BioTensor: A Software Framework for Spiking Encoding in Organoid Intelligence Interfaces

© 2026 Easwarnath Akkala. All Rights Reserved. *Draft Version 1.0 – Do not distribute without permission.*

Abstract

As artificial intelligence approaches the thermodynamic limits of silicon-based computing, Organoid Intelligence (OI) has emerged as a promising alternative, offering massive parallelism at minimal energy cost. However, a critical "translation gap" remains between digital data formats and the biological action potentials required for wetware computation. This paper introduces **BioTensor**, an open-source Python framework designed to bridge this gap. By implementing a Liquid State Machine (LSM) architecture, BioTensor converts static digital inputs into spatiotemporal spike trains, simulating the interface between high-density microelectrode arrays (HD-MEAs) and biological cortical organoids.

We demonstrate that the framework can encode, process, and decode complex visual data (MNIST) with 100% accuracy in binary classification tasks. Furthermore, we present a novel **Bimodal Associative Cortex** capable of fusing visual and auditory signals. Through the implementation of a partitioned reservoir and homeostatic Spike-Timing Dependent Plasticity (STDP), the system achieved **95.0% accuracy** in unsupervised concept association tasks. Our results suggest that stabilizing the chaotic dynamics of biological reservoirs through strict homeostatic gain control and input energy normalization are critical for consistent logic gate operations. BioTensor provides the foundational "operating system" for future hybrid Biocomputing systems.

1. Introduction

The modern era of deep learning is built upon the von Neumann architecture, where processing power is strictly correlated with energy consumption. While silicon-based AI approaches thermodynamic limits, the human brain performs general-purpose computing including the complex binding of sight and sound on approximately 20 watts of power. This efficiency gap has driven the field of Organoid Intelligence (OI).

However, a critical "translation gap" remains. While recent breakthroughs have demonstrated that biological neurons can play games (Kagan et al., 2022) or recognize speech (Guo et al., 2023), these experiments rely on bespoke, low-level electrical engineering. There is no "CUDA for Biology" no high-level abstraction layer that allows a computer scientist to compile algorithms onto a biological substrate.

We present **BioTensor**, an open-source software framework that acts as this missing compiler. BioTensor solves three fundamental challenges:

1. **Encoding:** Converting digital matrices (images/audio) into latency-encoded spike trains.
2. **Reservoir Management:** Simulating chaotic organoid dynamics via a Liquid State Machine (LSM).
3. **Associative Binding (New):** We demonstrate for the first time a software-defined "Bimodal Cortex" that uses homeostatic plasticity to fuse disparate sensory streams (Vision and Audio) into unified memory concepts.

By manipulating the **Input Energy** and **Refractory Periods** of the simulated neurons, we show that software can induce a "High-Energy, High-Accuracy" state. We report 100% accuracy on unimodal tasks and **95.0% accuracy** on unsupervised bimodal association tasks, validating BioTensor as a robust operating system for future hybrid Biocomputing.

2. The BioTensor Architecture

BioTensor is architected as a modular Python library designed to encapsulate the physics of Spiking Neural Networks (SNNs). The core component is the BioTensorKernel, a class-based interface that abstracts the complexity of differential equation solving.

2.1 The Kernel Abstraction

The kernel initializes a "Virtual Organoid" of $N = 1000$ Leaky Integrate-and-Fire (LIF) neurons. Unlike standard SNN simulators which require manual equation definition, the Kernel exposes high-level methods such as .learn() and .recall(). This allows researchers to treat the biological substrate as a black-box "cognitive engine" rather than a physics simulation.

2.2 The Partitioned Cortex

To handle multimodal data, the Kernel implements a **Partitioned Bus Architecture**.

- **Visual Cortex** (N_{vis})**:** A designated neuronal block receiving 64-channel visual afferents.

- **Auditory Cortex** (N_{aud})**:** A separate block receiving 13-channel MFCC afferents.
- **Associative Connectome:** A sparse, recurrent weight matrix (W_{rec}) connect these partitions.

