Transforming Systems with Neuromorphic Computing

Dr. Art Villanueva, ESEP
www.phronos.com
December 2025

PHRONOS
PRACTICAL AI

Art Villanueva, DEng, ESEP

PHRONOS
PRACTICAL AI

Dr. Art Villanueva operates at the intersection of artificial intelligence, emergence, and systems engineering. His research investigates complex adaptive systems (CAS), spanning topics from neural mechanisms in the human brain to autonomous unmanned aerial vehicles (UAVs). As the founder of Phronos, an AI services company, he is dedicated to applying AI and systems engineering principles across a wide range of domains. He also serves as a Senior Staff Engineer at General Atomics Aeronautical Systems, where he contributes to the advancement of next-generation aerospace technologies.

Previously, Dr. Villanueva served as Chief AI Technologist for Dell Technologies' Federal Strategic Programs, where he directed the company's AI strategy for government initiatives. His multidisciplinary career spans large-scale defense and transportation projects, as well as entrepreneurial ventures in clean technology and renewable energy.

An accomplished inventor and scholar, Dr. Villanueva holds multiple U.S. utility patents and has published several peer-reviewed papers. He earned a Doctor of Engineering in Systems Engineering from Colorado State University, focusing on meta-algorithmics for natural language processing. His academic background also includes dual master's degrees in Systems Engineering and Computer Science from UC San Diego, and a B.S. in Applied Mathematics from UCLA. Additionally, he is recognized as an Expert Systems Engineering Professional (ESEP) by INCOSE.

GENERAL ATOMICS AERONAUTICAL DELL Technologies CUBIC NG UCLA BRUINS UCSD Colorado State University

Disclaimer

The views and opinions expressed in this presentation are my own and do not represent those of General Atomics Aeronautical Systems, Inc. or any affiliated entities. This presentation is for informational purposes only and is unrelated to my professional role or work at General Atomics.

Agenda

I. Introduction: Why Neuromorphic Computing
II. A Primer on Neuroscience
III. Spiking Neural Networks
IV. Systems and Spiking Neural Networks
V. Expanding the Applications of Neuromorphic Computing
VI. Challenges and the Road Ahead
VII. Conclusion
VIII. Question-Answer-Discussion

PHRONOS
PRACTICAL AI

Introduction:
Why Neuromorphic Computing

PHRONOS
PRACTICAL AI

Limited Adaptability:

AI systems require retraining and cannot adapt dynamically to new environments.

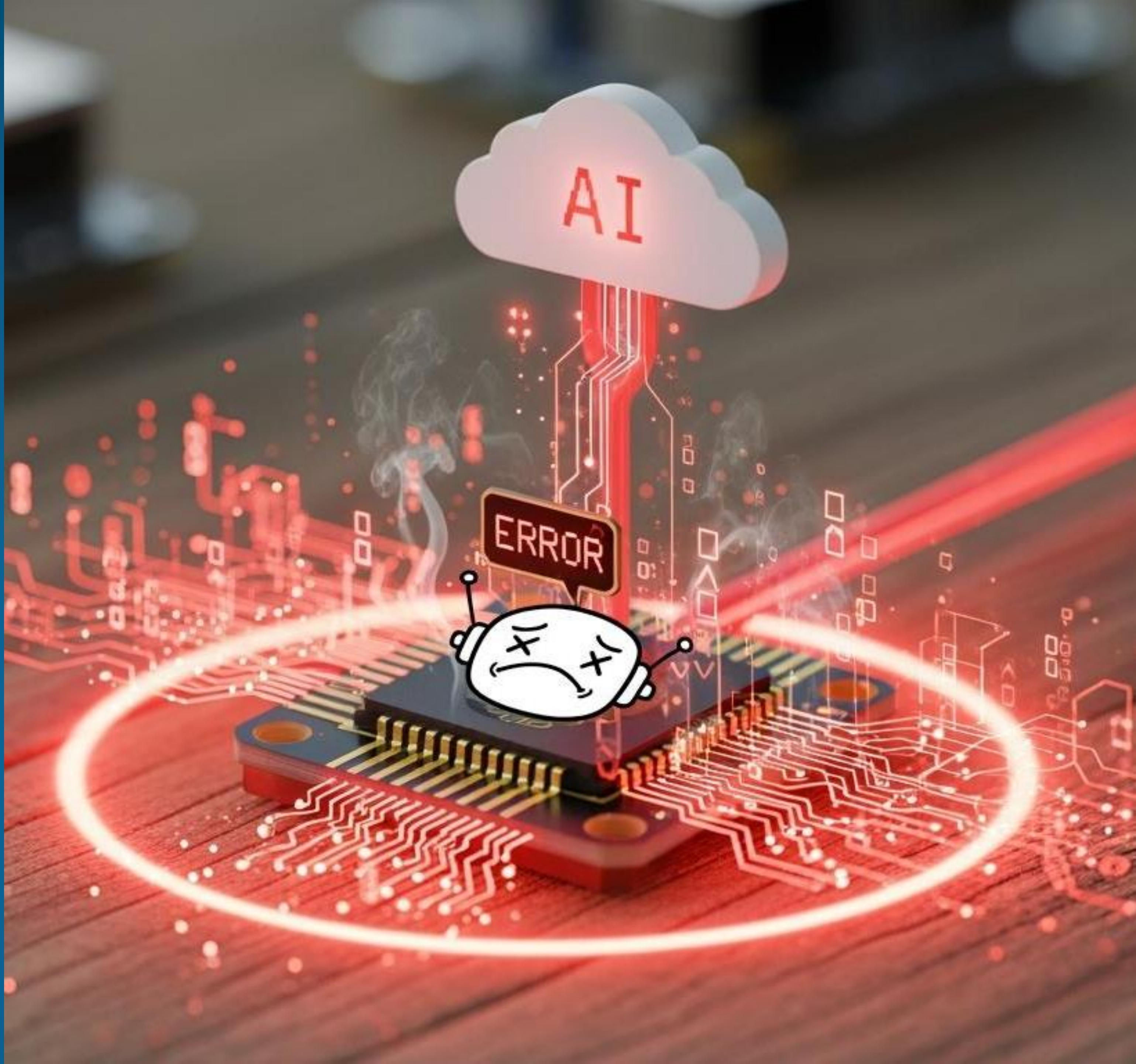
Slow Responses in Real-Time Tasks:

High latency affects applications like robotics and autonomous systems.

PHRONOS
PRACTICAL AI

Inefficiency in Edge Devices:

High power and computational demands make AI impractical for low-resource devices.



The Brain vs. GPU: A Lesson in Energy Efficiency

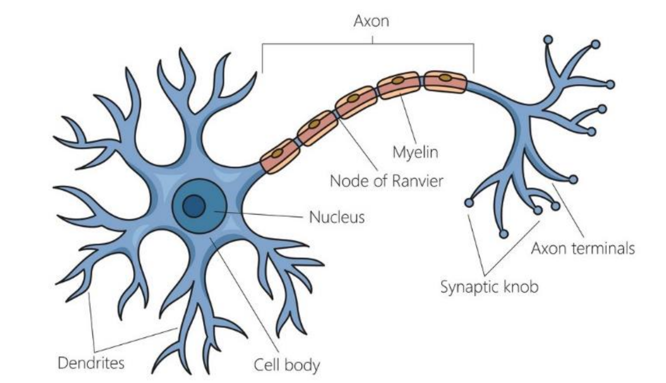
Sample Task: Learning and recognizing a face

Task	Human Brain	GPU (Training)	GPU (Inference)	Smartphone AI Accelerator
Power Consumption	~20W	300-700W per GPU (scaling to kW or MW in clusters)	100-300W per GPU	~500mW to 2W
Time Taken	Milliseconds (adaptive)	Hours/days for training large models	Milliseconds (fast inference)	Milliseconds
Efficiency	Extremely high	Very low (energy-intensive)	Moderate (optimized for inference)	High (specialized hardware)

