SLAYER for Loihi

Sumit Bam Shrestha
Mar 17, 2021

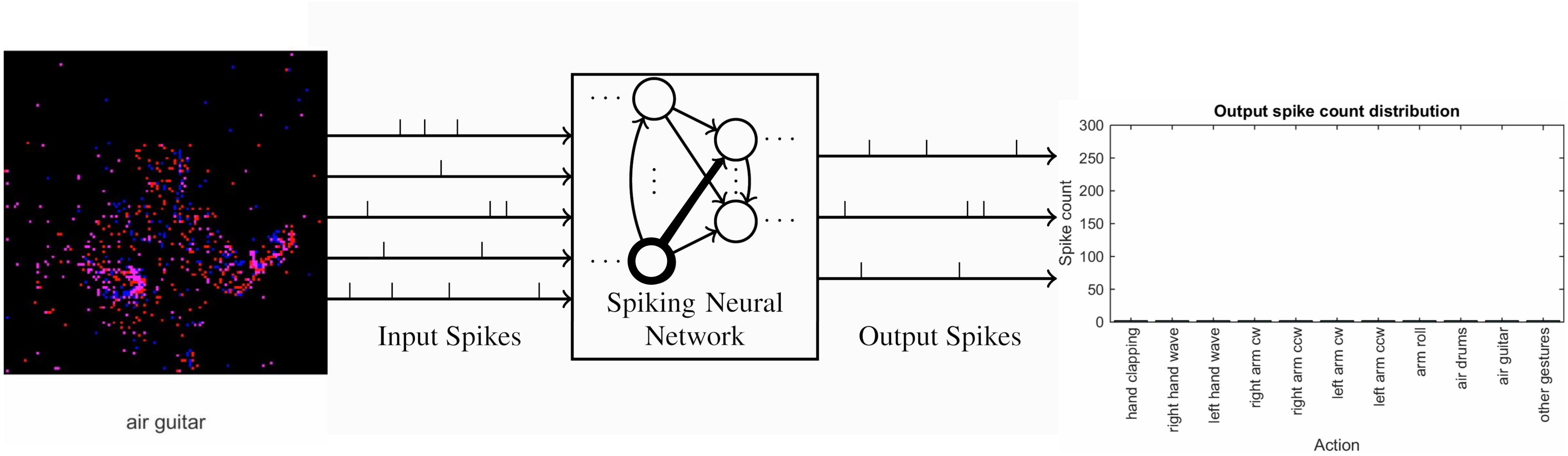
NICE 2021

Outline

- Computing with spikes
- SLAYER tools for Loihi
- Notebook Demos
  - SLAYER training notebook walkthrough (NMNIST)
  - Inference notebook in Loihi demo (NMNIST)
- Benchmark results

Computing with spikes

SLAYER

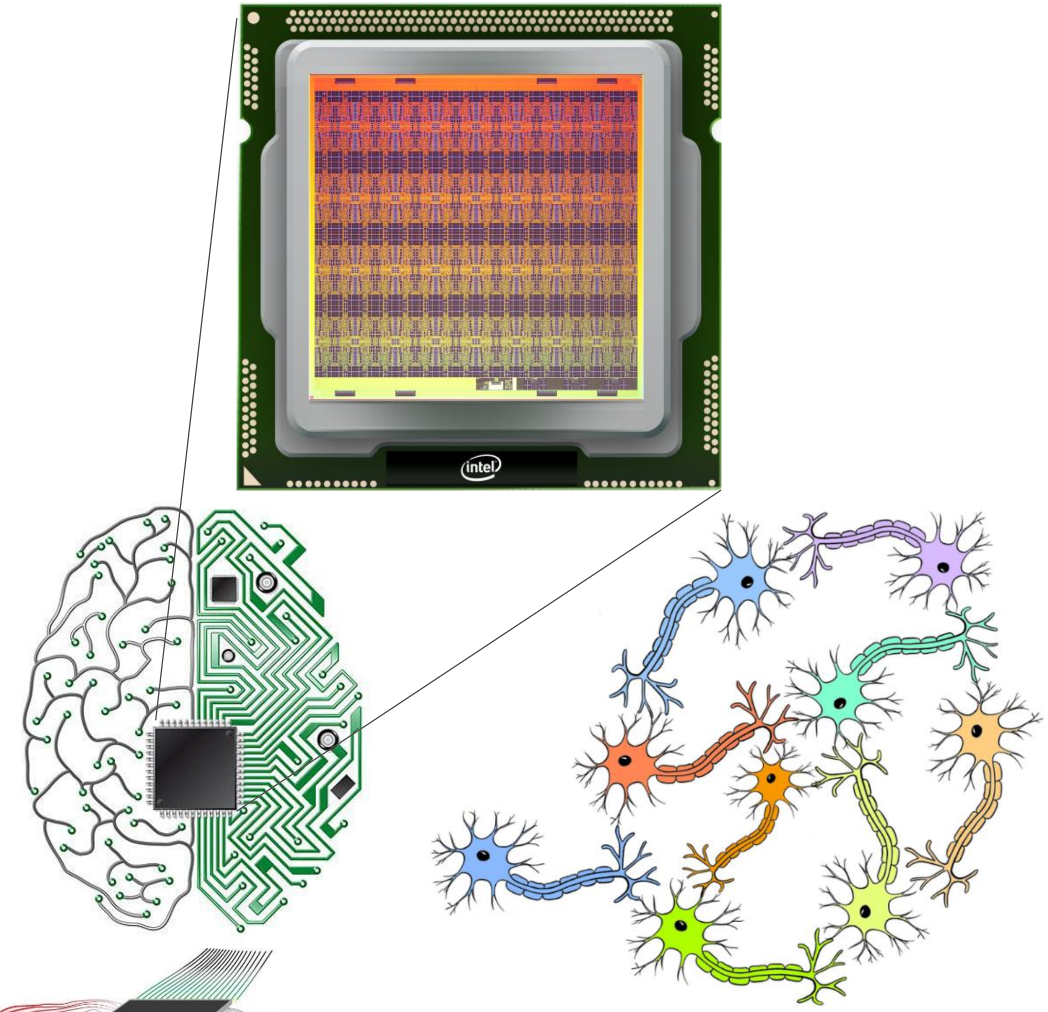


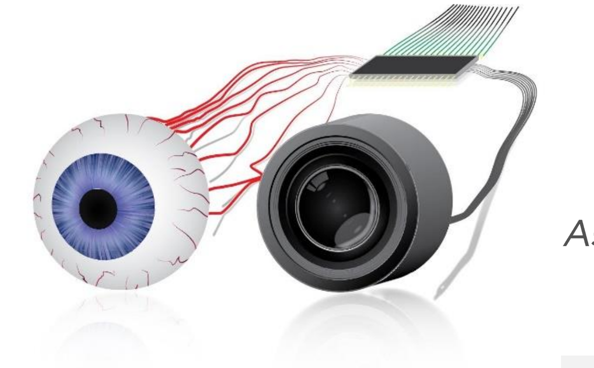
- Spike Backpropagation Method
- Custom PyTorch implementation
- dt-based learning rule
- Learns synaptic weights and axonal delays

Loihi: Computing With Spikes

Borrowing computing principles from biology

- Sparse communication with events
- Massively parallel
- Spatiotemporal interaction
- Local compute
- Low energy
- Versatile network

