A 28-nm Convolutional Neuromorphic Processor Enabling Online Learning with Spike-Based Retinas

Charlotte Frenkel*, Jean-Didier Legat and David Bol

ICTEAM Institute, Université catholique de Louvain, Belgium

*Now with Institute of Neuroinformatics, UZH and ETH Zürich, Switzerland
charlotte@ini.uzh.ch

2020 IEEE International Symposium on Circuits and Systems
Virtual, October 10-21, 2020

Background

Neuromorphic IC design strategies

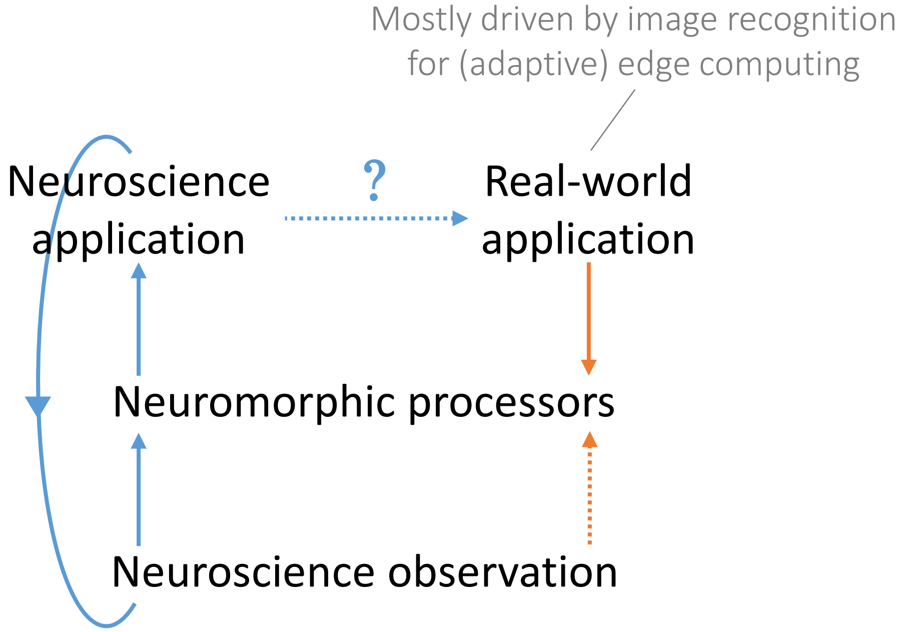
Bottom-up design
(exp. platforms)

Large-scale silicon integration

↑

Neuron & synapse building blocks

Versatility / efficiency tradeoff



Top-down design
(accelerators)

Algorithms

↓

Silicon integration

Accuracy / efficiency tradeoff

Top-down previous work with on-chip learning:

1. SNNs (LCAs), at the expense of reduced accuracy
(e.g., [Kim, VLSI-C, 2015], [Buhler, VLSI-C, 2017])
2. BNNs, at the expense of frame-based processing
(e.g., [Chen, VLSI-C, 2018], [Park, ISSCC, 2019])

Best of both worlds, from sensing to processing?

Outline

- SPOON – Proposed convolutional neuromorphic processor enabling online learning with spike-based retinas

