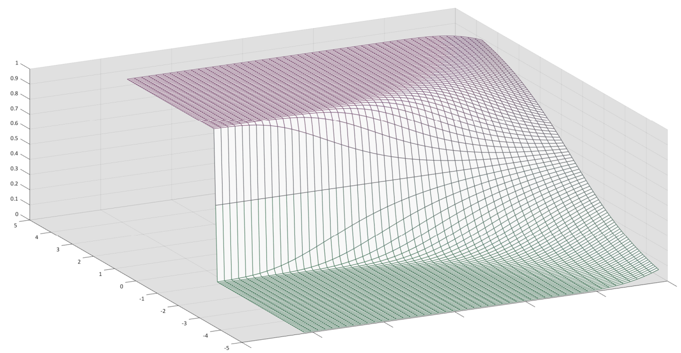
SAND2017-8513C

MLDL
Conference 2017



Steep Deep Spiking Networks

Ryan Dellana (09315)
William Severa (01462), Stephen Verzi (08832), Mike Smith (05851),
Fred Rothganger (01462), Conrad James (05228), Brad Aimone (01462),
Craig Vineyard (01462)

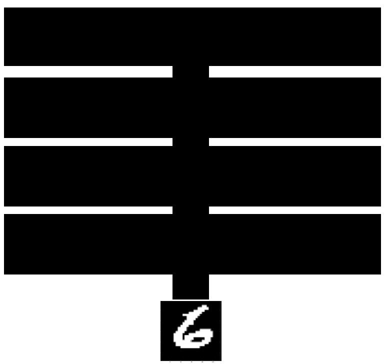
Sandia National Laboratories is a multi-mission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525. . SAND NO. XXXXXXXXXX

