Towards the Robust Image Recognition Using Spiking Neurons

Qian Liu

Abstract—

I. Introduction

To gain a better understanding of the brain and build biologically-inspired computers, increasing attention is being paid to research into spike-based neural computation. Within the field, the visual pathway and its hierarchical organisation have been extensively studied within the primate brain. Spiking Neural Networks (SNNs) inspired by the understanding of observed biological structure and function have been successfully applied to visual recognition/classification tasks. In addition, implementations on neuromorhpic hardware have made large-scale networks run in (or even faster than) real time, and accessible on mobile robots. Neuromorphic sensors, e.g. silicon retinas, are able to feed such a mobile system with real-time visual stimuli. A new series of vision benchmarks for spike-based neural processing are now needed to quantitatively measure progress within this rapidly advancing field. We propose that a large dataset of spike-based visual stimuli is needed to provide a baseline for comparisons on SNN models and algorithms, and some benchmarking network models are also required to validate the accuracy and cost of these neuromorphic hardware platforms.

First of all, an initial NE (Neuromorphic Engineering) dataset of input stimuli based on standard computer vision benchmarks consisting of digits (from the MNIST database) is presented according to the current research on spike-based image recognition. Within this dataset, all images are centre aligned and having similar scale. We describe how we intend to expand this dataset to fulfil the needs of upcoming research problems. For instance, the data should provide cases to measure position-, scale-, and viewing-angle invariance. The data are in Address-Event Representation (AER) format which is widely used in the neuromorphic engineering field unlike conventional images. These spike trains are produced by various techniques: rate-based Poisson spike generation, rank order encoding and recorded output from a silicon retina with both flashing and oscillating input stimuli. Furthermore a complementary evaluation methodology is also presented to assess both model-level and hardware-level performance. Finally, we provide two SNN models to validate their classification capabilities and to assess the performances of their hardware implementations as tentative benchmarks.

With this dataset we hope to (1) promote meaningful comparison between algorithms in the field of neural computation, (2) allow comparison with conventional image recognition methods, (3) provide an assessment of the state of the art in spike-based visual recognition, and (4) help researchers identify future directions and advance the field.

SDBN

II. Related Works

[1] proposed a quantitative modelling framework of object recognition with position-, scale- and view-invariance based on the units of MAX-like operations. The cortical-like model has been analysed on several datasets [2]. And recently [3] reported that their SNN implementation of the framework was capable of facial expression recognition with a classification accuracy (CA) of 97.35% on the JAFFE dataset [4] which contains 213 images of 7 facial expressions posed by 10 individuals. They employed simple integrate-and-fire neurons with rank order coding (ROC) where the earliest pre-synaptic spikes have the strongest impact on the post synaptic potentials. According to [5], the first wave of spikes carry explicit information through the ventral stream and in each stage meaningful information is extracted and spikes are regenerated. Using one spike per neuron, [6] reported 100% and 97.5% accuracies on the face identification task over changing contrast and luminance training (40 individuals $\times$ 8 images) and testing data (40 individuals $\times$ 2 images) respectively.

The Convolutional Neural Network (CNN), also known as the *ConvNet* developed by [7], is a well applied model of such a cortex-like framework. An early Convolutional Spiking Neural Network (CSNN) model identified faces of 35 persons with a CA of 98.3% exploiting simple integrate and fire neurons [8]. Another CSNN model [9] was trained and tested both with DVS raw data and Leaky Integrate-and-Fire (LIF) neurons. It

The author is with the School of Computer Science, University of Manchester, Manchester M13 9PL, U.K. (e-mail:qian.liu-3@manchester.ac.uk

was capable of recognising three moving postures with a CA of about 99.48% and 88.14% on the MNIST-DVS dataset (see Chapter **??**). As one step forward, [10] implemented a convolution processor module in hardware which could be combined with a DVS for high-speed recognition tasks. The inputs of the ConvNet were continuous spike events instead of static images or frame-based videos. The chip detected four suits of a 52 card deck while the cards were fast browsed in only 410 ms. Similarly, a real-time gesture recognition model [11] was implemented on a neuromorphic system with a DVS as a front-end and a SpiNNaker [12] machine as the back-end where LIF neurons built up the ConvNet configured with biological parameters. In this study's largest configuration, a network of 74,210 neurons and 15,216,512 synapses used 290 SpiNNaker cores in parallel and reached 93.0% accuracy.