- Conclusion and perspectives

Outline

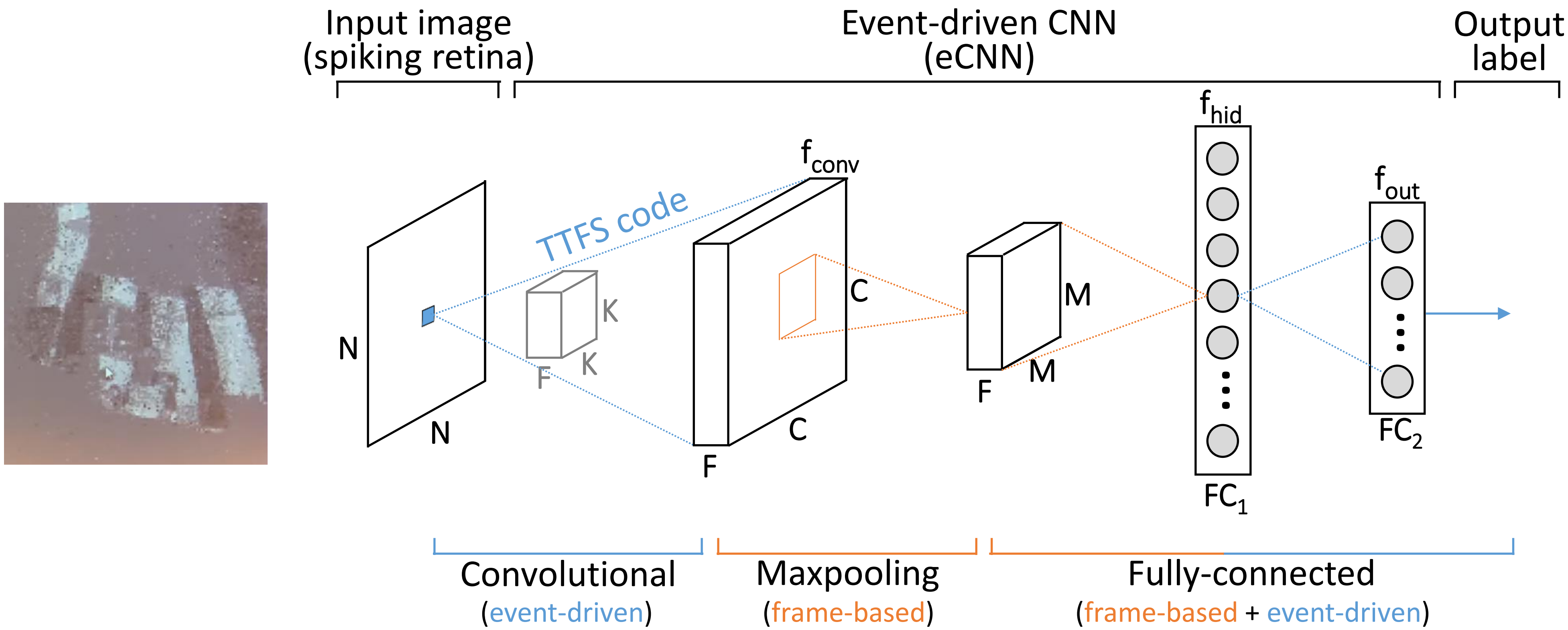
- SPOON – Proposed convolutional neuromorphic processor
  enabling online learning with spike-based retinas

