern discrimination and can substitute, in part, for network depth or recurrence in certain sequence-modeling tasks, motivating hardware support for programmable delay lines in recent neuromorphic chip designs, including Loihi 2, subject to a hardware-imposed maximum delay of several tens of timesteps.

D. Neural Coding Schemes

A central design question in spiking neural network engineering is how continuous-valued information, such as pixel intensities or sensor readings, should be encoded into discrete spike trains, and conversely how a population of output spikes should be decoded into a classification or control signal. Rate coding represents a value by the average firing frequency of a neuron or population over a fixed time window; it is robust to noise and straightforward to train but requires averaging over many timesteps, which increases latency and energy consumption and partially negates the efficiency benefits of sparse spiking. Temporal coding instead represents information in the precise timing of individual spikes, for example through time-to-first-spike coding, in which more strongly activated neurons fire earlier, or through phase or latency coding relative to a reference oscillation. Temporal codes can in principle convey substantial information in a single spike per neuron, yielding very low latency and high energy efficiency, but are more sensitive to jitter and are harder to train with standard gradient-based methods. Population coding distributes the representation of a variable across many neurons with overlapping tuning curves, improving robustness and resolution at the cost of increased neuron count. Practical neuromorphic systems frequently combine these schemes: sensory encoding stages often use temporal or event-based coding driven directly by physical transducers such as event cameras, while intermediate and output layers may rely on rate-like population codes to stabilize learning.

Neuromorphic Hardware Architectures

Neuromorphic hardware implements the co-located memory–compute and event-driven communication principles of biological neural tissue directly in the physical substrate, in contrast to software SNN simulation on conventional CPUs or GPUs, which must emulate spike-based dynamics through dense, clocked tensor operations. Existing neuromorphic platforms can be broadly classified along two axes: the physical representation of computation, spanning fully digital, mixed-signal analog-digital, and emerging memristive or photonic implementations; and the intended deployment regime, spanning large-scale research and data-center-class systems designed for brain-scale simulation, and compact, ultra-low-power edge processors designed for embedded deployment. Table I summarizes representative platforms discussed below.

A. Digital Neuromorphic Processors

Intel's Loihi 2 is among the most extensively studied digital neuromorphic research chips. Fabricated on the Intel 4 process node, Loihi 2 integrates approximately 120 to 128 fully asynchronous neuromorphic cores alongside six embedded Lakemont x86 processor cores, connected by an on-chip asynchronous network-on-chip that supports packet-switched spike routing. Each neurocore can host thousands of programmable neuron compartments, up to roughly 8,192 per core, with local SRAM storing membrane state and synaptic weights, giving a single chip capacity of up to one million neurons and on the order of one hundred twenty million synapses. A key architectural refinement relative to the original Loihi is support for graded, integer-valued spikes carrying up to thirty-two bits of payload rather than purely binary events, along with microcode-programmable neuron models that allow custom dendritic and somatic dynamics. Multiple Loihi 2 chips can be interconnected into systems ranging up to thousands of chips, exemplified by Intel's Hala Point system, and reported efficiency gains reach more than one hundred times over CPU baselines and roughly thirty times over GPU baselines for suitable event-driven workloads, with sub-millisecond, sub-millijoule inference achievable for small models.

IBM's neuromorphic research lineage began with TrueNorth, a fully digital, event-driven chip comprising 4,096 neurosynaptic cores fabricated in a 28 nm process, supporting approximately one million neurons and 256 million synapses at a power draw of roughly seventy milliwatts for the full chip, an efficiency profile that was, at the time

of its release, unprecedented among digital accelerators. IBM's more recent NorthPole chip extends this lineage with a substantially different design philosophy: rather than emulating a general-purpose neuromorphic substrate, NorthPole is a weight-stationary, memory-on-chip inference accelerator that eliminates off-chip DRAM access entirely, storing all model parameters across 256 cores fabricated in a 12 nm process with roughly twenty-two billion transistors. NorthPole achieves approximately four thousand times the throughput of TrueNorth, and in controlled comparisons has demonstrated roughly twenty-five times better energy efficiency, measured in frames processed per joule, than similarly-noded GPUs and CPUs on convolutional benchmarks such as ResNet-50 and YOLOv4. In subsequent large language model inference experiments using a three-billion-parameter Granite-derived model, a sixteen-chip NorthPole server achieved sub-millisecond per-token latency, reporting close to forty-seven times lower latency and over seventy times greater energy efficiency than the most competitive GPU baseline tested, illustrating that memory-centric, brain-inspired design principles can benefit not only spiking workloads but also conventional transformer-based inference when memory bandwidth is the binding constraint.