Deep Neural Networks (DNNs) together with deep learning are the most exciting research fields in vision recognition. The spiking deep network has great potential to combine remarkable performance with the energy efficient training and running. In the initial stage of the research, the study was focused on converting off-line trained deep network to SNNs [13]. The same network initially implemented on a FPGA achieved a CA of 92.0% [14], while a later implementation on SpiNNaker scored 95.0% [15]. Recent attempts have contributed to better translation by utilising modified units in a ConvNet [16] and tuning the weights and thresholds [17]). The later paper claims a state-of-the-art performance (99.1% on the MNIST dataset) comparing to original ConvNet. The current trend of training Spiking DNNs on-line using biologically-plausible learning methods is also promising. An event driven Contrastive Divergence (CD) training algorithm for RBMs (Restricted Boltzmann Machines) was proposed for Deep Belief Networks (DBN) using LIF neurons with STDP (Spike-Timing-Dependent Plasticity) synapses and verified on MNIST (91.9%) [18].

STDP as a biological learning process is applied to vision tasks. [19] demonstrated an unsupervised STDP learning model to classify car trajectories captured with a DVS retina. A similar model was tested on a Poissonian spike presentation of the MNIST dataset achieving a performance of 95.0% [20]. Theoretical analysis [21] showed that unsupervised STDP was able to approximate a stochastic version of Expectation Maximization, a powerful learning algorithm in machine learning. The computer simulation achieved 93.3% CA on MNIST and could be implemented in a memrisitve device [22].

III. Benchmarking spike-based visual recognition

The name of the first proposed dataset in the benchmarking system is NE15-MNIST which stands for Neuromorphic Engineering 2015 on MNIST. The original MNIST dataset is downloaded from the website[1] of THE MNIST DATABASE of handwritten digits [7]. The NE15-MNIST is converted into a spiking version of the original dataset consisting of four subsets which were generated for different purposes:

- *Poissonian* to benchmarking existing methods of rate-based spiking models.
- *FoCal* to promote the study of spatio-temporal algorithms applied to recognition tasks using few input spikes.
- *DVS recorded flashing input* to encourage research on fast recognition methods which are found in the primate visual pathway.
- *DVS recorded moving input* to trigger the study of algorithms targeting on continuous input from real-world sensors and to implement them on mobile neuromorphic robots.

The dataset can be found in the GitHub repository at: https://github.com/qian-liu/benchmarking.

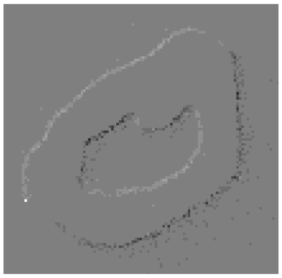
A. The Dataset: NE15-MNIST

1) File Formats: Two file formats are supported in the dataset: jAER format [23] (.dat or .aedat), and binary file in NumPy .npy format. The address event representation (AER) interface has been widely used in neuromorphic systems, especially for vision sensors. The spikes are encoded as time events with corresponding addresses to convey information. The spikes in jAER format, both recorded from a DVS retina and artificially generated, can be displayed in jAER software. Figure 1a is a snapshot of the software displaying an .aedat file which is recorded by a DVS retina [24]. The resolution of the DVS recorded data is 128×128. The other format of spikes used is a list of spike source arrays in PyNN [25], a description language for building spiking neuronal network models. Python code for converting one file format to and from the other is also provided.