  Dataflow
  Algorithm
  Architecture
  Implementation
  Benchmarking

- Conclusion and perspectives

Leveraging sparsity, event-driven sensing and maximizing data reuse

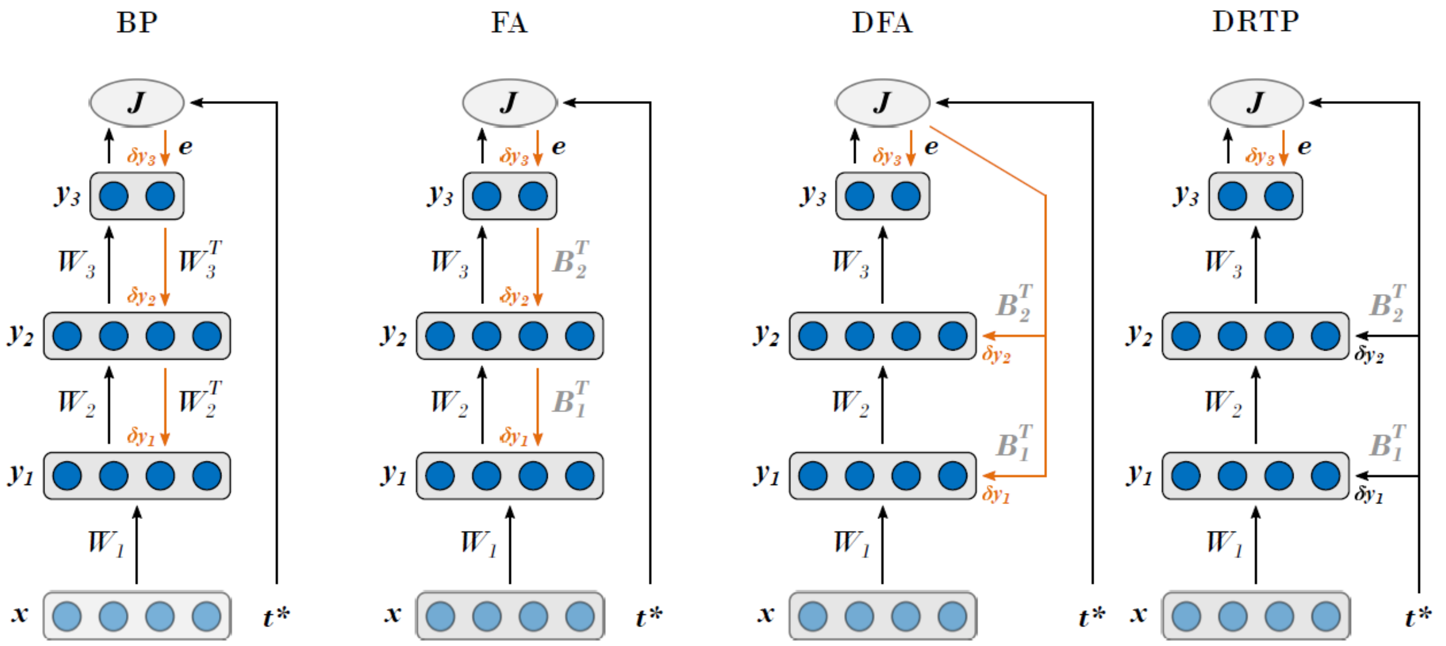
From sensing to processing



Learning-enabled

Learning algorithm – Direct random target projection (DRTP)

Releasing the weight transport and update locking of backprop



	BP	FA	DFA	DRTP
δy_k	$\frac{\partial J}{\partial y_k} = W_{k+1}^T \delta z_{k+1}$	$B_k^T \delta z_{k+1}$	$B_k^T e$	$\boxed{B_k^T t^*}$
Weight-transport-free	×	✓	✓	✓
Update-unlocked	×	×	×	✓

