Deep Convolutional Spiking Neural Networks for Image Classification

Ruthvik Vaila, John Chiasson, Vishal Saxena

Boise State Universiy, University of Idaho

ruthvikvaila@u.boisestate.edu

August 7, 2019

Spike Timing Dependant Plasticity (STDP)

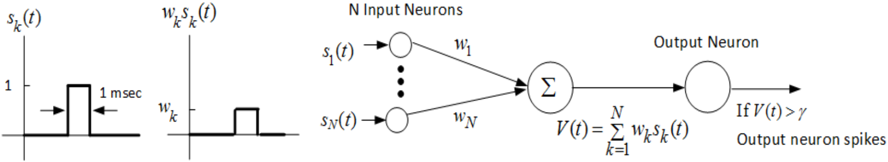


Figure 1: A simple fully connected spiking network.

- $V(t)$ is the membrane potential.
- Spike timing dependant plasticity (STDP) (Markram, Gerstner, and Sjöström 2012) has been shown to be able to detect hidden (in noise) patterns in spiking data (Masquelier, Guyonneau, and Thorpe 2008).
- Figure 1 shows a simple 2 layer fully connected network with N input (pre-synaptic) neurons and 1 output (post-synaptic) neuron.

Spike Timing Dependant Plasticity (STDP)

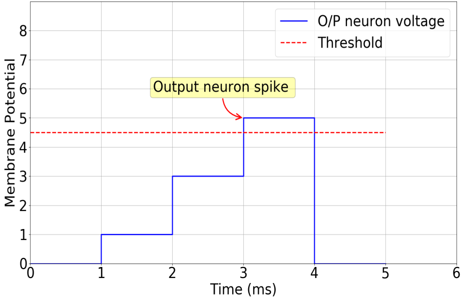


Figure 2: Membrane potential buildup.

$$w_i \leftarrow w_i + \Delta w_i, \quad \Delta w_i = \begin{cases} -a^- w_i(1-w_i), & \text{if } \; t_{out} - t_i < 0 \\ +a^+ w_i(1-w_i), & \text{if } \; t_{out} - t_i \geq 0. \end{cases}$$

- t_i is the spike time of the input (pre-synaptic) neuron.
- t_{out} is the spike time of the output (post-synaptic) neuron.

Spike Timing Dependant Plasticity (STDP)

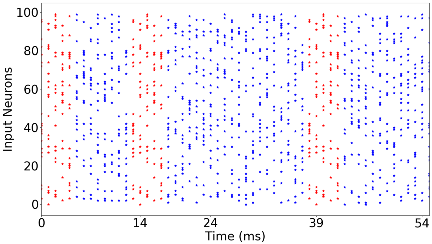


Figure 3: The red pattern of spikes is 5 ms. Blue spikes are random.

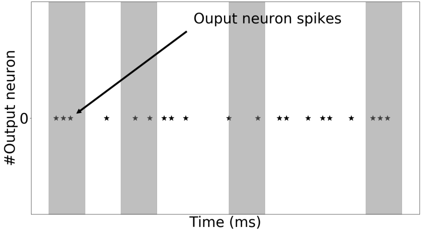
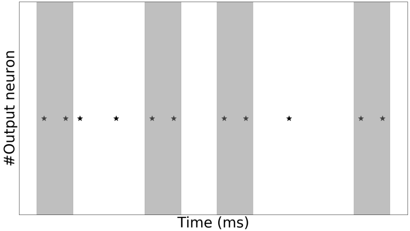
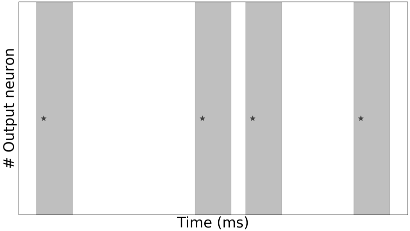


Figure 4: The grey box indicates the presence of the red pattern. Asterisk (*) indicates the output neuron spike.

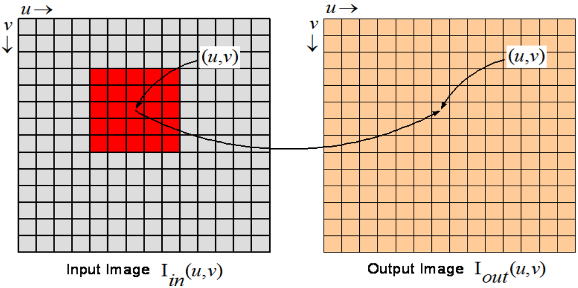
Spike Timing Dependant Plasticity (STDP)





- Left: After some training, the figure shows the output neuron starts to spike selectively (though it incorrectly spikes at times when the pattern is not presented).
- Right: After completion of the training the output neuron spikes only when the pattern is present.

