Synaptic Plasticity Dynamics for Deep Continuous Local Learning

Jacques Kaiser – FZI Research Center For Information Technology, Karlsruhe, Germany
Hesham Mostafa – Department of Bioengineering, University of California San Diego, La Jolla, USA
Emre Neftci – Department of Cognitive Sciences, University of California Irvine, Irvine, USA

Scan me

Abstract

Understanding and deriving neural and synaptic plasticity rules that can enable hidden weights to learn is an ongoing quest in neuroscience and neuromorphic engineering. From a machine learning perspective, **locality** and **differentiability** are key issues of the spiking neuron model operations. While the latter problem is now being tackled with surrogate gradient approaches, how to achieve this in deep networks in a scalable and local fashion is still an open question. Here, we demonstrate that deep learning algorithms that locally approximate the gradient backpropagation updates using locally synthesized gradients overcome this challenge. Our approach, called **Deep Continuous Local Learning (DCLL)**, results in highly efficient spiking neural networks and synaptic plasticity capable of training deep neural networks.

Contact Information

github.com/nmi-lab/dcll
arxiv.org/abs/1811.10766
nmi-lab.org
jkaiser@fzi.de, hmmostafa@ucsd.edu,
eneftci@uci.edu

References

[1] Arnon Amir et al. "A Low Power, Fully Event-Based Gesture Recognition System". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2017.

[2] P. Lichtsteiner et al. "An 128x128 120dB 15μs-latency temporal contrast vision sensor". In: *IEEE J. Solid State Circuits* (2008).

[3] Timothy P Lillicrap et al. "Random feedback weights support learning in deep neural networks". In: *arXiv preprint arXiv:1411.0247* (2014).

[4] Friedemann Zenke et al. "SuperSpike: Supervised learning in multi-layer spiking neural networks". In: *arXiv preprint arXiv:1705.11146* (2017).

Deep Continuous Local Learning (DCLL)

In three factor rules, both pre-synaptic and post-synaptic terms are local, meaning that all the variables to compute them are available and the neuron and synapse. The third factor – the error – generally involves non-local terms, including the activity of other neurons and the targets, and their history. Feedback alignment [3] has shown that approximations to the back-propagated errors are possible, but how to maintain their history efficiently remains a challenging problem. SuperSpike [4] deals with it by explicitly computing this history at the synapse and scales quadratically in the number of state variables.

The DCLL rule combines SuperSpike with deep local learning to solve the temporal and spatial credit assignment problem in continuous spiking neural networks. To achieve this, we organize neurons per layers and train each layer to predict a pseudotarget using random local readout, reminiscent of readout neurons in liquid state machines. The loss function is the sum of the layerwise loss functions defined against the local readout predictions. The absence of a temporal convolution term in the readout enables linear scaling.

As in SuperSpike [4], we rely on a deterministic Spike Response Model (SRM) model combined with a soft threshold function for computing a surrogate gradient:

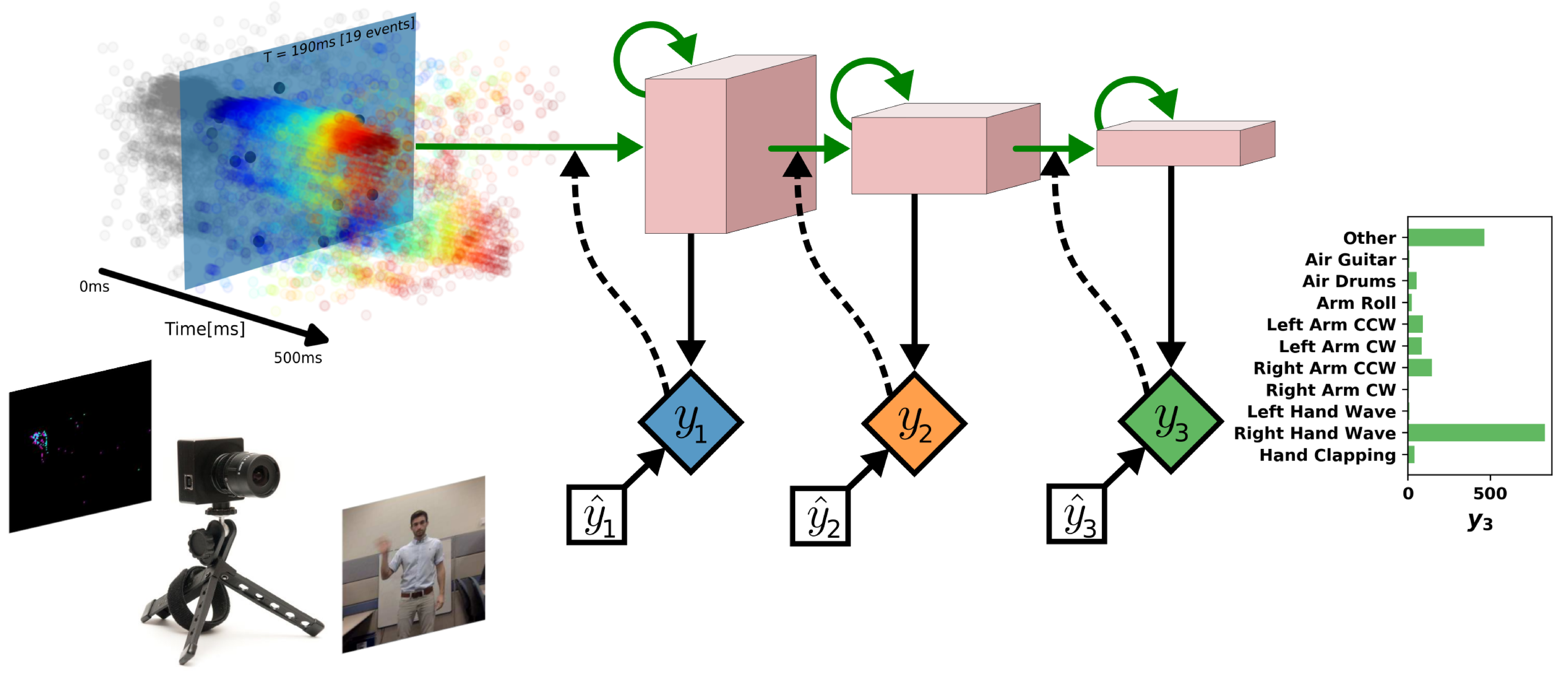
$$\begin{aligned} \textit{potential}: u_i &= \sum_j w_{ij}(\epsilon * s_j) + \eta * s_i, \\ \textit{spike}: s_i &= \Theta(u_i), \\ \textit{surrogate}: a_i &= \sigma(u_i), \end{aligned}$$

with Θ the unit step function and σ the sigmoid function. The surrogate network can be differentiated with respect to the neuron parameters. This enables a gradient-based optimization of a target loss L as a function of a_i:

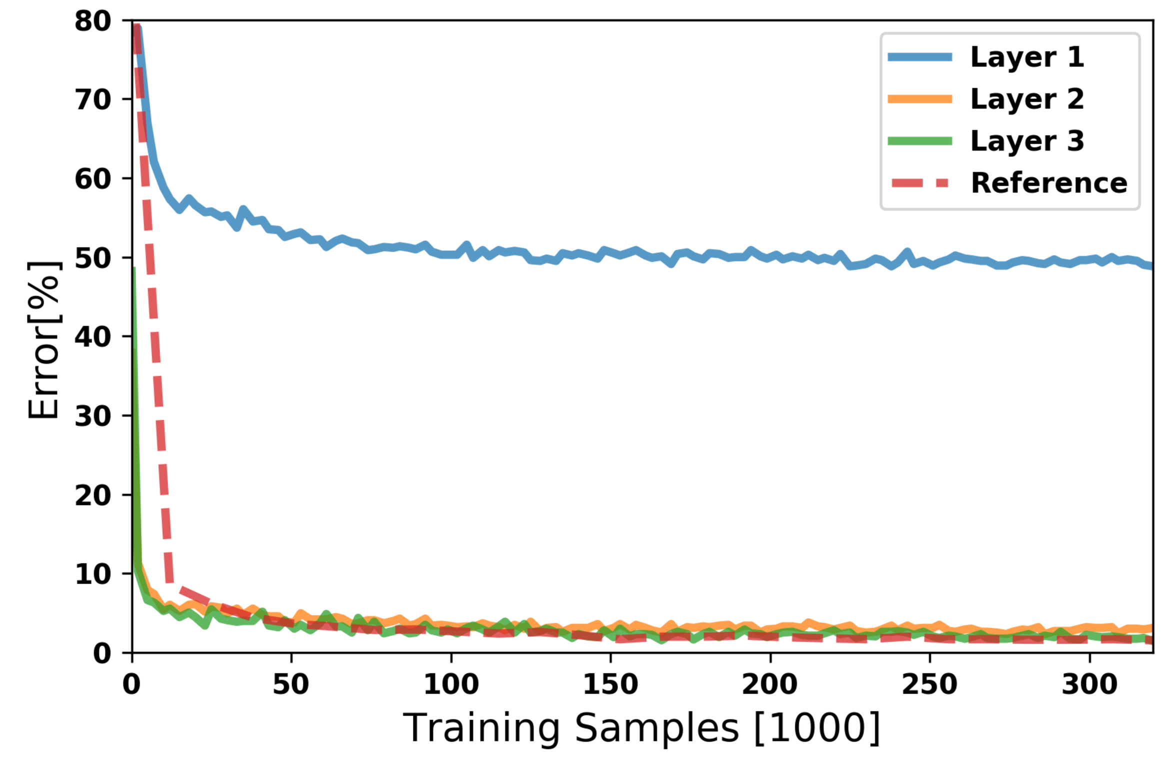
$$\frac{\partial}{\partial w_{ij}} L(a_i) = \frac{\partial}{\partial a_i} L(a_i) \frac{\partial}{\partial w_{ij}} a_i.$$