B. Large-Scale Simulation Platforms

SpiNNaker2, developed at TU Dresden in collaboration with the University of Manchester and commercialized through the spin-off SpiNNcloud, takes a hybrid approach in which each chip integrates roughly 152 low-power ARM Cortex-M4F cores together with dedicated multiply-accumulate and exponential/logarithm accelerators, fabricated in 22 nm FDSOI technology with nineteen megabytes of on-chip SRAM and two gigabytes of off-chip DRAM per chip. Unlike purely event-driven asynchronous designs, SpiNNaker2 relies on software-defined neuron and synapse models executed on general-purpose cores, giving it flexibility to run both spiking and conventional deep neural network workloads, and employs adaptive body-biasing and dynamic voltage-frequency scaling to enable near-threshold operation, yielding roughly a tenfold improvement in simulation capacity per watt over the original SpiNNaker system. Systems built from tens of thousands of SpiNNaker2 chips can model billions of neurons in real time, positioning the platform for large-scale computational neuroscience as well as commercial brain-inspired supercomputing.

C. Commercial Edge Neuromorphic Processors

Where research chips such as Loihi 2 and TrueNorth prioritize architectural flexibility and scale, a distinct class of commercial neuromorphic processors targets ultra-low-power embedded deployment. BrainChip's Akida is a fully digital, event-based neural processor fabricated in mature, low-cost process nodes and designed for on-chip, incremental learning without cloud connectivity; its second generation supports on the order of 1.2 million neurons and ten billion synapses in event-domain representation, targeting always-on sensing applications such as keyword spotting, gesture recognition, and vibration-based predictive maintenance at milliwatt-scale power consumption. Competing edge-focused designs include Innatera's Pulsar/T1 mixed-signal processor and SynSense's Speck and Dynap-family chips, which combine analog neuron circuits with digital routing fabrics to minimize static power. These commercial parts generally trade the raw neuron capacity and programmability of research chips for smaller silicon area, lower unit cost, and software toolchains, such as BrainChip's MetaTF, that integrate more directly with existing embedded development workflows.

D. Memristive and Emerging Analog Substrates

A parallel research direction seeks to implement synaptic weight storage and the associated multiply-accumulate operation directly within emerging non-volatile memory devices, most notably memristors based on resistive RAM (RRAM) or phase-change memory (PCM). A memristor's conductance can be set incrementally through applied voltage pulses and persists without power, allowing a single device to store an analog synaptic weight. When arranged in a crossbar array, a grid of memristors performs an entire matrix–vector multiplication in a single analog step, exploiting Ohm's and Kirchhoff's laws to compute a weighted sum of input voltages as an output current, which offers the prospect of extreme energy efficiency for the dominant operation in neural network inference. However, memristor-based crossbars face significant device-level challenges, including cycle-

to-cycle and device-to-device conductance variability, limited endurance, nonlinear and asymmetric weight-update characteristics that complicate on-chip learning, and the need for precise peripheral analog-to-digital and digital-to-analog conversion circuitry that can itself dominate system-level power and area. These challenges currently confine most memristor-based neuromorphic demonstrations to research prototypes and small-scale arrays rather than commercial-scale deployment, though the technology is advancing rapidly and is frequently cited as a promising path toward the still-unrealized goal of fully analog, in-memory spiking computation at brain-comparable density.

E. System-Level Design Considerations

Beyond the choice of neuron circuit and memory technology, several system-level design decisions materially affect the practical efficiency and usability of a neuromorphic platform. Spike routing architecture determines how events are communicated both within and between chips: address-event representation (AER) protocols, in which each spike is tagged with the address of its source neuron and routed asynchronously through a shared digital bus or network-on-chip, are the dominant approach and allow routing infrastructure to be shared efficiently across many low-activity neurons, at the cost of arbitration and potential congestion when local activity bursts occur. Synchronization strategy is a second key design axis: fully asynchronous designs maximize energy proportionality to activity but complicate deterministic timing and debugging, whereas designs such as Loihi 2 retain a global algorithmic timestep enforced through periodic barrier synchronization across cores, trading a small synchronization overhead for substantially simplified programming semantics and reproducibility. Precision is a third consideration; while early neuromorphic designs such as TrueNorth used strictly binary spikes and low-precision weights to minimize circuit complexity, more recent chips increasingly support graded, multi-bit spikes and higher-precision synaptic state, reflecting an empirical finding that a small amount of additional numerical precision substantially improves achievable task accuracy at limited additional energy cost, narrowing, without eliminating, the historical accuracy gap between spiking and conventional deep networks.

Table 1: Comparison of Representative Neuromorphic Hardware Platforms.