Convolution



- $W_C(i,j)$: $\quad 0 \leq i,j \leq 4$ denotes a 5×5 convolution kernel. The red square in the figure above.
- With the kernel centered on the location (u, v) of the image $\mathbf{I}_{in}(u, v)$ the value $\mathbf{I}_{out}(u, v)$ of the output image at (u, v) is

$$\mathbf{I}_{out}(u, v) = \sum_{j=-2}^{j=2} \sum_{i=-2}^{i=2} \mathbf{I}_{in}(u + i, v + j) W_C(i, j).$$

Convolution for feature detection

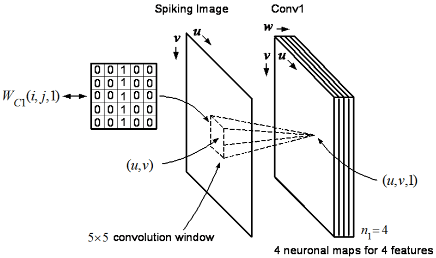


Figure 5: Feature detection.

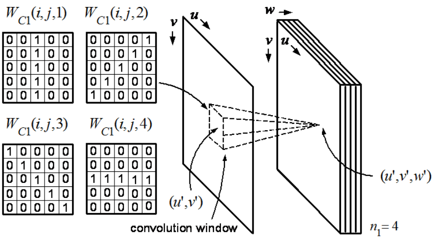


Figure 6: Feature detection.

- The $W_{C1}(i,j,1)$ kernel detects vertical line of spikes at any location of the (spiking) input image.
- At time t the neuron at (u,v) of map1 of the Conv1 **spikes** if $V_m(u,v,t,1) \geq \gamma_{C1}$ where γ_{C1} is the threshold.

$$V_m(u,v,t,1) = \sum_{\tau=0}^{t}\left(\sum_{j=-2}^{2}\sum_{i=-2}^{2} s_{in}(u+i,v+j,\tau)W_{C1}(i,j,1)\right).$$

- We say that the neuron at (u,v) of map1 in Conv1 detected a vertical line in the input.

Silicon retina input (ATIS/eDVS spikes)

- Silicon Retina (eDVS)
- Silicon Retina (ATIS) (Posch et al. 2010)
- Spikes from ATIS camera are recorded with 3 saccades [1] per image. We used pre-processed spikes (corrected for saccades) and direct spikes from the data set in this research.
- In this research we used spikes from the ATIS camera.
- ATIS: Asynchronous Time-based Image Sensor
- eDVS: Embedded Dynamic Vision Sensor

[1] a rapid movement of the eye between fixation points.

Convert photographic image into a spiking image using a Difference of Gaussian kernel (DoG)

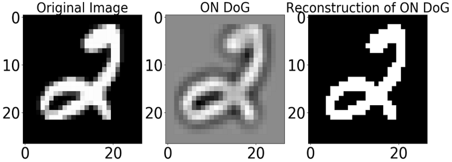


Figure 7: Left: Original grey-scale image. Center: Output of the ON centred DoG filter. Right: Accumulation of spikes (white indicates a spike black indicates no spike).

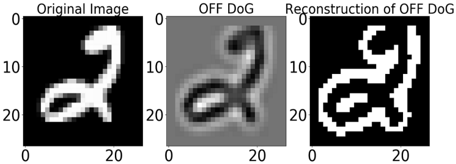


Figure 8: Left: Original grey-scale image. Center: Output of the OFF centred DoG filter. Right: Accumulation of spikes.

Network Architecture

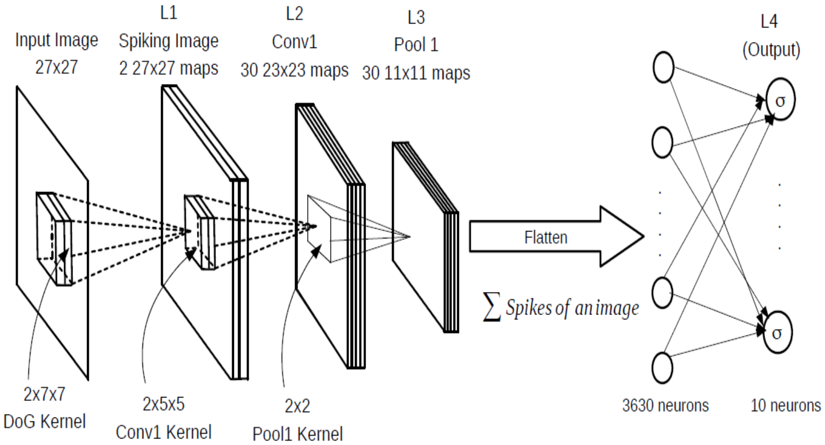


Figure 9: Deep spiking convolutional network architecture for classification of the MNIST data set.