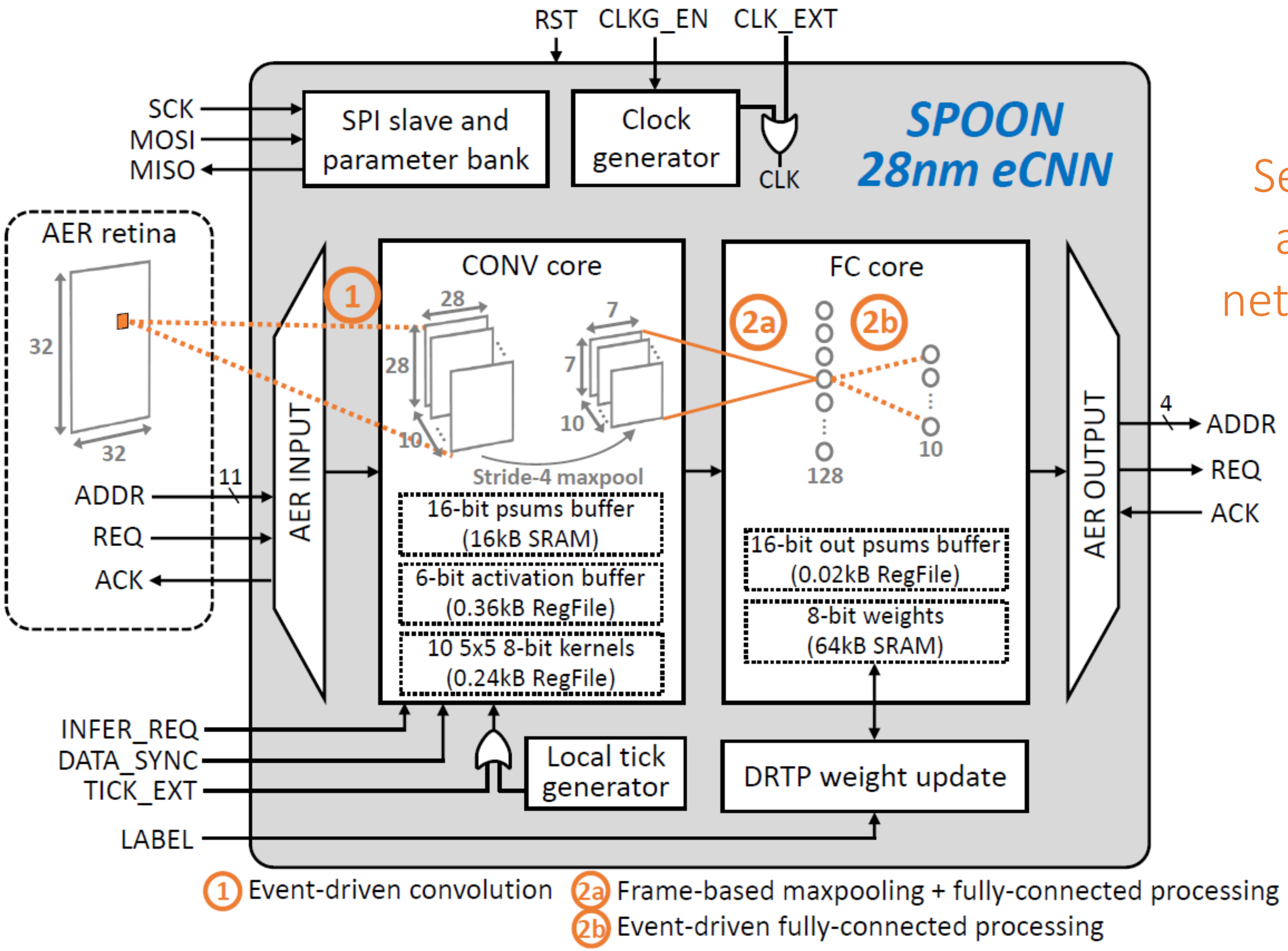
Feedforward local training

↘ Computational and memory cost ↘

[Lillicrap, *Nat. Comms.*, 2016] [Nokland, *NeurIPS*, 2016] [Frenkel&Lefebvre, arXiv, 2019]

Architecture of SPOON

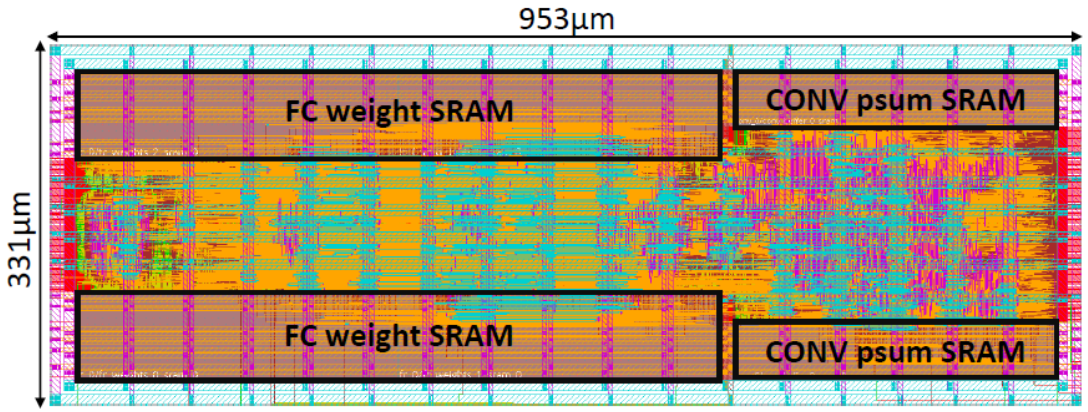
SPOON – A Spiking Online-Learning Convolutional Neuromorphic Processor



See the paper for circuit and architectural details of the network and of the DRTP block.