PHRONOS
PRACTICAL AI

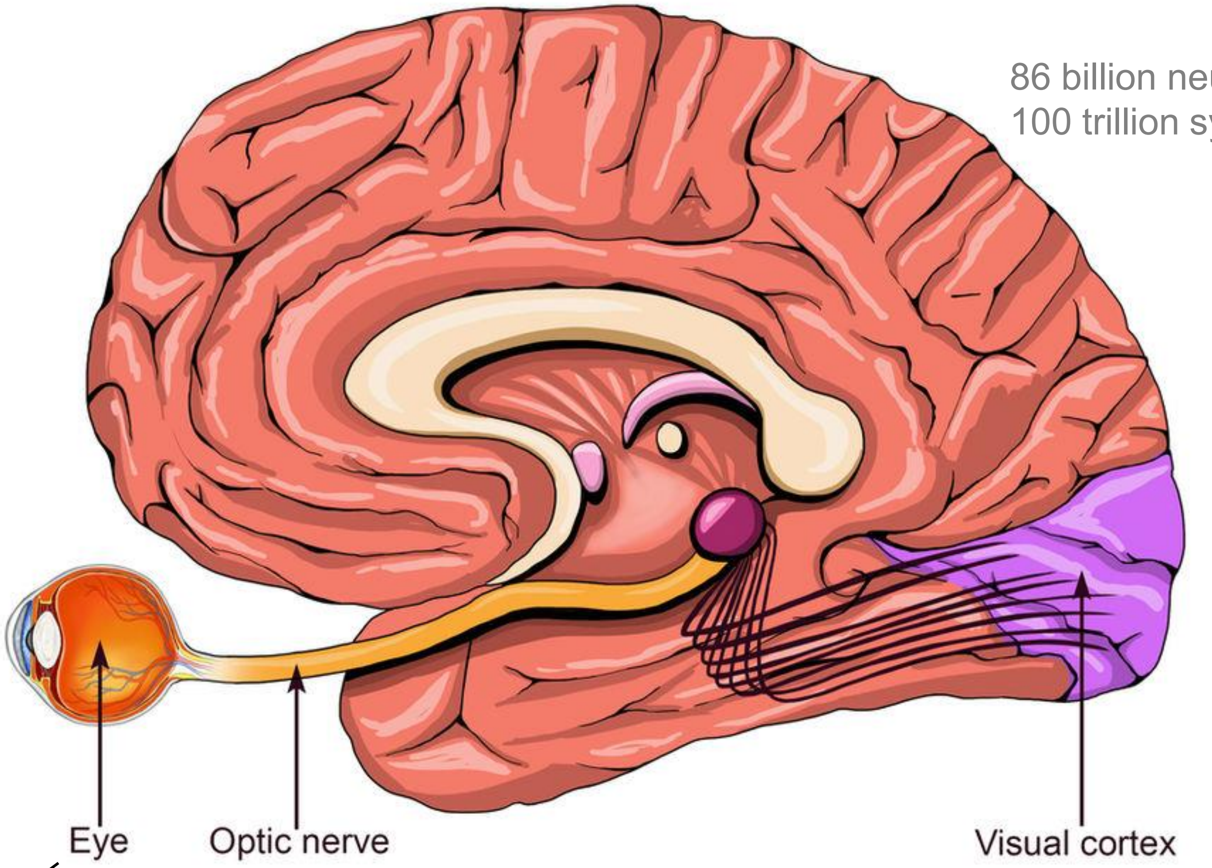
A Primer on Neuroscience

Neuronal Activity (Vision Example)



NEURON

86 billion neurons
100 trillion synapses

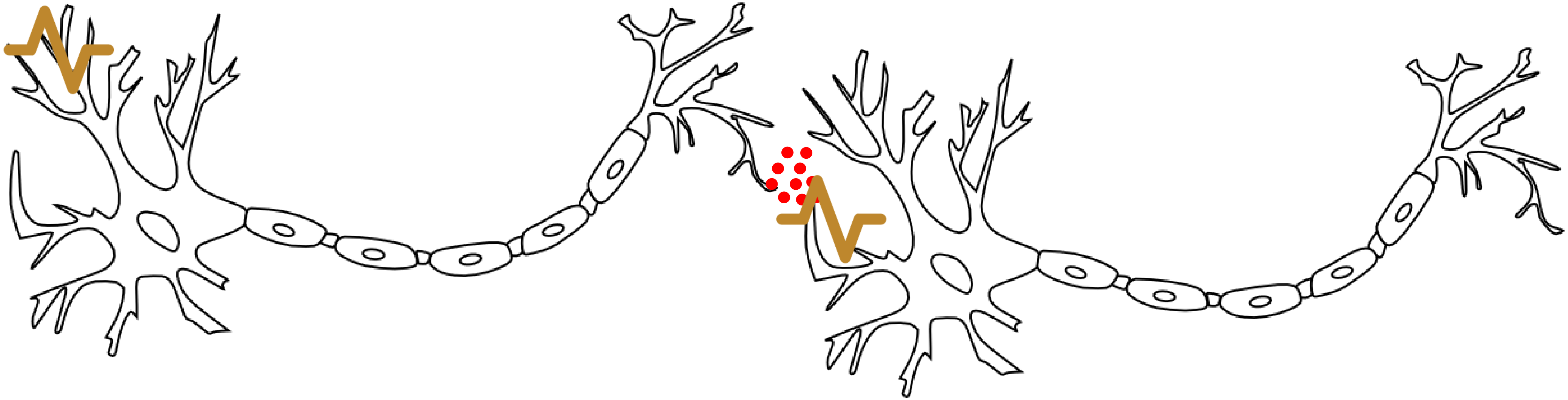


Photoreceptors (rods and cones) and other neurons in parallel and series

Bundled set of neurons in parallel

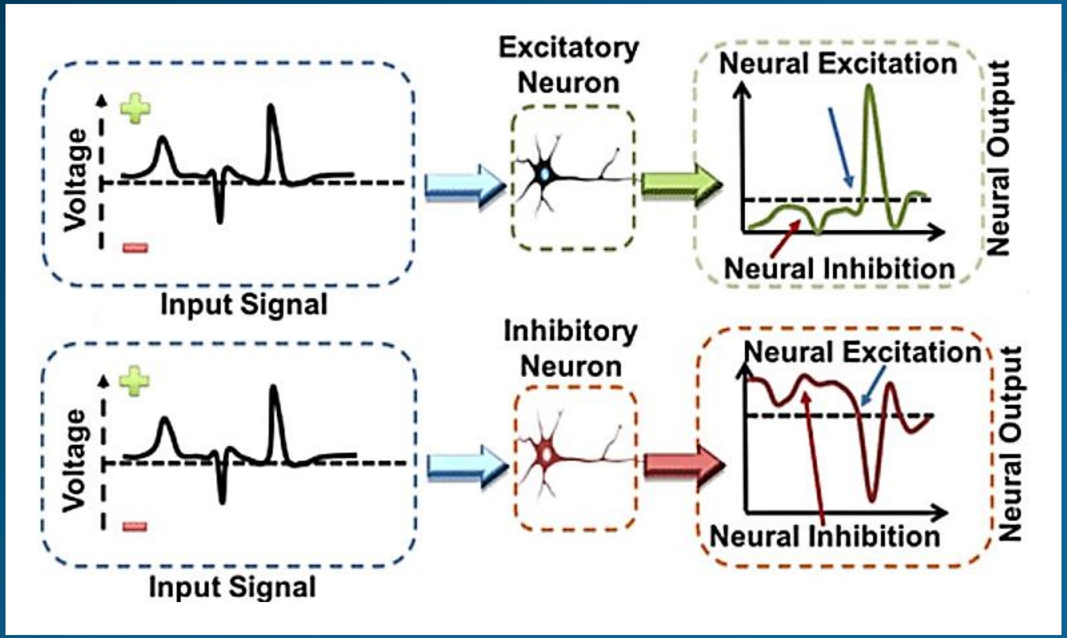
Complex infrastructure of neurons in parallel and series

How Neurons Work (Biological Vision Example)



1. **Signal Starts in a Neuron**:
   - Neuron receives a signal (electrical pulse) from another neuron or a stimulus (e.g., light hitting the eye) through its dendrites.
   - **If the signal is strong enough**, the neuron "activates" and generates an electrical pulse called a **spike**.
2. **Signal Travels Down the Axon and Reaches the Synapse (Gap)**:
   - Carries coded information (via shape, timing, frequency, etc.)
   - At the end, the signal needs to jump to the next neuron.
3. **Chemical Messengers Take Over**:
   - Chemical messengers called neurotransmitters are triggered.
   - These neurotransmitters travel across the synapse and attach to the next neuron.
4. **Next Neuron is Activated**:
   - When neurotransmitters bind to the next neuron, they pass on the message.
   - **If the signal is strong enough**, the process starts over.