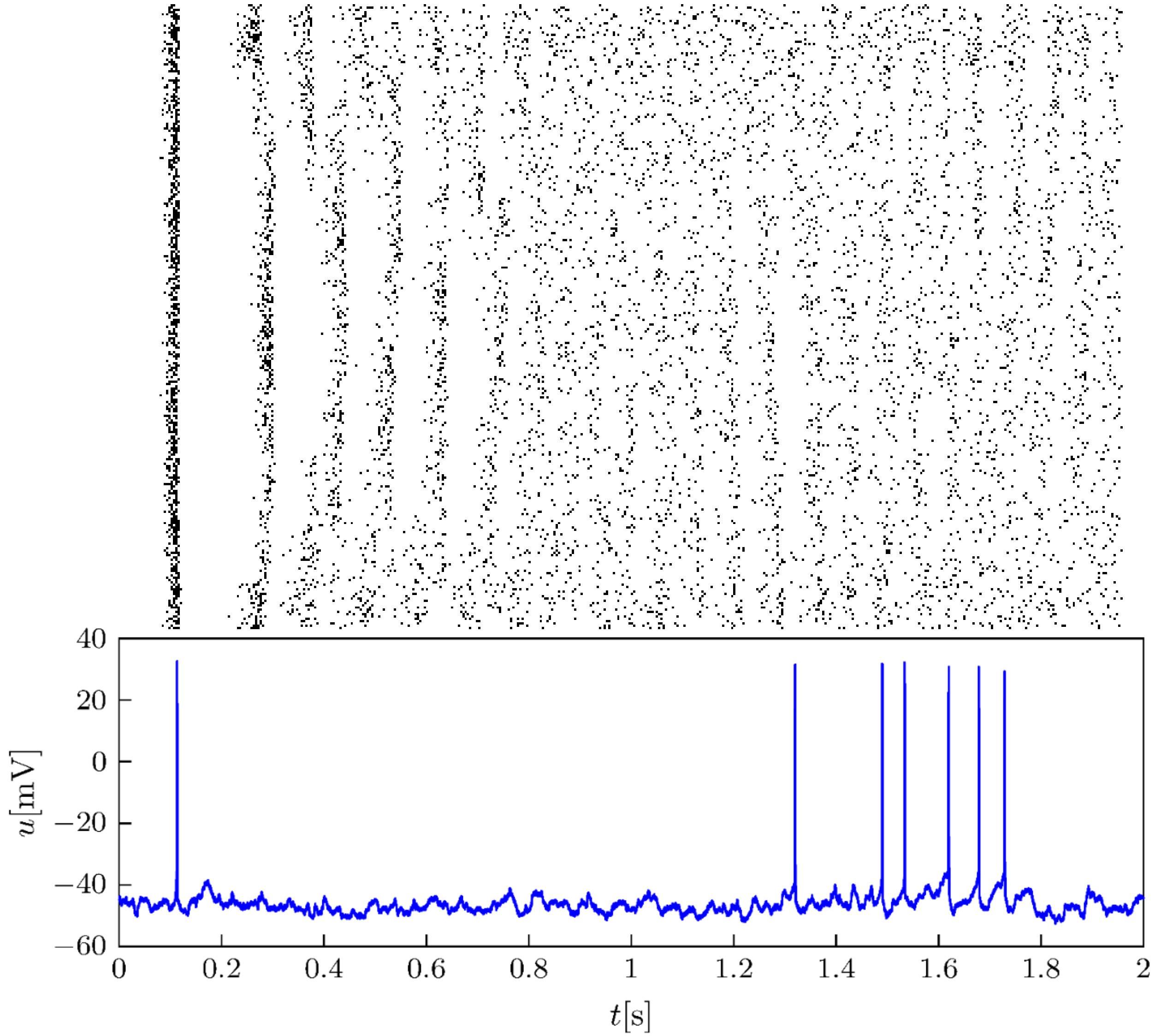




As engineers, we would be foolish to ignore the lessons of a billion years of evolution.
—Carver Mead, 1993

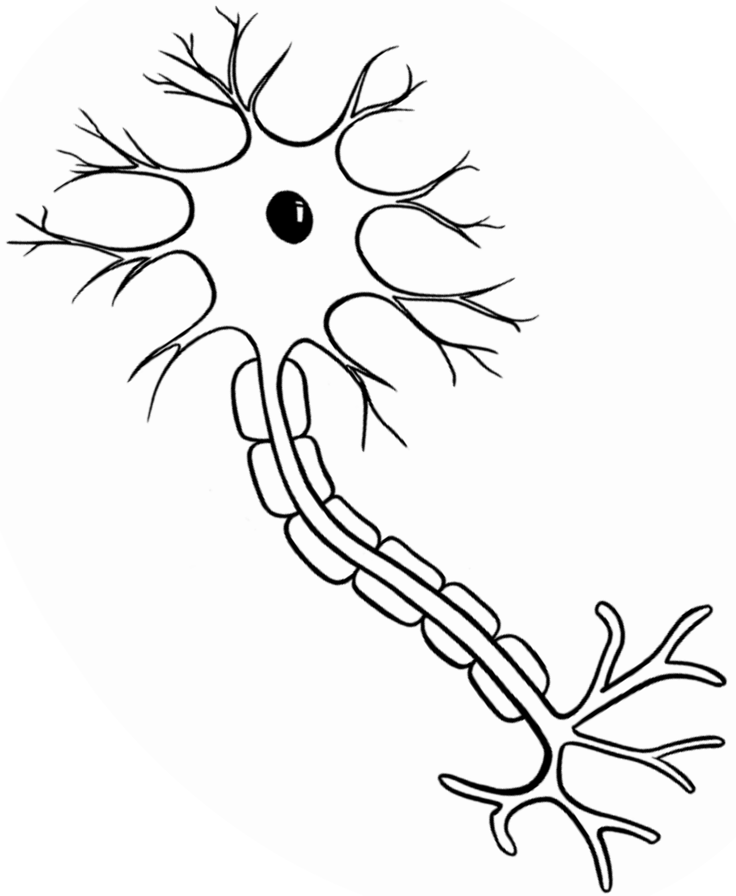
Sparsity with Spikes

- Biology uses spikes
- Events in time
- Magnitude is less relevant
- Spike only when necessary