SPOON implementation results

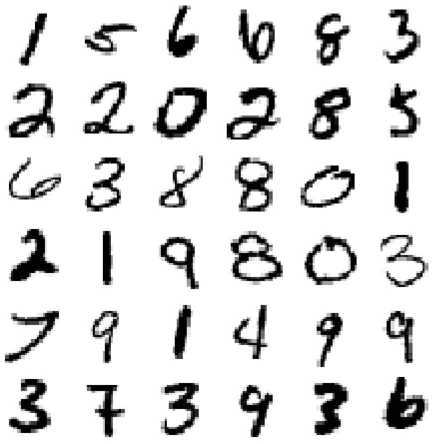
Specifications and pre-silicon performance metrics



Technology	28nm FDSOI CMOS
Implementation	Digital
Area	0.32mm^2 (0.26mm^2 excl. rails)
Topology	C5×5@10–FC128–FC10
Online learning	Stochastic DRTP, 8-bit weights
Time constant	Biological to accelerated
Supply voltage	0.6V – 1.0V
Max. clock frequency	150MHz
Leakage power	61μW at 0.6V
Energy for CONV core	1.7nJ/event at 0.6V
Energy for FC core	55nJ/inference at 0.6V
Online learning overhead	16.8% in power, 11.8% in area

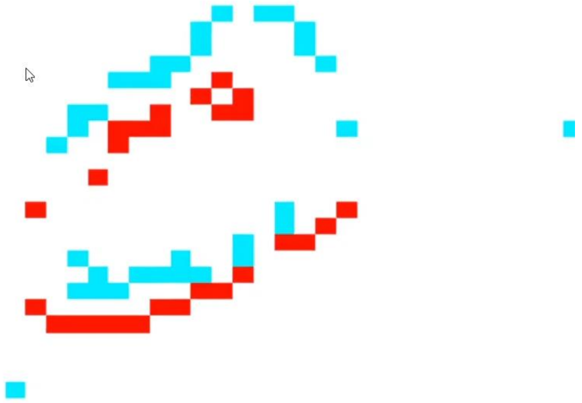
SPOON benchmarking

Temporal code for edge detection also demonstrated on N-MNIST



MNIST (TTFS-encoded):

- off-chip BP training: 97.5% test-set accuracy
- on-chip online DRTP training: 92.8% accuracy (one epoch)
  95.3% accuracy (100 epochs)
- energy efficiency: 313nJ/inference