Excitatory and Inhibitory Neurons



- Excitatory Neurotransmitters
  - **Glutamate**: Primary excitatory neurotransmitter, for learning and memory
  - **Dopamine**: Reward, motivation, and motor control
  - **Acetylcholine**: Stimulates muscle contraction and supports attention and memory in the brain
- Inhibitory Neurotransmitters
  - **GABA** (Gamma-Aminobutyric Acid): promotes relaxation and reduces anxiety
  - **Glycine**: Helps with motor and sensory control.
  - **Serotonin**: Regulates mood, appetite, and sleep

PHRONOS
PRACTICAL AI

Connectome

- A map of neural connections (neurons + synapses)
- Explains brain function, behavior, and learning
- Governs how the network processes input, stores memory, and learns
- Dynamic:
  - Changes with experience and aging
  - New connections can form, or old ones can weaken or disappear, enabling learning and memory

Aspect	Genome	Connectome
Nature	Static (DNA sequence)	Dynamic (neuroplasticity)
Components	Genes and nucleotides	Neurons and synapses
Mapping	Fully mapped (linear)	3D, partially mapped
Purpose	Blueprint for life	Brain's wiring for function

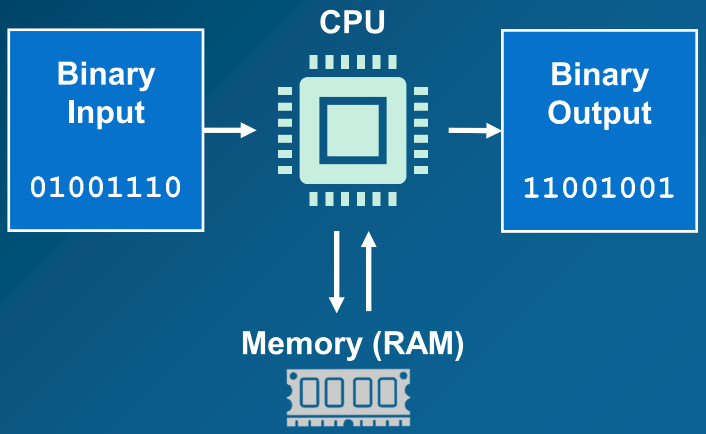
PHRONOS
PRACTICAL AI

Spiking Neural Networks

PHRONOS
PRACTICAL AI

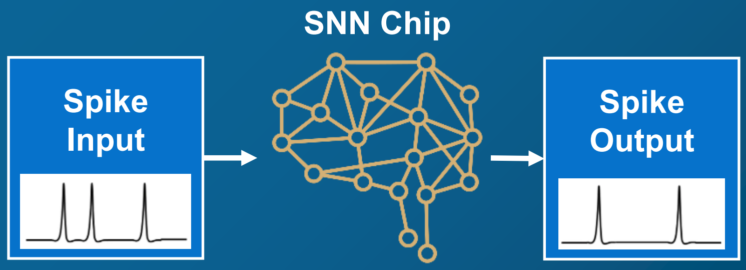
Traditional Computing vs. Neuromorphic Computing

Von Neumann Architecture (Traditional Computing)



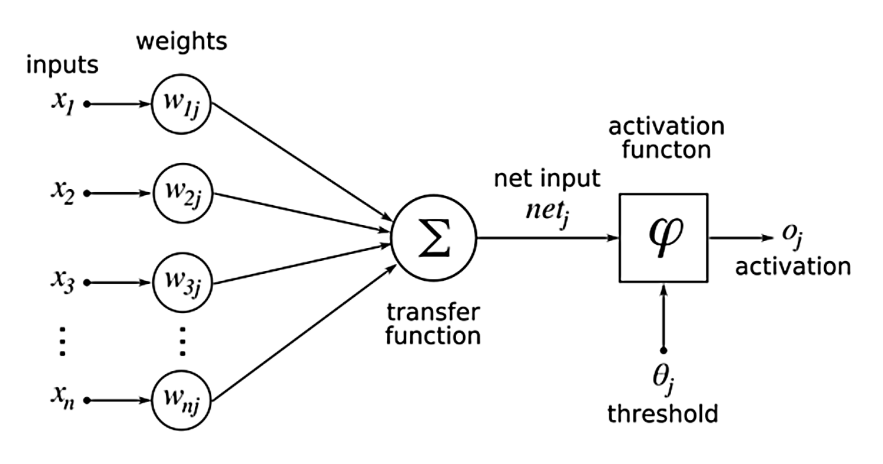
- Separate CPU and memory with sequential, clock-driven processing

Neuromorphic Architecture



- Mimics the brain with neurons (computation) and synapses (memory) co-located.
- Event-driven, asynchronous processing using spikes.

Artificial Neural Networks (ANNs)



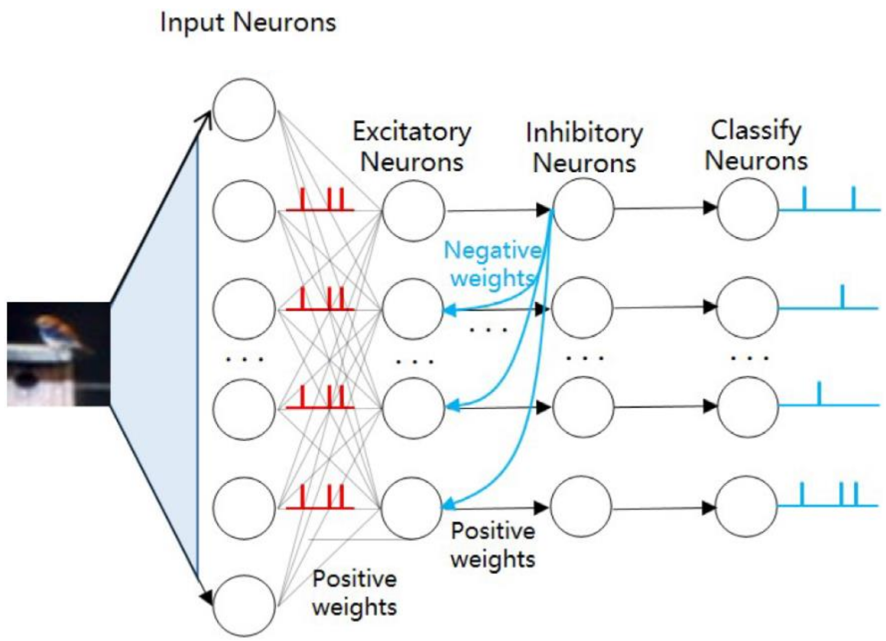
- Inputs $(x_1, x_2, x_3, \ldots, x_n)$:
  - Features of the input data.
  - Each input is associated with a weight (w_{ij}) that determines importance.
- Weights (w_{ij}):
  - Scalar values that scale the contribution of each input to the neuron.
  - Learned during training to optimize the network's performance.
- Transfer Function (Σ):
  - Computes the weighted sum of inputs (net_j) by summing $w_{ij} \cdot x_i$ for all inputs.
- Activation Function (φ):
  - Applied a non-linear transformation to the net input (net_j).
  - Common examples include ReLU, sigmoid, or tanh functions.
  - Introduces non-linearity, enabling the network to model complex patterns.
- Threshold (θ_j)
  - Bias term or threshold that shifts the activation function.
  - Helps the neuron fire only when the input signal exceeds a specific value.
- Output (o_j):
  - The neuron's output after applying the activation function.
  - Serves as input to the next layer in the network.

PHRONOS
PRACTICAL AI

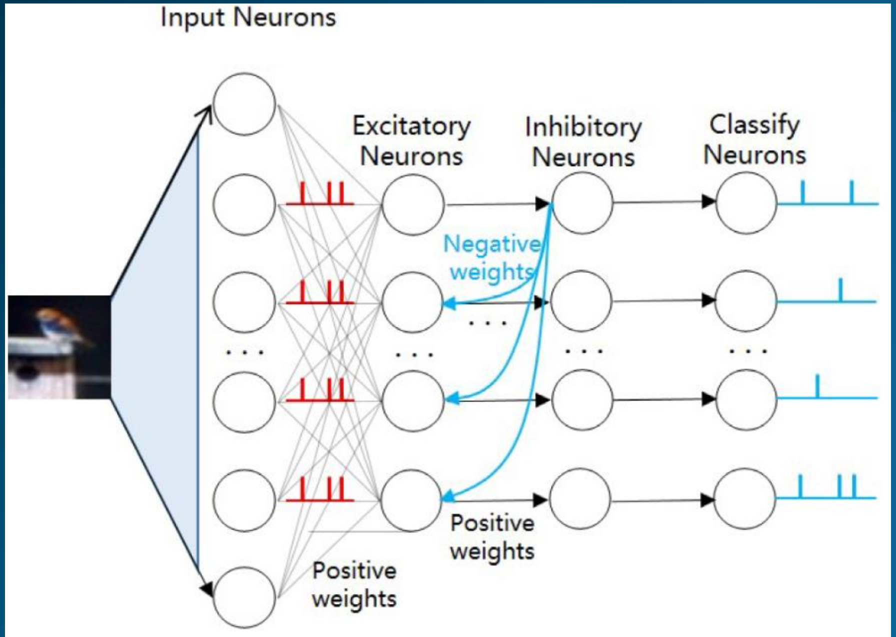
https://aoguedao.github.io/neural-computing-book/chapters/scientific_machine_learning/neural_networks_part1.html