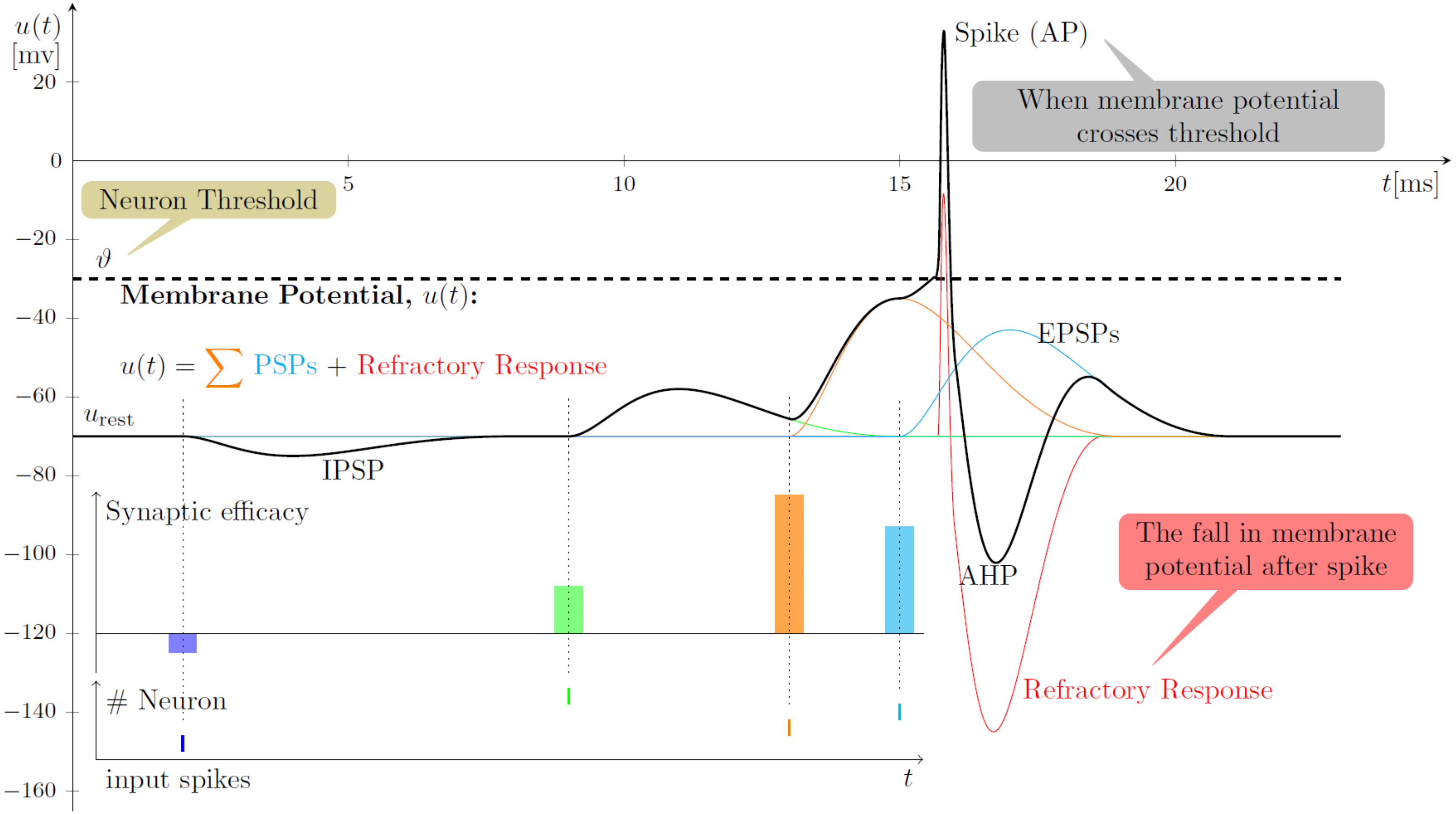


Spiking Neuron: Temporal Computing

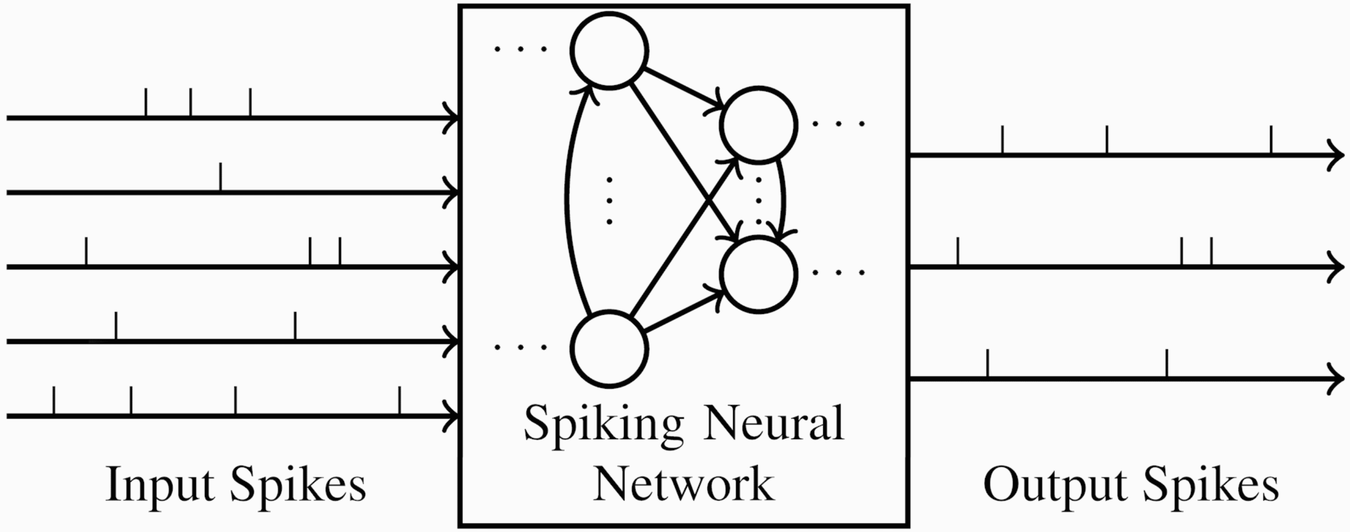
- Mathematical model of biological neuron
- Input and output are both spikes
- Like activation functions



Spiking Neuron: Temporal Computing



SNN: Sparse, Local, Parallel and Spatiotemporal

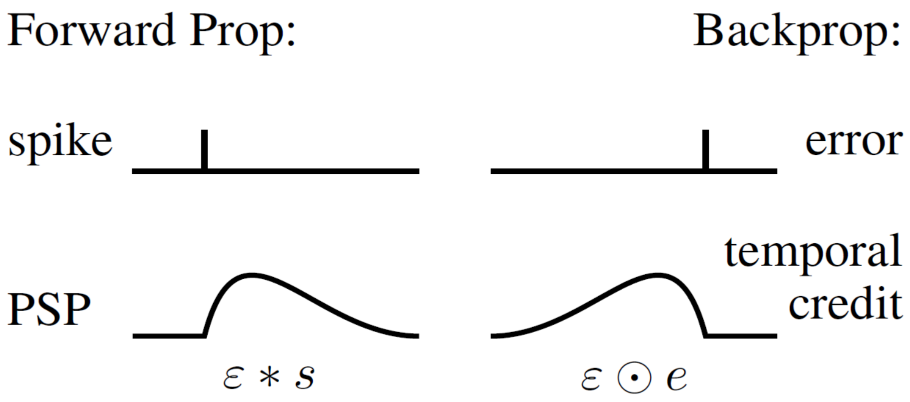


SLAYER tools for Loihi

Error Backpropagation with SLAYER

Two key principles:

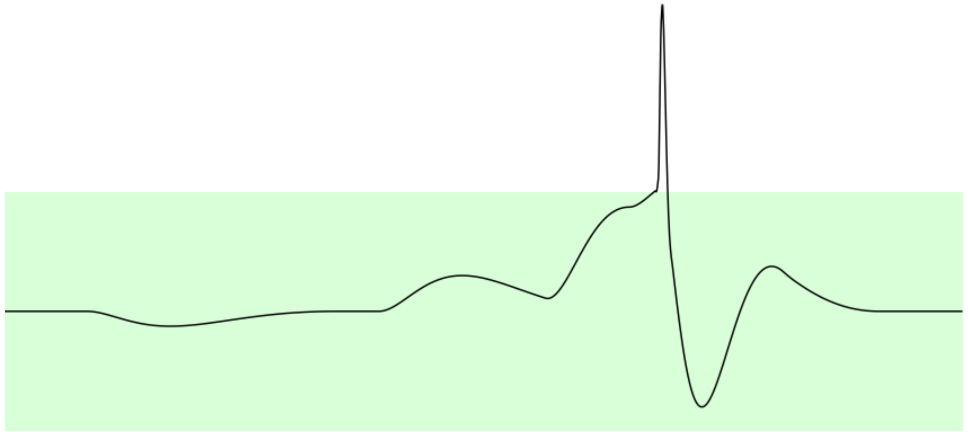
- Temporal Error credit assignment
  - Rewind actions in history