2) Poissonian: In the cortex, the timing of spikes is highly irregular [26]. It can be interpreted that the inter-spike interval reflects a random process driven by the instantaneous firing rate. If the generation of each spike is assumed to be independent of all the other spikes, the spike train is seen as a Poisson process. The spiking rate can be estimated by averaging the pooled responses of the neurons.

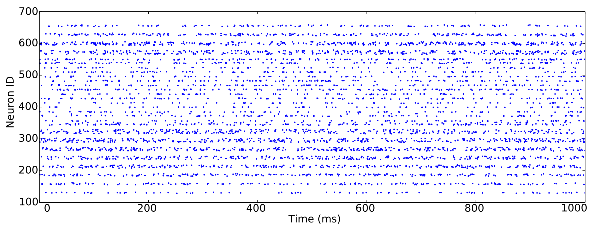
As stated above, rate coding is exclusively used in presenting images with spikes. The spiking rate of each

[1] http://yann.lecun.com/exdb/mnist/



(a) A snapshot of jAER playing the DVS recorded spikes.

(b) A snapshot of jAER playing Poissonian spike trains.



(c) The raster plot of the Poissonian spike trains.

Fig. 1: Snapshots of jAER software playing spike presented videos. The same image of digit “0” is transformed to spikes by DVS recording and the Poissonian generation respectively. A raster plot of the Poissonian spike trains is also provided.

neuron is in accordance with its corresponding pixel intensity. Instead of providing exact spike arrays, we share the Python code for generating the spikes. Every recognition system may require different spiking rates and various lengths of their durations. The generated Poissonian spikes can be in the formats of both jAER and PyNN spike source array. Thus, it is easy to visualise the digits and also to build spiking neural networks. Because different simulators generate random Poissonian spike trains with various mechanisms, languages and codes, using the same dataset enables performance evaluation on different simulators without the interference created by non-unified input. The same digit displayed in Fig. 1a is converted to Poissonian spike trains, see Fig. 1b. The raster plot can be found in Fig. 1c, indicating the intensities of the pixels.

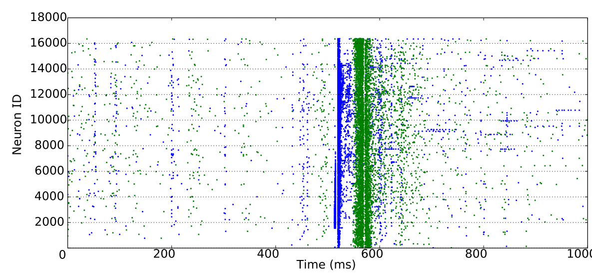
3) Rank-Order-Encoding: A different way of encoding spikes is using a rank-order code; this means keeping just the order in which those spikes were fired and disregarding the exact timing. Rank-ordered spike trains have been used in vision tasks under a biological plausibility constraint, making them a viable way of image encoding for neural applications [27], [28], [29].

4) DVS Sensor Output with Flashing Input: The purpose of including the subset of DVS recorded flashing digits is to promote the application of Rank-Order-Coding to DVS output, and accelerate the fast on-set recognition by using just the beginning part of spike trains within less than 30 ms.

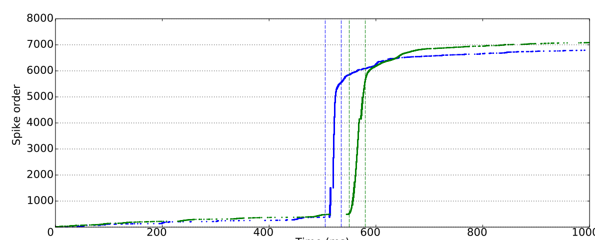
Each digit and a blank image was shown alternately and each display lasted one second. The digits were displayed on an LCD monitor in front of the DVS retina [24] and were placed in the centre of the visual field of the camera. Since there are two polarities of the spikes: ’ON’ indicates the increase of the intensity while ’OFF’ reflects the opposite, there are ’ON’ and ’OFF’ flashing recordings respectively per digit. In Fig. 2, the burstiness of the spikes is illustrated where most of the spikes occur in a 30 ms slot. In total, the subset of the database contains $2{\times}60,000$ recordings for training and $2{\times}10,000$ for testing.

