Turn Down That Noise: Synaptic Encoding of Afferent SNR in a Single Spiking Neuron

Saeed Afshar, *Member, IEEE*, Libin George, *Student Member, IEEE*, Chetan Singh Thakur, *Student Member, IEEE*, Jonathan Tapson, *Member, IEEE*, André van Schaik, *Fellow, IEEE*, Philip de Chazal, *Senior Member, IEEE*, and Tara Julia Hamilton, *Member, IEEE*

***Abstract*—We have added a simplified neuromorphic model of Spike Time Dependent Plasticity (STDP) to the previously described Synapto-dendritic Kernel Adapting Neuron (SKAN), a hardware efficient neuron model capable of learning spatio-temporal spike patterns. The resulting neuron model is the first to perform synaptic encoding of afferent signal-to-noise ratio in addition to the unsupervised learning of spatio-temporal spike patterns. The neuron model is particularly suitable for implementation in digital neuromorphic hardware as it does not use any complex mathematical operations and uses a novel shift-based normalization approach to achieve synaptic homeostasis. The neuron's noise compensation properties are characterized and tested on random spatio-temporal spike patterns as well as a noise corrupted subset of the zero images of the MNIST handwritten digit dataset. Results show the simultaneously learning common patterns in its input data while dynamically weighing individual afferents based on their signal to noise ratio. Despite its simplicity the interesting behaviors of the neuron model and the resulting computational power may also offer insights into biological systems.**

***Index Terms*—Delay plasticity, neuromorphic engineering, spatio-temporal spike pattern recognition, spiking neural network, synaptic plasticity, temporal coding.**

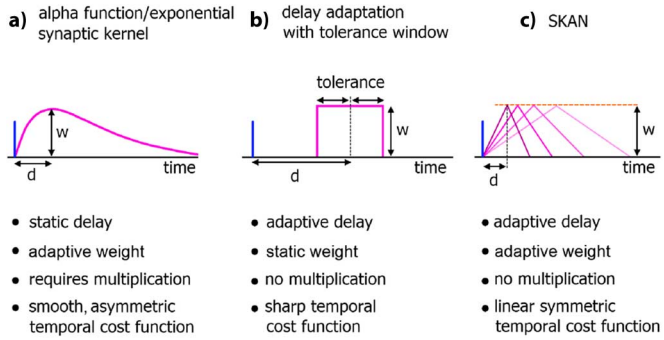

Fig. 1. Comparison of neuromorphic implementations of synapto-dendritic kernels. The characteristics of realized Excitatory PostSynaptic Potential (EPSP) kernels are computationally important just prior to being summed at the soma. These kernels represent the cost function used to translate the temporal error in spatiotemporal spike patterns at the synapse to the somatic membrane potential. Due to the large number of synapses neural network systems require, the complexity, functionality, and hardware cost of these kernels is a critical feature of neuromorphic spiking networks. (a) An exponentially decaying kernel typically used for modelling synapses. (b) A simplified synaptic model with the kernel modelled as a binary delayed window. (c) SKAN. Adapted from [8].

I. Introduction

SYNAPSES are by far the most numerous computational elements in the brain and in neuromorphic systems. Due to their large numbers, the return on investment on synapses, i.e., how much functional computation they perform versus how much hardware resources they take up, becomes a defining feature of any neural system whether evolved or designed by an engineer [1]–[3]. Thus the extraction of the most functionality from the fewest, simplest synapses is often a central focus for the neuromorphic engineer [4], [5].

In the context of neuromorphic systems the synapse serves three essential functions. The first is simply to form a connection from one neuron to the next. The second is to spread the energy of the presynaptic input spike over time via the synaptic kernel and the third is to weigh this kernel such that when it is added to other similarly weighted synaptic kernels, the resulting summation, called the somatic membrane potential, is a useful signal encoding functionally relevant information such as how well an input spatio-temporal pattern matches those commonly seen in the past.

While the realization and use of these distributed kernel-based processing units has evidently been mastered by evolution, despite significant recent progress, our best engineered systems still find the large-scale realization of these three functions challenging. The first and simplest function of the synapse, that of acting as the connection between neurons represents the greatest hardware challenge. Limitations in network bandwidth or connectivity are often the most serious obstacles restricting full utilization of neuromorphic hardware resources. Innovative approaches such as time multiplexing [3] and Address Event Representation (AER) [6], [7] can create virtual all-to-all connected networks, however, these advantages come at the expense of reduced operating speed.

The second function of the synapse, that of spreading an input spike's energy over time can be realized via a range of synaptic kernels with varying levels of complexity, hardware cost and computational utility. Two extremes include an exponentially decaying kernel shown in Fig. 1(a) which is

Manuscript received November 02, 2014; revised March 09, 2015; accepted March 21, 2015. Date of publication April 22, 2015; date of current version April 29, 2015. This paper was recommended by Associate Editor V. Gruev.

S. Afshar, L. George, C. S. Thakur, J. Tapson, A. van Schaik, and T. J. Hamilton are with the Biomedical Engineering and Neuroscience Program, The MARCS Institute, University of Western Sydney, Kingswood 2751, NSW, Australia (e-mail: s.v.afshar@uws.edu.au).