Motivation

- Run existing deep networks on spiking neuromorphic hardware

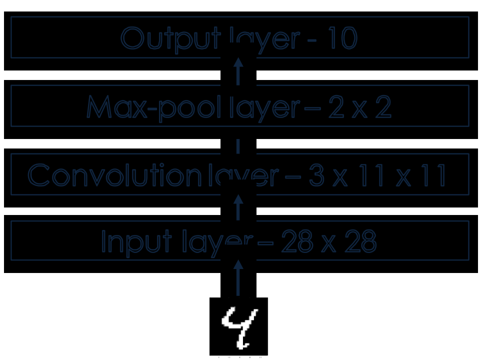
Non-convolutional

0 0 0 0 0 0 1 0 0 0

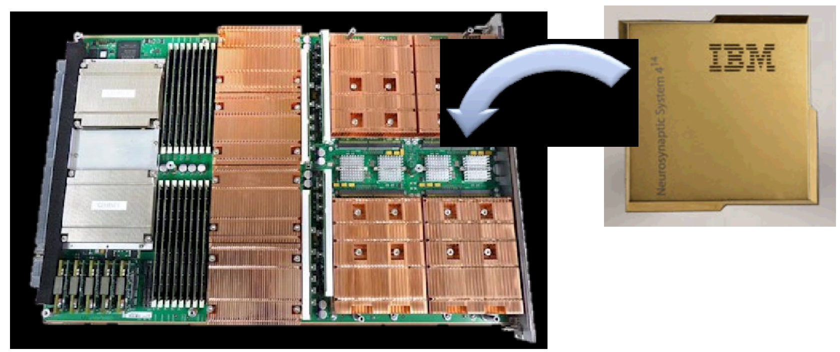


Convolutional

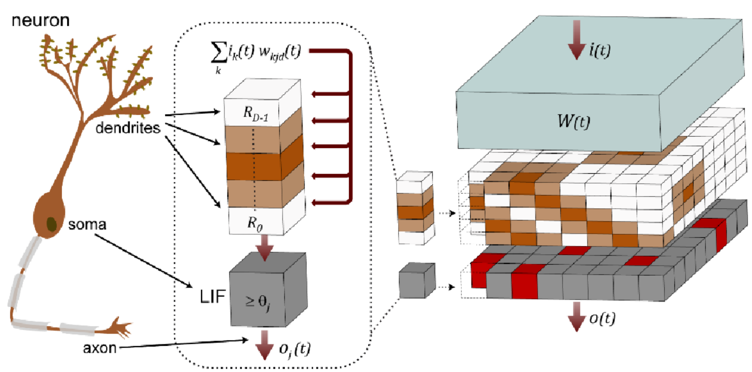
0 0 0 0 1 0 0 0 0 0



IBM TrueNorth



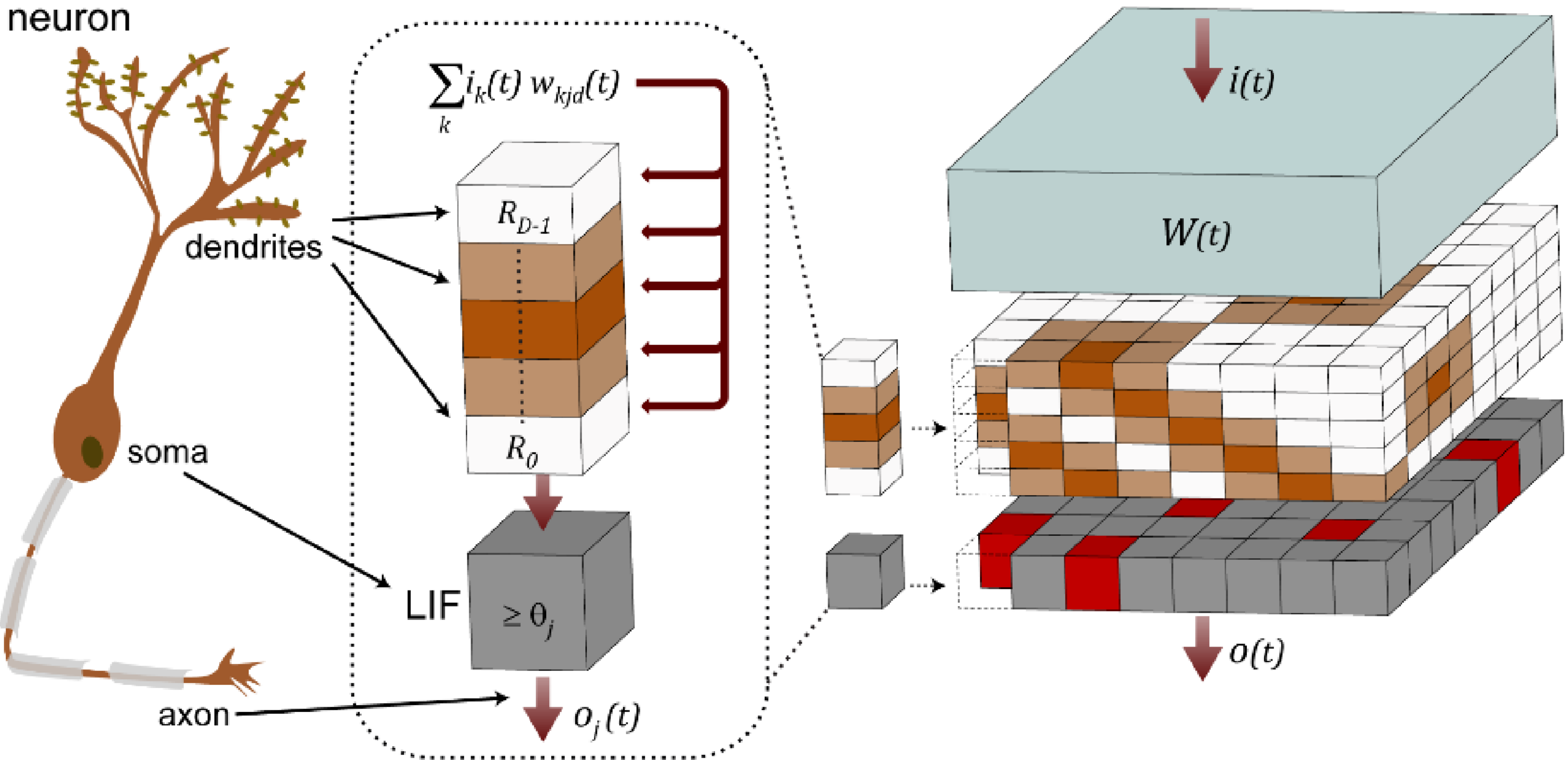
Spiking Temporal Processing Unit (STPU)



Collaboration with Lewis Rhodes Labs (D Follett)

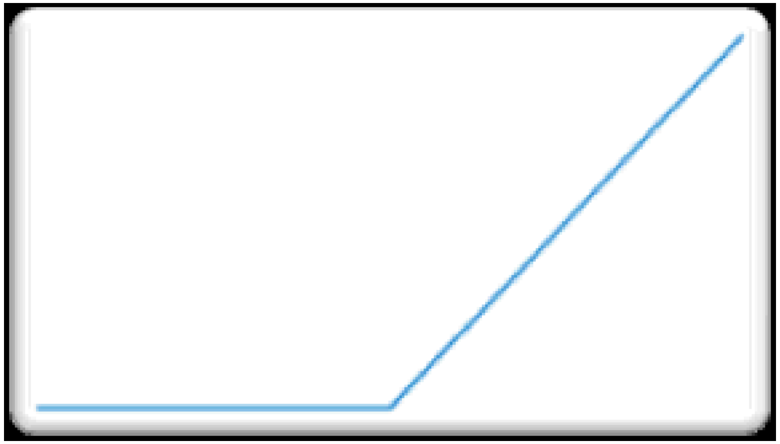
Spiking neural network architecture

- Spiking Temporal Processing Unit (STPU)

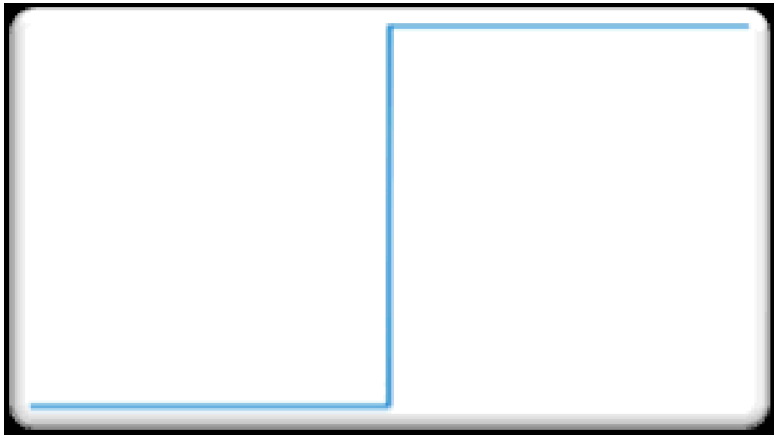


Spiking network training challenges

- Spiking neurons promise size, weight and power (SWaP) as well as throughput advantages
- Current state-of-the-art deep networks are non-spiking trained using backpropagation
- Methods for training spiking neural networks (SNNs) still open for debate