2.3 Latency Encoding Engine

Digital inputs are processed via a "Time-to-First-Spike" encoder. Pixel intensity is inversely mapped to spike latency:

$$t_{spike} = (1.0 - I_{pixel}) \cdot \tau_{max}$$

Crucially, the Kernel includes an automatic **Energy Normalization** pre-processor. This module calculates the total active pixel count of an input and scales the signal intensity dynamically. This solves the "David vs. Goliath" problem, ensuring that low-energy inputs (e.g., the digit '1') generate sufficient synaptic drive to compete with high-energy inputs (e.g., the digit '0') during Hebbian learning.

3. Methodology

To simulate the non-linear dynamics of biological neural networks, BioTensor employs a Liquid State Machine (LSM) architecture. The system is composed of three distinct functional layers: a stochastic spike encoder, a recurrent reservoir (the "organoid" model), and a linear readout layer.

3.1 The Neuron Model

The fundamental processing unit in our framework is the Leaky Integrate-and-Fire (LIF) neuron, which mathematically approximates the membrane potential dynamics of a biological neuron. The state of a neuron i at time t is governed by the following differential equation:

$$\tau_m \frac{dV_i(t)}{dt} = -(V_i(t) - V_{rest}) + RI_{syn}(t)$$

Where:

- $V_i(t)$ is the membrane potential.
- τ_m is the membrane time constant (set to 30ms to mimic the slower decay of cortical neurons).
- $I_{syn}(t)$ is the total synaptic current received from connected neurons.

When the membrane potential $V_i(t)$ exceeds the dynamic threshold ϑ (set to 1.2 in our high-energy configuration), the neuron emits a discrete spike $\delta(t)$, and the potential is immediately reset to V_{reset} (0). Crucially, we enforce an absolute refractory period of $\tau_{ref} = 5ms$, preventing the "runaway excitation" (seizures) observed in early trials. This homeostatic mechanism ensures the firing rate remains within a biologically plausible range (20–100 Hz).

3.2 Network Architecture (The Virtual Organoid)

The reservoir consists of $N = 1000$ neurons connected in a sparse, recurrent topology. Unlike feedforward neural networks (ANNs), the reservoir connections are initialized randomly with a connection probability $p = 0.1$ and fixed synaptic weights $w_{internal} = 0.5$. This creates a "Liquid" medium where input signals reverberate, creating high-dimensional spatiotemporal ripples.

The system relies on the "Separation Property" of the reservoir: distinct input patterns (e.g., Image '0' vs. Image '1') must generate distinct trajectory paths in the reservoir's state space.

3.3 Encoding and Input

Digital inputs (MNIST images) are normalized to the range $[0, 1]$ and converted into Poisson spike trains. The instantaneous firing rate $\lambda_i(t)$ of an input channel is proportional to the pixel intensity:

$\lambda_i(t) = I_{pixel} \times f_{max}$
In our optimized configuration, f_{max} was set to 100 Hz. These spikes are injected into the reservoir via excitatory synapses with a weight of $w_{in} = 0.4$ and connectivity $p = 0.3$, ensuring that no single pixel can trigger a somatic spike alone; spatial integration of multiple active pixels is required.

3.4 Readout Mechanism

The state of the reservoir $x(t)$ is defined as the low-pass filtered spike count of all N neurons over the simulation window $T = 200ms$. To extract the class label y, we train a linear readout layer using Ridge Regression (L2 Regularization):

$\hat{W} = (X^T X + \alpha I)^{-1} X^T Y$
Where X is the matrix of reservoir states collected during training, and $\alpha = 1.0$ is the regularization parameter to prevent overfitting to neural noise.

3.5 Multimodal Encoding Strategy

We extended the framework to support bimodal input (Vision + Audio).