Platform	Cores/Units	Process Node	Neurons/Synapses (max)	Comm. Model	Reported Efficiency
Intel Loihi 2	120–128 neurocores	Intel 4 (7 nm)	≤1M neurons / 120M synapses	Async NoC, graded spikes	>100× vs CPU, ~30× vs GPU
IBM TrueNorth	4,096 cores	Samsung 28 nm	1M neurons / 256M synapses	Async, binary spikes	~70 mW/chip
IBM NorthPole	256 cores	12 nm (GF)	On-chip weight-stationary	Digital, memory-on-chip	25× vs 12 nm GPU/CPU
SpiNNaker 2	152 ARM cores/chip	22 nm FDSOI	Millions (system-scale)	Packet-switched AER	10× perf/W vs SpiNNaker1
BrainChip Akida (2nd gen)	Event-domain NPUs	28 nm (commercial)	1.2M neurons / 10B synapses	Fully digital, on-chip STDP	mW-range inference
Memristor crossbar (research)	Analog PE array	Emerging (RRAM/PCM)	Array-size dependent	Analog VMM, in-memory	Single-step matrix–vector multiply

Event-Driven Sensing

Neuromorphic processors realize their full energy and latency advantage only when paired with sensing modalities that natively produce sparse, asynchronous data, since converting dense frame-based sensor output into spikes after the fact reintroduces much of the redundant computation neuromorphic hardware is designed to avoid. The dynamic vision sensor (DVS), also called an event camera, is the most mature example of such a sensor. Rather than capturing full image frames at a fixed clock rate, each pixel in a DVS operates independently and asynchronously, emitting a discrete event only when the logarithmic photocurrent at that pixel changes by more than a threshold amount, together with a microsecond-resolution timestamp and the polarity (increase or decrease) of the change. Because static regions of a scene generate no events at all, DVS output is extremely sparse for typical natural scenes, and because each pixel operates independently, event cameras achieve microsecond-scale temporal resolution and very high dynamic range without the motion blur that afflicts conventional frame-based cameras under fast motion or challenging lighting.

Commercial event-camera vendors such as Prophesee and iniVation have demonstrated that DVS output pairs naturally with spiking neural network processing pipelines: events can be routed directly into a neuromorphic chip's asynchronous spike fabric without an intermediate frame-buffering or format-conversion stage, preserving both the sparsity and the fine temporal structure of the sensor data end to end. This sensor-to-processor pipeline has been demonstrated in low-power object tracking, gesture recognition, high-speed robotics, and fully neuromorphic drone flight controllers that perform visual odometry and obstacle avoidance using only spiking computation from sensor to actuator, reporting power consumption for the full perception-and-control pipeline in the range of hundreds of milliwatts rather than the several watts typical of frame-based vision pipelines on embedded GPUs. Beyond vision, analogous event-driven principles have been applied to auditory sensing through silicon cochlea models that convert sound into spike trains organized by frequency channel, and to tactile and olfactory sensing in robotic and prosthetic applications, extending the event-driven paradigm across sensory modalities.

Training Methodologies for Spiking Neural Networks

The discrete, non-differentiable nature of the spike-generation function is the central obstacle to training spiking neural networks, since the standard backpropagation algorithm that underlies modern deep learning requires differentiable operations at every layer. The neuromorphic computing community has developed three broad families of methods to address this obstacle, each with distinct trade-offs in accuracy, hardware compatibility, and training cost.

A. Unsupervised Local Learning: Spike-Timing-Dependent Plasticity

Spike-timing-dependent plasticity (STDP) is a biologically observed synaptic learning rule in which the change in synaptic weight depends on the relative timing of pre- and post-synaptic spikes: when a presynaptic spike consistently precedes a postsynaptic spike within a short window, the synapse is strengthened (long-term potentiation), whereas the reverse temporal order weakens the synapse (long-term depression). STDP is attractive for neuromorphic hardware because it is a purely local rule, requiring only information available at each synapse, and can be implemented directly in analog or mixed-signal circuitry, including memristor-based synapses whose conductance update characteristics naturally resemble the STDP learning curve. However, STDP alone is an unsupervised rule that discovers correlational structure in input spike trains rather than optimizing a task-specific objective, and networks trained purely with STDP typically achieve substantially lower accuracy than supervised alternatives on standard classification benchmarks, which has limited its use to unsupervised feature learning, novelty detection, and on-chip adaptation layered on top of otherwise pre-trained networks.

B. ANN-to-SNN Conversion

A widely used practical strategy sidesteps the non-differentiability problem entirely by first training a conventional, high-accuracy artificial neural network using standard backpropagation, and then converting the trained network into a functionally equivalent spiking network for deployment on neuromorphic hardware. Conversion typically replaces rectified linear (ReLU) activations with rate-coded integrate-and-fire neurons and applies weight and threshold rescaling (normalization) so that the average post-conversion spike rate approximates the corresponding pre-conversion analog activation. This approach can achieve near-original ANN accuracy on tasks such as image classification, and is attractive because it reuses the mature training infrastructure and pretrained model zoo of conventional deep learning. Its principal drawback is that accurately approximating analog activations with spike rates generally requires averaging over many timesteps, which increases inference latency and reduces the energy advantage of sparse, single-spike temporal coding; conversion-based networks are also inherently rate-coded and therefore forgo the potential efficiency of aggressive temporal coding. Reported accuracy degradation from conversion is typically small on well-studied benchmarks but can range from under one percent to several percent depending on architecture, timestep budget, and target dataset.