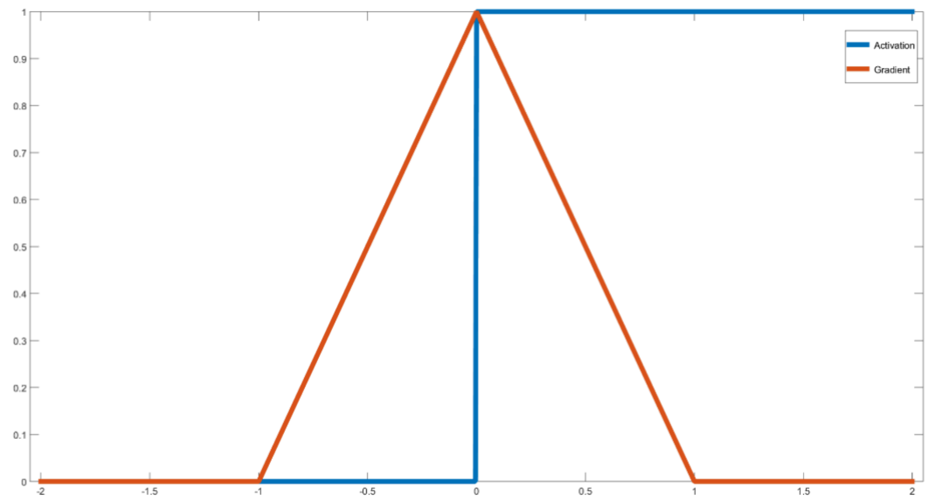
Rectified Linear



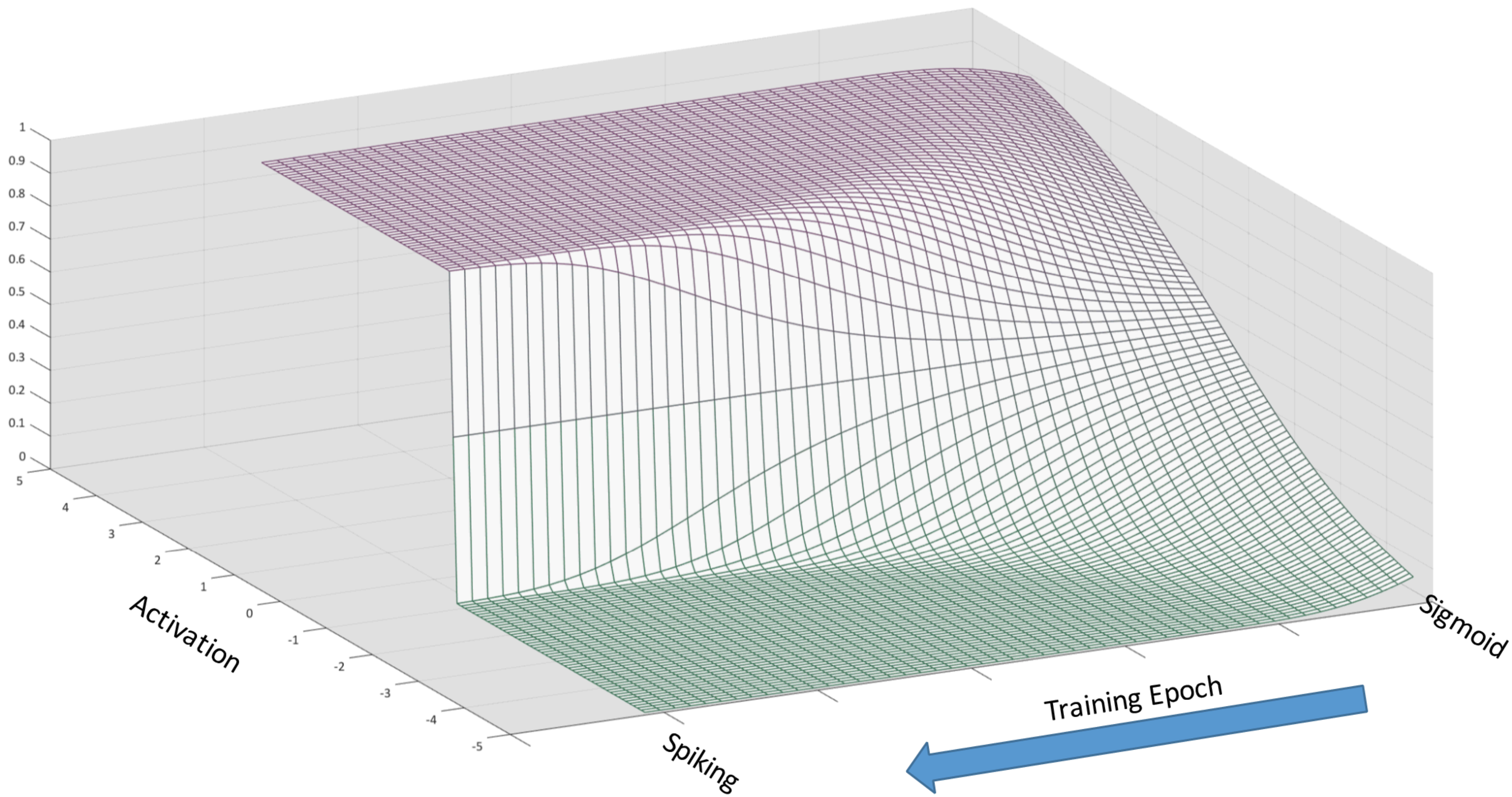
Spiking Neuron

Other training methods for SNNs

- Spiking-Native Training
  - Spike Timing Dependent Plasticity
  - Linear discriminator or SVM on activity
  - Evolutionary Algorithms
- Spiking-Conversion Training
  - Rate-coded or temporally coded spiking neurons
  - ‘Real-world Use’ spiking benefits may be lost
  - Synthetic gradients
  - Requires
    - Modified workflow
    - Modified networks
    - Hardware specific



Activation functions converge to spiking during training



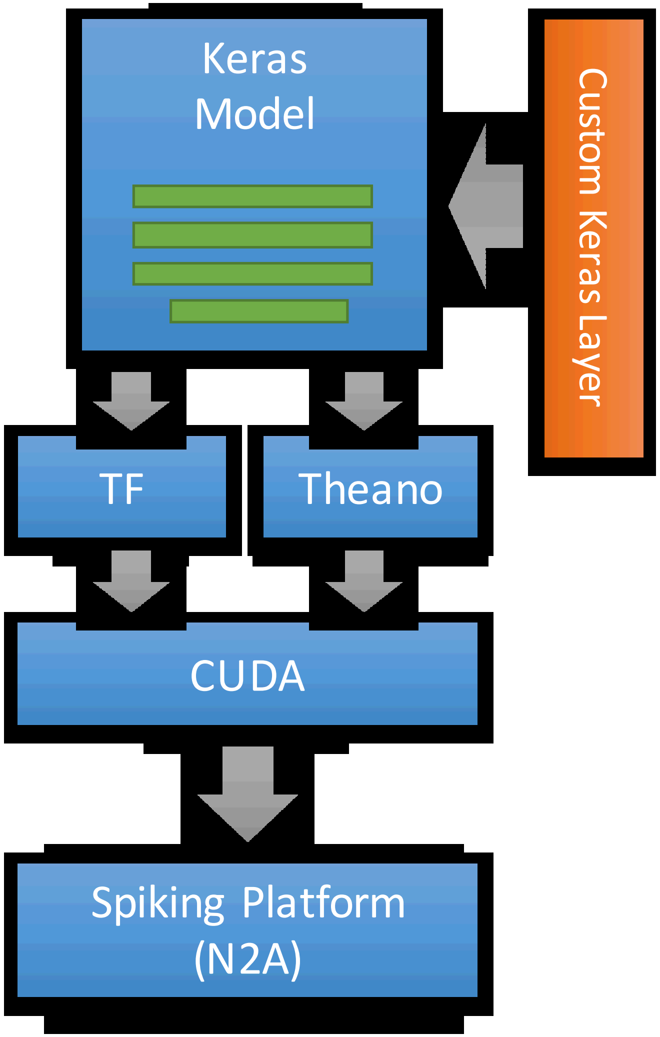
Algorithmic approach

- Augment existing deep network with extra sigmoid layer (one-to-one) for each deep layer to be operated using LIF neurons on neuromorphic hardware