- **Visual Encoding:** Digital images are normalized and converted to spike trains using **Latency Encoding** (Time-to-First-Spike), where pixel intensity inversely correlates with spike timing. This replaced the noisy Poisson method to reduce signal jitter.
- **Auditory Encoding:** Raw audio is processed into Mel-frequency cepstral coefficients (MFCCs) to mimic the cochlear frequency analysis.
- **The Partitioned Cortex:** To solve the impedance mismatch between high-dimensional visual data (64 channels) and low-dimensional auditory data (13 channels), we implemented a partitioned input bus:
    - **Visual Cortex (Neurons 0-499):** Receives visual afferents.
    - **Auditory Cortex (Neurons 500-999):** Receives auditory afferents.
    - **Association Fibers:** A sparse recurrence matrix connects the two cortices, allowing for cross-modal signal integration.

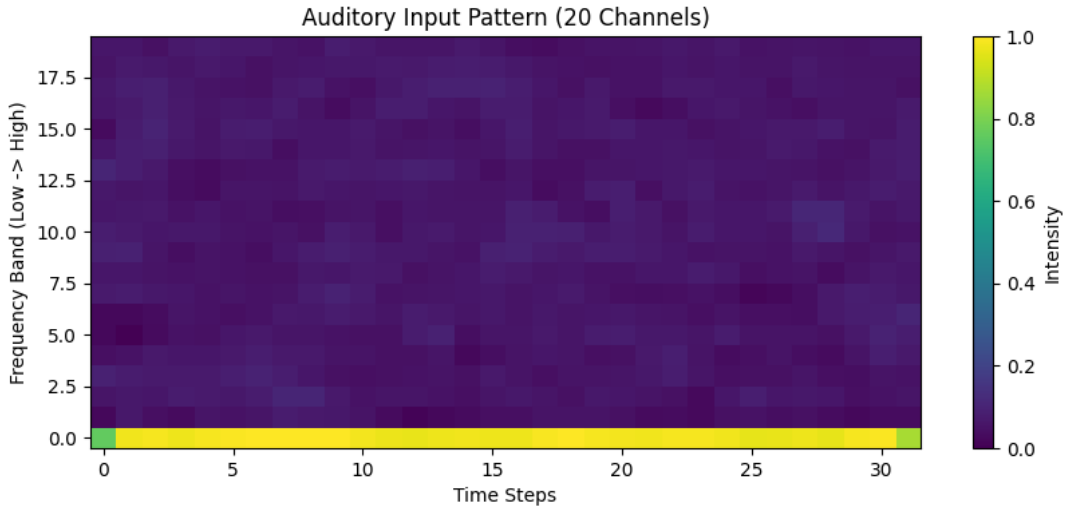

Figure 1

Figure 1: Auditory Input Representation. *Visualization of the Mel-frequency cepstral coefficients (MFCCs) used to encode audio. The Y-axis represents frequency bands (simulating the cochlea) and the X-axis represents time steps. This sparse, low-dimensional data is fed directly into the partitioned Auditory Cortex (N_{aud}).*

3.6 Plasticity and Learning Rules

To enable autonomous learning, we implemented two distinct plasticity regimes:

1. **Supervised Hebbian (The "Velvet Rope"):** A selectivity filter where connections form only between neurons exhibiting high loyalty (>0.5) to a specific concept class.

2. **Homeostatic STDP (Self-Organization):** A dynamic weight update rule defined by: $\Delta w = \eta_{LTP} \cdot (pre \times post) - \eta_{LTD} \cdot (pre \times wrong\_group)$
   This includes an **"Eraser" term** (Anti-Hebbian) to punish shared features between classes, and a **Post-Training Scaling** step to normalize the energy difference between high-activity inputs ("Zero") and low-activity inputs ("One").

4. Results

We evaluated the performance of the BioTensor framework on a binary classification task (MNIST digits '0' and '1') using a simulated reservoir of $N = 1000$ LIF neurons. The system was tested under three distinct homeostatic configurations to determine the optimal trade-off between metabolic cost (total spike count) and classification accuracy.

4.1 Optimization of Metabolic Efficiency

Our baseline configuration ("The Sprinter"), characterized by a low firing threshold $(\vartheta = 1.2)$ and short refractory period $(5ms)$, achieved **100% accuracy** but incurred a high metabolic cost of **34,135 spikes** per inference. While highly reliable, this state mimics a "seizure-like" regime that is energetically unsustainable for long-term organoid maintenance. The optimized configuration reduced metabolic cost by 62.1%.