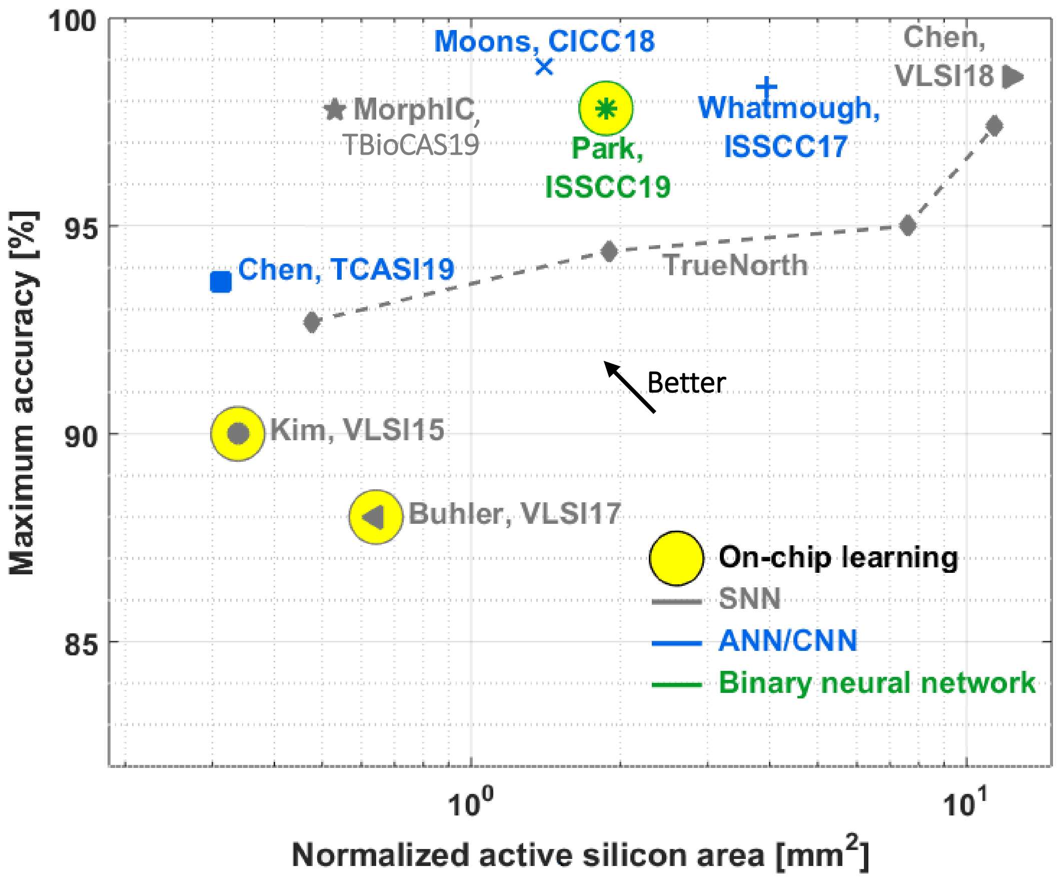


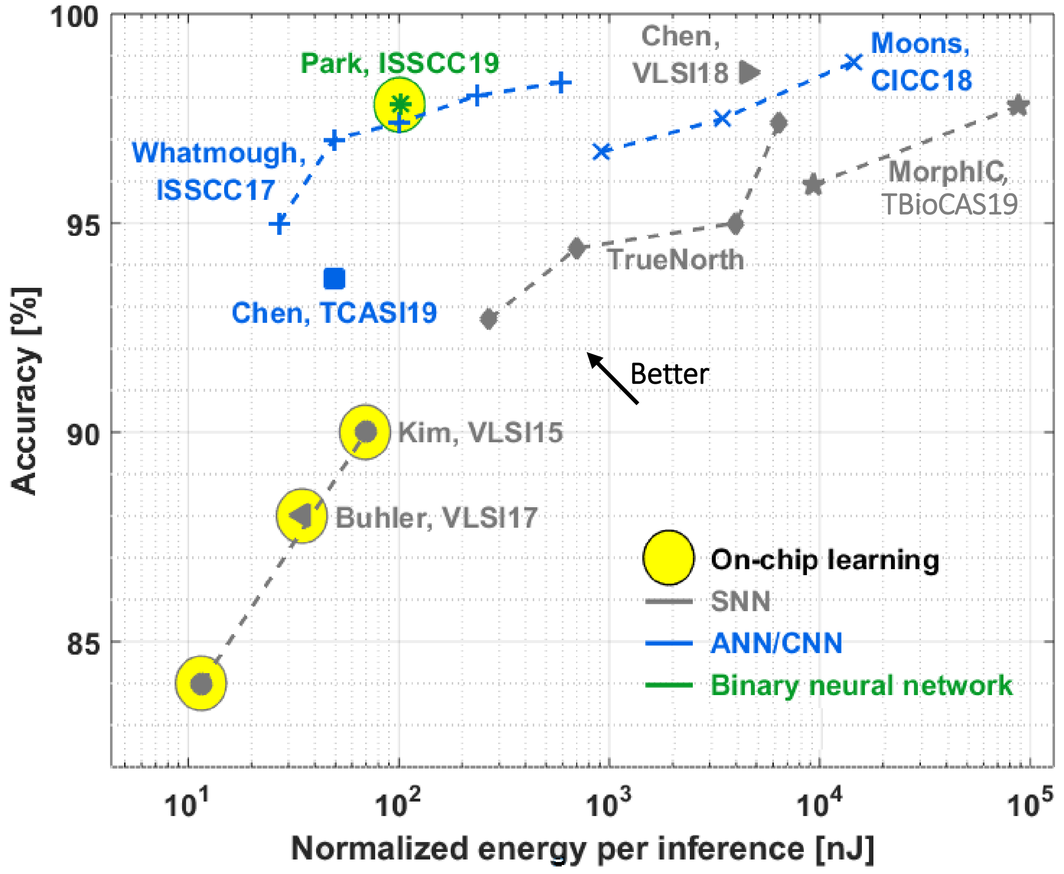
N-MNIST (only the first saccade, only the first event of each pixel):