- As network is trained sharpen sigmoid neurons to the point where they essentially become (spike-like) step functions
- Assumption: we can implement deep activation function using LIF neurons

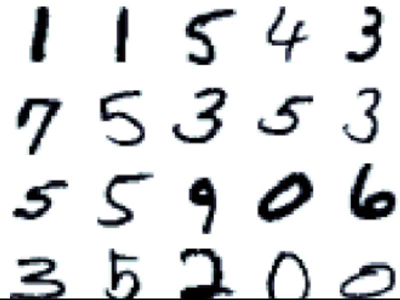
Advantages

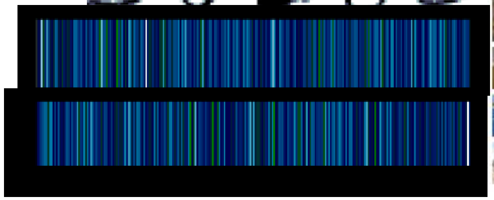
- No restrictions on network model
- Integrates with existing deep learning/neural network libraries
- Immediately useable on hardware after training
- Independent from hardware destination
- 1 Frame = 1 Time-step
- Compatible with multiple origin activation functions:
- Sigmoid, ReLu, Leaky ReLu, etc.
- Software implemented via a custom keras activation layer

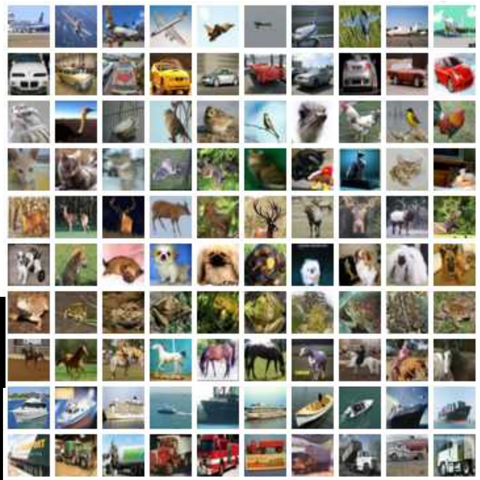


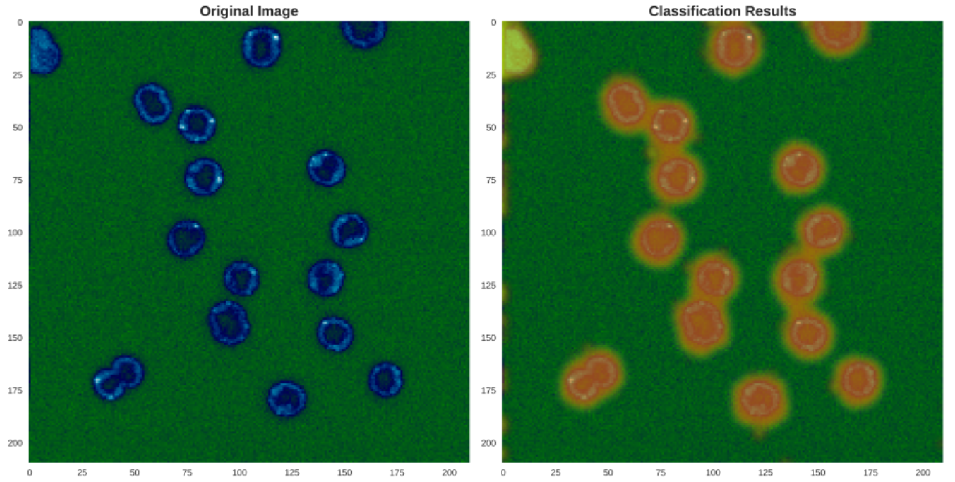
Datasets

- MNIST
  - 28x28 images of digits
  - 10 classes
- Cifar-10
  - 32x32 images of common objects
  - 10 classes
- Hyperspectral Dataset
  - 512 dimension hyperspectral pixels
  - 3 classes



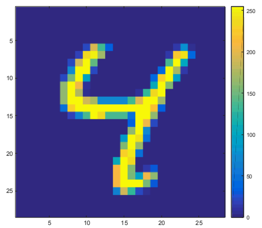
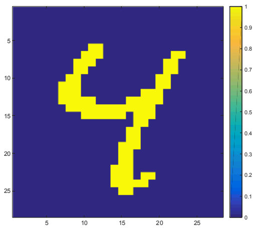






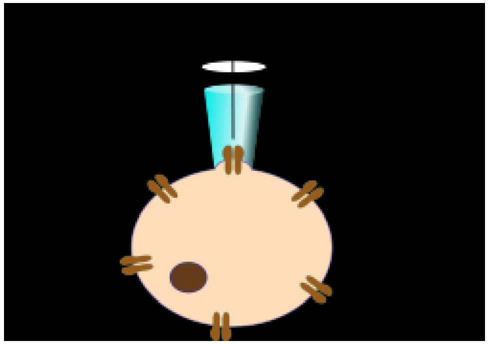
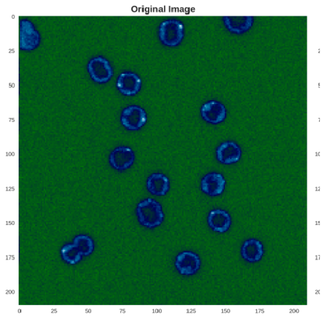
Data conversion

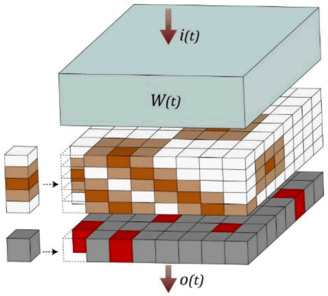
- We need to convert inputs to spikes
  - Binarized MNIST