Lateral inhibition and STDP competition

The number 5 is presented to the network. The total number of spikes produced by all the maps at each location.

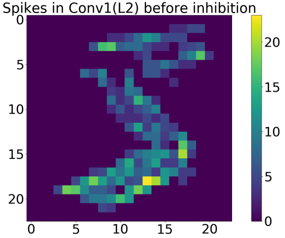


Figure 10: Spikes are accumulated in L2 *without* lateral inhibition and STDP competition.

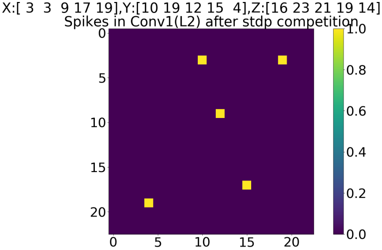


Figure 11: Spikes are accumulated in L2 *with* lateral inhibition and STDP competition.

Feature extraction

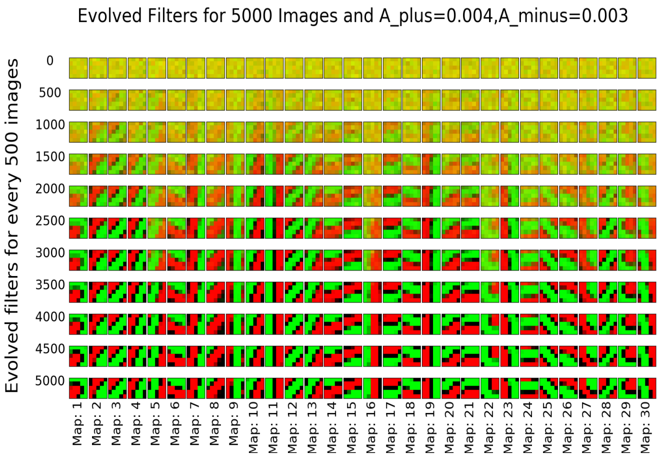


Figure 12: Plot of the kernels (weights) of the 30 maps of L2. The ON (green) 5×5 filter and the OFF (red) 5×5 filter are superimposed on top of each other.

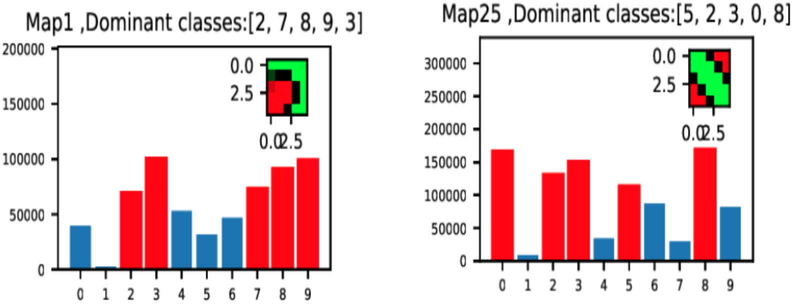
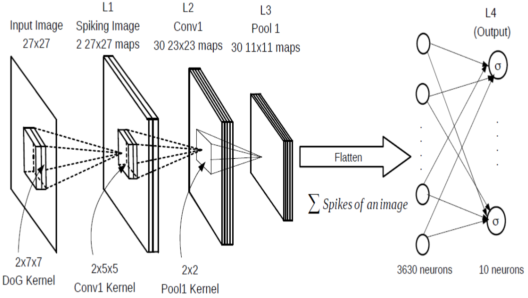


Figure 13: Spikes per map per digit. Headings for each of the sub-plots indicate the dominant (most spiking) digit for respective features.

Feature classification

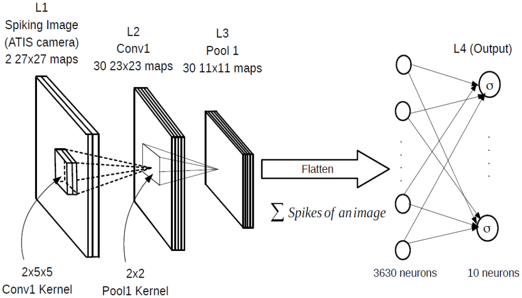


Classifier	Test Acc	Valid Acc	Training Time	λ	η	Epochs
2 Layer FCN	98.4%	98.5%	10 minutes	1/10	$0.1/(1.007)^{\#Epoch}$	20
RBF SVM	98.8%	98.87%	150 minutes	1/3.6	-	-
Linear SVM	98.41%	98.31%	100 minutes	1/0.012	-	-

Table 1: Spike count vectors extracted at layer L3 were used for classification.