Spiking Neural Networks (SNNs)

- SNN neurons mimic biological neurons by:
  - Receiving input as spikes
  - Accumulating input until a threshold is reached (leaky integrate-and-fire (LIF) model)
  - Firing spikes to communicate with other neurons
- Event-driven:
  - SNN neurons enable adaptive, event-driven systems



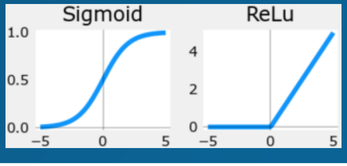
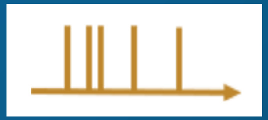
How SNNs Work

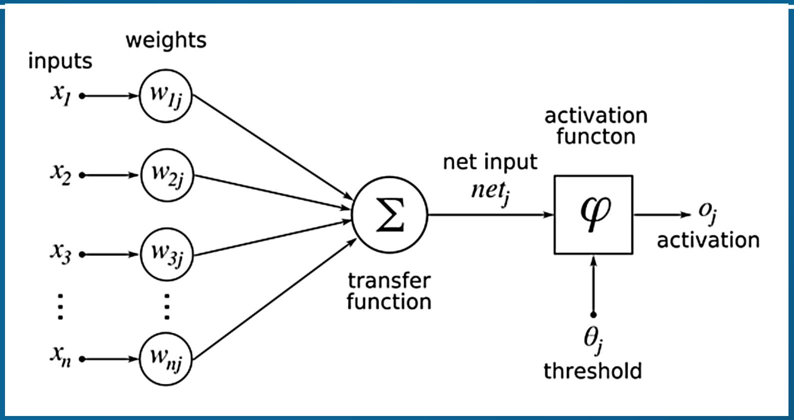


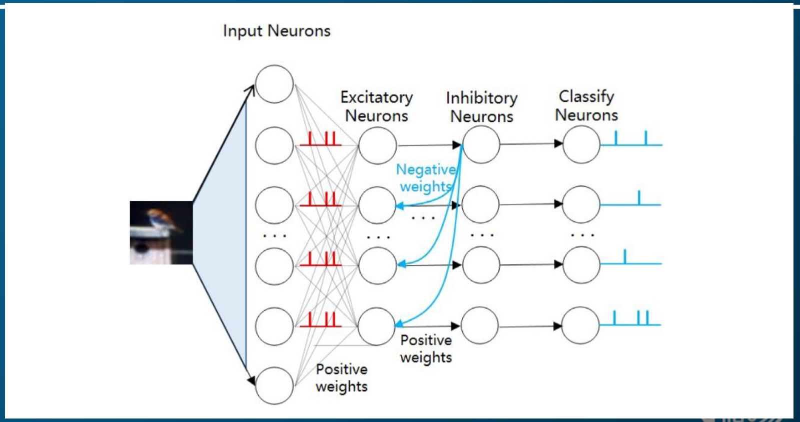
- Input Neurons:
  - Receive input as spike trains (discrete, time-dependent spikes) rather than continuous values.
  - Spike trains encode information over time.
- Excitatory and Inhibitory Neurons:
  - Excitatory neurons amplify signals using positive weights.
  - Inhibitory neurons suppress signals using negative weights, mimicking biological synaptic behavior.
- Spike-Based Communication:
  - Information is processed through spike trains, with neurons firing based on incoming spikes and their membrane potential.
- Temporal Dynamics:
  - Event-driven information processing based on the timing and frequency of spikes.
  - Outputs are also represented as spike trains.

https://aoguedao.github.io/neural-computing-book/chapters/scientific_machine_learning/neural_networks_part1.html

Artificial Neural Networks vs. Spiking Neural Networks

	Artificial Neural Networks (ANNs) (e.g., Deep Neural Networks (DNNs))	**Spiking Neural Networks (SNNs)**
Inspiration	Abstraction of biological neurons	Mimics biological neurons
Computation	Continuous, requires static inputs	Event-driven
Data Processing	Real-valued (continuous) activations (e.g., rectified linear unit (ReLU), sigmoid) 	Dynamic, processes time-sensitive inputs 
Hardware	Neural Processing Units (NPUs) like Intel Nervana, Graphical Processing Units (GPUs) like Nvidia A100	Specialized neuromorphic chips like Intel Loihi
Limitations	Energy-intensive and less effective for time-varying data	Harder to train due to spike-based dynamics



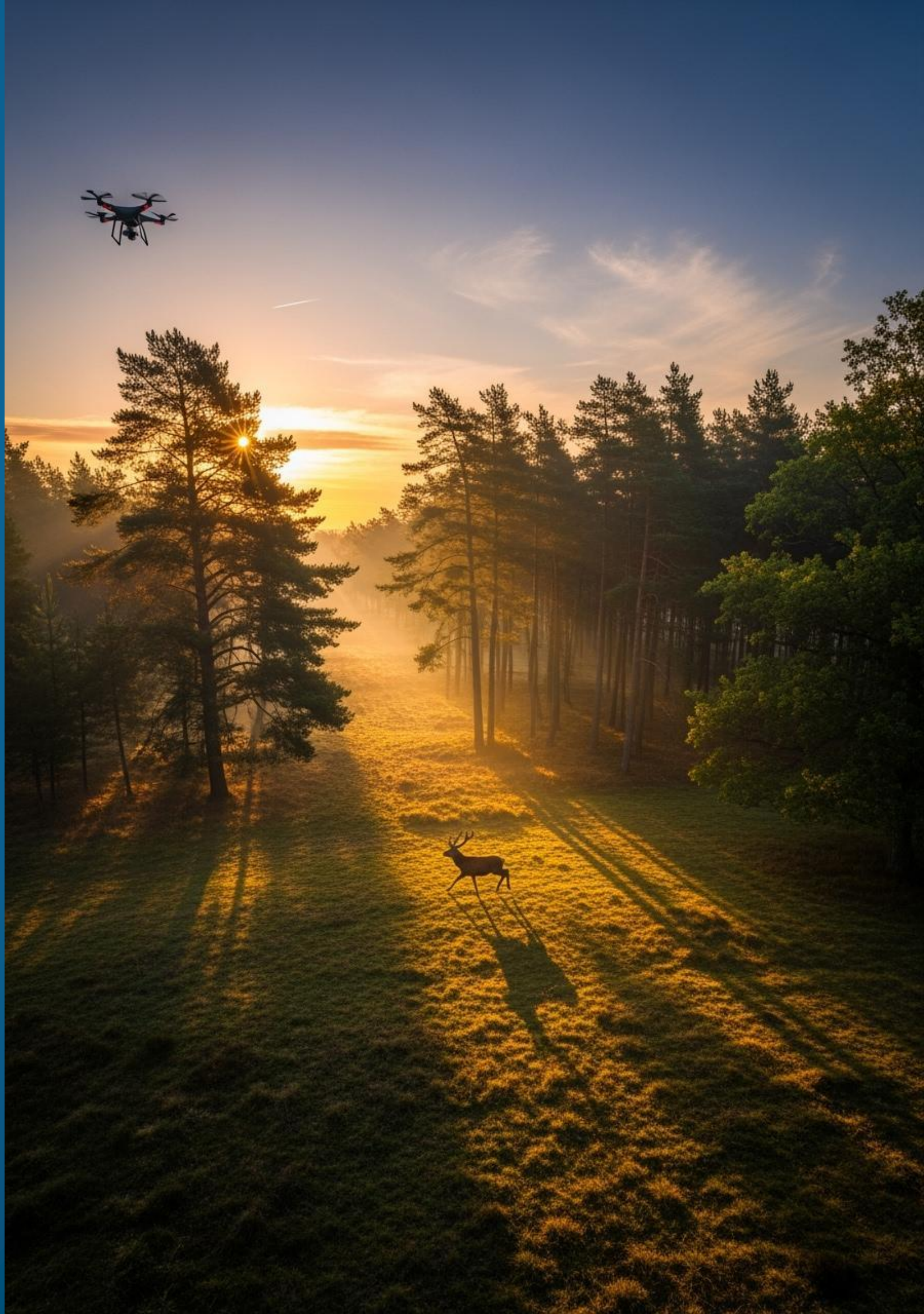


PHRONOS
PRACTICAL AI

SNN Input Encoding

- Converts raw data (e.g., pixels in an image) into spike trains:
  - Bright regions or edges → Frequent spikes.
  - Dark or uniform regions → Few or no spikes.