P. de Chazal is with the Biomedical Engineering and Neuroscience Program, The MARCS Institute, University of Western Sydney, Kingswood 2751, NSW, Australia, and also with the School of Electrical and Information Engineering, University of Sydney, Sydney, NSW 2006, Australia.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TBCAS.2015.2416391

typically used to model biological synapses [9], and the simple delay learning approach with a binary kernel and a temporal tolerance window shown in Fig. 1(b) [10]. In the ideal synaptic model, a real-valued synaptic alpha function is multiplied by a real-valued synaptic weight with the later adapting to input spikes to model synaptic weight adaptation. However the cost of implementing this synaptic kernel in large numbers in digital hardware is substantial as it requires the realization of a multiplier at each synapse. In addition, the alpha function does not model the computationally useful peak delay adaptation effects observed in biology [11] which necessitates the realization of multiple synapses for learning arbitrary delays.

The third function of the synapse, that of weighing the kernel, requires another multiplication operation between the synaptic weight and the instantaneous value of the EPSP kernel. For real-valued synapses and kernels the hardware cost of this multiplication operation can be prohibitive [4]. At the other end of the spectrum, rather than implement complex synaptic weight adaptation, other neuromorphic implementations of spiking networks have focused on the adjustment of explicit propagation delays along the neural signal path to encode memory [10], [12], [13]. Here the energy of the spike is spread via a binary valued tolerance window as shown Fig. 1(b). This discarding of synaptic weights significantly simplifies implementation and allows more synapses to be realized. The down side is that explicit window-based delay learning schemes can produce "sharp" systems with lower tolerance for the dynamically changing temporal variance they inevitably encounter in applications where neuronal systems are expected to excel: noisy, dynamic and unpredictable environments [14]. In addition while use of these simplified kernels allow more synapses to be realized, limited network bandwidth can sometimes mean that these larger numbers cannot actually be fully utilized.

In this context, the proposed Synapto-dendritic Kernel Adapting Neuron (SKAN) model, which is the focus of this work and is shown Fig. 1(c), uses fully adaptable yet simple accumulator based kernels. Here it should be noted that the kernels of SKAN do not directly model individual synapses, which have kernels that are approximately static. Instead the kernels are simplified models of multiple synapses with different kernels combined with the adaptive properties of dendritic tree on which they reside [11], with the assumption that the function of the whole system is to create a mapping from the afferents to the soma such that commonly presented spatio-temporal patterns are preferentially transmitted. In this way SKAN is an attempt to model the entire input to soma coupling system in as simplified and hardware amenable way possible. Having made this distinction, the word synapse will be used to refer to the entire system.

These adaptable kernels allow each synapse to learn any delay, such that the number of synapses required equals only the number of input channels or afferents [15]. SKAN's live unsupervised hand gesture learning and recognition has been demonstrated in [16] using a neuromorphic visual scene to spatio-temporal spike pattern transformation [17]. Although the SKAN neuron exhibits robustness to noise this robustness is achieved through the adaptation of its soma to overall input noise which means that the neuron has a single measure for the noisiness of its inputs and assumes a homogenous signal to noise ratio at all input synapses. In this work, by introducing a simplified Spike Time Dependent Plasticity (STDP) rule to each synapse, the neuron model is enhanced such that the synapses adapt independently to their noise environment, improving spatio-temporal spike pattern recognition.

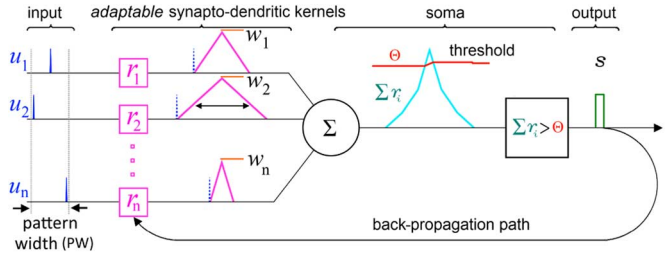

Fig. 2. Schematic of the elements and information paths in SKAN. The input spikes $u_i(t)$ trigger adaptable synapto-dendritic kernels which rise up to the synaptic weight $u_i(t)$ and are summed to form the neuron's somatic membrane potential $\Sigma r_i(t)$. This signal is then compared to an adaptive somatic threshold $\Theta(t)$ which, if exceeded, results in an output pulse $s(t)$. The output pulse also feeds back to adapt the kernels. Note that the back propagating signal does not travel beyond the synapto-dendritic structures of the neuron to previous neural layers. Adapted from [8].

II. Synapto-Dendritic Kernel Adapting Neuron

A. Kernel and Threshold Adaptation Mechanisms

The dynamics of SKAN have been previously described in detail for the special case of static weights [8]. Fig. 2 illustrates the functional diagram of the neuron.

At each synapse the presynaptic input spike $u_i(t)$ triggers the EPSP kernel, which is the central element of SKAN and is modeled as an adaptable kernel in the form of a simple ramp-up-ramp-down sequence $r_i(t)$. After being triggered $r_i(t)$ rises with slope $\Delta r_i(t)$ until it reaches the synaptic weight variable $w_i(t)$. In the previously proposed SKAN model these weights are static and do not change over time. After reaching $w_i(t)$ the kernel ramps down with the same slope and returns to zero. These EPSP kernels spread the energy of the input spike signals over time allowing them to be summed at the soma to generate the somatic membrane potential $\Sigma r_i(t)$, which is compared to the threshold $\Theta(t)$ to generate an postsynaptic output pulse $s(t)$. The back propagation of this output pulse in turn adapts the slope of the kernels $\Delta r_i(t)$ such that, if a kernel is rising at the time of the output pulse, it is deemed to be too late and so its slope is increased, making the kernel sharper. Alternatively if the kernel is falling at the time of the output pulse, it is deemed to be too early and its slope is decreased such that the kernel becomes wider. This simple kernel slope adaptation rule aligns the peaks of many synaptic kernels in response to repeated presentations of the same spatio-temporal pattern. As the kernel peaks become ever more aligned their summation at the somatic membrane potential forms an ever higher and narrower peak as shown in Fig. 3.