  - Image contours


  - Direct current injection



Results

- MNIST

STPU DNN	Input 784	Dense 512	Dense 512	Output 10		
STPU CNN	Input 784	Conv 3x11x11	MaxPool 2x2	Output 10		
CNN Restricted	Input 784	Conv 8x3x3	Conv 8x3x3	Dense 100	Dense 100	Output 10
CNN Unrestricted	Input 784	Conv 32x10x10	Conv 32x10x10	Dense 100	Dense 100	Output 10

STPU-bound networks are size-restricted

	STPU DNN	STPU CNN	CNN Restricted	CNN Unrestricted	TrueNorth
Original Acc	97.3	97.1	98.5	99.1	
Spiking Acc	97.3	94.9	97.3	98.5	99.4*

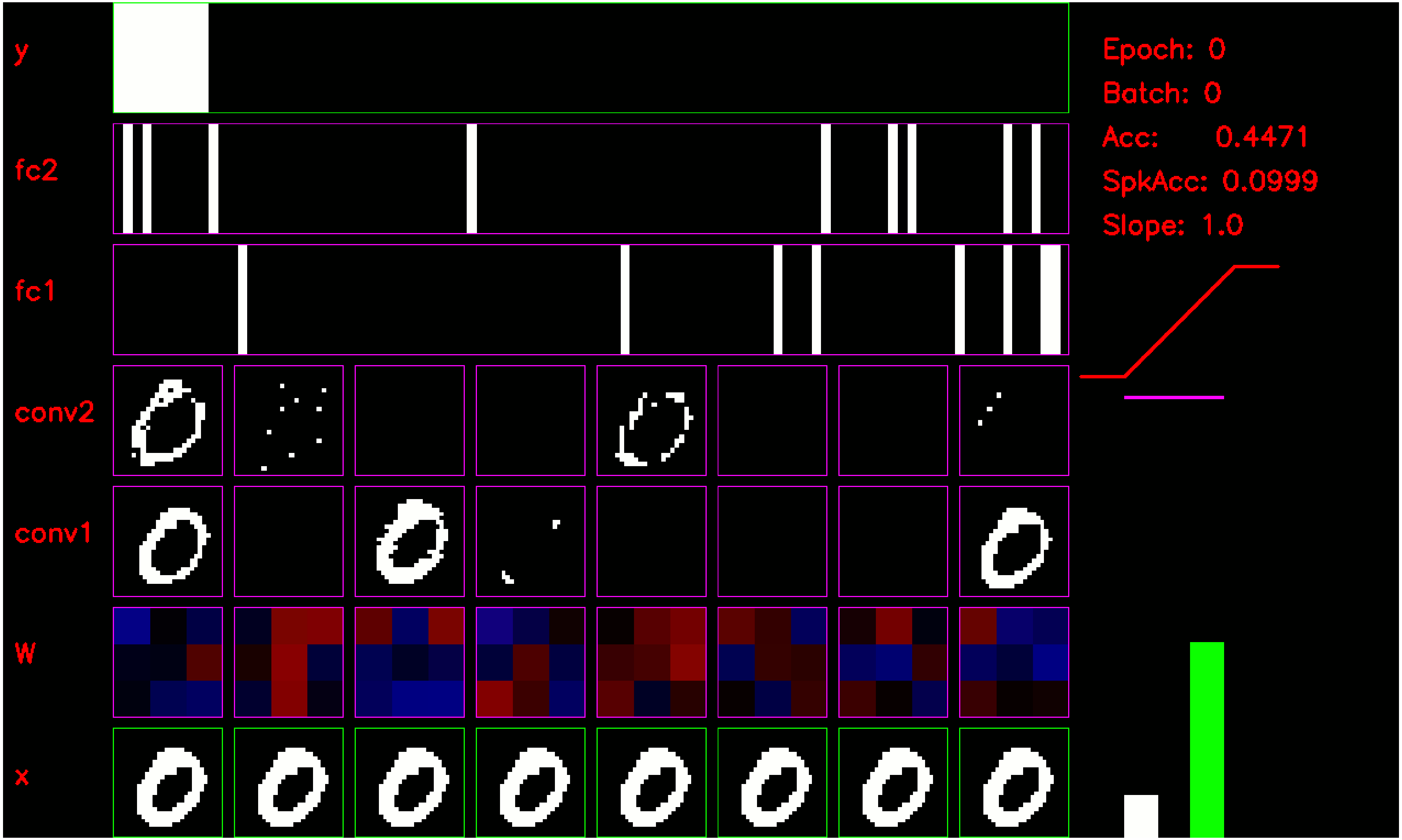
*TrueNorth 4uJ/frame network 95% accurate