C. Surrogate Gradient Learning

The dominant approach in current SNN research is direct training with backpropagation-through-time using a surrogate gradient: during the forward pass, the network uses the true, non-differentiable spike function, while during the backward pass, the discontinuous derivative of the spike function is replaced with a smooth surrogate, such as a scaled sigmoid, fast-sigmoid, or piecewise-linear function, that provides a useful gradient signal in the vicinity of the firing threshold. Surrogate gradient training allows spiking networks to be optimized end to end for a specific task and timestep budget, generally achieves higher accuracy than conversion-based approaches at matched or lower timestep counts, and naturally exploits temporal structure in the input, making it well-suited to genuinely temporal tasks such as speech recognition, event-camera classification, and time-series forecasting. Its main costs are increased training-time memory and compute relative to standard ANN training, since gradients must be backpropagated through every simulated timestep, and a degree of training instability that has motivated auxiliary techniques such as membrane potential regularization, batch normalization variants adapted for spiking activations, and learnable time constants. Related biologically motivated alternatives, including eligibility-propagation (e-prop) and forward-mode online learning rules, aim to approximate the effect of backpropagation-through-time using only local, causal information, which is attractive for on-chip learning but currently trails surrogate gradient backpropagation in achievable accuracy on complex benchmarks.

D. Hybrid and Knowledge-Distillation Approaches

A growing body of work combines these paradigms, for example by initializing surrogate-gradient training from an ANN-to-SNN converted network to accelerate convergence, or by using knowledge distillation from a large, high-accuracy conventional teacher network to guide the training of a compact spiking student network intended for deployment on resource-constrained edge neuromorphic hardware. Such hybrid strategies have proven particularly effective for deploying SNNs derived from large pretrained models, including recent demonstrations of transformer-derived and large-language-model-inspired architectures adapted to spiking, event-driven execution on platforms such as Loihi 2, indicating that the algorithmic gap between conventional deep learning and neuromorphic computation is narrowing.

Software Ecosystems and Tool-chains

Historically, one of the most significant barriers to neuromorphic adoption has been the absence of software tooling comparable in maturity to CUDA, cuDNN, and the broader PyTorch/TensorFlow ecosystem that underpins conventional deep learning. This gap has narrowed considerably in recent years. Intel's Lava is an open-source framework that allows researchers to describe spiking and hybrid neural networks using high-level Python

application programming interfaces and to compile them for execution on either conventional processors or Loihi 2, with lower-level access to custom neuron microcode and embedded processor code for advanced users. Nengo provides a higher-level, biologically grounded modeling framework based on the Neural Engineering Framework, supporting deployment across CPUs, GPUs, and multiple neuromorphic backends. snnTorch and Norse extend the familiar PyTorch programming model with spiking neuron layers, automatic differentiation through surrogate gradients, and utilities for rate and temporal spike encoding, substantially lowering the barrier for deep learning practitioners to experiment with SNNs using existing skills and infrastructure. BrainChip's MetaTF toolchain similarly targets practitioners with an existing TensorFlow/Keras workflow, providing quantization-aware training and direct mapping to Akida silicon.

A parallel and increasingly important development is the emergence of hardware-agnostic intermediate representations for spiking networks, exemplified by the Neuromorphic Intermediate Representation (NIR), which defines a common computational graph format that can be exported from training frameworks such as Norse or snnTorch and imported into diverse deployment targets, including SpiNNaker2, SynSense Speck, and simulation backends, in a manner conceptually analogous to ONNX for conventional deep learning. Standardized intermediate representations, together with emerging cross-platform benchmarking suites discussed in Section VII, are widely regarded within the community as prerequisites for neuromorphic computing to transition from a collection of vendor-specific research toolchains into a broadly interoperable computing ecosystem.

Benchmarking and Energy Efficiency

Quantifying and comparing the energy efficiency of neuromorphic systems is complicated by heterogeneity in reporting methodology: published figures may reflect only core logic power or full system power including I/O and host communication, may or may not include the energy cost of spike encoding, and are frequently measured on different workloads, precisions, and batch sizes than competing conventional accelerators, making naive cross-paper comparison unreliable. The NeuroBench framework has been proposed specifically to address this problem, defining standardized workloads, metrics, and reporting harnesses that allow neuromorphic algorithms and hardware to be benchmarked on a comparable basis, spanning both algorithmic metrics, such as accuracy and activation sparsity, and hardware-level metrics, such as energy per inference and latency, and is increasingly adopted alongside general-purpose edge benchmarks such as MLPerf Tiny for cross-platform comparison.

Notwithstanding these methodological caveats, a consistent qualitative pattern emerges across independently reported studies: neuromorphic platforms deliver their largest advantages on genuinely sparse, temporally structured, and low-batch-size workloads, where conventional accelerators cannot fully exploit their massive parallelism and pay a fixed energy overhead for data movement regardless of activity level. Table II summarizes representative comparative results drawn from the literature discussed in preceding sections, spanning sensor-fusion, computer-vision, large-language-model, keyword-spotting, and brain-simulation workloads. Reported efficiency gains range from roughly one order of magnitude for latency-dominated inference tasks to several orders of magnitude for highly sparse, always-on sensing applications, with the important caveat that these gains typically apply to inference on models and tasks that are well matched to event-driven, sparse computation; dense, compute-bound workloads such as large-batch training of conventional convolutional or transformer architectures generally remain more efficiently served by GPUs and purpose-built dense accelerators.