In addition to the kernel adaptation, which captures the temporal information of the input pattern, the neuron's threshold also adapts: At every time step during an output pulse the threshold rises and every time the membrane potential returns to zero the threshold falls, as indicated by the grey circle in Fig. 3.

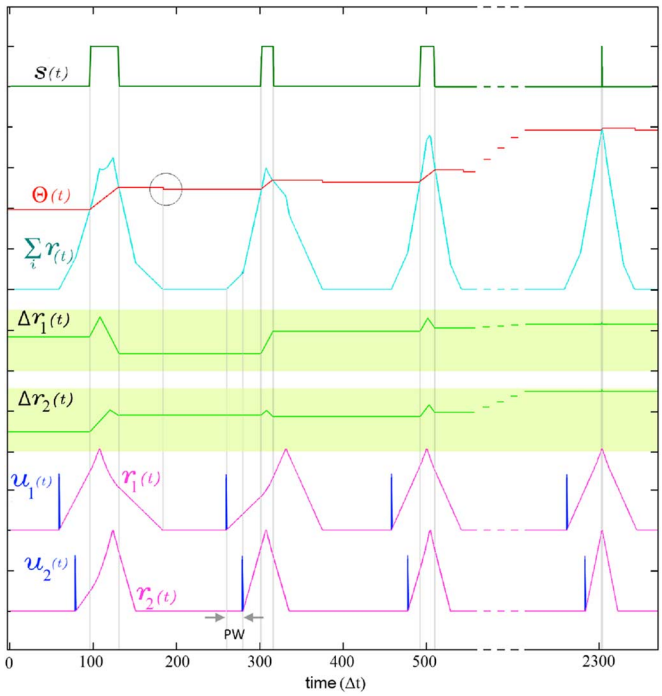

Fig. 3. Kernel adaptation in SKAN with static synaptic weights. The kernels and the threshold of SKAN adapt in response to repeated spatio-temporal pattern presentations. The kernels have captured the ISI information by the third presentation of the pattern. With each subsequent presentation the threshold $\Theta(t)$ increases making the neuron more selective as the kernel step sizes $\Delta r_i(t)$ increase, making the kernels narrower. As a result, each pattern presentation increases the neuron's confidence about the underlying process producing the ISI's, narrowing the neuron's receptive field around the target ISI, and producing a smaller output pulse $s(t)$. By the 11th presentation $(t = 2300\Delta t)$, the Θ_{rise} during the output spike and Θ_{fall} balance each other such that the $\Theta_{before} \approx \Theta_{after}$. The soma output spike $s(t)$ is now a finely tuned unit delta pulse which indicates high certainty. When the membrane potential returns to zero, the neuron's threshold falls as indicated by the grey circle. Adapted from [8].

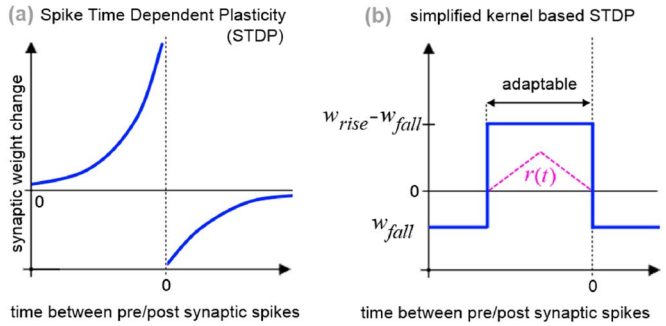

Fig. 4. Comparison of the classical STDP synaptic weight update curve with the simplified kernel based STDP used in this work. (a) In the classical STDP update rule, a synapse has its highest weight increase if it receives a presynaptic input spike just prior to a postsynaptic output spike. Conversely a postsynaptic output spike that precedes a presynaptic input spike causes the greatest decrease in synaptic weight. Both effects decay with longer inter-spike intervals. (b) In the simplified model used in this work, the adaptable EPSP kernel not only learns the commonest pre/post synaptic spike interval, but also doubles as a flag that enables the increase in synaptic weights in the event of a postsynaptic spike.

As the neuron spikes more in response to a particular pattern, its threshold rises, making the neuron ever more selective for the pattern that triggered it and narrowing its spatio-temporal receptive field. Conversely, unrecognized input patterns, which do not cause an output spike, reduce the threshold, making the neuron more receptive to new patterns. Through this feedback mechanism the neuron automatically maintains a balance between selectivity and generalization in response to the statistics of its environment.

In the following sections the synaptic weight adaptation of SKAN and the resulting behaviors of the neuron are described.