Results

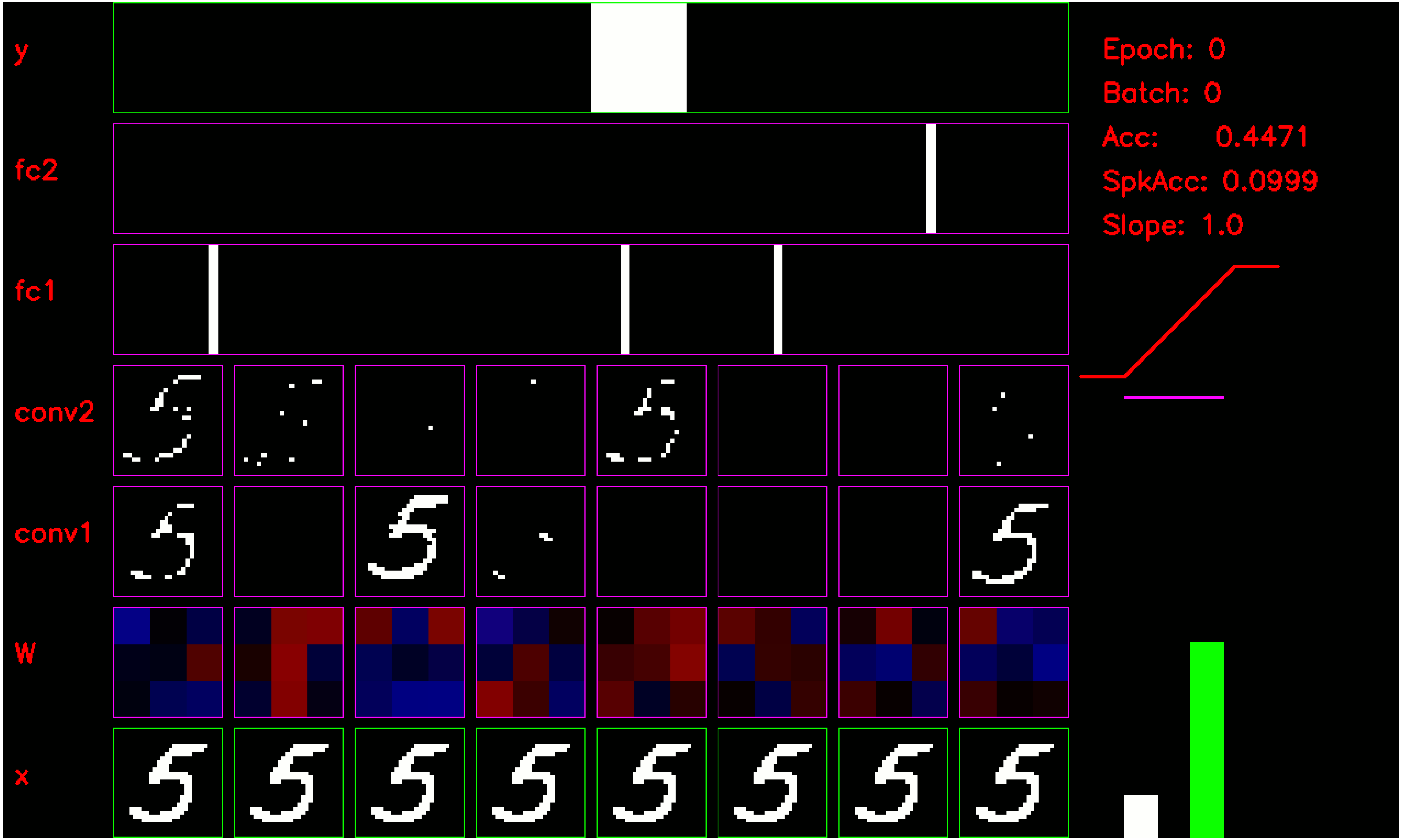
- Cifar-10
  - Cifar-10 spiking vs non-spiking has ~1-2% accuracy degradation
  - Challenge converting images into spikes
  - Edge detection forces loss of information content and accuracy

- Hyperspectral cell dataset
  - Accuracy degradation 98.2% to 97.7%
  - Raw values ‘current injection’ into first layer
  - Network is STPU-compatible