- Example Scenario:
  - UAV is flying over a forest at sunrise
  - Shadows are moving as the sun rises
  - Trees are swaying in the wind
  - A deer suddenly runs across the scene

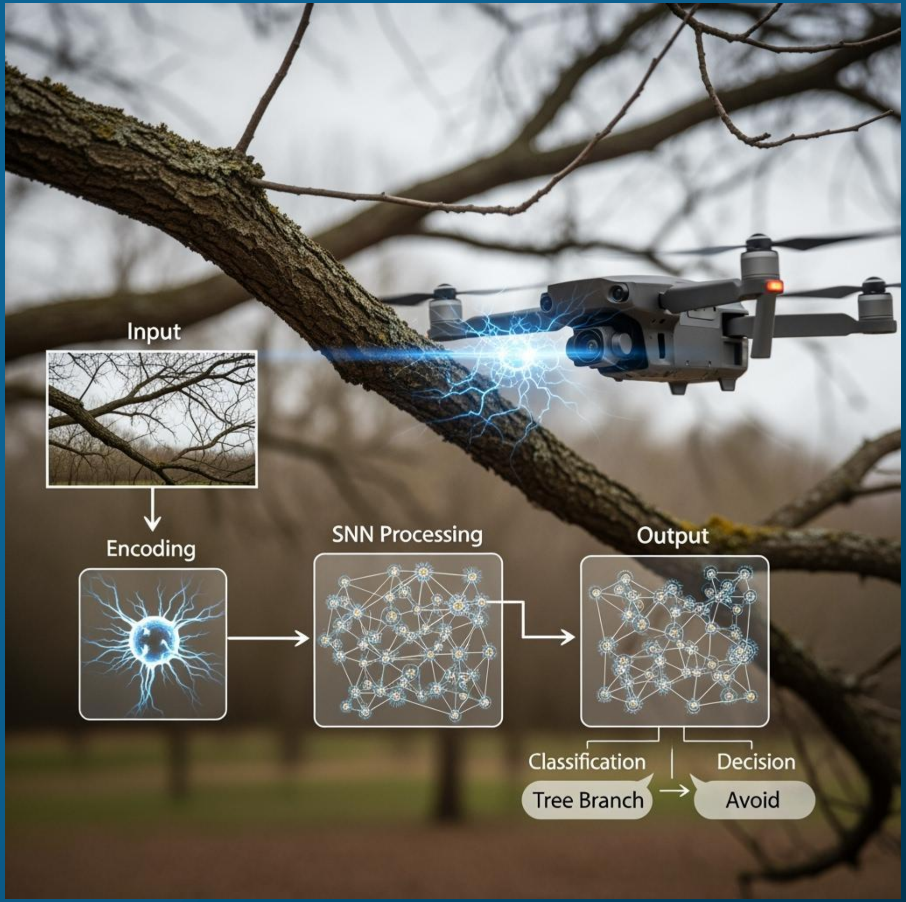


SNN Spike Transmission and Processing

- Neuron Behavior:
  - Neurons collect input spikes (e.g., edges or motion) and fire only when a threshold is crossed.
  - Timing matters: Earlier spikes (more urgent or important information) carry more weight.
- Example Scenario:
  - UAV flying through a dynamic environment, such as a dense forest or an obstacle-filled area.
  - Obstacle Detection: The UAV camera detects edges (e.g., tree trunk).
  - Neurons accumulate spikes and fire.
  - Spike Timing:
    - Far tree (early spike) → gentle path adjustment
    - Fast bird (early spike) → emergency swerve
    - Cloud shadow (late spike) → low-priority, minimal or no reaction

SNN Connectomes and Memory

- Connectome:
  - Network of neurons underpinning all neural processing.
  - Modified during learning (e.g., spike-timing dependent plasticity, or STDP).
- Memory in SNNs
  - Short-Term Memory (STM):
    - Temporarily encoded in spike patterns within the connectome during computation.
    - Transient – exists only while stimuli or active processing occur.
  - Long-Term Memory (LTM):
    - Stored as synaptic weight changes in the connectome, retaining learned behaviors.
    - Persistent within the connectome but requires external saving for power-loss protection.
- Example: UAV Learning
  - A UAV learns to associate tree branches (spike patterns) with avoidance behavior.
  - STM: Temporary spike activity during real-time processing
  - LTM: Synaptic changes encoding the knowledge for future flights

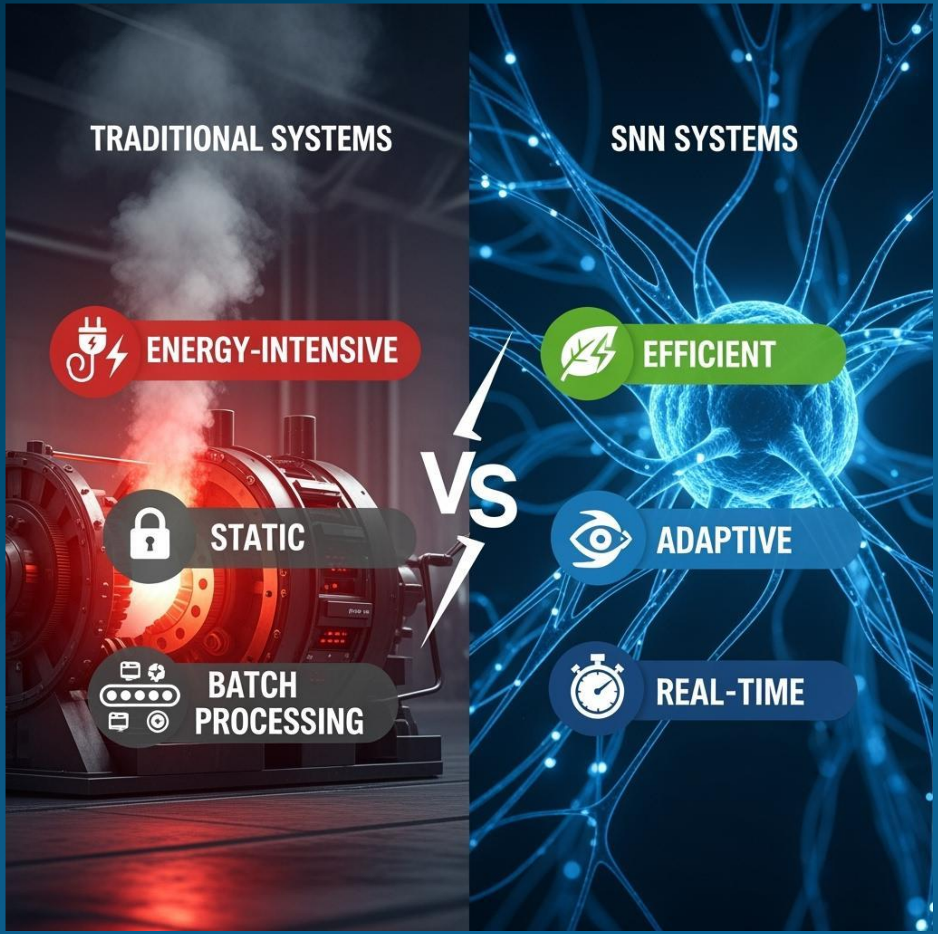


NOTE: The connectome supports STM and LTM, but RAM is still needed for buffering and computation in most setups.

Systems and SNNs

PHRONOS
PRACTICAL AI

Why SNNs Matter for Systems



Key Advantages of SNNs:

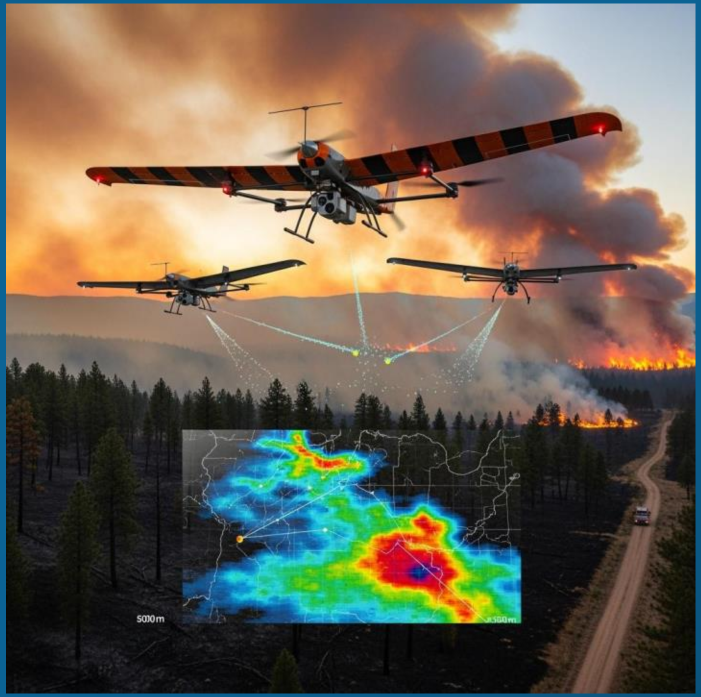
- **Energy Efficiency**: Lower power consumption, ideal for edge devices.
- **Real-Time Responsiveness**: Processes dynamic, time-sensitive inputs instantly
- **Adaptability**: Handles unpredictable environments with event-driven processing
- **Scalability**: Supports modular and distributed systems

PHRONOS
PRACTICAL AI

Applications in Modern Systems

Autonomous Navigation
UAVs avoiding obstacles in real-time (e.g., forests, disaster zones).