III. Weight Adaptation via Simplified STDP

A. Simplified Weight Update Rule

In biology the rules governing synaptic weight adaptation vary enormously both in degree and in type across species, brain regions, synapse types, cell types, within individual cells, over short time scales, and as a function of organism development [18]. However, the STDP rule shown in Fig. 4(a) is by far the best studied synaptic plasticity rule in neuroscience today due to its reproducibility and neurocomputational utility in selectively strengthening synapses such that given a large enough number of hetergenous synapses with different intrinsic delays, any arbitrary spatio-temporal pattern can be learnt by a single neuron [19]. As a consequence, the faithful modeling of this rule in hardware is now a major focus in neuromorphic engineering [9]. In this model of synaptic plasticity the strengthening or weakeing of synaptic transmission efficiency is typically modelled by an exponential decaying function of the inter-spike interval between the pre- and post- synaptic spikes as shown Fig. 4(a). As with the smooth synaptic alpha function, such accurate modelling of neurobiological processes can incur additional hardware costs while providing little computational improvement compared to even highly simplified models [20]. Therefore, as was the case with SKANs simplified kernels, in this report, the classic STDP rule is replaced with the simiplified weight update rule shown in Fig. 4(b). The rule is designed so that it reuses the same signals and flags that are already present in the static weight SKAN system, such that the presynaptic input spike $u_i(t)$, which triggers the EPSP kernel $r_i(t)$, also triggers a binary weight adjustment flag $d_i(t)$ If a back-propogating postsynaptic output spike, $s(t)$, arrives while this flag is high, then the synaptic weight $w_i(t)$ is increased by w_{rise} and the flag is reset to zero. Alternatively if the membrane potential $\Sigma r_i(t)$, returns to zero before an output spike arrives, then the synaptic weight is decreased by w_{fall} and the flag is returns to zero. These two rules are described in (1) and (2).

$$w_i(t) = \begin{cases} w_i(t-1) + w_{rise}, & \text{if } d_i(t-1) \wedge \downarrow s(t) \\ w_i(t-1) - w_{fall}, & \text{if } d_i(t-1) \wedge \downarrow \Sigma r_i(t) \end{cases} \quad (1)$$

$$d_i(t) = \begin{cases} 1, & \text{if } u_i(t) \\ 0, & \text{if } \downarrow \Sigma r_i(t) \vee \downarrow s(t) \end{cases} \quad (2)$$

Where $\downarrow s(t)$ is the falling edge of the postsynaptic output spike and $\downarrow \Sigma r_i(t)$ is the return of the membrane potential to zero $(\Sigma r_i(t-1) > 0 \wedge \Sigma r_i(t) = 0)$.

As a result of (1) and (2), every time the membrane potential rises due to input spikes, the synaptic weights of the activated

inputs either rise in response to the resultant output spike or they fall when the membrane potential returns to zero.

This simplification of the STDP model significantly reduces hardware costs. By using the EPSP kernel as a binary flag of adaptable duration, the need for realization of the exponentially decaying function of Fig. 4(a) is eliminated and the use of the constant update terms w_{rise} and w_{fall} replaces the addition of two arbitrary values, $w_i(t)$ and $\Delta w_i(t)$, which would otherwise be required at each synapse and which is significantly more costly in terms of hardware resources in comparison to constant terms which can be hardwired.

Additionally unlike in classical STDP, in the proposed model, if an output spike were to be triggered just after the membrane had returned to zero (say by an external stimulator) there would be no change to the synaptic weight, but in the normal operation of the neuron, this simplification of the model does not affect the overall behavior. Similarly, in the SKAN model multiple input spikes that arrive within a short time or in bursts are covered by the EPSP kernel of the leading spike and are invisible to the system. This reduction is arguably desirable as it correlates well to real world event driven stimuli where relative stimulus onset times across afferents carry salient information.

B. Synaptic Weight Normalization Without Division

An additional layer of complexity arises through the need for synaptic homeostasis which is required to keep the synaptic weights within some limited dynamic range while preserving their *relative* strengths. In the biological context a number of homeostasis models have been proposed [21], [22]. The common feature of these models is divisive normalization where the strength of all synapses in a neuron are rescaled via division by a global signal which keeps all synaptic weights within their physiological dynamic range while preserving their relative transmission efficiency. The implementation of this normalization operation in digital hardware again involves multiplication operation at each synapse.

In this work we propose a novel digital approach to this problem that eliminates the need for this multiplication. The synaptic weights and other variables in the neuron all of which are implemented using unsigned integers are normalized not via multiplication but bit shift operations. Here, instead of normalizing the synaptic weights such that the max of the weights, $\max(w_i(t))$, or the sum of the weights, $\Sigma w_i(t)$, is clamped to a specific value, the max signal is allowed to vary within the top half of a digital range, updated by the weight update rule of (1) and (2). When the update rule pushes the $\max(w_i(t))$ signal beyond this digital range, all the neuron's signals, i.e., the weights $w_i(t)$, EPSP kernels $r_i(t)$, EPSP kernel step-sizes $\Delta r_i(t)$ and the threshold $\Theta(t)$, are right shifted, (division by 2). Conversely if the signal falls below half its range, all the signals of the neuron are left shifted (multiplication by 2) as shown for a synapse case. These two conditions simplify to (3) and (4).