Table 2: Representative Energy and Latency Comparisons from Recent Literature.

Task / Workload	Conventional (GPU/CPU)	Neuromorphic Platform	Reported Gain
Sensor fusion (Oxford RadarCar)	RTX 3060: 3.80 GOP/s/W	Loihi 2: 103.9 GOP/s/W	~27× power efficiency
ResNet-50 image inference	12 nm GPU baseline	IBM NorthPole	25× frames/Joule
3B-parameter LLM token inference	Comparable GPU	IBM NorthPole (16-chip 2U)	46.9× lower latency, 72.7× more efficient
Always-on keyword spotting	MCU (mW-active)	BrainChip Akida	Sub-mW event-driven inference
Large-scale brain simulation	HPC cluster (kW-MW)	SpiNNaker2 (million-core)	Order-of-magnitude perf/W gain

A. Sources of Efficiency Gains

The efficiency advantages summarized in Table II arise from a combination of distinct, partially separable mechanisms rather than a single architectural feature. Event-driven activation sparsity avoids computation entirely for inactive neurons and synapses, an advantage that scales with how sparse the underlying data and task genuinely are; memory-compute co-location avoids the energy cost of transporting weights and activations across a memory hierarchy, an advantage that persists even for dense workloads, as demonstrated by NorthPole's efficiency gains on conventional convolutional and transformer inference; low-precision, fixed-point or graded-spike arithmetic reduces per-operation energy relative to floating-point computation; and asynchronous, clockless or near-threshold circuit operation avoids the fixed energy overhead of global clock distribution and enables voltage scaling unavailable to conventionally clocked designs. Because these mechanisms are only partially overlapping, a platform need not implement all of them to realize meaningful efficiency gains, which explains why architecturally distinct systems, from the fully asynchronous, spike-native Loihi 2 to the synchronous, weight-stationary NorthPole, can each report substantial, if differently sourced, efficiency improvements over conventional baselines.

B. Interpreting Reported Comparisons

Readers of neuromorphic benchmark literature should treat vendor-reported efficiency multipliers with appropriate caution, since the choice of baseline accelerator, process node, precision, and whether reported power includes full system overhead or only core logic can each shift a reported gain by a factor of two or more without any change in the underlying algorithm or workload. Comparisons that hold process node approximately constant between the neuromorphic and conventional baseline, as in the twelve-nanometer NorthPole-versus-GPU comparisons and the Loihi-2-versus-embedded-CPU/GPU sensor-fusion comparison summarized in Table II, are generally more informative than comparisons across widely different process generations, since more advanced process nodes independently reduce power consumption regardless of architecture. With this caveat in mind, the consistent direction, if not always the precise magnitude, of reported gains across independent research groups and vendors lends credibility to the qualitative claim that event-driven, memory-centric architectures offer genuine and substantial efficiency advantages for appropriately matched workloads.

A further consideration in interpreting these results is the distinction between algorithmic sparsity and hardware-exploitable sparsity. A spiking network may exhibit low average firing rates in simulation, yet realize little energy benefit if the underlying hardware cannot skip computation or communication for inactive neurons and synapses at fine granularity. This observation has motivated increasing attention, both in benchmark design and in hardware

architecture, to metrics such as effective synaptic operations per inference and measured, rather than merely theoretical, energy per spike, since the latter can vary by more than an order of magnitude depending on routing distance, fan-out, and memory access patterns even on the same chip.

Applications

A. Always-On Edge Sensing

The clearest commercial application of neuromorphic computing to date is always-on sensing in battery-constrained devices, including voice-activated keyword spotting in hearables and smart-home devices, vibration-based predictive maintenance in industrial equipment, and gesture and presence detection in wearables. In these settings, the device must continuously monitor its environment for a rare triggering event while consuming minimal average power, a requirement that maps naturally onto the event-driven, sparse-activity operating regime of neuromorphic processors such as BrainChip Akida and Innatera's Pulsar/T1, both of which report sub-milliwatt to low-milliwatt active power for representative always-on classification tasks.

B. Robotics and Autonomous Systems

Robotic platforms, particularly small unmanned aerial vehicles with tight size, weight, and power constraints, benefit from the combination of event-camera perception and neuromorphic processing to achieve low-latency reactive control without the power budget of GPU-based perception stacks. Demonstrated systems include fully neuromorphic optical-flow-based obstacle avoidance and landing controllers, as well as neuromorphic implementations of simultaneous localization and mapping subcomponents, that report closed-loop control latencies in the low milliseconds and total system power consumption compatible with sub-100-gram aerial platforms, a regime largely inaccessible to conventional embedded GPU solutions.