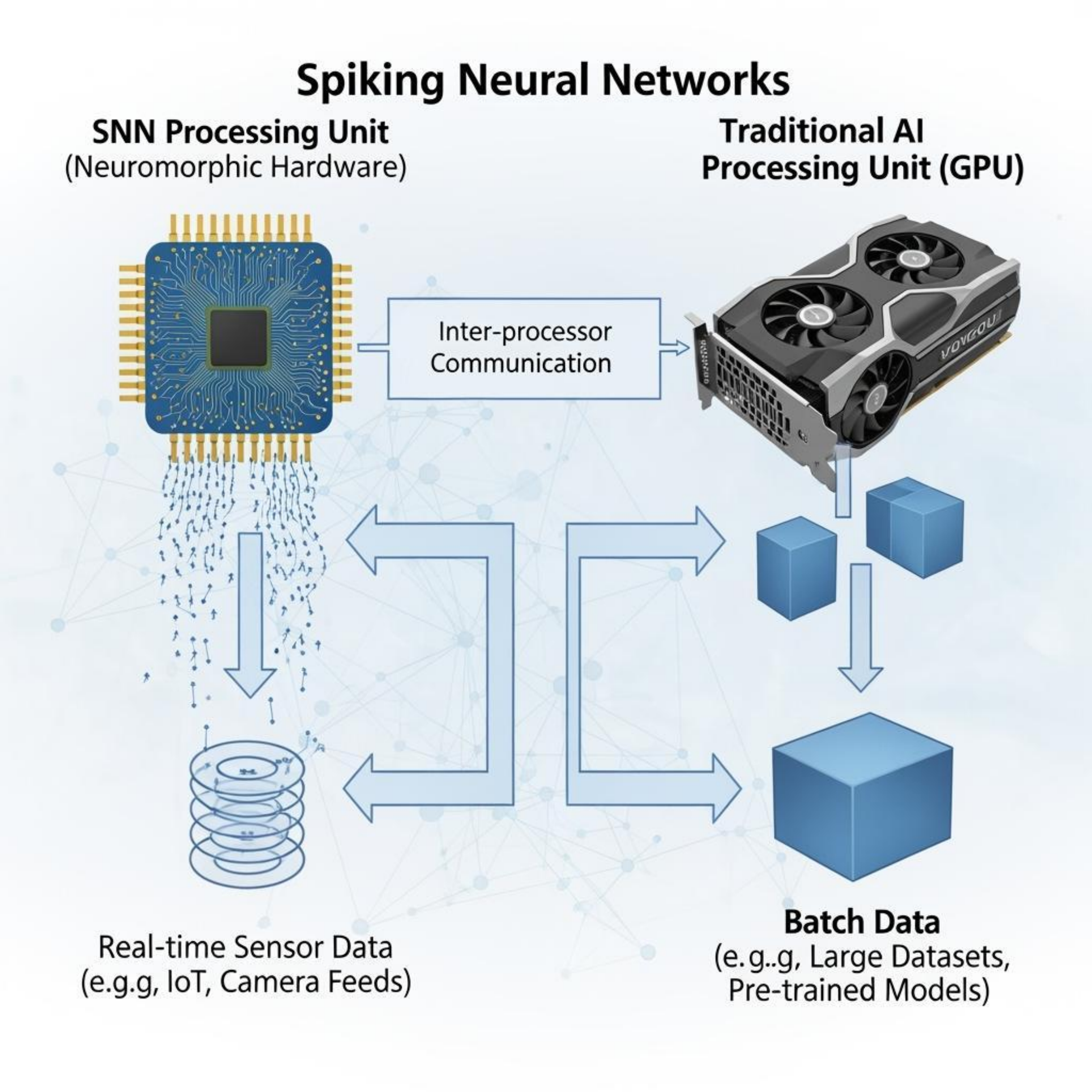
Swarm Intelligence
UAV swarms dynamically coordinate for tasks such as wildfire mapping



IoT & Edge Devices
Real-time processing for smart cities, adaptive robotics, and factories.

PHRONOS
PRACTICAL AI

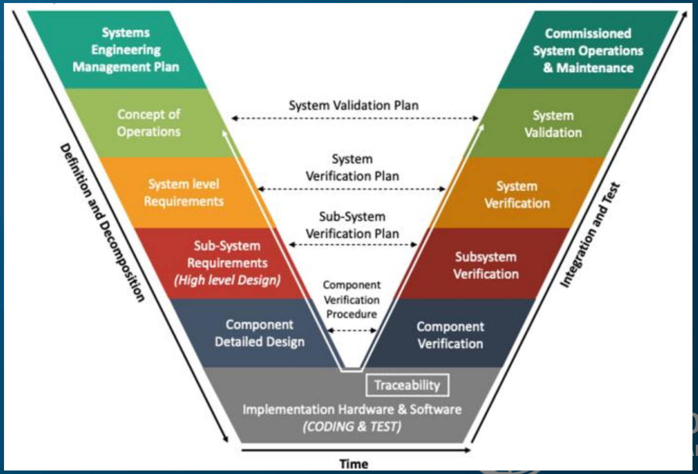
Integrating SNNs into Systems



- Combining SNNs with traditional AI reduces power consumption and improves adaptability.
- Example: Real-time obstacle detection (SNN) + long-term data analysis (DNN).

Integration of neuromorphic computing should start as early in the lifecycle as possible:

Example: UAV navigation algorithms at the Design stage.



Expanding the Applications of Neuromorphic Computing

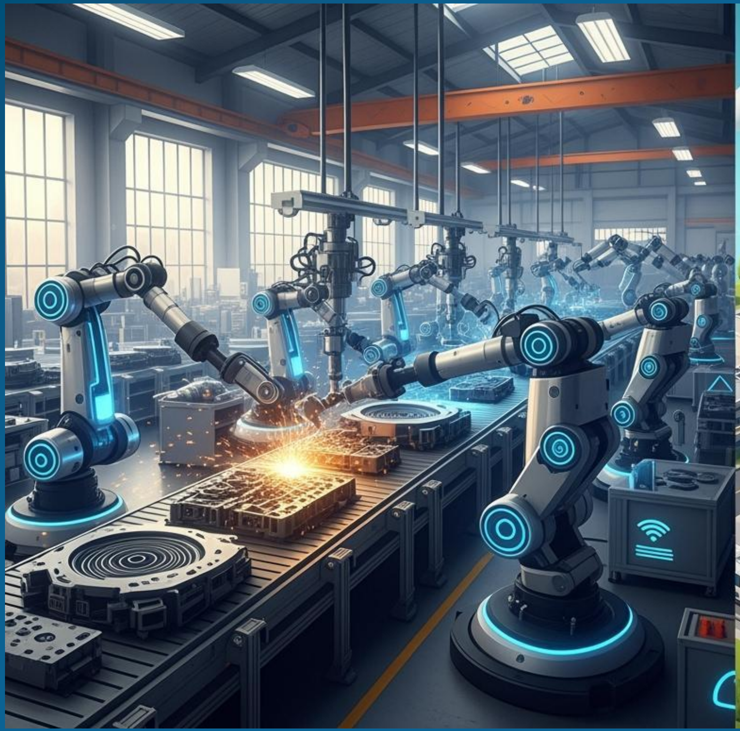
PHRONOS
PRACTICAL AI

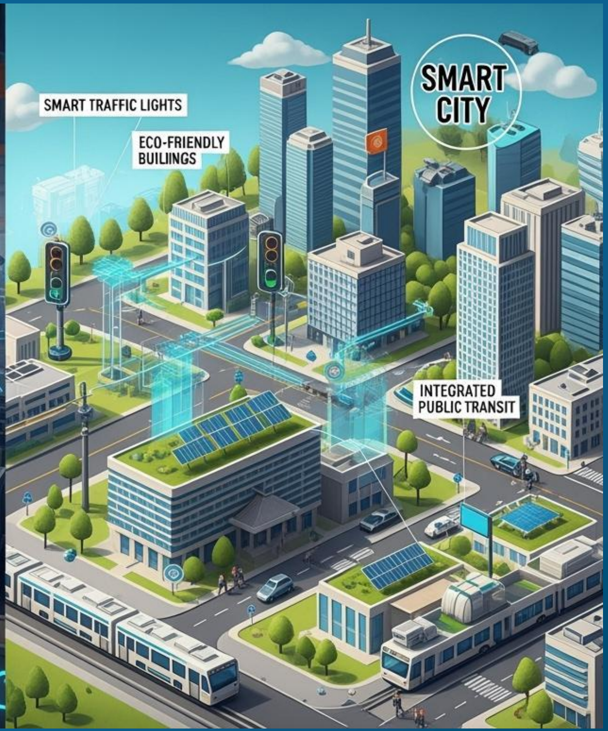
Expanding Applications Across Industries