$$\cup_i \, overflow(w_i(t)) \rightarrow right\ shift\ neuron\ values \quad (3)$$

$$\cap_i \, MSB(w_i(t)) = 0 \rightarrow left\ shift\ neuron\ values \quad (4)$$

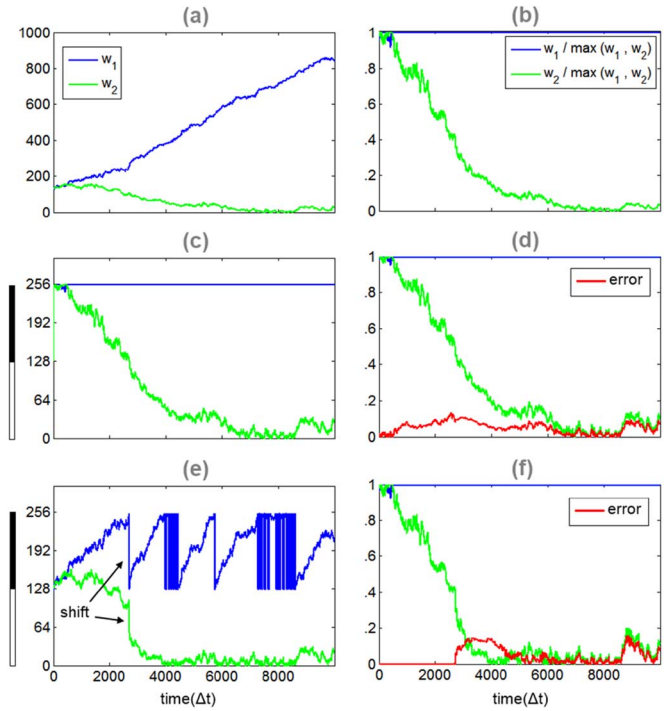

Fig. 5. Comparison of fixed point division and shift based normalization for encoding the relative strengths of synaptic weights within an 8-bit dynamic range. The two synapses with weights w_1 and w_2 are updated by two independent non-zero mean stochastic processes. The panels on the left show the encoded weights while the panels on the right show the relative weights with the strongest synapse normalized to 1 (calculated via floating point division). (a) The original weights of w_1 and w_2 with no bound on their dynamic range: $w_i(t+1) = w_i(t) + \Delta w_i(t)$. (b) The true relative strengths of the original synaptic weights. (c) Limiting the two weights to between 0–255 (8-bit unsigned integer) via fixed point division: $w_i(t+1) = w_i(t) + \Delta w_i(t) / \max_i(w_i(t) + \Delta w_i(t))$. The black and white bars indicate the top and bottom half of the digital range (0–127 and 128–255). (d) Relative strength of the bounded 8-bit synaptic weights. The error plot shows RMS error with respect to the original relative strengths shown in (b). (e) 8-bit Shift based normalization showing the stronger signal w_1 triggering shifts in both synapses as it exceeds the bounds of the top half of its digital range: $w_i(t+1) = f(w_i(t) + \Delta w_i(t))$. Where $f()$ denotes equations (3) and (4). (f) The relative synaptic strengths encoded via shift based normalization and the associated error.

The fact that *all* neuronal parameters, $w_i(t)$, $r_i(t)$ and $\Theta(t)$ are also shifted means that the neuron is essentially not affected by the shift operations. As demonstrated in Fig. 5 for a two synapse neuron, the overall effect of the shifting operations described is to continuously generate more dynamic range such that all weights become normalized while the max signal remains within the range described by (5)

$$2^{b-1} \leq \max\left(w_i(t)\right) \leq 2^b - 1 \quad (5)$$

where b is the number of bits used to represent the synaptic weights.

Fig. 6 shows the error introduced in the relative strength of synaptic weights through the use of shift based normalization with varying bit-widths. In addition to the quantization noise introduced, an important edge case occurs in the 'right shift neuron values' operation, which requires a design decision in terms of any weak synapses which go to zero. One option is to not allow any weights to go to zero. This can be implemented

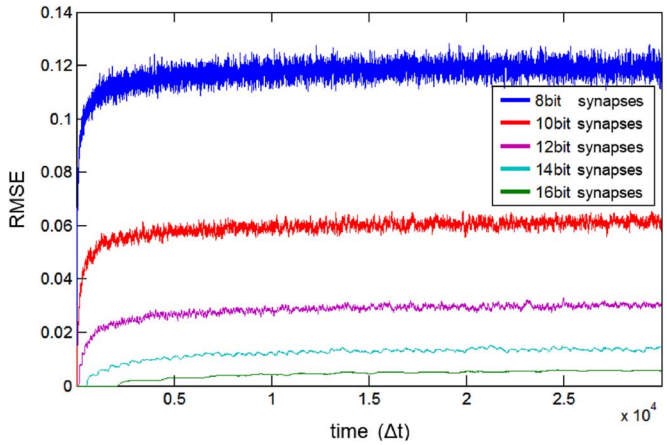

Fig. 6. RMS error of shift based normalization with respect to normalization via double-precision floating-point division. Random synaptic weight updates with $\mathrm{P}(w_{rise}) = \mathrm{P}(w_{fall})$ were performed on a simulated 16 synapse neuron with synaptic bit-width of 8 to 16. The RMS error of the relative weights of the shift based synapses was calculated against synapses which were normalized via double-precision division. Increasing bit-width in the shift based synapses resulted in lower error but even at the lower bit-lengths the relative order of the synaptic weights followed the floating-point implementation.

either by checking all bits of every synapse and setting to 1 any that go to zero or simply by assuming the LSB of all synapses is set to high without any zero checking. Another option in dealing with synaptic weights that go to zero is to disable them completely. This can potentially allow re-allocation of the synapses to other neurons. An application of this option is discussed in Section IV.