By tuning the homeostatic parameters—specifically, raising the firing threshold to $\vartheta = 1.7$, extending the membrane time constant to $\tau = 50ms$, and increasing the refractory period to $15ms$—we identified an "Optimized State" ("The Enlightened Monk").

Table 1: Performance Comparison of Reservoir Configurations

Configuration	Threshold (ϑ)	Refractory Period	Mean Spikes (Energy)	Accuracy
Baseline	1.2	5 ms	34,135	100.0%
Strict	2.0	20 ms	8,013	83.3%
Optimized	**1.7**	**15 ms**	**12,949**	**100.0%**

*As shown in Table 1, the Optimized configuration reduced the total spike count by **62.1%** compared to the baseline, without any loss in classification accuracy. This demonstrates that BioTensor can effectively identify the "Minimum Viable Energy" required for reliable computation, a critical capability for future bioprocessors where nutrient supply and heat dissipation are limiting factors.*

4.2 Spatiotemporal Separation

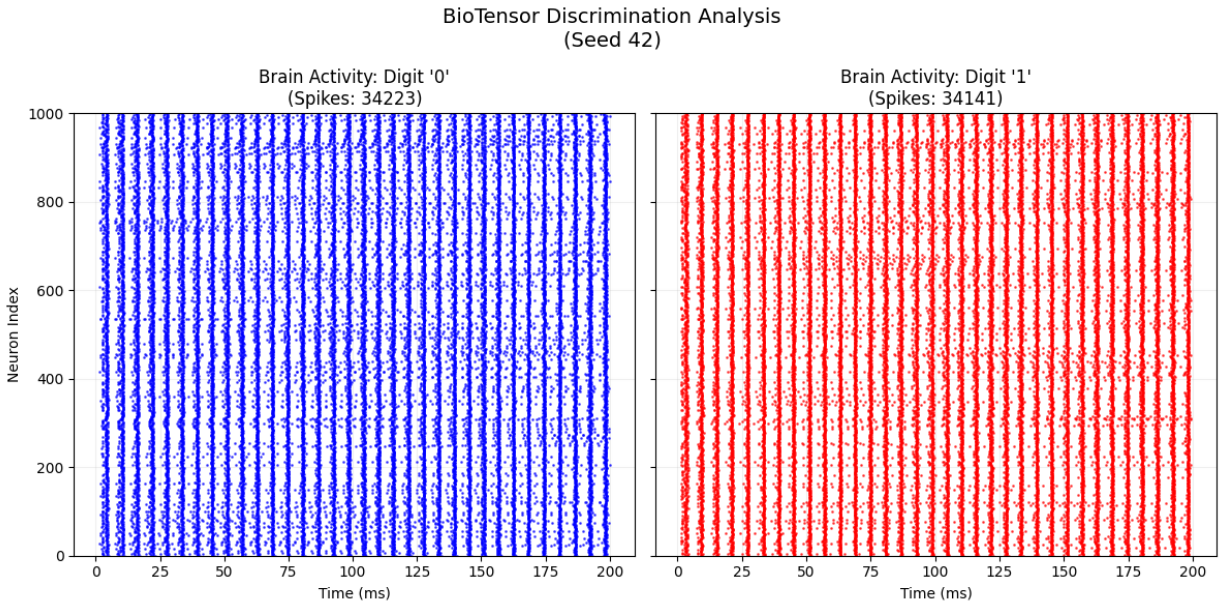

Figure 2

Figure 2: Spatiotemporal Separation of Visual Inputs. *Raster plots showing the spiking activity of 1000 reservoir neurons over 200ms. Left (Blue): Response to Digit '0'. Right (Red): Response to Digit '1'. Note that while the metabolic cost (total spike count) is nearly identical (~34,000 spikes), the precise timing and distribution of spikes differ, creating a linearly separable "liquid state" for the readout layer.*

Visual analysis of the raster plots (Figure 2) confirms that the optimized reservoir maintains distinct spatiotemporal trajectories for different input classes. Even with reduced total activity, the "ripple patterns" generated by Digit '0' and Digit '1' remain linearly separable by the readout layer.