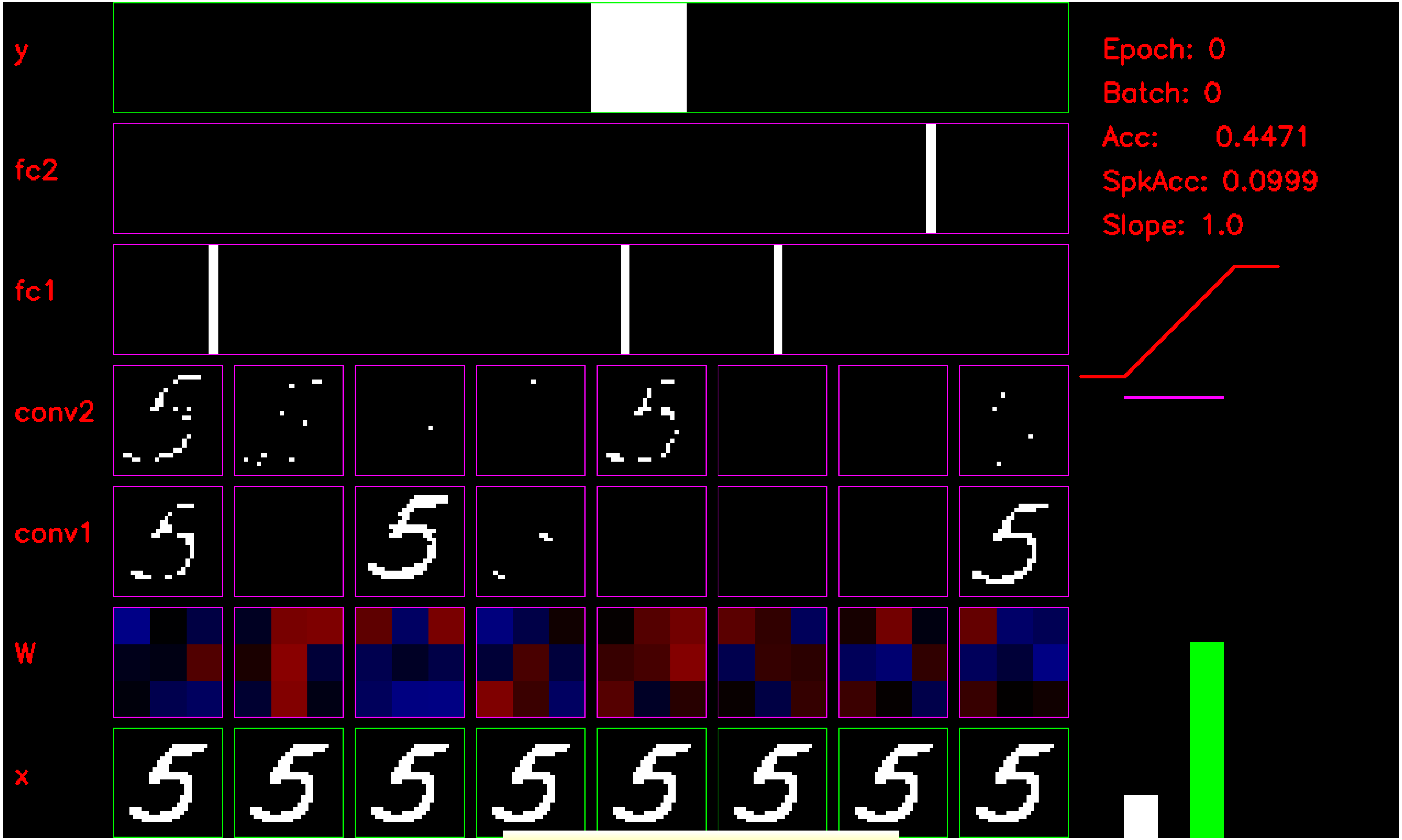
Visualizing deep spiking CNNs



Visualizing deep spiking CNNs



Visualizing deep spiking CNNs

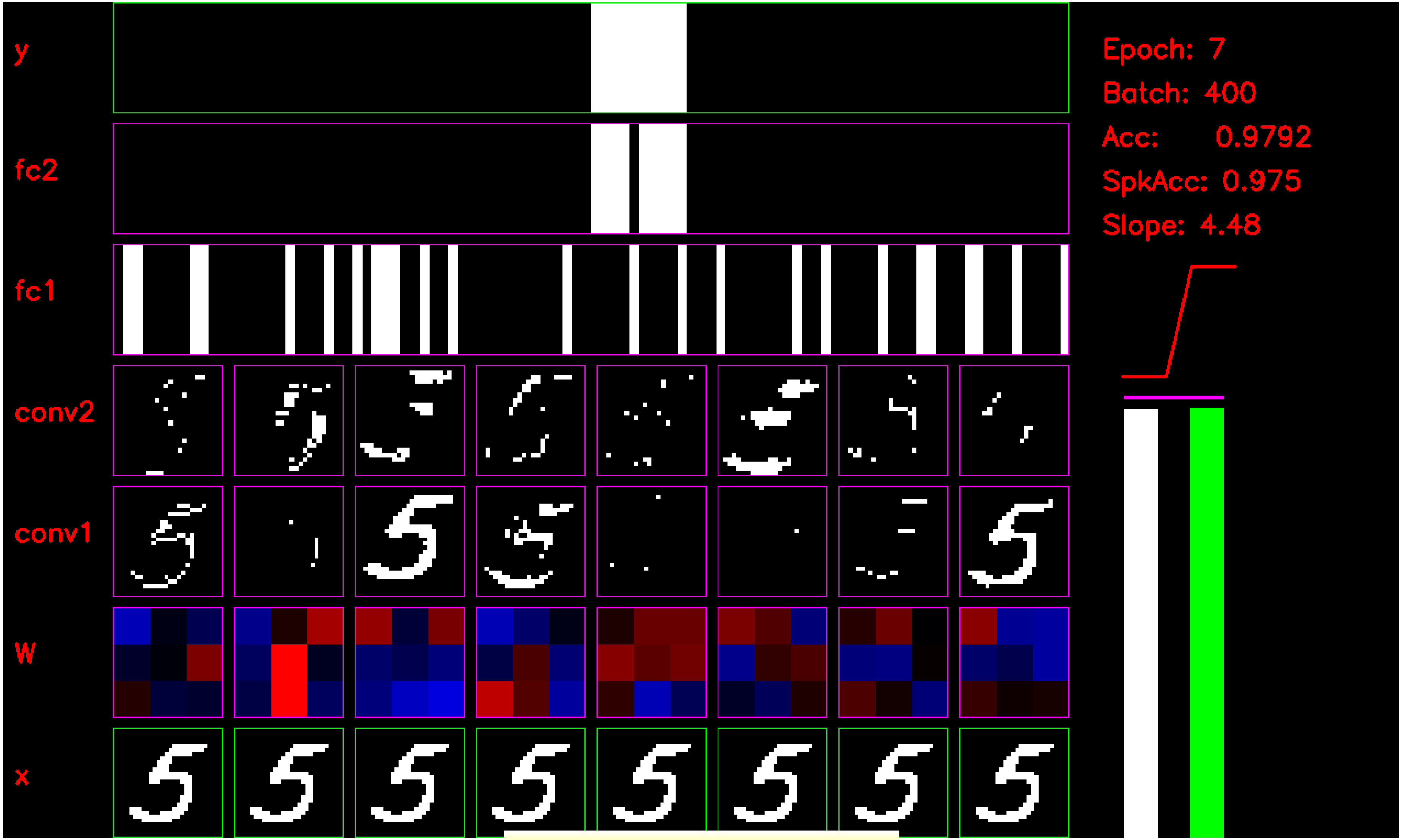


Undertrained network

Visualizing deep spiking CNNs

Overtrained network

Visualizing deep spiking CNNs

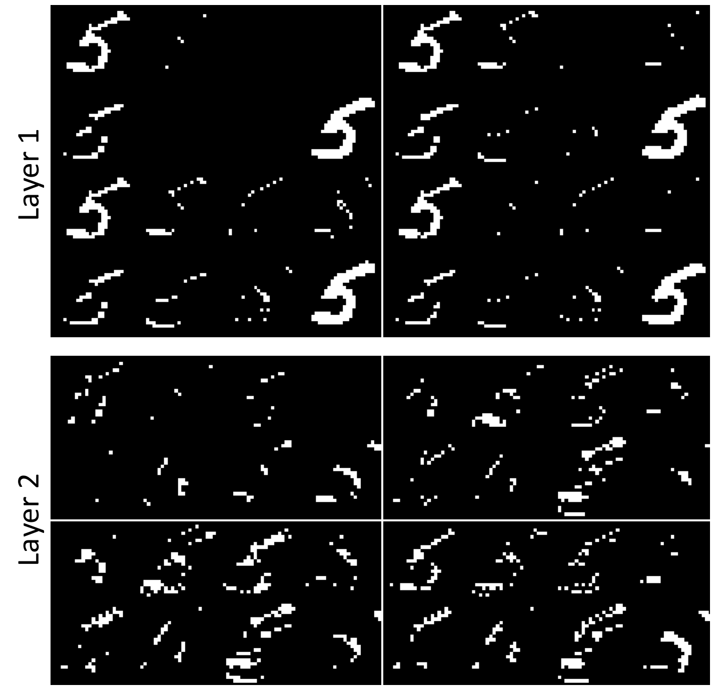


Optimally trained network

Lessons learned and future work

- Best practices for training stability
  - RMSprop best performing optimizer
  - Custom optimizer could be designed to be 'activation-aware'
  - Neurogenesis Deep Learning (NDL) for dead neurons
- Effective spiking networks differ from CNNs
  - Larger patch sizes help performance and stability
  - Additive noise may help regulate activity
  - Winner-take-all to replace softmax
  - Sparse distributed representations