IV. Results

A. STDP and SKAN Combine to Produce Synaptic Encoding of Afferent Signal to Noise Ratio (SNR)

Given static synaptic weights $w_i(t) = w_i(t_0)$, the simple kernel adaptation of SKAN can perform unsupervised learning of common spatio-temporal patterns in noisy environments as detailed in [8]. When this static weight model of SKAN is combined with the synaptic weight update and the normalization operation of the previous section, the neuron not only finds and learns the most common spatio-temporal pattern, but additionally adjusts its synaptic weights independently to compensate for the signal-to-noise ratio of individual afferents and thus improves recognition performance. In this work the corruptive noise is defined as additive noise spikes generated by a homogeneous Poisson process with rate λ/T. The signal represents spatio-temporal spike patterns which are presented every T time steps as shown in Fig. 6. The SNR is thus defined as the ratio of the target pattern presentation rate 1/T, and the noise process rate λ/T giving

$$SNR = 1 : \lambda. \tag{6}$$

To demonstrate the SNR encoding effect, consider the case where the neuron is presented with repeated spatio-temporal spike patterns that are received via noise corrupted afferents. After several pattern presentations the neuron's kernels 'see through' this noise and adapt their slopes $\Delta r_i(t)$ so that they

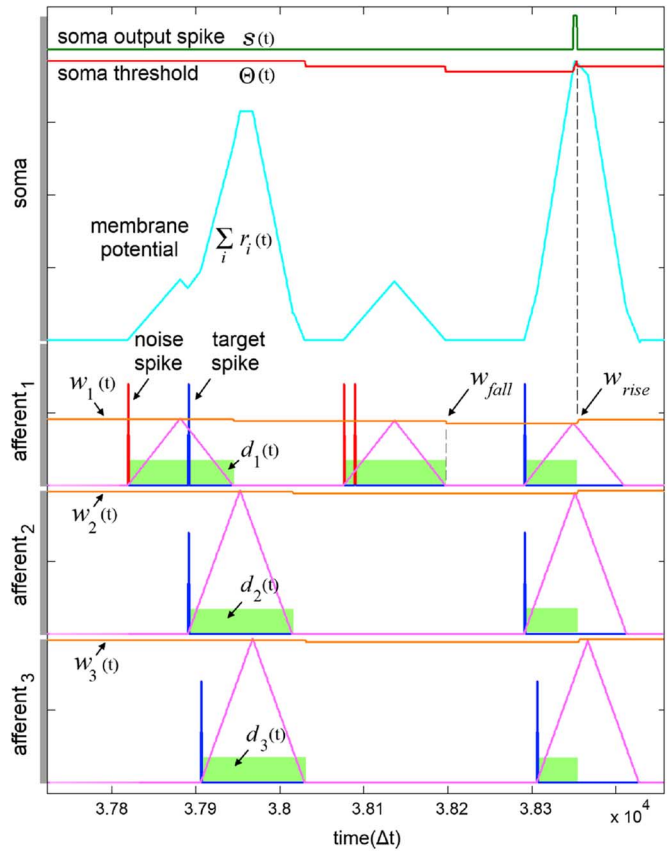

Fig. 7. Synaptic weight adaptation in a neuron with three synaptic inputs. The first input channel $(\mathrm{afferent}_1)$ is noisy $(\mathrm{SNR} = 1:1)$ while the other two input channels $(\mathrm{afferent}_{2:3})$ are noise free. Input spikes trigger both the kernel $r_i(t)$ and the weight update flag $d_i(t)$. The output spike $s(t)$ causes a rise in the synaptic weight of all synapses by w_{rise}, while the return of the membrane potential to zero induces a fall by w_{fall}, in the activated synapses with $d_i(t) = 1$. During the time interval shown the noisy afferent with weight w_1 experiences two weight falls and one weight rise while the two noise free afferents experience only one weight fall and one weight rise. As a result of many such adjustments w_1 falls to its steady state value of approximately half of w_2 and w_3. The lower weighted synapses contribute less and less to the membrane potential The full dynamic range of the synaptic are indicated by the grey regions marked $\mathrm{afferent}_{1:3}$.

align with the pattern. This is because the noise is uncorrelated with the pattern and it is just as likely to increase the slopes as it is to decrease them such that the noise is averaged out, leaving only the target pattern for the kernels to train on. Fig. 7 shows what happens next for a case where one of the three afferents is corrupted by $\mathrm{SNR} = 1:1$, that is, where the probability of the presence of a Poisson noise spike during any time period equals the probability of a target spike belonging to the target spatio-temporal pattern.

Noise spikes, being uncorrelated with the target pattern and with each other, typically arrive on their own or in such a way that their EPSPs are not enough to push the membrane potential $\Sigma r_i(t)$ past the threshold $\Theta(t)$ to cause a postsynaptic output spike. Such noise spikes do however reduce their respective synaptic weight by w_{fall}. In the case where a clean target spatio-temporal pattern arrives without any neighboring noise, all the synaptic weights are increased equally by w_{rise}. The combined result of these changes is that synapses that receive target input spikes more often accumulate higher and higher weights while