- Kheradpisheh et al. 2018 reported an accuracy of 98.4% by using two convolution layers for feature extraction and an SVM for a classifier.

Transfer learning with N-MNIST (ATIS) spikes

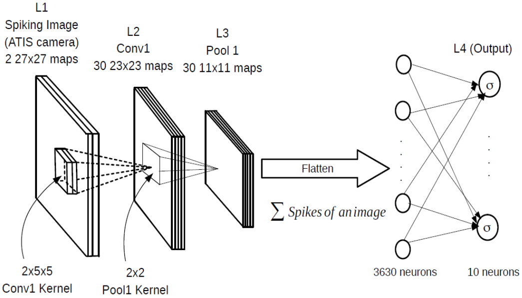


Classifier	Test Acc	Valid Acc	Training Time	λ	η	Epochs
2 Layer FCN	97.45%	97.62%	5 minutes	$\frac{1}{10.0}$	$\frac{0.1}{1.007^{\#Epoch}}$	20
RBF SVM	98.32%	98.40%	200 minutes	$\frac{1}{3.6}$	-	-
Linear SVM	97.64%	97.71%	100 minutes	$\frac{1}{0.012}$	-	-

Table 2: Classification accuracies of N-MNIST data set.

- Stromatias et al. 2017 reported an accuracy of 97.23% accuracy by using artificially generated features for the kernels of the first convolutional layer and training a 3 layer fully connected (1 hidden layer) neural network classifier on spikes collected at the first pooling layer.

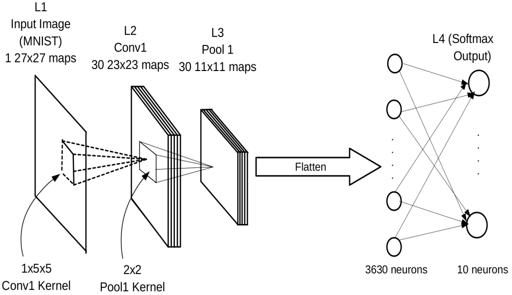
Learning with N-MNIST (ATIS) spikes



Classifier	Test Acc	Valid Acc	Training Time	λ	η	Epochs
2 Layer FCN	97.21%	97.46%	5 minutes	$\frac{1}{10.0}$	$\frac{0.1}{1.007^{\#Epoch}}$	20
RBF SVM	98.16%	98.2%	150 minutes	$\frac{1}{3.6}$	-	-
Linear SVM	97.38%	97.44%	100 minutes	$\frac{1}{0.012}$	-	-

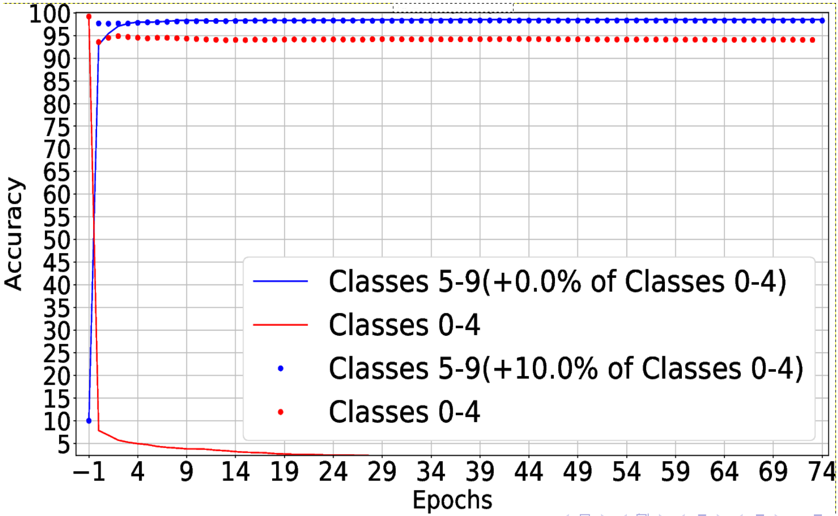
Table 3: Classification accuracies of N-MNIST data set.