Automation & Robotics:

Adaptive robotics in factories

Real-time adjustments in industrial IoT systems





Smart Infrastructure:

Traffic signal optimization in smart cities

Energy-efficient public transportation systems

PHRONOS
PRACTICAL AI

Space Exploration & Healthcare

Space Exploration:

Adaptive navigation systems for autonomous rovers (e.g., Mars exploration)

Fault-tolerant spacecraft systems

Healthcare:

Wearable devices for real-time health monitoring (e.g., detecting irregular heartbeats)

Intelligent, energy-efficient prosthetics

PHRONOS
PRACTICAL AI

Neuromorphic-Inspired Search Accelerator

Local Compute (In-Memory Execution)

- Computation occurs directly inside storage arrays
- Minimizes data movement energy and latency
- Enables associative, event-driven lookup

Parallel Query Evaluation

- Entire dataset evaluates a query in one operation
- Hardware-level parallelism vs. sequential CPU loops
- Supports large-scale nearest-neighbor and pattern match tasks

LEWIS RHODES LABS

Performance Advantages

- Sub-millisecond search
- Low power per operation
- Scalable vector/feature lookup throughput

PHRONOS
PRACTICAL AI

Neuromorphic-Inspired Search Accelerator

Feature	Traditional CPU-Based Search	GPU-Accelerated Search	Neuromorphic/ASIC Search (e.g., Lewis Rhodes Labs)
Processing Location	CPU, separate from memory	GPU, separate from memory	Integrated with memory (in-memory computing) [1][2]
Parallelism	Limited	High	Massively parallel [1][3]
Data Movement	High	High	Minimal [1]
Latency	High	Moderate	Ultra-low [1]
Power Consumption	High	High	Low [1][3]
Scalability	Limited	Good	Excellent [3]
Best Use Cases	Small datasets, simple queries	AI model training, moderate-sized datasets	Large datasets, complex pattern matching, real-time search [1][3]
Query Flexibility	High	Moderate	Application-Specific [4]
Search Speed	Slower	Faster	Extremely Fast [3][5]
Indexing Requirement	Often Required	Varies	No Indexing Burden [5]
Data Ingestion	Can be slow	Can be slow	Rapid [3]

[1] SmartSSD LLR
[2] Comparison of Neuromorphic Computing versus Quantum Materials - Patsnap Eureka
[3] Extreme Search® Technology - Lewis Rhodes Labs, Inc.
[4] AI Accelerator Chips Overview and Comparison - HardwareBee
[5] Lewis Rhodes Labs, Inc.: Home

The Road Ahead

PHRONOS
PRACTICAL AI

Why has adoption been slow?

Competition from GPUs and TPUs

- GPUs and TPUs are already highly optimized for AI and machine learning tasks
- Neuromorphic hardware doesn't yet offer enough of an advantage to justify switching.

Limited Applications

- Neuromorphic systems are highly specialized and excel in niche tasks and low-power sensory processing.

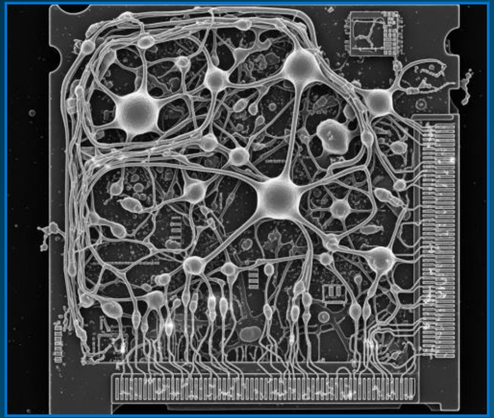
Complexity and Expertise

- Neuromorphic systems requires specialized expertise, and the talent pool is small.
- Mapping real-world problems to neuromorphic architectures is challenging.

PHRONOS
PRACTICAL AI

Key Challenges

Hardware Scalability

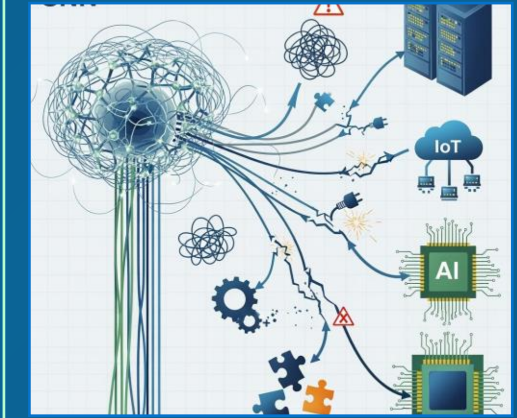


- Current neuromorphic chips are not yet scalable for large, complex systems like UAV fleets or smart cities.
- **Challenge**: Miniaturization and mass production.

Programming Paradigms

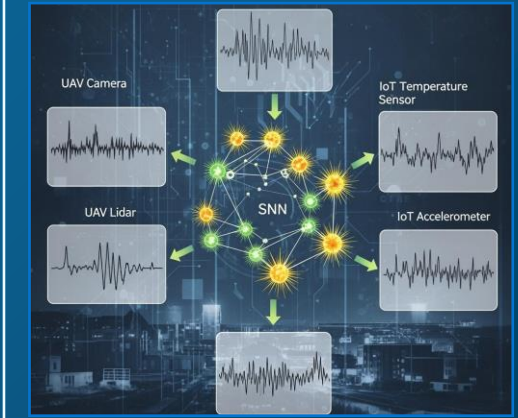
- Lack of standardized tools and frameworks for designing and deploying SNNs in real-world applications.
- **Challenge**: Need for developer-friendly tools and hybrid architectures.

Integration with Legacy Systems



- Bridging SNN-based systems with existing architectures (e.g., traditional AI or IoT infrastructure) is complex.
- **Example**: Combining SNNs with traditional AI for UAV navigation.

Environmental Uncertainty

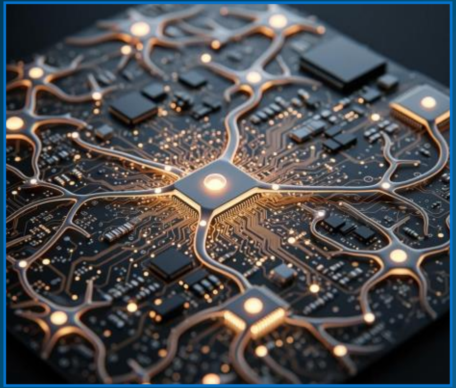


- SNNs must handle noisy, unpredictable inputs (e.g., sensor noise in UAVs or IoT devices).
- **Challenge**: Designing robust, error-tolerant systems.

Is the Tide Changing?

Active Research and Future Trends

Neuromorphic Hardware



- Advancements in neuromorphic chips like Intel Loihi 2 and IBM TrueNorth.
- Focus: Miniaturization, scalability, and energy efficiency for deployment in large systems.

Neuromorphic Algorithms

- Development of new learning paradigms, such as spike-timing dependent plasticity (STDP) and unsupervised learning.
- Focus: Enabling SNNs to "learn" from dynamic, real-time environments.