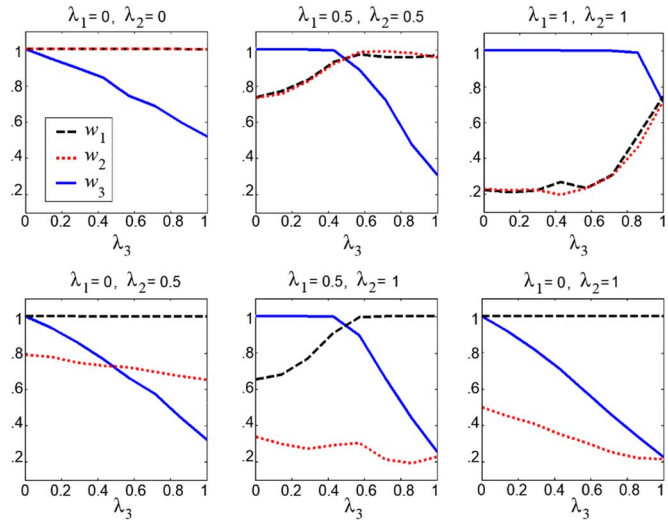

Fig. 8. Mean normalized synaptic weights w_i as a function of noise spike rate showing the synaptic weights of a three synapse neuron encoding the *relative* level of input SNR such that the noisier afferents receive a lower weighting than less noisy afferents. Where $\mathrm{SNR} = 1 : \lambda_i$ and λ_i is the noise spike generation rate at the ith input. The neuron was receiving a random spatio-temporal target pattern corrupted with varying level of Poisson noise for each of the three input channels. In all panels the Poisson noise rate across the first two channels, λ_1 and λ_2, was kept constant while λ_3 was swept from $\mathrm{SNR} = 0$ to $\mathrm{SNR} = 1$ and the average steady state value of the synaptic weights, w_1, w_2 and w_3 are plotted.

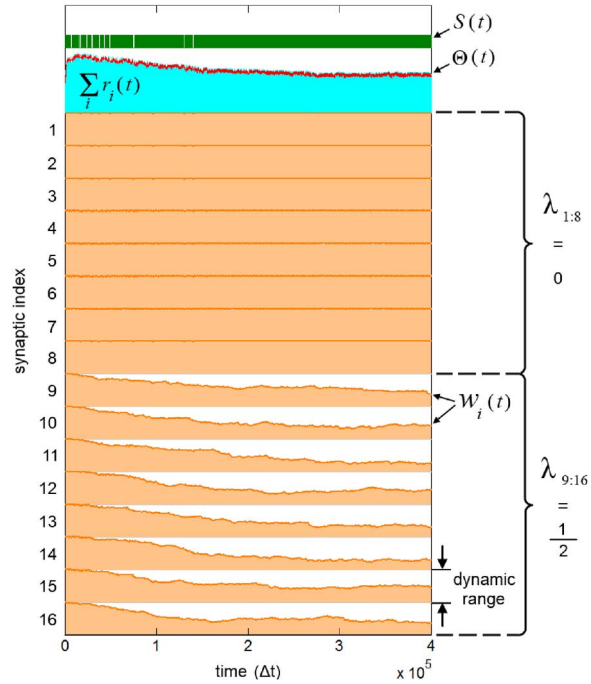

Fig. 9. Time series plot showing the evolution of synaptic weights of a sixteen input neuron from an initial equal value their steady state. Half the inputs, have a noise spike rate $\lambda_{9:16} = 1/2$, ($\mathrm{SNR} = 1 : 1/2$) while the other half are noise free $\lambda_{1:8} = 0$. The equally sized rectangles indicate the equal normalized dynamic range of the synaptic weights while the sixteen indexed plots show the weight of each synapse $w_i(t)$. The noise free synapses, $w_{1:8}(t)$ are all nearly equal and at the top of the normalized dynamic range. The weight of the noisy synapses $w_{9:16}(t)$ fall to approximately half that of the noise free synapses where they remain in a steady state in response to the noise environment. By lowering the synaptic weight of the noisy afferents the neuron reduces their contribution to the somatic membrane potential making the later a less noisy and thus more useful signal during recognition.

synapses with greater noise spikes have their weights pushed down. Over time the forces pushing a synaptic weight up (post-synaptic spikes following presynaptic spikes), and down (presynaptic noise spikes without a postsynaptic output spike) come into balance with each other and the weight normalization produced by the neuron's shift operations. If the Poisson processes generating the noise spikes is homogeneous, i.e. constant over time, then the synaptic weights converge to a steady state value which encodes (and compensates for) the relative signal to noise ratio of each afferent as shown in Fig. 8.

To demonstrate how the synaptic weights of a neuron evolve to their steady state over time, a sixteen input neuron with half its afferents corrupted with noise is shown in Fig. 9. Here, the neuron's input and kernel signals have been removed to more clearly show the relative synaptic weight encoding over time. Additionally the plot shows the neuron's membrane potential $\Sigma r_i(t)$ leveling off at a low steady state value with lower noise and more consistent output spikes $S(t)$.

B. Recognition Performance of the Synaptic SNR Encoding Neuron on Noise Corrupted Data

To quantify the recognition performance of the neuron under various noise regimes, the recognition error of the synaptic weight adapting neuron was measured against a neuron without synaptic weigh adaptation. Both neurons were presented with a random sequence one thousand patterns populated by two random spatio-temporal patterns. For each test some of the input channels were noise corrupted at varying levels as indicated in Fig. 9. The neurons were given one hundred presentations of this noisy randomized data stream within which to perform unsupervised learning of one of the two random patterns after which the pattern for which the neuron spiked most was designated as its target pattern. In the following nine hundred presentations the error in recognition was measured, defined as the number of missed target patterns plus false positive output spikes divided by the total number of target presentations. Fig. 10 demonstrates the effect of the SNR encoding synapse, where over a wide range of noise environments the neuron effectively removes all corrupting input noise and delivers near perfect unsupervised learning and recognition performance. An interesting feature of the neuron is the initial increase in error at the low noise level for the weight adapting neuron, where a minimum noise threshold must be reached to trigger the weight adaptation system. For these tests the w_{rise}/w_{fall} ratio was deliberately chosen to clearly illustrate this behavior. This initial rise can be brought down by choosing a larger w_{fall} term making the neuron more aggressive in terms of shutting down noisy afferents.