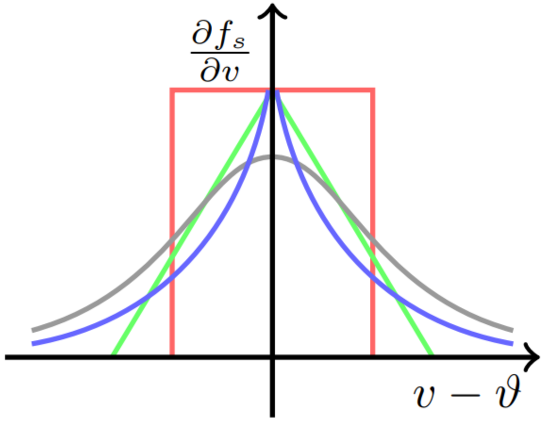


Error Backpropagation with SLAYER

Two key principles:

- Temporal Error credit assignment
- Spike function derivative
  - Look beneath the surface
  - Spike Escape Rate function





SLAYER PyTorch

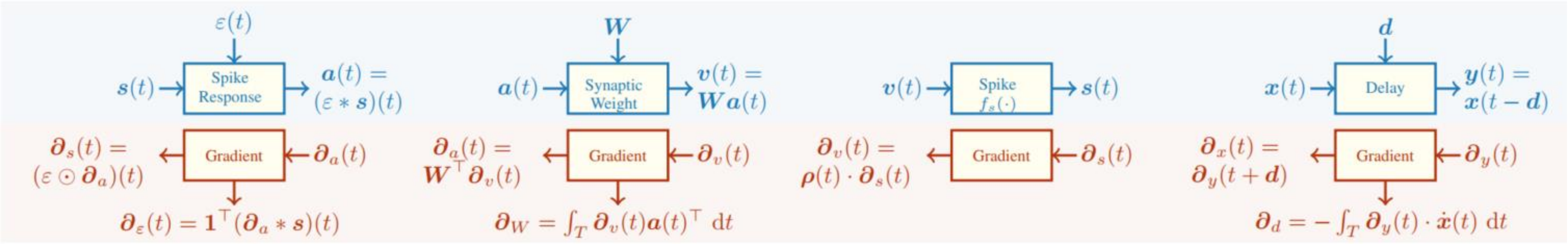
- Lego Like computational Blocks
- Fully auto-grad compatible
- dense, convolution, pooling, transposed convolution, unpooling
- Axonal delay

$$\boldsymbol{a}^{(l-1)}(t) = \left(\varepsilon * s_{\boldsymbol{d}}^{(l-1)}\right)(t)$$

$$\boldsymbol{v}^{(l)}(t) = \boldsymbol{W}^{(l-1)}\boldsymbol{a}^{(l-1)}(t) + \left(\nu * \boldsymbol{s}^{(l)}\right)(t)$$

$$\boldsymbol{s}^{(l)}(t) = f_s(\boldsymbol{v}^{(l)}(t))$$

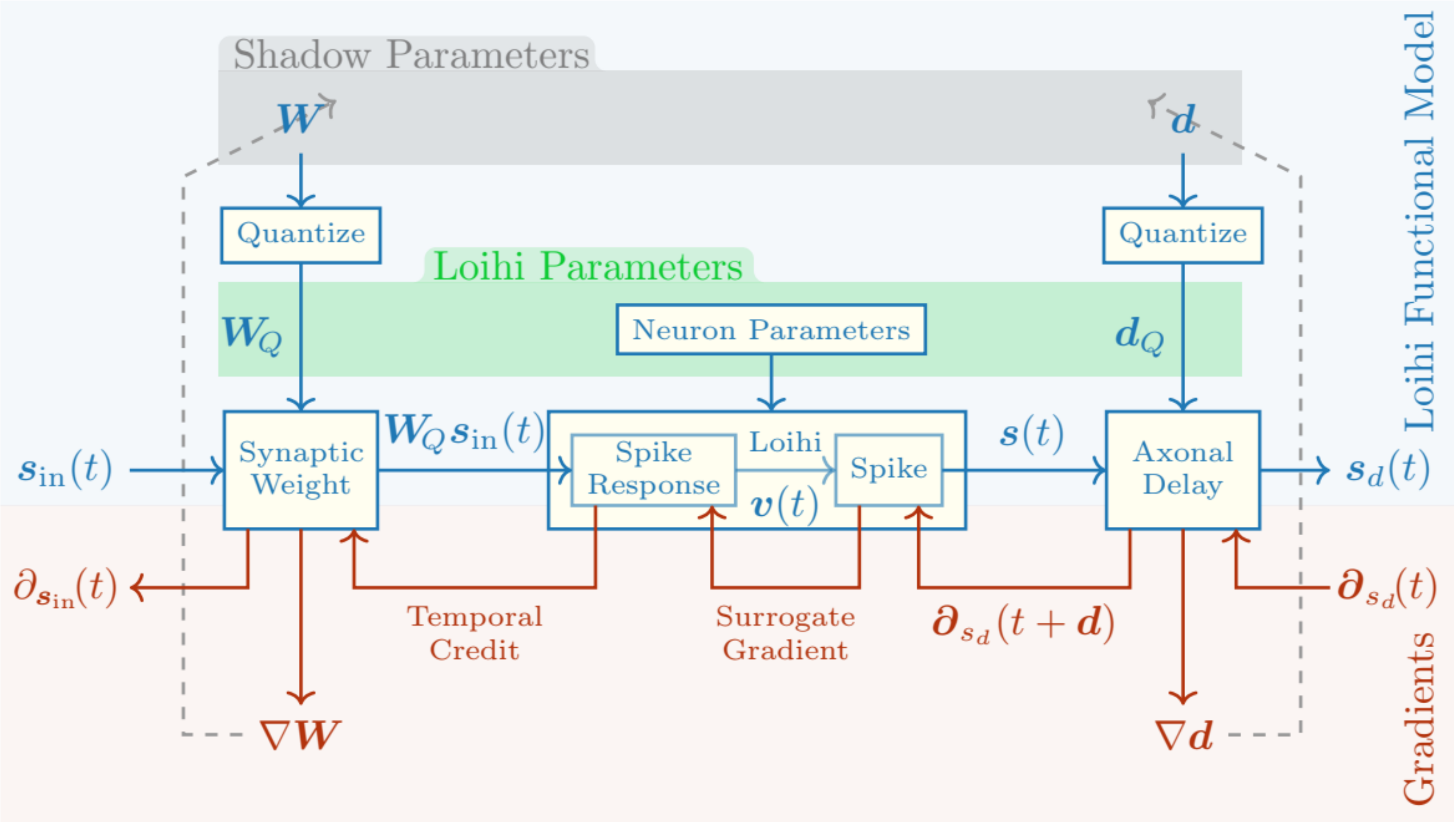
$$\boldsymbol{s}_{\boldsymbol{d}}^{(l)}(t) = \boldsymbol{s}^{(l)}(t - \boldsymbol{d})$$