C. Sensor Fusion and Autonomous Vehicles

Beyond aerial robotics, neuromorphic sensor-fusion pipelines have been evaluated for ground-vehicle perception tasks using automotive radar and LiDAR datasets, where studies comparing Loihi 2 against embedded CPU and GPU baselines on identical fusion workloads have reported power-efficiency improvements exceeding an order of magnitude, as summarized in Table II, suggesting a potential role for neuromorphic co-processors in always-on, low-latency perception subsystems for advanced driver-assistance and autonomous-vehicle applications, particularly where full-scale GPU compute is reserved for less time-critical, higher-level planning tasks.

D. Large-Model Inference and Data-Center Efficiency

While neuromorphic computing is most closely associated with edge deployment, IBM's NorthPole results on large language model inference demonstrate that memory-centric, brain-inspired architectural principles can also benefit latency- and energy-sensitive data-center inference for conventional, non-spiking transformer models, suggesting a broader role for neuromorphic-adjacent memory-compute co-location techniques beyond spiking workloads specifically, particularly for inference-serving scenarios in which per-token latency and energy, rather than raw training throughput, are the dominant cost drivers.

E. Biomedical and Neuroprosthetic Applications

Neuromorphic hardware's native compatibility with biological timescales and event-based signal representations has also motivated its use in implantable and wearable biomedical devices, including closed-loop neural-interface systems that decode motor intent from spike-like neural recordings for prosthetic control, and low-power seizure-detection systems that process electroencephalographic signals as sparse events, both of which benefit from the millisecond-scale, sub-milliwatt operating regime achievable with dedicated neuromorphic silicon in a form factor compatible with implantation or continuous wearable use.

F. Illustrative Case Study: Event-Driven Gesture Recognition

A frequently cited benchmark that illustrates the practical advantages of the full neuromorphic stack is dynamic gesture recognition using the DVS-Gesture dataset, in which a DVS camera records eleven categories of hand and

arm gestures performed by multiple subjects under varying lighting conditions. Because a DVS reports only pixel-level brightness changes, a static or slowly moving hand produces almost no events, while a rapid gesture produces a dense, temporally structured burst of events that a spiking network can classify using only a handful of algorithmic timesteps. Reported implementations of spiking convolutional networks for this task on neuromorphic hardware such as Loihi and TrueNorth-class processors have demonstrated classification accuracy exceeding ninety percent at power consumption several orders of magnitude below an equivalent frame-based convolutional pipeline evaluated on an embedded GPU, primarily because the frame-based pipeline must process every frame at a fixed rate regardless of scene activity, whereas the event-driven pipeline's computation and communication scale directly with the amount of motion actually present in the scene. This case study is frequently used as a minimal but representative demonstration of the central neuromorphic value proposition: energy consumption that is proportional to informational activity rather than to a fixed sampling clock.

Challenges and Open Problems

A. Device Variability and Analog Non-Idealities

Analog and mixed-signal neuromorphic circuits, and memristive synaptic arrays in particular, suffer from device-to-device and cycle-to-cycle conductance variability, limited write endurance, and nonlinear, asymmetric weight-update characteristics that degrade both inference accuracy and on-chip learning stability relative to idealized simulation. Mitigating these effects typically requires either additional calibration and error-correction circuitry, which erodes area and energy advantages, or training algorithms explicitly robust to weight perturbation, an active area of ongoing research.

B. Training Difficulty and the Accuracy Gap

Despite substantial progress in surrogate gradient learning, spiking neural networks trained from scratch still generally trail state-of-the-art conventional deep networks in accuracy on complex, large-scale benchmarks, and training remains more memory- and compute-intensive than standard backpropagation because gradients must be propagated through every simulated timestep. Reported ANN-to-SNN conversion accuracy losses in recent surveys range from under one percent to roughly five percent depending on architecture and timestep budget, and closing this residual gap without sacrificing the latency and sparsity advantages of low-timestep, temporally coded operation remains an open problem.

C. Software and Toolchain Immaturity

Although frameworks such as Lava, Nengo, snnTorch, and NIR have meaningfully improved accessibility, the neuromorphic software ecosystem remains fragmented and considerably less mature than the CUDA-centric conventional deep learning stack, with many advanced hardware features, such as Loihi 2's custom neuron microcode, gated behind restricted-access developer programs. This fragmentation increases the engineering cost of porting a given application across vendors and slows the feedback loop between algorithmic and hardware innovation.

D. Absence of Standardized, Comparable Benchmarks

As discussed in Section VII, heterogeneous reporting methodology across vendors and research groups makes rigorous cross-platform comparison difficult, and most published neuromorphic benchmark results still rely on relatively small-scale datasets such as MNIST, CIFAR-10, or DVS-Gesture, with large-scale, ImageNet-class evaluations remaining comparatively rare. Continued adoption of standardized suites such as NeuroBench and MLPerf Tiny, along with more consistent system-level (rather than core-only) power reporting, will be necessary for the field to make verifiable efficiency claims comparable to those routinely made for conventional accelerators.