5) DVS Sensor Output with Moving Input: In order to address the problems of position- and scale- invariance, a subset of DVS recorded moving digits is presented.

MNIST digits were scaled to three different sizes, by using smooth interpolation algorithms to increase their size from the original 28x28 pixel size, and displayed on the monitor with slow motion. The same DVS [24] used in Section III-A4 captured the movements of the digits and generated spike trains for each pixel of its $128{\times}128$ resolution. A total of 30,000 recordings were made: 10 digits, at 3 different scales, 1000 different handwritten samples for each.



(a) Spikes recorded in the order of neuron ID during 1s of time.



(b) Spikes plotted in the sequence of appearing time during 1s of time. Bursty spikes apeer in slots less than 30 ms.

Fig. 2: The bursty of spikes is illustrated where most of the spikes occur in a 30 ms slot. Blue for ’ON’ events and green for ’OFF’.

B. Evaluation

A complementary evaluation methodology is essential to provide common metrics and assess both the model-level and hardware-level performance.

IV. Case Studies

A. 2-Layer STDP

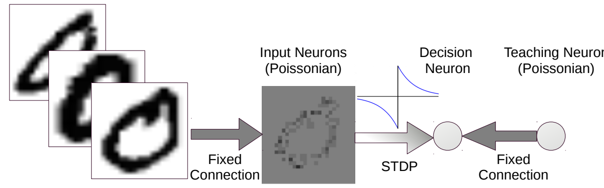
The case study is a simple two-layered network where the input neurons receive Poissonian presented spike trains from the dataset and form an FC network with the decision neurons. The model utilises LIF neurons, and the parameters are all with biological means.

In order to fully assess the performance, different settings have been configured on the network, such as network size, input rate and testing images duration. For simplicity of describing the system, one standard configuration is set as the example in the following sections.

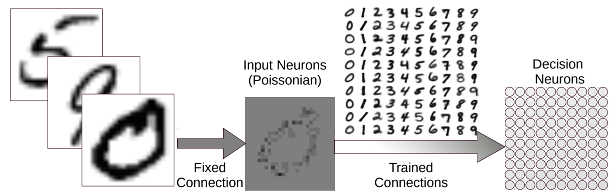
1) Training: There are two layers in the model: 28×28 input neurons fully connect to 100 decision neurons. Each decision neuron responds to a certain template of a digit. In the standard configuration, there are 10 decision neurons answering to the same digit with slightly different templates. Those templates are embedded in the connection weights between the two layers. Fig. 3a shows how the connections to a single decision neuron are tuned.

The training set of $60,000$ hand written digits are firstly classified into 100 classes, 10 subclasses per digit, using K-means clusters. So the images in a certain subclass are used to train one corresponding decision neuron. The firing rates of the input neurons are assigned linearly according to their intensities and normalised with a total firing rate of $2,000$ Hz. All the images together are presented for $18,000$ s (about 300 ms per image) during training and at the same time a teaching signal of 50 Hz is conveyed to the decision neuron to trigger STDP learning. The trained weights are plotted in accordance with the positions of the decision neurons in Fig. 3b.

2) Testing: After training the weights of the plastic synapses are set to static, keeping the state of the weights at the last moment of training. The weak weights were set to inhibitory connections with an identical strength. The feed-forward testing network is shown in Fig. 3b where Poissonion spike trains are generated the same way as in the training with a total firing rate of $2,000$ Hz per image. The input neurons convey the same spike trains to every decision neuron through its responding trained synaptic weights. Every testing image ($10,000$ images in total) is presented once and lasts 1 s with a silence of 200 ms between them. The output neuron with the highest firing rate decides what digit was recognized.