In the case of Mean Square Error loss for each layer, the learning rule can be derived as:

$$\Delta w_{ij}(t) = -\eta \left(\sigma'(u_i^n)(\epsilon * s_j^{n-1})\right) \sum_k g_{ki}^n error_k(t).$$

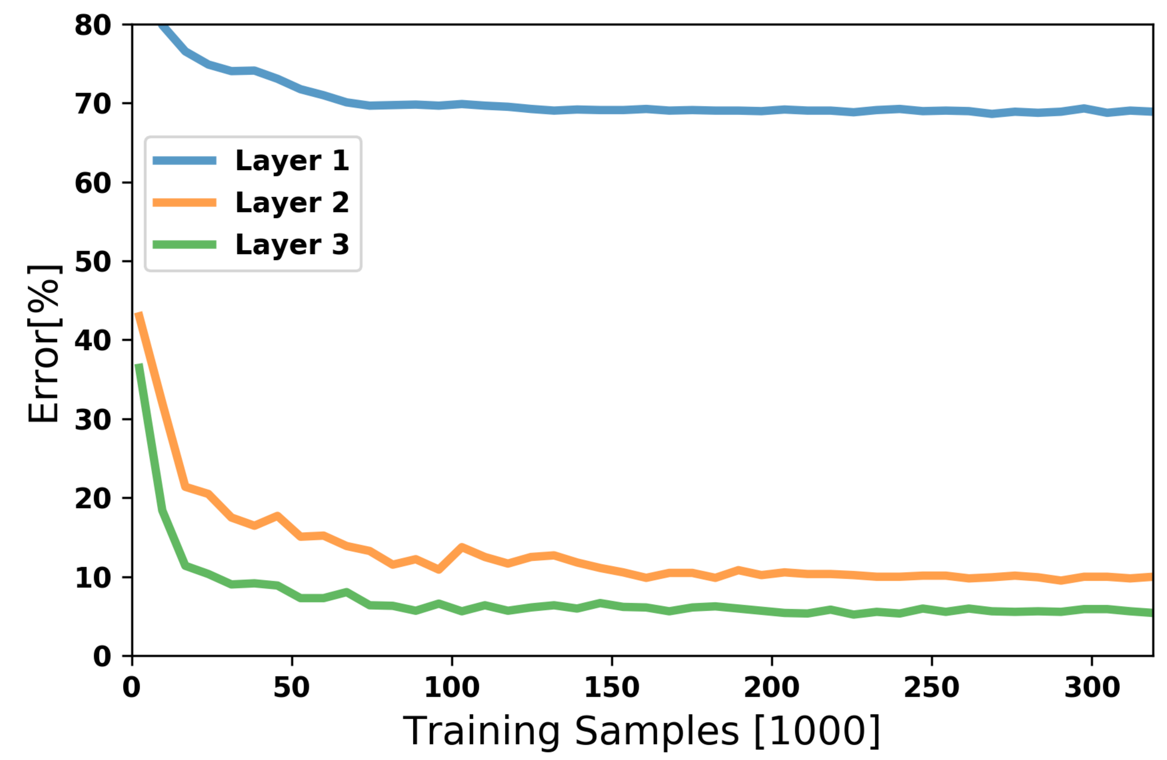


Results on Poisson-MNIST



We rely on a simple network architecture consisting of three convolutional layers of 16, 24 and 32 channels respectively with 7×7 kernels interleaved with max pooling layers. Each digit is converted into a 500ms spiketrain, yielding 450 gradient steps per minibatch. The final error is 1.23% for the third layer of spiking DCLL. An identical conventional convnet achieves 0.91% error.

Results on DvsGesture



Amir *et al.* recorded the DvsGesture dataset [1] using a Dynamic Vision Sensor (DVS) [2], consisting of 11 hand and arm gestures. DCLL reaches a 5.819% error, a performance comparable to the IBM EEDN case (5.51%). Moreover, DCLL converged after a much smaller number of iterations, and weight updates were performed online.