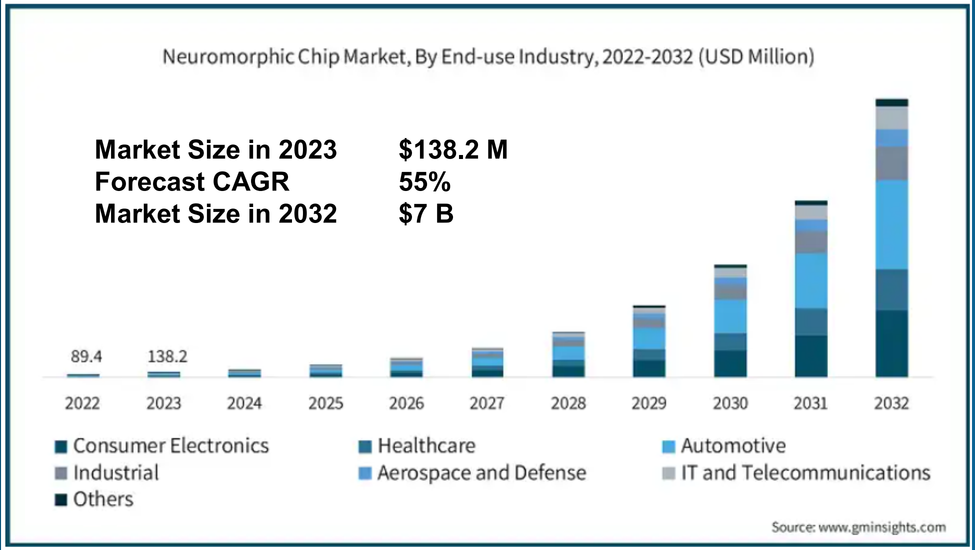
Materials Science

- Exploration of memristors, phase-change materials, and other technologies for more efficient neuromorphic hardware.
- Focus: Mimicking biological synapses more effectively.

Emerging Technologies

- Integration of photonics for faster, light-based neuromorphic computing.
- Exploration of biological computing for next-gen systems inspired by the brain

Neuromorphic Chip Market



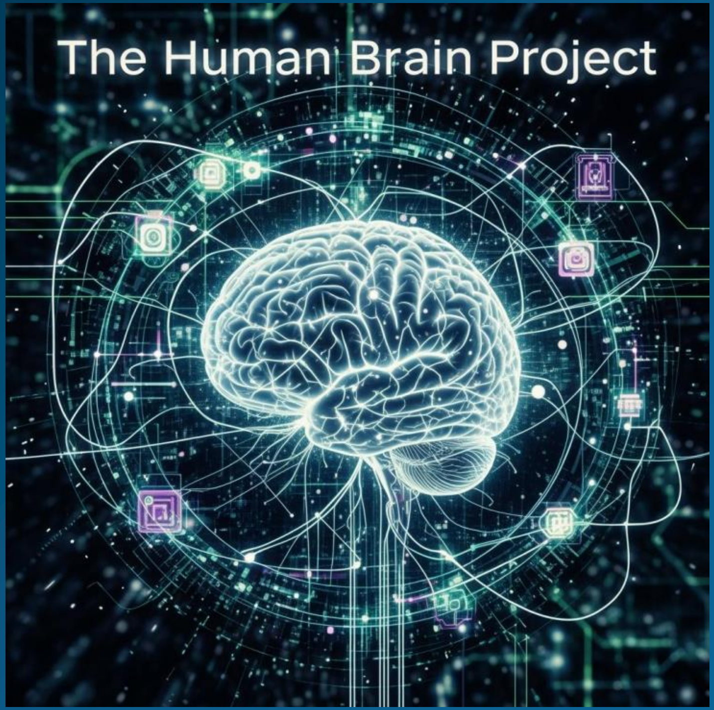
Company	Chip/ Technology	Availability	Approx. Cost	Applications
Intel Corporation	Loihi	Research only	Not disclosed	Robotics, AI research, pattern recognition
IBM Corporation	TrueNorth	Research collaborations	Likely tens of thousands USD	Real-time data processing, neuroscience
BrainChip, Inc.	Akida	Commercially available	$250/PCIe board	Edge AI, IoT, robotics, health monitoring
SynSense AG	Dynap-CNN	Commercially available	>$300/board	Smart sensors, vision-based AI, wearable devices
Prophesee.ai	Event-based Vision Sensors	Commercially available	Not disclosed	Vision-based AI, industrial automation, robotics

https://www.gminsights.com/industry-analysis/neuromorphic-chip-market

PHRONOS
PRACTICAL AI

The Human Brain Project (2013-2023)

To understand the brain and apply insights to medicine, AI, and computing



- EU-funded, €1 billion
- Key Focus Areas
  - Neuroscience: Study brain structure and function.
  - Brain Simulation: Use supercomputers to model the brain.
  - Neuromorphic Computing: Develop brain-inspired hardware (e.g., SpiNNaker).
  - Medical Applications: Advance treatments for neurological diseases.
- Major Achievements
  - BigBrain Atlas: 3D brain map
  - EBRAINS: Digital platform for neuroscience tools.
  - Neuromorphic Systems: SpiNNaker and BrainScaleS
- Legacy
  - Fostered neuroscience-computing collaboration
  - Tools and platforms (EBRAINS) continue driving research.

Other Technologies

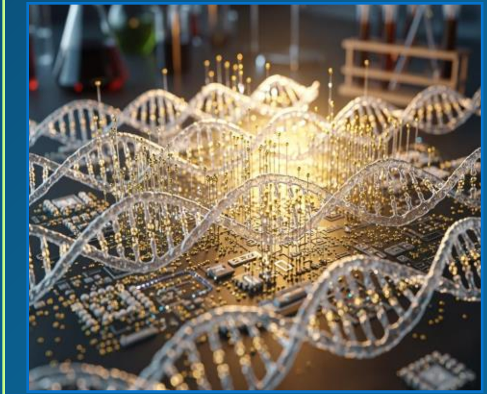
Quantum Computing

Uses qubits for parallel computation

Excels at optimization and simulation

Limited by stability and scalability

DNA Computing (Biocomputing)

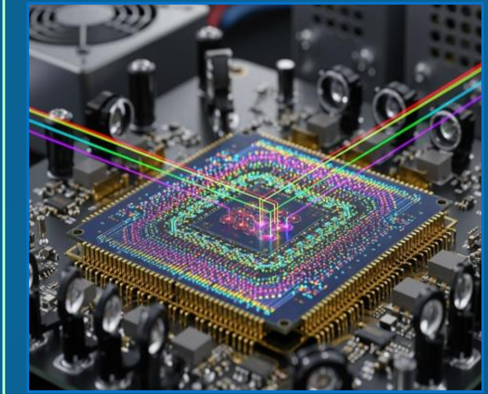


Computes with DNA molecules

Ultra-dense storage and parallelism

Slow and hard to integrate

Optical Computing (Photonic Computing)



Computes with light instead of electrons

Very fast, low-loss data processing

Hardware integration is challenging

Edge Computing with Swarm Intelligence

Distributed processing on edge devices

Swarm algorithms enable coordination

Good for IoT and low-latency tasks

PHRONOS
PRACTICAL AI

Conclusion

PHRONOS
PRACTICAL AI

Transforming Systems with Neuromorphic Computing

- Key Takeaways
  - Inspired by the brain, SNNs offer energy-efficient, real-time, and adaptive solutions
  - **Existing** Real applications include autonomous navigation to IoT, healthcare, smart cities, and search
- Call to Action:
  - Explore and assess how neuromorphic computing can transform your systems

Question-Answer-Discussion

Unlock the Power of AI for Your Organization

- Gain a competitive edge
- Improve efficiency and automate routine tasks
- Enhance decision-making with data-driven recommendations

Expert AI Services

- Strategic AI
- Market research and analysis
- Grant research
- AI-assisted chatbots
- AI-driven proposal writing
- Predictive analytics and forecasting
- AI Education

Art Villanueva, DEng, ESEP
Principal
art@phronos.com

PHRONOS
PRACTICAL AI

Unlock the true potential of AI. While 95% of companies struggle to generate returns from their AI investments, our AI consulting services provide a clear path to success.

We specialize in aligning AI strategies with your business goals, implementing tailored solutions, and optimizing AI performance to deliver measurable results.

Transforming Systems with Neuromorphic Computing

Keywords:
neuromorphic computing, spiking neural networks, SNNs, energy-efficient computing, brain-inspired systems, adaptive systems

PHRONOS
PRACTICAL AI

As the complexity of modern systems grows, traditional computing architectures struggle to meet the demands of real-time adaptability, energy efficiency, and scalability. Neuromorphic computing, inspired by the brain's architecture and processes, offers a novel approach to designing intelligent, adaptive systems. Using spiking neural networks (SNNs) -- a computational model that mimics biological neurons -- neuromorphic systems enable energy-efficient, event-driven processing optimized for dynamic environments.

This presentation explores the contrast between traditional and neuromorphic architectures, the capabilities of SNNs in real-time processing, and practical applications such as autonomous UAV navigation, swarm intelligence, and smart infrastructure. It also addresses the integration of neuromorphic technologies into existing workflows, along with challenges like scalability, interoperability, and performance trade-offs.

By situating neuromorphic computing within the context of modern engineering challenges, this presentation provides valuable insights into the potential of brain-inspired systems. Attendees will learn how to leverage this emerging approach to develop more adaptive, efficient, and intelligent systems suited for the complexities of the future.