(a) Training model of a single decision neuron.



(b) Testing model of the spiking neural network.

Fig. 3: The training and testing model of the two-layered spiking neural network.

Taken the trained weights from the NEST simulation, the accuracy of the recognition on NEST reaches 90.03% with the standard configuration, while the result drops slightly to 89.97% using SpiNNaker. In comparison, both trained and tested on SpiNNaker the recognition accuracy is 87.41%, and with the same weights applied to NEST the result turns out to be 87.25%.

V. Preliminary and Future Work

In order to implement training of Spiking Deep Belief Networks (SDBNs) on SpiNNaker, this paper studies the layer-by-layer training of spiking Restricted Boltzmann Machine (RBM) of a DBN. The study starts from understanding the original problem, Products of Experts (PoE), which was solved by using Contrastive Divergence (CD). It involves utilising Markov Chain Monte Carlo (MCMC) sampling to present the distribution of a certain untraceable high-dimensional probability model function, e.g. PoE. Among these sampling algorithms, Gibbs method is introduced and used in PoE problem. Instead of minimising the original objective function of Kullback-Leibler divergence, the contrastive divergence is exploited to solve PoE. Then the study continues on applying CD to RBM. Finally, on-line learning methods only spiking neurons used are explored to train spiking RBM.

Need to draw a figure! And Gantt Chart.

Appendix A
Thesis Outline

The following section-level outline gives the planned thesis structure for this project. Sections which are reliant on upcoming work are indicated with a star (*);

1) Introduction

TABLE I: Hardware independent comparison

	Preprocessing	Network	Training	Recognition
[30]	None	Two layer, LIF neurons	Semi-supervised, STDP, calcium LTP/LTD	96.5%
[31]	None	V1 (edge), V4 (orientation), and competitive decision, Izhikevich neurons	Semi-supervised, STDP, calcium LTP/LTD	91.6% 300 ms per test
[18]	Thresholding	Two layer RBM, LIF neurons	Event-driven contrastive divergence, supervised	91.9% 1 s per test
[20]	None	Two layers, LIF neurons, inhibitory feedback	Unsupervised, exp. STDP, $3,000,000$ s of training $200,000$ s per iteration	95%
[17]	None	ConvNet or FCnet, LIF neurons	Off-line trained with ReLU, weight normalization	99.1% (ConvNet), 98.6% (FCnet); 0.5 s per test
[9]	Thresholding or DVS	Simple (Gabor), Complex (MAX) and Tempotron	Tempotron, supervised	Thresholding 91.3%, 11 s per test DVS 88.1%, 2 s per test
This paper	None	Four layer RBM, LIF neurons	Off-line trained, unsupervised	94.94% 16 ms latency
This paper	None	FC decision layer, LIF neurons	K-means clusters, Supervised STDP $18,000$ s of training	92.98% 1 s per test 10.70 ms latency

TABLE II: Hardware dependent comparison

	System	Neuron Model	Synaptic Plasticity	Precision	Simulation Time	Energy/Power Usage
SpiNNaker [32]	Digital, Scalable	Programmable Neuron/Synapse, Axonal delay	Programmable learning rule	11- to 14-bit synapses	Real-time Flexible time resolution	8 nJ/SE 54.27 MSops/W
TrueNorth [33]	Digital, Scalable	Fixed models, Config params, Axonal delay	No plasticity	122 bits params & states, 4-bit synapse (4 signed int + on/off state)	Real-time	46 GSops/W
Neurogrid [34]	Mixed-mode, Scalable	Fixed models, Config params	Fixed rule	13-bit shared synapses	Real-time	941 pJ/SE
HI-CANN [35]	Mixed-mode, Scalable	Fixed models, Config params	Fixed rule	4-bit synapses	Faster than real-time	198 pJ/SE 13.5 MSops/W (network only)
iAER-IFAT [36]	Mixed-mode, Scalable	Fixed models, Config params	No plasticity	Analogue neuron/synapse	Real-time	20GSops/W