Learnable PSP response | Learnable Weights | Learnable Delays

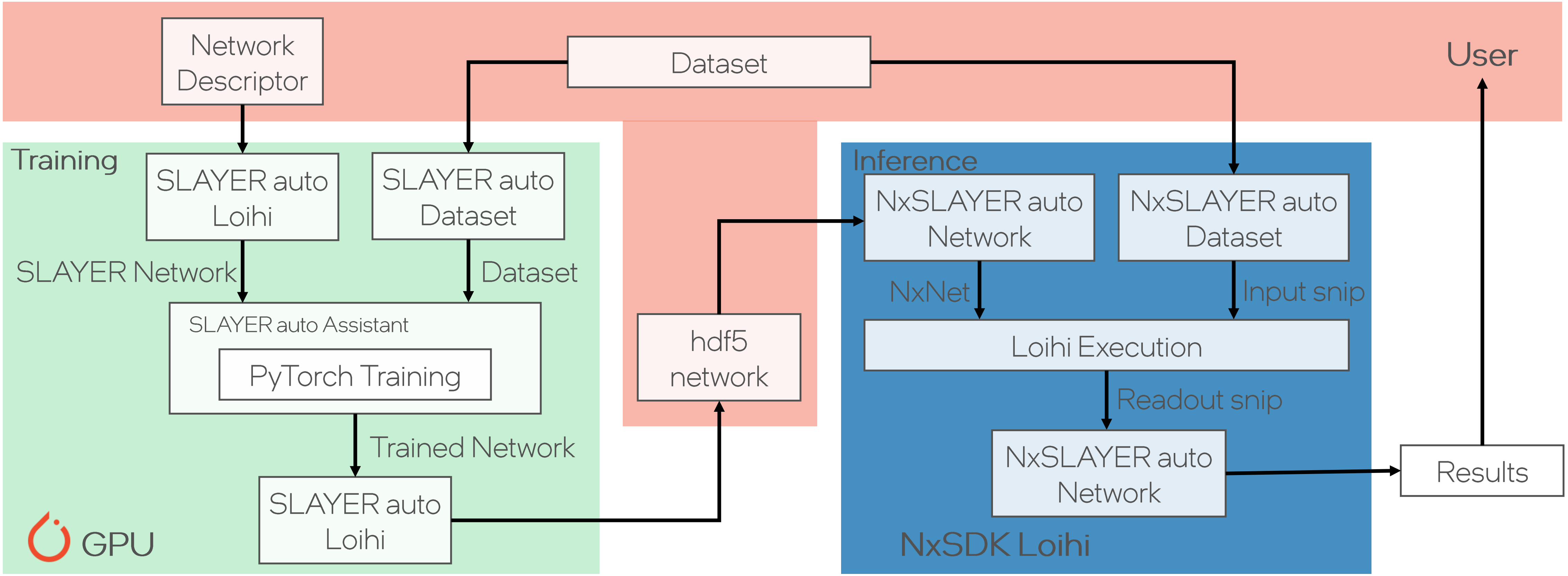
SLAYER-Loihi

- For CUBA LIF neuron model in Loihi

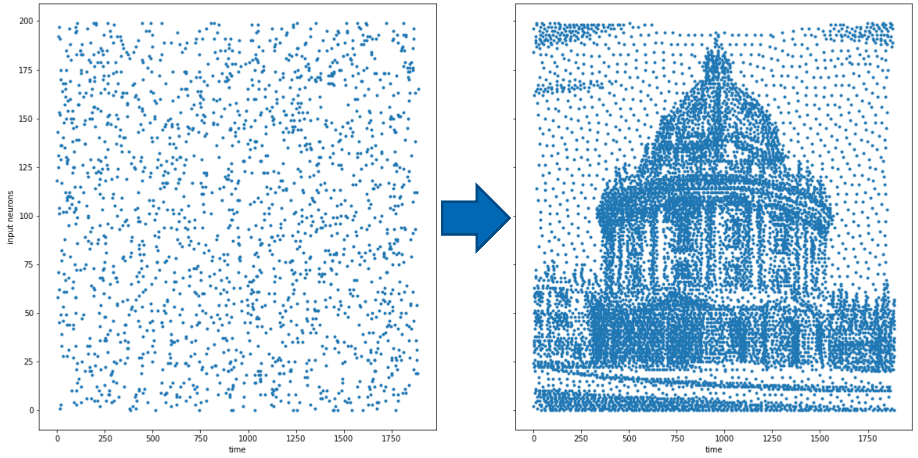


SLAYER Auto Modules

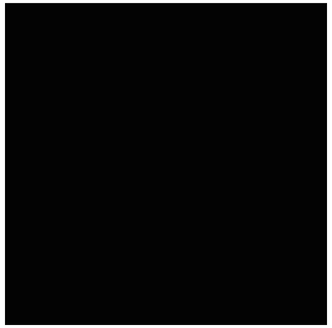
- Easy network creation for training on PyTorch and executing on Loihi.



SLAYER capabilities

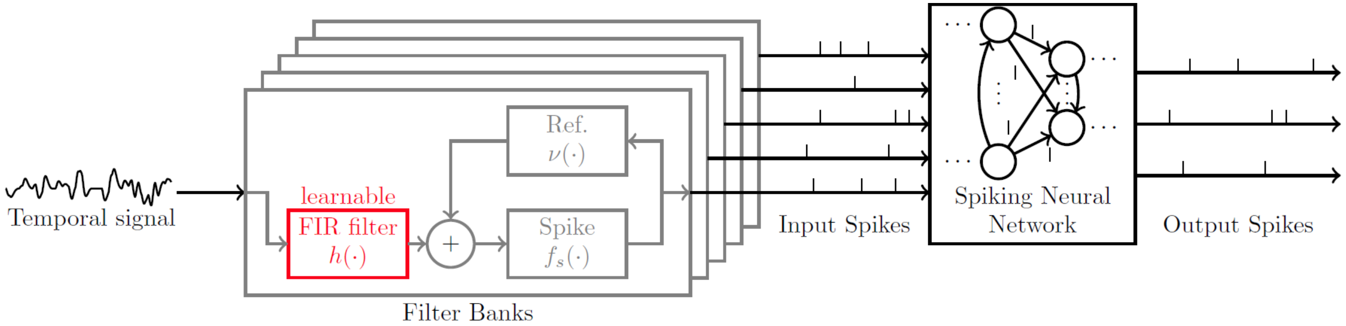


- Precise learning of temporal spikes
- Processing dynamic spatiotemporal data
- Learning synaptic weights and axonal delays
- One-to-one network mapping to Loihi
- End-to-end learning with spike encoding
  - Numeric temporal data injection to the spiking network
  - Generalized linear model



Not the best suited for

- Rate coded models



Notebook Demo

SLAYER-Loihi : Tutorials

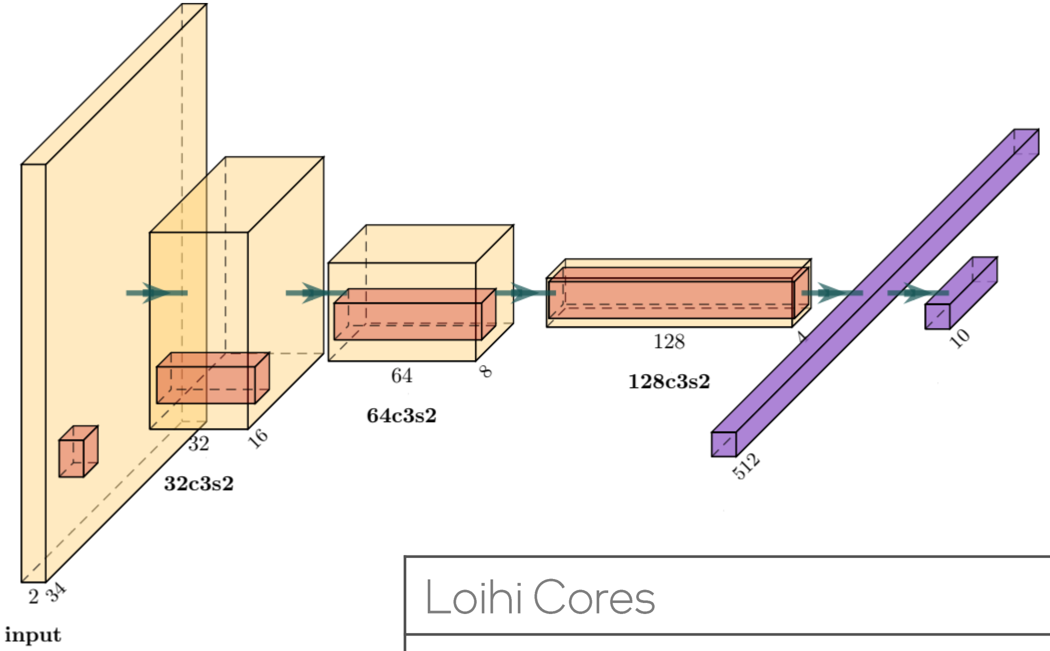
- Examples for SLAYER Loihi training are here:
  - https://github.com/bamsumit/slayerPytorch/tree/master/exampleLoihi