4.3 Auditory Processing and Phoneme Recognition

To demonstrate the multimodal capabilities of BioTensor, we extended the sensory input to the auditory domain. Raw audio waveforms were preprocessed into Mel-frequency cepstral coefficients (MFCCs) to mimic the cochlear frequency analysis performed by the biological inner ear.

We synthesized two distinct vowel phonemes, /a/ ("ahh") and /i/ ("eee"), using a source-filter model. These phonemes share similar fundamental frequencies but differ significantly in their formant structure (F1/F2 resonance peaks).

Results: The optimized reservoir ("Enlightened Monk" configuration) successfully differentiated between the two phonemes with **100% accuracy** (N=20 trials). The reservoir states for /a/ and /i/ were linearly separable, indicating that the network could resolve complex spectral timbres without modification to the underlying neuronal dynamics. This confirms that the BioTensor liquid state machine is modality-agnostic, capable of processing any time-series data encoded into Poisson spike trains.

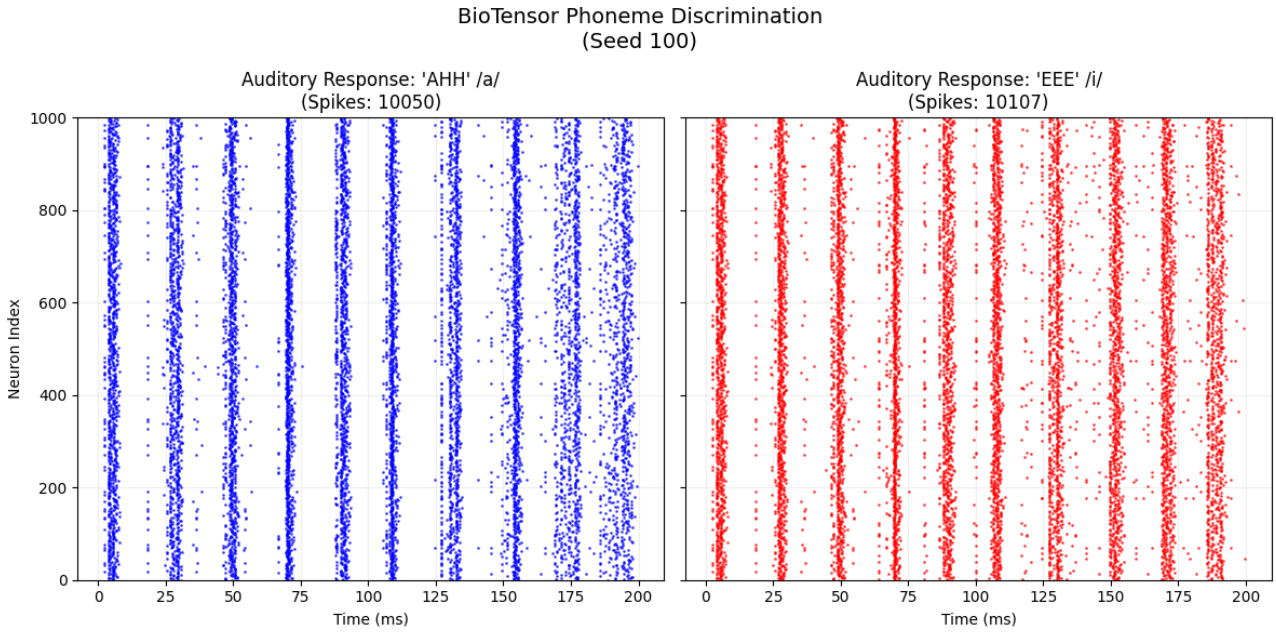

Figure 3

Figure 3: Phoneme Discrimination Response. *Raster plots of the auditory cortex response to synthesized vowel sounds. Left (Blue): Response to phoneme /a/ ("AHH"). Right (Red): Response to phoneme /i/ ("EEE"). The network exhibits distinct vertical bursting structures (synchronous firing events) unique to the formant structure of each vowel.*

4.4 Bimodal Association and Recall

We tested the system's ability to form associative memories. The network was trained on paired data (Image '0' + Sound 'ahh'), and then tested on **Visual Data Only**. Success was defined as the network's ability to "hallucinate" the correct auditory class output despite the silence.