2) Background
   a) Neural Network on Image Recognition
   b) Neuron Models and Spiking Neural Network
   c) Spiking Neural Network Simulation
   d) Neuromorphic Simulators
3) Related Works
   a) Vision Databases and Benchmarks
   b) Deep Neural Networks
   c) Spike-Based Image Recognition
   d) Real-Time Neuromorphic Vision System
4) Benchmarking Spike-Based Visual Recognition
   a) Database
   b) Evaluation Methodology
5) Spiking Deep Belief Network
   a) Restricted Boltzmann Machine
   b) Deep Belief Network
   c) Spiking RBM and DBN *

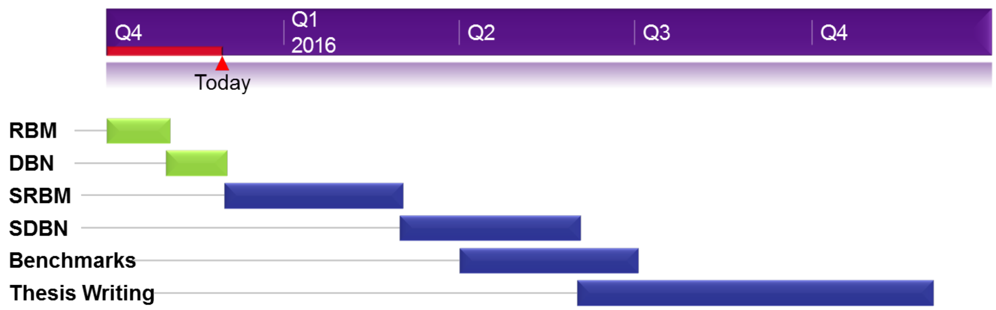


Fig. 4: Gantt Chart of the future work.

   d) Formalisation of SDBN *
6) Benchmarks
   a) STDP Learned 2-Layer Network
   b) Spiking DBN *
   c) Off-line Trained Convolutional Network *
7) Discussions *
   a) Benefits of Spike-based Processing *
   b) Scalability of Hardware SDBN *
   c) Future NE Database Development *
8) Future Work *
   a) SDBN Toolbox on SpiNNaker *
   b) Learning on Spiking Convolutional Network *
   c) Video-Based Recognition and Benchmarks *
   d) Natural Language Processing and Benchmarks *

References

[1] M. Riesenhuber and T. Poggio, "Hierarchical models of object recognition in cortex," *Nature neuroscience*, vol. 2, no. 11, pp. 1019–1025, 1999.

[2] T. Serre, L. Wolf, S. Bileschi, M. Riesenhuber, and T. Poggio, "Robust object recognition with cortex-like mechanisms," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 29, no. 3, pp. 411–426, 2007.

[3] S.-Y. Fu, G.-S. Yang, and X.-K. Kuai, "A spiking neural network based cortex-like mechanism and application to facial expression recognition," *Computational intelligence and neuroscience*, vol. 2012, p. 19, 2012.

[4] M. Lyons, S. Akamatsu, M. Kamachi, and J. Gyoba, "Coding facial expressions with gabor wavelets," in *Automatic Face and Gesture Recognition, 1998. Proceedings. Third IEEE International Conference on*, pp. 200–205, 1998.

[5] R. Van Rullen and S. J. Thorpe, "Surfing a spike wave down the ventral stream," *Vision research*, vol. 42, no. 23, pp. 2593–2615, 2002.

[6] A. Delorme and S. J. Thorpe, "Face identification using one spike per neuron: resistance to image degradations," *Neural Networks*, vol. 14, no. 6, pp. 795–803, 2001.