- off-chip BP training: 93.8% test-set accuracy
- on-chip online DRTP training: 90.2% accuracy (one epoch)
  93.0% accuracy (100 epochs)
- energy efficiency: 665nJ/inference

SPOON benchmarking

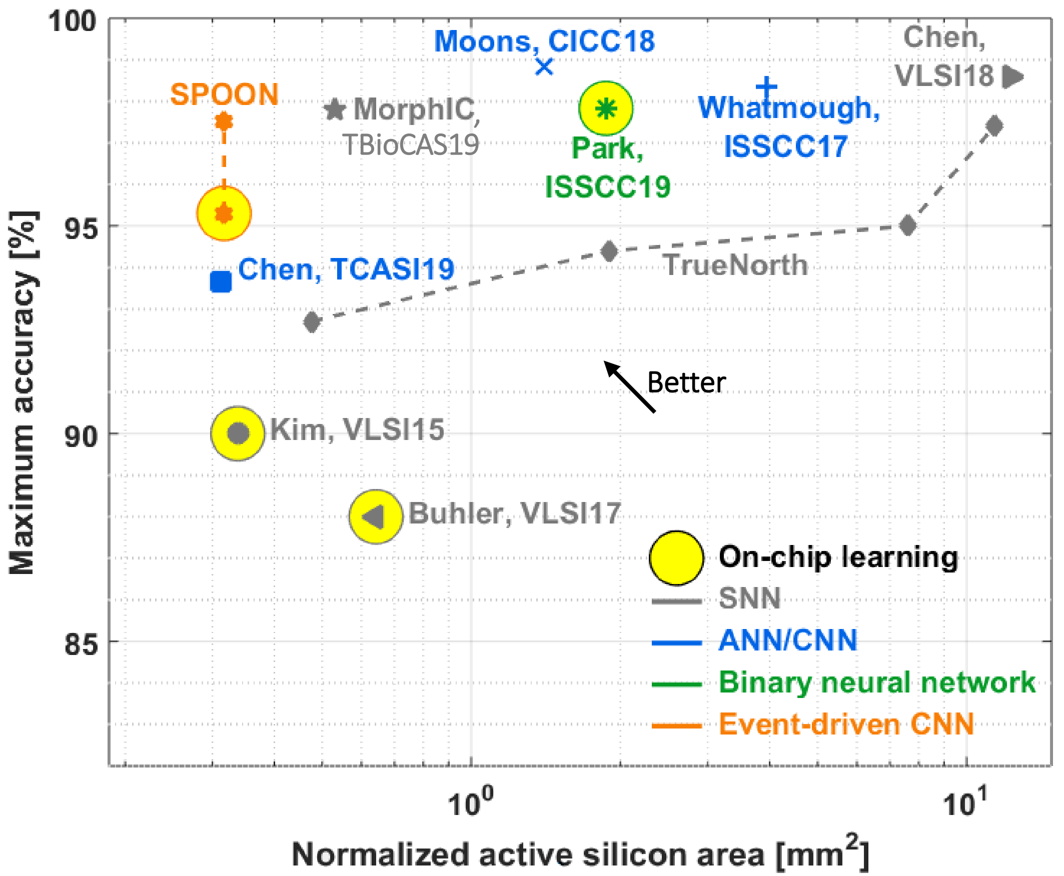
Against state-of-the-art SNNs, ANNs, CNNs and BNNs on MNIST