E. Scalability and Integration

Scaling neuromorphic systems from single-chip demonstrations to the multi-chip, brain-scale systems required for the most ambitious applications introduces non-trivial engineering challenges in inter-chip spike routing bandwidth, barrier synchronization overhead across asynchronous domains, and the diminishing marginal efficiency benefit of event-driven computation as models grow large enough that most neurons are active in a typical timestep, eroding the sparsity assumption on which neuromorphic efficiency claims fundamentally rest.

G. Talent, Tooling, and Adoption Inertia

A less technical but nonetheless significant barrier to adoption is the comparatively small pool of engineers and researchers trained in event-driven, temporally extended programming models, relative to the very large community fluent in conventional tensor-based deep learning frameworks. Organizations evaluating neuromorphic hardware for a new product must typically weigh the prospective energy and latency benefits against the cost of retraining teams, rewriting data pipelines around asynchronous event streams, and accepting a smaller, less battle-tested software ecosystem, an adoption calculus that tends to favor conventional accelerators except in applications where energy or latency constraints are so severe that no conventional solution is viable. Lowering this adoption barrier through improved documentation, more capable simulators that allow algorithm development without physical hardware access, and closer integration with mainstream machine learning frameworks is likely to be as consequential for the field's near-term trajectory as further hardware innovation.

F. Encoding and Interfacing Overhead

Many practical AI workloads originate as dense, frame-based, or otherwise conventionally formatted data, and converting such data into spike trains suitable for neuromorphic processing, or converting spike-based outputs back into a form usable by downstream conventional systems, introduces encoding and decoding overhead that can offset part of the core computational efficiency advantage unless the encoding stage is itself implemented in low-power, ideally sensor-native, event-driven hardware such as a dynamic vision sensor or silicon cochlea.

Future Research Directions

A. Photonic and Optoelectronic Neuromorphic Computing

Photonic neuromorphic architectures, in which spiking or analog neural computation is performed using light rather than electrical current, promise extremely low propagation latency and, in principle, minimal resistive energy dissipation for signal routing, and have demonstrated promising early results for high-bandwidth analog matrix multiplication; extending these demonstrations to fully integrated, densely connected spiking photonic systems compatible with existing electronic control and memory remains a substantial engineering challenge and an active research frontier.

B. Three-Dimensional and Monolithic Integration

Advanced packaging and monolithic three-dimensional integration techniques, which stack memory and logic layers with dense vertical interconnects, offer a path toward substantially reducing the residual data-movement energy that persists even in memory-on-chip designs such as NorthPole, and are likely to be an important lever for the next generation of neuromorphic accelerators seeking brain-comparable synaptic density within a practical silicon footprint.

C. In-Sensor and Near-Sensor Computing

Pushing computation even closer to the physical transducer, for example by embedding simple spiking computation directly within the pixel array of an event camera or the transduction stage of a microphone, can further reduce the energy and latency cost of the sensing-to-inference pipeline by eliminating off-sensor data transfer for the majority of trivial or redundant frames, and early in-sensor computing prototypes have demonstrated meaningful power reductions for tasks such as always-on visual wake-word detection.

D. Hybrid Von Neumann–Neuromorphic Systems

Rather than viewing neuromorphic and conventional accelerators as competing alternatives, an increasingly favored systems-level direction treats them as complementary co-processors within a heterogeneous system-on-chip, in which a neuromorphic front end performs continuous, low-power event filtering and coarse triggering, waking a more capable but power-hungry conventional accelerator only when a task-relevant event is detected, combining the strengths of both paradigms within a single power and latency budget.

E. Standardization and Ecosystem Maturation

Continued development of hardware-agnostic intermediate representations such as NIR, standardized benchmarking methodology through frameworks such as NeuroBench, and open, cross-vendor programming interfaces are likely to be as important to the field's trajectory as any single hardware innovation, since they directly determine whether algorithmic advances developed on one platform can be readily transferred to another, a prerequisite for the kind of broad ecosystem effects that have driven the rapid maturation of conventional deep learning hardware and software over the past decade.

Conclusion

Neuromorphic computing represents a fundamentally different answer to the energy-efficiency problem confronting modern artificial intelligence, one grounded in architectural principles, sparse asynchronous spiking, co-located memory and compute, and event-driven sensing, that are directly borrowed from biological nervous systems rather than incrementally optimized from the conventional von Neumann accelerator design space. The evidence surveyed in this paper, spanning digital research chips such as Intel Loihi 2 and IBM TrueNorth and NorthPole, large-scale hybrid systems such as SpiNNaker2, and commercial edge processors such as BrainChip Akida, together with emerging memristive and photonic substrates, indicates that these architectural principles translate into consistent, often substantial, real-world energy and latency advantages for sparse, temporally structured, low-batch-size inference workloads, with reported efficiency gains ranging from roughly one to several orders of magnitude depending on task and platform. At the same time, meaningful obstacles remain, including device-level non-idealities, a persistent training-time accuracy gap relative to conventional deep networks, an immature and fragmented software ecosystem, and the absence of fully standardized, system-level benchmarking practice. As algorithmic techniques for training spiking networks continue to mature, as hardware-agnostic software toolchains reduce cross-platform friction, and as emerging device technologies push synaptic density and energy efficiency closer to biological benchmarks, neuromorphic computing is well positioned to become an increasingly indispensable component of the AI hardware landscape, not as a wholesale replacement for GPU-centric deep learning, but as a complementary, ultra-low-power substrate purpose-built for the always-on, latency-critical, resource-constrained inference workloads that conventional accelerators are least equipped to serve efficiently.

