Neuromorphic intermediate representation

Artyom Grishin, Victor Mazanov, Vladislav Merkulov, Remal Gareev

Project Idea

The idea is to implement the Neuromorphic Intermediate Representation (NIR) [1] and run the single Leaky Integrate-and-Fire (LIF) neuron experiment. Our goal is to validate the NIR framework by replicating the LIF neuron behavior and comparing our results with those reported by the authors in the original paper.

Method/Technique

We will use Google Colab as the primary computing resource, leveraging its GPU for efficient simulation. The experiment will be implemented using Python and relevant neuromorphic computing libraries, such as PyTorch, snnTorch, or Norse. We will define the LIF neuron using NIR's computational primitives and structure the model according to the NIR framework. Input to the neuron will be a predefined current stimulus, as described in the paper, and we will analyze the resulting spike train and membrane potential dynamics.

Dataset explanation and accessible link

Neuromorphic MNIST (N-MNIST) – Used for training and evaluating a Spiking Convolutional Neural Network (SCNN). This dataset is an event-based version of the classic MNIST dataset, designed for neuromorphic vision processing. Instead of conventional pixel-based images, N-MNIST uses spike-based event data captured by a Dynamic Vision Sensor (DVS), which records changes in light intensity over time. Link to the dataset

1 Project Timeline (1 March - 18 April)

Dates	Task
1–7 Mar	**Research & Planning** - Study NIR framework and LIF dynamics - Review PyTorch/snnTorch/Norse libraries - Create shared summary document
8–14 Mar	**LIF Prototype Development** - Implement basic LIF neuron in Python - Configure Google Colab environment - Generate test
15–21 Mar	**NIR Integration** - Map LIF model to NIR primitives - Validate against paper results - Visualize membrane potential
22 Mar–4 Apr	**SCNN Implementation** - Design network architecture - Preprocess N-MNIST dataset - Integrate NIR components
5–11 Apr	**Training & Tuning** - Train SCNN model - Optimize hyperparameters - Validate accuracy (>85% target)
12–18 Apr	**Final Preparation** - Create presentation slides - Prepare demo video - Final documentation

References

[1] Jens E. Pedersen, Steven Abreu, Matthias Jobst, Gregor Lenz, Vittorio Fra, Felix Christian Bauer, Dylan Richard Muir, Peng Zhou, Bernhard Vogginger, Kade Heckel, Gianvito Urgese, Sadasivan Shankar, Terrence C. Stewart, Sadique Sheik, and Jason K. Eshraghian. Neuromorphic intermediate representation: A unified instruction set for interoperable brain-inspired computing. *Nature Communications*, 15(1):8122, September 2024.