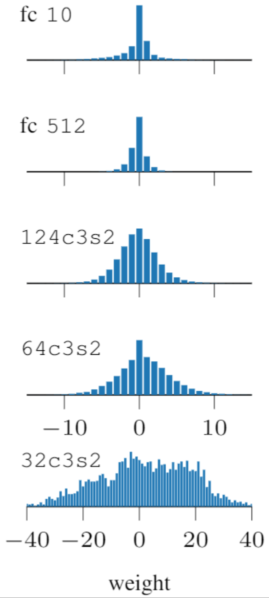
- Network inference demo:

Benchmark Results

SLAYER-Loihi : Classification Benchmarks

NMNIST‡





Loihi Cores	37
Dynamic Energy (mJ/sample)	1.98 ± 0.11
Inference Speedup (x)	9.86 ± 0.83
Sample Length (ticks)	300
Energy Delay Product (μJs/sample)	60.54 ± 6.60

	Accuracy (%)	Parameters
Loihi	99.20 ± 0.10 (Best: 99.33)	1,147k
SNN[1]	99.53	17,664k

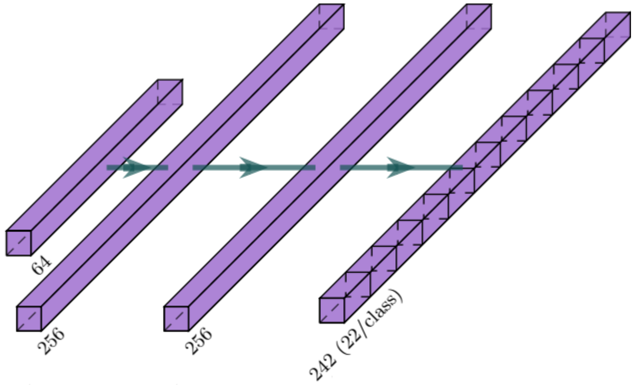
Intel Loihi measurements obtained using NxSDK v0.9.9 running on Nahuku 32. Performance results are based on testing as of Jan 2021 and may not reflect all publicly available security updates. Results may vary.

[1] Wu et al. *Direct training for spiking neural networks: Faster, larger, better.*

‡ NMNIST dataset is available for public use under CC4.0 at https://www.garrickorchard.com/datasets/n-mnist

SLAYER-Loihi : Classification Benchmarks

NTIDIGITS‡



	Accuracy (%)	Parameters
Loihi	76.46	143,872
SNN[1]	93.63	351,241
LSTM[2]	91.25	610,500

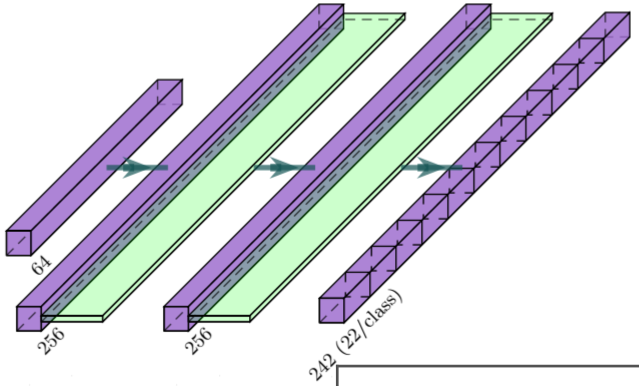
Intel Loihi measurements obtained using NxSDK v0.9.9 running on Nahuku 32. Performance results are based on testing as of Jan 2021 and may not reflect all publicly available security updates. Results may vary.

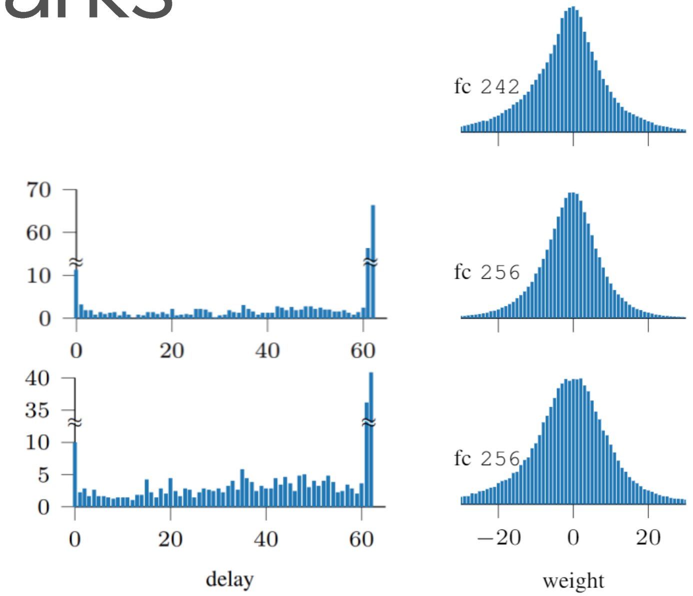
[1] Zhang et al. *Spike-train level backpropagation for training deep recurrent spiking neural networks*.
[2] Anumula et al. *Feature representations for neuromorphic audio spike streams*.

‡ NTIDIGITS dataset is available for public use under CC4.0 at http://sensors.ini.uzh.ch/databases.html

SLAYER-Loihi : Classification Benchmarks

NTIDIGITS‡





	Accuracy (%)	Parameters
Loihi	76.46	143,872
Loihi(Delay)	92.40 ± 0.19 (Best: 92.72)	144,384
SNN[1]	93.63	351,241
LSTM[2]	91.25	610,500

Loihi Cores	6
Dynamic Energy (mJ/sample)	0.51 ± 0.08
Inference Speedup (x)	33.10 ± 2.83
Sample Length (ticks)	3000
Energy Delay Product (μJs/sample)	46.96 ± 10.38

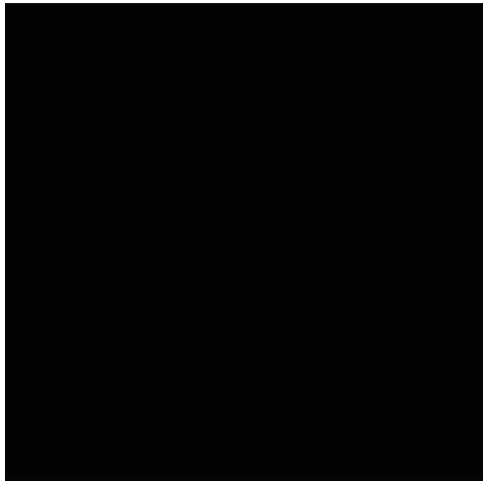
Intel Loihi measurements obtained using NxSDK v0.9.9 running on Nahuku 32. Performance results are based on testing as of Jan 2021 and may not reflect all publicly available security updates. Results may vary.