[7] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[8] M. Matsugu, K. Mori, M. Ishii, and Y. Mitarai, "Convolutional spiking neural network model for robust face detection," in *Neural Information Processing, 2002. ICONIP'02. Proceedings of the 9th International Conference on*, vol. 2, pp. 660–664, 2002.

[9] B. Zhao, R. Ding, S. Chen, B. Linares-Barranco, and H. Tang, "Feedforward categorization on AER motion events using cortex-like features in a spiking neural network," *Neural Networks and Learning Systems, IEEE Transactions on*, vol. 26, pp. 1963–1978, Sept 2015.

[10] L. Camunas-Mesa, C. Zamarreño-Ramos, A. Linares-Barranco, A. J. Acosta-Jiménez, T. Serrano-Gotarredona, and B. Linares-Barranco, "An event-driven multi-kernel convolution processor module for event-driven vision sensors," *Solid-State Circuits, IEEE Journal of*, vol. 47, no. 2, pp. 504–517, 2012.

[11] Q. Liu and S. Furber, "Real-time recognition of dynamic hand postures on a neuromorphic system," in *Artificial Neural Networks, 2015. ICANN. International Conference on*, vol. 1, p. 979, 2015.

[12] S. B. Furber, F. Galluppi, S. Temple, L. Plana, *et al.*, "The SpiNNaker Project," *Proceedings of the IEEE*, vol. 102, no. 5, pp. 652–665, 2014.

[13] P. O'Connor, D. Neil, S.-C. Liu, T. Delbruck, and M. Pfeiffer, "Real-time classification and sensor fusion with a spiking deep belief network," *Frontiers in neuroscience*, vol. 7, 2013.

[14] D. Neil and S.-C. Liu, "Minitaur, an event-driven FPGA-based spiking network accelerator," *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on*, vol. 22, no. 12, pp. 2621–2628, 2014.

[15] E. Stromatias, D. Neil, F. Galluppi, M. Pfeiffer, S.-C. Liu, and S. Furber, "Scalable energy-efficient, low-latency implementations of trained spiking deep belief networks on SpiNNaker," in *Neural Networks (IJCNN), The 2015 International Joint Conference on*, p. to be published, IEEE, 2015.

[16] Y. Cao, Y. Chen, and D. Khosla, "Spiking deep convolutional neural networks for energy-efficient object recognition," *International Journal of Computer Vision*, vol. 113, no. 1, pp. 54–66, 2015.

[17] P. Diehl, D. Neil, J. Binas, M. Cook, S.-C. Liu, and M. Pfeiffer, "Fast-classifying, high-accuracy spiking deep networks through weight and threshold balancing," in *Neural Networks (IJCNN), The 2015 International Joint Conference on*, p. to be published, 2015.

[18] E. Neftci, S. Das, B. Pedroni, K. Kreutz-Delgado, and G. Cauwenberghs, "Event-driven contrastive divergence for spiking neuromorphic systems," *Frontiers in neuroscience*, vol. 7, 2013.

[19] O. Bichler, D. Querlioz, S. J. Thorpe, J.-P. Bourgoin, and C. Gamrat, "Extraction of temporally correlated features from dynamic vision sensors with spike-timing-dependent plasticity," *Neural Networks*, vol. 32, pp. 339–348, 2012.

[20] P. U. Diehl and M. Cook, "Unsupervised learning of digit recognition using spike-timing-dependent plasticity," *Frontiers in Computational Neuroscience*, vol. 9, p. 99, 2015.

[21] B. Nessler, M. Pfeiffer, L. Buesing, and W. Maass, "Bayesian computation emerges in generic cortical microcircuits through spike-timing-dependent plasticity," *PLoS Comput Biol*, 2013.

[22] J. Bill and R. Legenstein, "A compound memristive synapse

model for statistical learning through STDP in spiking neural networks," *Frontiers in neuroscience*, vol. 8, 2014.