References

[1] C. Mead, Analog VLSI and Neural Systems. Reading, MA, USA: Addison-Wesley, 1989.
[2] F. Akopyan et al., "TrueNorth: Design and tool flow of a 65 mW 1 million neuron programmable neurosynaptic chip," IEEE Trans. Comput.-Aided Design Integr. Circuits Syst., vol. 34, no. 10, pp. 1537–1557, 2015.
[3] M. Davies et al., "Loihi: A neuromorphic manycore processor with on-chip learning," IEEE Micro, vol. 38, no. 1, pp. 82–99, 2018.
[4] G. Orchard, E. P. Frady, D. B. D. Rubin, S. Sanborn, S. B. Shrestha, F. T. Sommer, and M. Davies, "Efficient neuromorphic signal processing with Loihi 2," in Proc. IEEE Workshop on Signal Processing Systems (SiPS), 2021, pp. 254–259.

[5] D. S. Modha et al., "Neural inference at the frontier of energy, space, and time," Science, vol. 382, no. 6668, pp. 329–335, 2023.
[6] C. Mayr, S. Hoeppner, and S. Furber, "SpiNNaker 2: A 10 million core processor system for brain simulation and machine learning," in Communicating Process Architectures 2017 & 2018, IOS Press, 2019, pp. 277–280.
[7] H. A. Gonzalez et al., "SpiNNaker2: A large-scale neuromorphic system for event-based and asynchronous machine learning," arXiv preprint, 2024.
[8] BrainChip Holdings Ltd., "Akida neural processor SoC technical reference manual," 2024.
[9] E. M. Izhikevich, "Simple model of spiking neurons," IEEE Trans. Neural Netw., vol. 14, no. 6, pp. 1569–1572, 2003.
[10] A. L. Hodgkin and A. F. Huxley, "A quantitative description of membrane current and its application to conduction and excitation in nerve," J. Physiol., vol. 117, no. 4, pp. 500–544, 1952.
[11] W. Gerstner and W. M. Kistler, Spiking Neuron Models: Single Neurons, Populations, Plasticity. Cambridge, U.K.: Cambridge Univ. Press, 2002.
[12] G. Bi and M. Poo, "Synaptic modifications in cultured hippocampal neurons: Dependence on spike timing, synaptic strength, and postsynaptic cell type," J. Neurosci., vol. 18, no. 24, pp. 10464–10472, 1998.
[13] B. Rueckauer, I.-A. Lungu, Y. Hu, M. Pfeiffer, and S.-C. Liu, "Conversion of continuous-valued deep networks to efficient event-driven networks for image classification," Front. Neurosci., vol. 11, art. 682, 2017.
[14] E. O. Neftci, H. Mostafa, and F. Zenke, "Surrogate gradient learning in spiking neural networks," IEEE Signal Process. Mag., vol. 36, no. 6, pp. 51–63, 2019.
[15] G. Bellec et al., "A solution to the learning dilemma for recurrent networks of spiking neurons," Nat. Commun., vol. 11, art. 3625, 2020.
[16] J. K. Eshraghian et al., "Training spiking neural networks using lessons from deep learning," Proc. IEEE, vol. 111, no. 9, pp. 1016–1054, 2023.
[17] P. Lichtsteiner, C. Posch, and T. Delbruck, "A 128×128 120 dB 15 μs latency asynchronous temporal contrast vision sensor," IEEE J. Solid-State Circuits, vol. 43, no. 2, pp. 566–576, 2008.
[18] F. Paredes-Vallés et al., "Fully neuromorphic vision and control for autonomous drone flight," Sci. Robot., vol. 9, no. 90, art. eadi0591, 2024.
[19] J. Yik et al., "The NeuroBench framework for benchmarking neuromorphic computing algorithms and systems," Nat. Commun., vol. 16, art. 1545, 2025.
[20] D. R. Muir and S. Sheik, "The road to commercial success for neuromorphic technologies," Nat. Commun., vol. 16, art. 3586, 2025.
[21] A. Isik et al., "Neuromorphic principles for efficient large language models on Intel Loihi 2," arXiv preprint arXiv:2503.18002, 2025.
[22] IBM Research, "IBM's NorthPole achieves new speed and efficiency milestones," IBM Research Blog, 2024.
[23] Open Neuromorphic, "Neuromorphic hardware guide," open-neuromorphic.org, 2025.
[24] Y. Khacef et al., "Sensor fusion accelerated with Loihi-2: A case study," arXiv preprint arXiv:2408.16096, 2024.