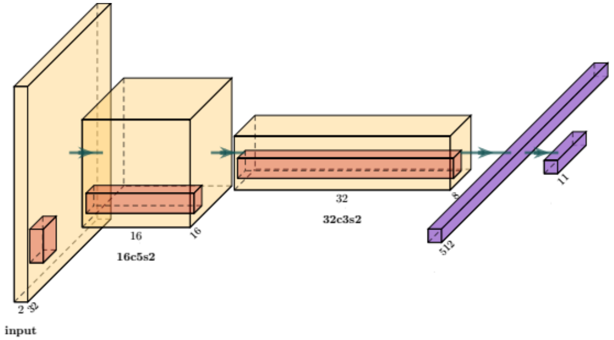
[1] Zhang et al. *Spike-train level backpropagation for training deep recurrent spiking neural networks*.
[2] Anumula et al. *Feature representations for neuromorphic audio spike streams*.

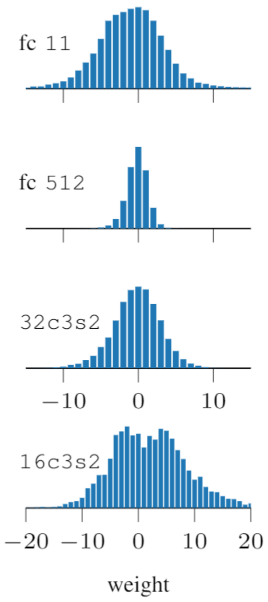
‡ NTIDIGITS dataset is available for public use under CC4.0 at http://sensors.ini.uzh.ch/databases.html

SLAYER-Loihi : Classification Benchmarks

DVS Gesture‡ (Resource)







	Accuracy (%)	Parameters
Loihi(Res.)	95.98 ± 0.21 (Best: 96.59)	1,059k
SNN[1]	95.54 ± 0.16	1,246k
Overall[2]	< 97.75	2,117k

Loihi Cores	18
Dynamic Energy (mJ/sample)	9.88 ± 1.40
Inference Speedup (x)	11.13 ± 1.24
Sample Length (ticks)	1500
Energy Delay Product (μJs/sample)	1340.86 ± 197.22

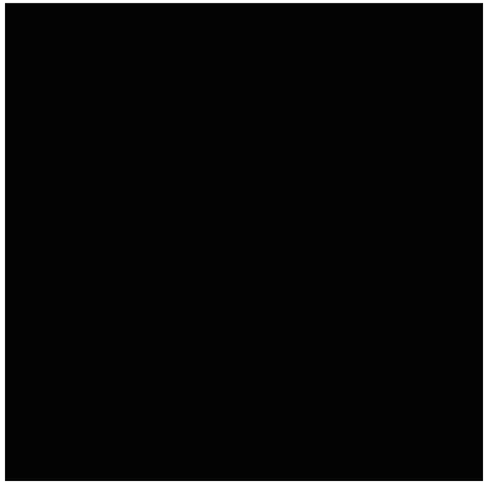
Intel Loihi measurements obtained using NxSDK v0.9.9 running on Nahuku 32. Performance results are based on testing as of Jan 2021 and may not reflect all publicly available security updates. Results may vary.

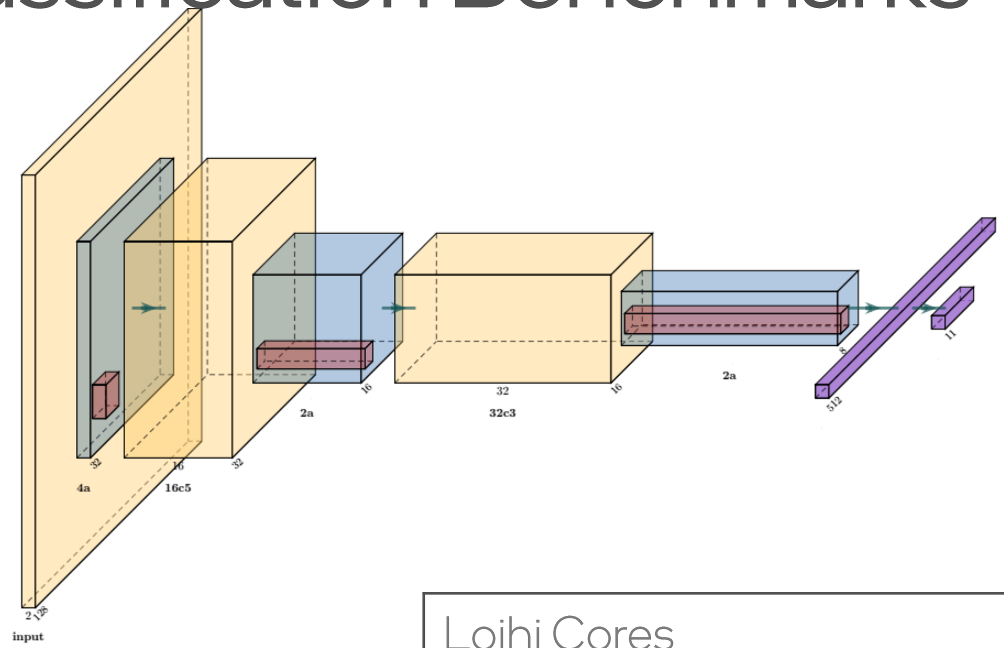
[1] Kaiser et al. *Synaptic plasticity dynamics for deep continuous local learning (DECOLLE).*
[2] Ghosh et al. *Spatiotemporal Filtering for Event-Based Action Recognition.*

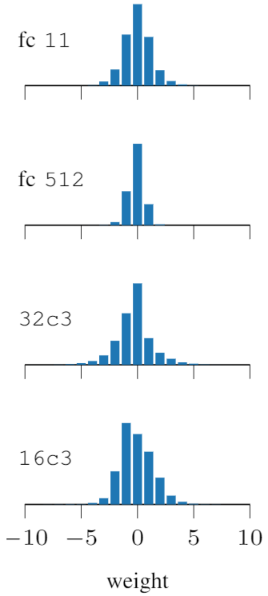
‡ DVS Gesture dataset is available for public use under CC4.0 at https://www.research.ibm.com/dvsgesture/

SLAYER-Loihi : Classification Benchmarks

DVS Gesture‡ (Performance)







	Accuracy (%)	Parameters
Loihi(Res.)	95.98 ± 0.21 (Best: 96.59)	1,059k
Loihi(Perf.)	96.44 ± 1.09 (Best: 97.73)	1,066k
SNN[1]	95.54 ± 0.16	1,246k
Overall[2]	< 97.75	2,117k

Loihi Cores	79
Dynamic Energy (mJ/sample)	15.42 ± 2.66
Inference Speedup (x)	12.71 ± 3.95
Sample Length (ticks)	1500
Energy Delay Product (μJs/sample)	2071.34 ± 1053.81

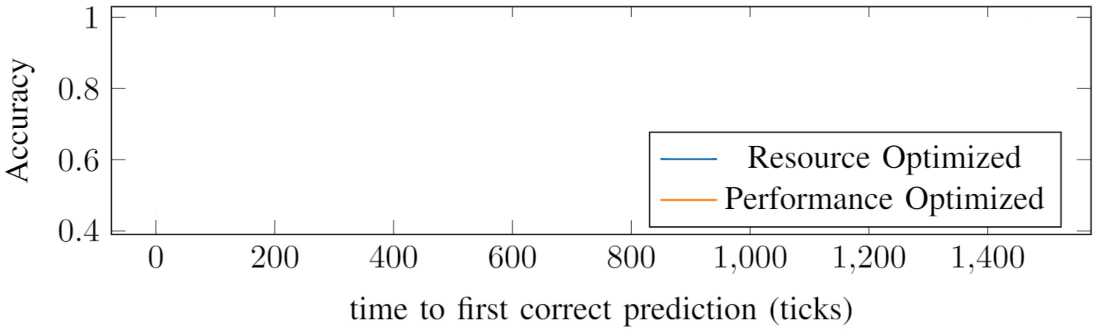
Intel Loihi measurements obtained using NxSDK v0.9.9 running on Nahuku 32. Performance results are based on testing as of Jan 2021 and may not reflect all publicly available security updates. Results may vary.