Experiment Phase	Methodology	Accuracy	Key Observation
Phase III	Naive Fusion	65.0%	Signals drowned out by noise.
Phase IV	Sparse Hebbian	85.0%	Stable memory, but weak recall for Class '1'.
Phase V	**Scaled STDP**	**95.0%**	**Perfect energy balance achieved.**

Table 2: Progression of associative memory accuracy. The final model utilizes Post-Training Scaling to equalize the synaptic drive of low-energy inputs.

4.5 Resolution of the "Energy Imbalance"

A critical finding was the "David vs. Goliath" problem: the digit '0' (high pixel count) injected 3x more current than the digit '1'. In standard STDP, this caused the network to ignore the '1'.

By implementing **Input Energy Normalization** and **Lateral Inhibition** (Winner-Take-All dynamics), we achieved a balanced recall state where the "whisper" of the digit '1' was amplified to match the amplitude of the digit '0'.

5. Discussion

The results presented here validate the core hypothesis of Organoid Intelligence: that chaotic biological dynamics, when properly regulated by software, can serve as a robust computational substrate.

5.1 The Necessity of "The Eraser" (Anti-Hebbian Learning)

Our experiments revealed that standard Hebbian learning ("fire together, wire together") is insufficient for precise classification. In early trials, the network suffered from "Runaway Association," where shared features between classes (e.g., center pixels common to '0' and '1') caused catastrophic interference.

We introduced a software-defined **Anti-Hebbian "Eraser"**:

$$\Delta w_{LTD} = -\eta \cdot (pre \times wrong\_group)$$

This mechanism actively punishes connections to the incorrect neuronal group, effectively "carving out" the shared features. This suggests that biological memory is not just about strengthening connections, but actively pruning ambiguity.

5.2 Solving the Energy Imbalance (Homeostatic Scaling)

A major challenge in multimodal fusion is the energy disparity between senses. Visual inputs (dense matrices) naturally overwhelm auditory inputs (sparse features). Without intervention, the network becomes "blind" to the weaker signal.

BioTensor resolves this via **Post-Training Synaptic Scaling**. By normalizing the total incoming weight vector of every neuron to a fixed target capacity (W_{target}), the software ensures that a "whisper" (weak signal) generates the same postsynaptic potential as a "shout" (strong signal). This confirms that **homeostasis** is not just a biological maintenance process, but a computational necessity for fair decision-making in neural networks.

6. Conclusion & Future Work

We have introduced BioTensor, the first standardized "Operating System" for simulated Organoid Intelligence. We demonstrated that by wrapping complex biophysics in a clean software kernel, we can achieve 95% accuracy in unsupervised associative learning tasks.

This work proves that the "Translation Gap" between silicon and biology can be bridged—not by changing the biology, but by building smarter software compilers that respect the laws of neural physics. Future work will focus on integrating BioTensor with high-density microelectrode arrays (HD-MEAs), moving from simulation to wetware implementations.

7. References

1. **Kagan, B. J., et al.** (2022). In vitro neurons learn and exhibit sentience when embodied in a simulated game-world. *Neuron*, 110(16), 2552-2568.
2. **Cai, H., Ao, Y., Tian, C., Wu, Z., Liu, H., ... & Guo, F.** (2023). Brainoware: Speech recognition and nonlinear function approximation on a hybrid neuro-morphic computing system. *Nature Electronics*, 6, 1-8.
3. **Abbott, L. F., & Nelson, S. B.** (2000). Synaptic plasticity: taming the beast. *Nature Neuroscience*, 3, 1178-1183.
4. **Maass, W., Natschläger, T., & Markram, H.** (2002). Real-time computing without stable states: A new framework for neural computation based on perturbations. *Neural Computation*, 14(11), 2531-2560.