Activation of Convolution Layers



Thank you

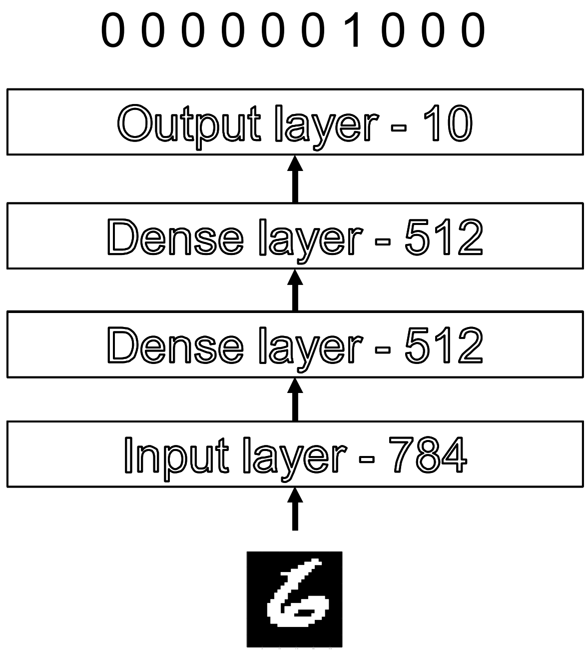
- Questions?

Backup slides

Comparing spiking (STPU) and non-spiking (keras) deep networks using the MNIST dataset

- Non-convolutional

non-spiking (dense) network test accuracy: 97.3

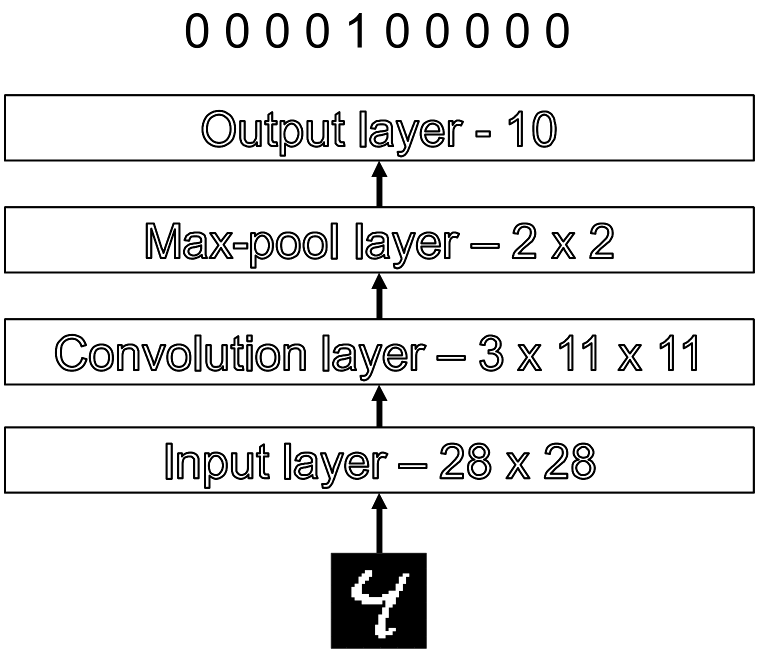


spiking (LIF neuron) network test accuracy: 97.28

Comparing spiking (STPU simulator) and non-spiking (keras) deep networks using the MNIST dataset

- Convolutional

non-spiking (CNN) network test accuracy: 97.09

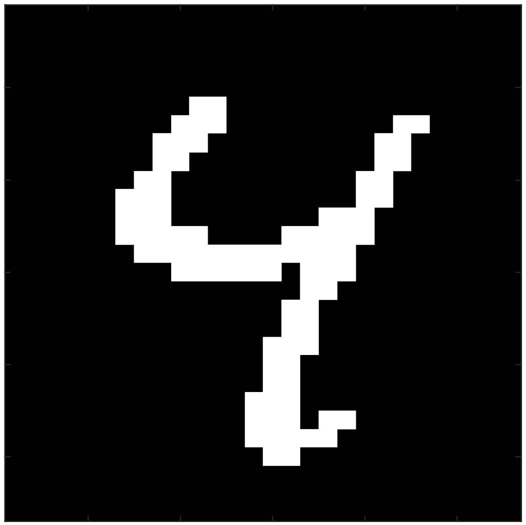


spiking (LIF neuron) network test accuracy: 94.85

This represents work-in-progress – we feel we can improve spiking accuracy with more time and better hardware

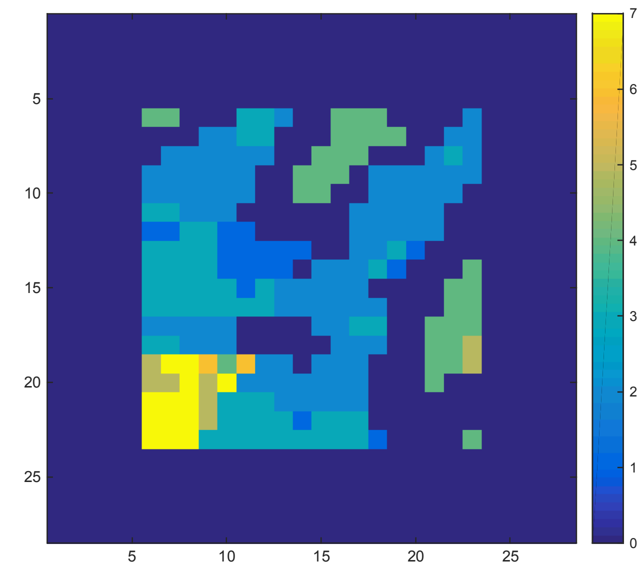
Spiking convolution representation of example image

Original image



Aggregation of convolution filters

Each filter is 11x11 but is represented by its center pixel



Active filters	Aggregated value
1st, 2nd and 3rd	7
2nd and 3rd	6
1st and 3rd	5
3rd	4
1st and 2nd	3
2nd	2
1st	1
none	0

Aggregation of filters seems to be ”learning” the white-space around the digit possibly in addition to the digit – Note this is unweighted aggregation