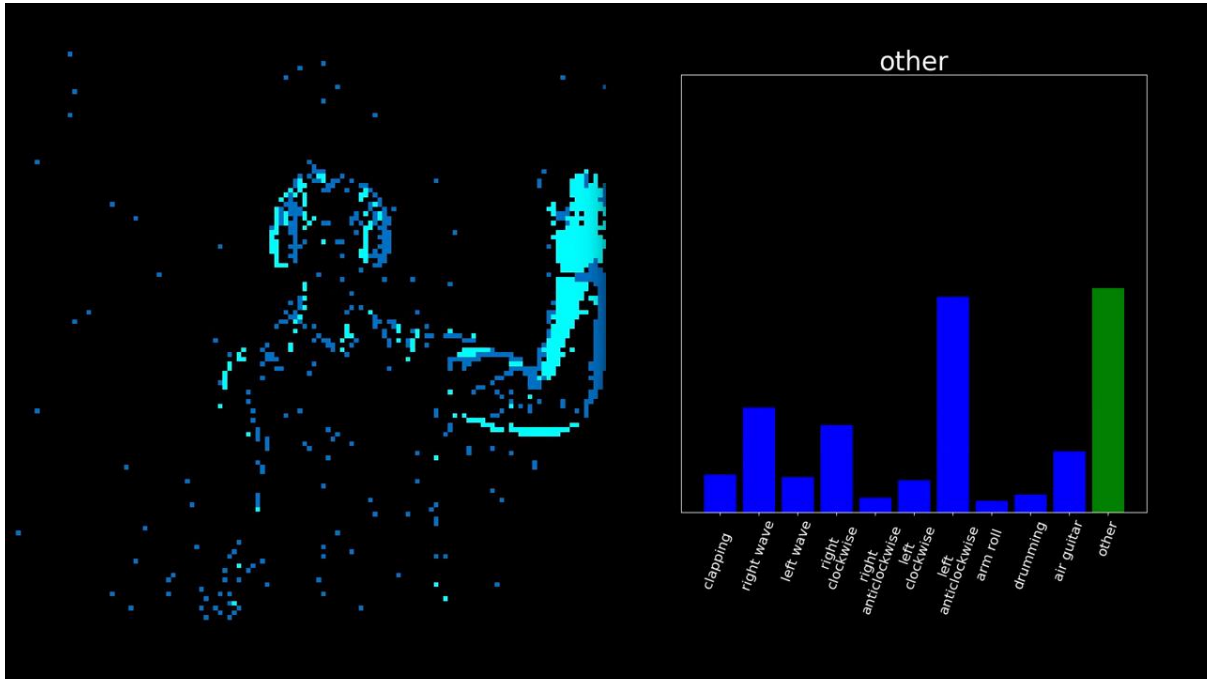
[1] Kaiser et al. *Synaptic plasticity dynamics for deep continuous local learning (DECOLLE).*
[2] Ghosh et al. *Spatiotemporal Filtering for Event-Based Action Recognition.*

‡ DVS Gesture dataset is available for public use under CC4.0 at https://www.research.ibm.com/dvsgesture/

SLAYER-Loihi : Classification Benchmarks

DVS Gesture‡: Inference Latency





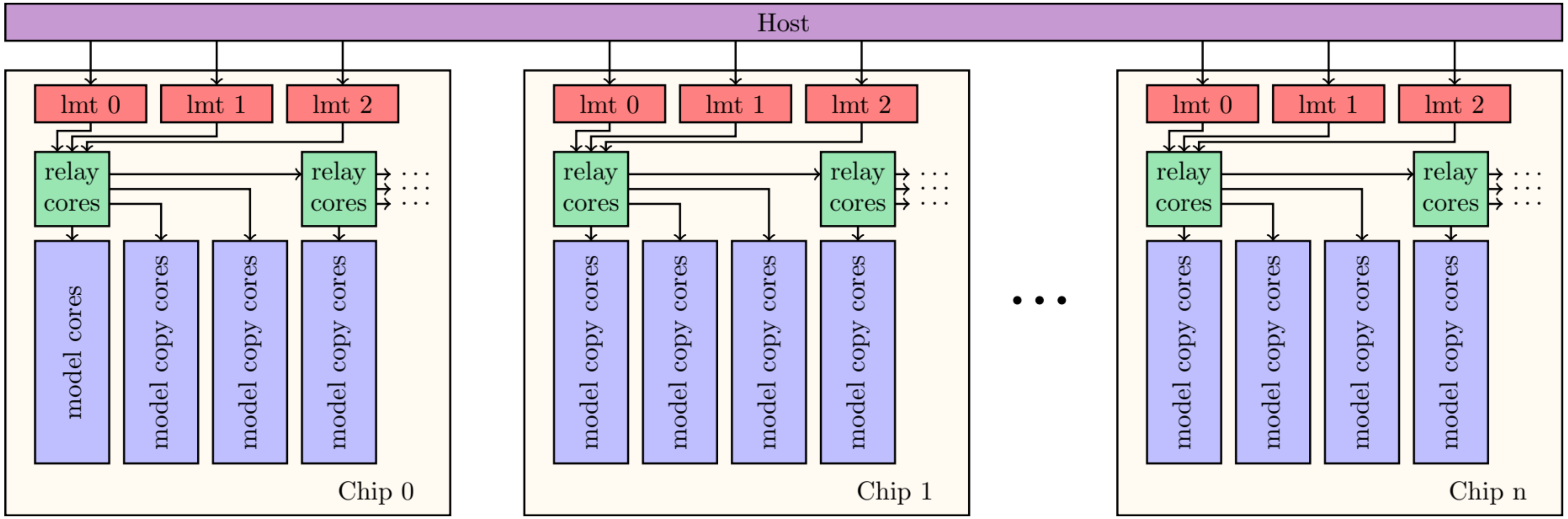
Average Latency:	65.90 ms
Dynamic Power (Loihi):	9.95 ± 1.61 mW†
Dynamic Power(DAVIS240C):	5mW*

* iniVation DAVIS 240C performance numbers obtained from published specifications.
† Intel Loihi measurements obtained using NxSDK v0.9.9 running on Nahuku 32. Performance results are based on testing as of Jan 2021 and may not reflect all publicly available security updates. Results may vary.

‡ DVS Gesture dataset is available for public use under CC4.0 at https://www.research.ibm.com/dvsgesture/

Energy Benchmark

- Typically, the networks don't fill the entire board.
- Network replication tools for accurate energy estimation.
- More details on benchmarking in our next talk.



Legal Information

Performance varies by use, configuration and other factors. Learn more at www.Intel.com/PerformanceIndex.

Performance results are based on testing as of dates shown in configurations and may not reflect all publicly available updates. See backup for configuration details. No product or component can be absolutely secure.

Your costs and results may vary.

Results have been estimated or simulated.

Intel technologies may require enabled hardware, software or service activation.

Intel does not control or audit third-party data. You should consult other sources to evaluate accuracy.

Intel disclaims all express and implied warranties, including without limitation, the implied warranties of merchantability, fitness for a particular purpose, and non-infringement, as well as any warranty arising from course of performance, course of dealing, or usage in trade.

Thank You!