[23] T. Delbruck, "Frame-free dynamic digital vision," in *Proceedings of Intl. Symp. on Secure-Life Electronics, Advanced Electronics for Quality Life and Society*, pp. 21–26, 2008.

[24] T. Serrano-Gotarredona and B. Linares-Barranco, "A 128×128 1.5% contrast sensitivity 0.9% FPN 3μs latency 4 mW asynchronous frame-free dynamic vision sensor using transimpedance preamplifiers," *Solid-State Circuits, IEEE Journal of*, vol. 48, no. 3, pp. 827–838, 2013.

[25] A. P. Davison, D. Brüderle, J. Eppler, J. Kremkow, E. Muller, D. Pecevski, L. Perrinet, and P. Yger, "PyNN: a common interface for neuronal network simulators," *Frontiers in neuroinformatics*, vol. 2, 2008.

[26] L. R. Squire and S. M. Kosslyn, *Findings and current opinion in cognitive neuroscience*. MIT Press, 1998.

[27] R. Van Rullen and S. J. Thorpe, "Rate coding versus temporal order coding: what the retinal ganglion cells tell the visual cortex," *Neural computation*, vol. 13, no. 6, pp. 1255–1283, 2001.

[28] B. Sen and S. Furber, "Evaluating rank-order code performance using a biologically-derived retinal model," in *Neural Networks, 2009. IJCNN. International Joint Conference on*, pp. 2867–2874, IEEE, 2009.

[29] K. Masmoudi, M. Antonini, P. Kornprobst, and L. Perrinet, "A novel bio-inspired static image compression scheme for noisy data transmission over low-bandwidth channels," in *Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on*, pp. 3506–3509, 2010.

[30] J. M. Brader, W. Senn, and S. Fusi, "Learning real-world stimuli in a neural network with spike-driven synaptic dynamics," *Neural computation*, vol. 19, no. 11, pp. 2881–2912, 2007.

[31] M. Beyeler, N. D. Dutt, and J. L. Krichmar, "Categorization and decision-making in a neurobiologically plausible spiking network using a STDP-like learning rule," *Neural Networks*, vol. 48, pp. 109–124, 2013.

[32] E. Stromatias, F. Galluppi, C. Patterson, and S. Furber, "Power analysis of large-scale, real-time neural networks on SpiNNaker," in *Neural Networks (IJCNN), The 2013 International Joint Conference on*, pp. 1–8, 2013.

[33] P. A. Merolla, J. V. Arthur, R. Alvarez-Icaza, A. S. Cassidy, J. Sawada, F. Akopyan, B. L. Jackson, N. Imam, C. Guo, Y. Nakamura, *et al.*, "A million spiking-neuron integrated circuit with a scalable communication network and interface," *Science*, vol. 345, no. 6197, pp. 668–673, 2014.

[34] B. V. Benjamin, P. Gao, E. McQuinn, S. Choudhary, A. R. Chandrasekaran, J.-M. Bussat, R. Alvarez-Icaza, J. V. Arthur, P. Merolla, K. Boahen, *et al.*, "Neurogrid: a mixed-analog-digital multichip system for large-scale neural simulations," *Proceedings of the IEEE*, vol. 102, no. 5, pp. 699–716, 2014.

[35] J. Schemmel, D. Bruderle, A. Grubl, M. Hock, K. Meier, and S. Millner, "A wafer-scale neuromorphic hardware system for large-scale neural modeling," in *Circuits and Systems (ISCAS), Proceedings of 2010 IEEE International Symposium on*, pp. 1947–1950, 2010.

[36] T. Yu, J. Park, S. Joshi, C. Maier, and G. Cauwenberghs, "65k-neuron Integrate-and-Fire array transceiver with address-event reconfigurable synaptic routing," in *Biomedical Circuits and Systems Conference (BioCAS), 2012 IEEE*, pp. 21–24, 2012.