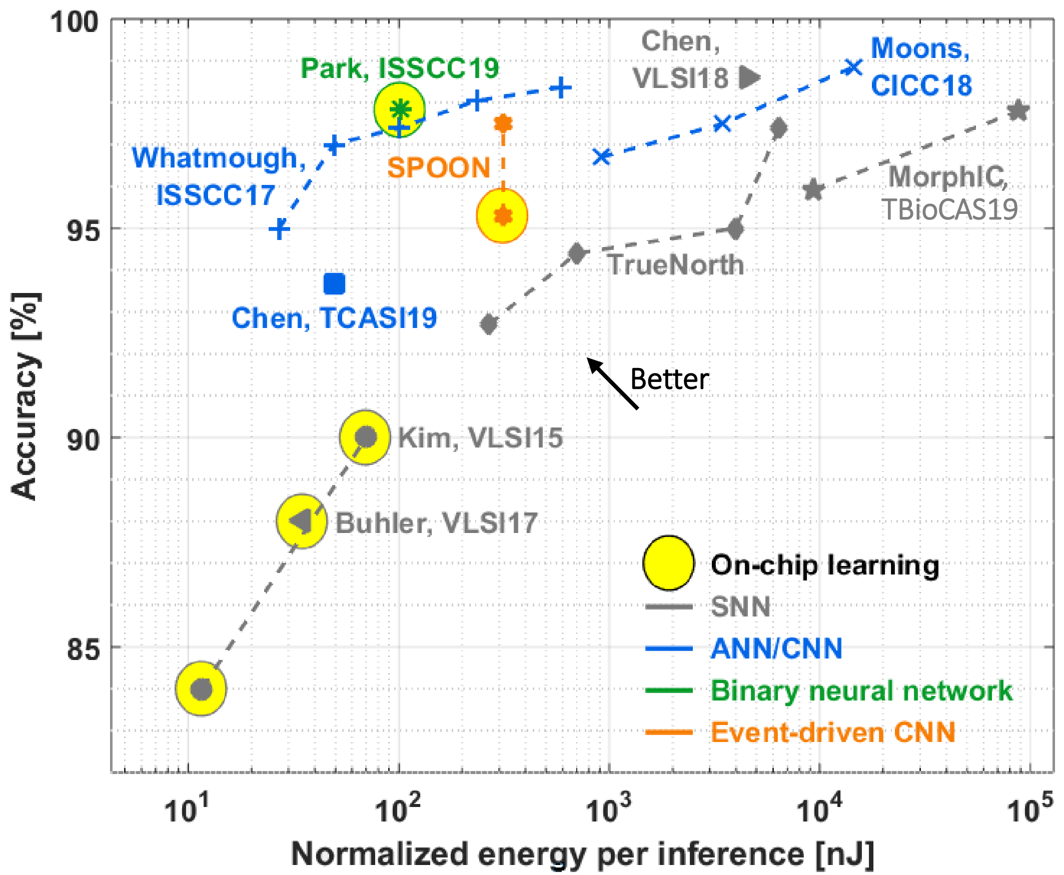




SPOON benchmarking

Against state-of-the-art SNNs, ANNs, CNNs and BNNs on MNIST





Outline

- SPOON – Proposed convolutional neuromorphic processor enabling online learning with spike-based retinas

- Conclusion and perspectives

  Summary of the key messages

Conclusion

Background

Neuroscience application ? Real-world application

Image recognition for adaptive edge computing

DRTP

Maximize data reuse (combine event-driven and frame-based)

Neuromorphic processors

Event-driven / sparse sensing, temporal code

Neuroscience observation

Proposed SPOON eCNN

Only SPOON allows reaching the efficiency of ANN/CNN/BNN accelerators while enabling online learning with event-based sensors.

Thank you!

Further resources:

The corresponding ISCAS paper (more info on the architecture)

The DRTP preprint: https://arxiv.org/pdf/1909.01311.pdf

DRTP PyTorch code open-sourced on https://github.com/chfrenkel