Catastrophic forgetting

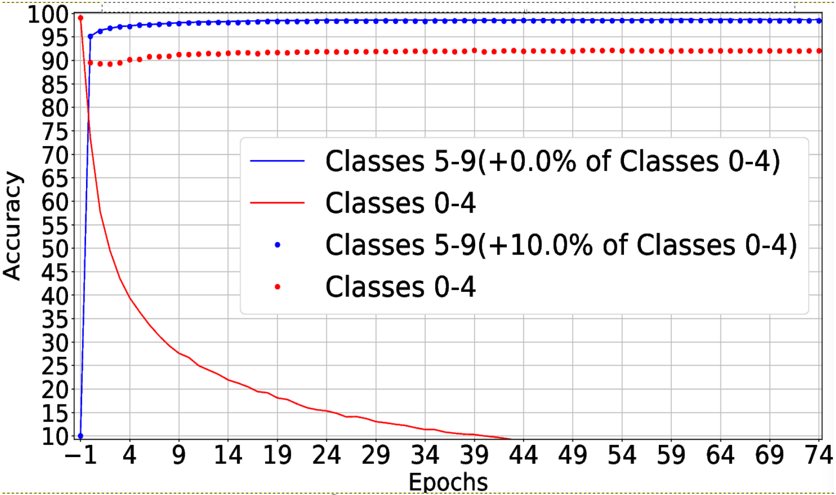


- **Step1:** Train on $0, 1, 2, 3, 4$, fix the weights of the convolution layer.
- **Step2:** Train on $5, 6, 7, 8, 9$ but back propagate till the flattened pooling layer only. (keeping the convolution layer unchanged)
- **Step3:** While training on $5, 6, 7, 8, 9$ accuracy test on $0, 1, 2, 3, 4$ results in less than 10% accuracy after one epoch.

Catastrophic forgetting in conventional CNNs with backprop



Catastrophic forgetting in a Spiking CNNs with STDP



- Note that solid red line. Spiking CNN forgets but not catastrophically. At the end of one epoch, spiking CNN retained almost 77% accuracy.

Catastrophic forgetting in a Spiking CNNs with STDP

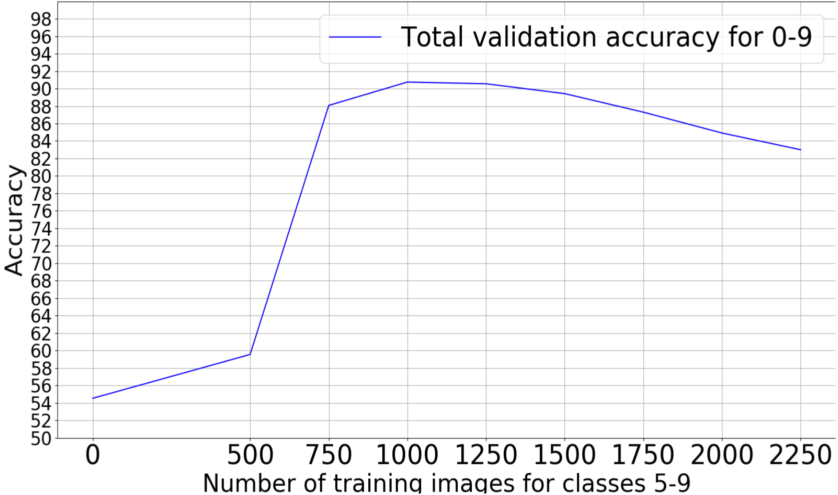


Figure 14: Note that as the number of training images for the classes 5-9 increases the total accuracy drops.

Acknowledgements

- We would like to express our gratitude to Professor Timothée Masquelier,[2] Dr. Saeed Reza Kheradpisheh[3] and Dr. Milad Mozafari[4] for answering our many questions about their work.

[2] Timothée Masquelier and Simon J. Thorpe (2007). "Unsupervised Learning of Visual Features through Spike Timing Dependent Plasticity". In: *PLoS Computational Biology* 3, pp. 1762 –1776.

[3] Saeed Reza Kheradpisheh et al. (2018). "STDP-based spiking deep convolutional neural networks for object recognition". In: *Neural Networks* 99, pp. 56 –67. ISSN: 0893-6080. DOI: https://doi.org/10.1016/j.neunet.2017.12.005. URL: http://www.sciencedirect.com/science/article/pii/S0893608017302903.

[4] M. Mozafari et al. (2018). "First-Spike-Based Visual Categorization Using Reward-Modulated STDP". In: *IEEE Transactions on Neural Networks and Learning Systems* 29.12, pp. 6178–6190. ISSN: 2162-237X. DOI: 10.1109/TNNLS.2018.2826721.