C. Implementation in FPGA

To evaluate the hardware requirements of the proposed combined SKAN-STDP system. The system was implemented in an Altera Cyclone-V GX FPGA, a low-end FPGA containing 77,000 programmable logic elements (LEs) in Verilog hardware description language. There were no approximations used in the FPGA implementation as the ideal neuron model is deterministic and was designed using unsigned integer operation. Accordingly all signals in the FPGA and software implementations

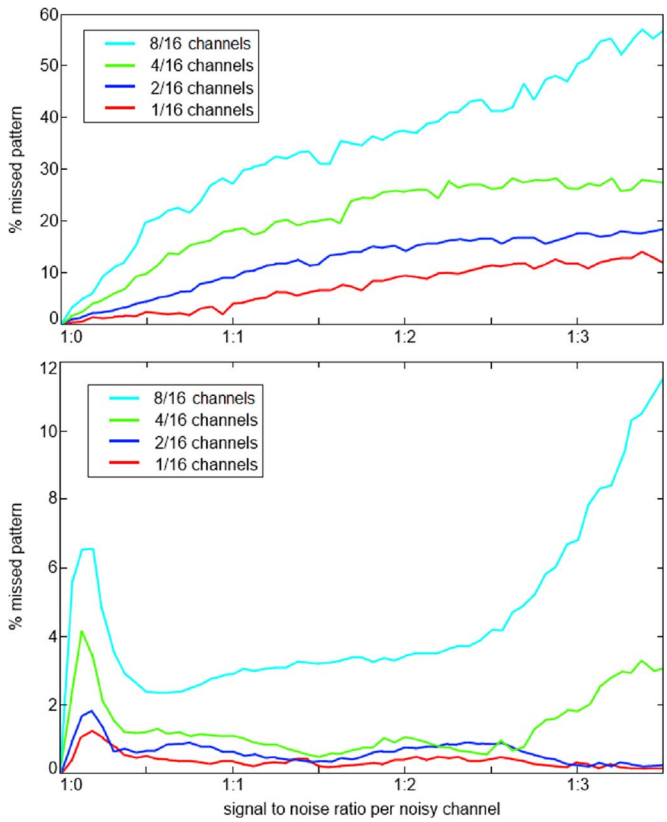

Fig. 10. Enhanced recognition performance via synaptic signal to noise ratio encoding. (top) Recognition error as a function of increasing noise in a kernel adapting neuron with static synaptic weights. The four plots demonstrate increasing recognition error rate both as the number of noisy channels increase and as the SNR per noisy channel deteriorates. Note that as long as the noise corrupted channels are few in number the static SKAN can provide a moderate level of unsupervised recognition performance. (bottom) The same noise regime being applied to the same neuron this time with the dynamically adaptive synaptic weights (note the change in scale for the vertical axis). After perfect performance in the noiseless environments the error rates raise rapidly (SNR $= 1:0$–$1{:}0.25$). The reason for this initial rise in error is that the relative level of noise is simply too low to trigger the neuron's weight adaptation system such that the recognition profile is almost the same as for the static weighted neuron. As the noise level increases the neuron's SNR encoding system switches off the noisy channels and the recognition performance returns to near perfect. At very high noise levels (SNR $< 1:3$), the error rate begins to rise again, this time because the neuron's learning of its "target pattern" during the unsupervised learning period begins to deteriorate.

were determined to be identical. Figures in this work were generated using software simulations of the model which was identical to that realized in hardware. The model's synaptic and somatic parameters were implemented using 12-bit and 20-bit unsigned integers respectively which is at the high end of synaptic signal precision [4]. The hardware usage of the two proposed systems is presented in Table I.

For the static weighted SKAN system additional synapses require 24 single bit registers and approximately 50 ALMs each. For the combined SKAN-STDP system additional synapses require 40 single bit registers and approximately 150 ALMs each. To provide a comparison to the resource usage of the 12-bit SKAN and SKAN-STDP synapses, a single 12-bit unsigned multiplier and its input registers were synthesized on the same device. The resource usage of the multiplier consisted of 24

TABLE I
ALTERA CYCLONE V FPGA RESOURCE USAGE FOR THE STATIC WEIGHT SKAN AND DYNAMIC WEIGHT SKAN MODELS, WITH VARYING NUMBER OF SYNAPSES

Synapses/Neuron		1	2	4	8	16
Single bit Registers	SKAN	38	62	110	206	398
	SKAN + STDP	62	102	182	342	662
Logic utilization (in ALM)	SKAN	69	102	179	358	693
	SKAN + STDP	136	285	521	1004	2030

*An Adaptive Logic Module (ALM) is equivalent to 2.65 logic elements.

single bit registers and 65 ALMs. While utilizing approximately twice the hardware resources of a single multiplier of the same precision, the SKAN-STDP synapse is capable of learning an arbitrary spike delay, generating a membrane potential at the soma, encoding its input signal to noise ratio in its synaptic weight and normalizing the dynamic range of its signals.

D. Example Application: Unsupervised Feature Learning Using a Camera With Noise Corrupted Pixels

Cameras can often suffer from noisy pixels and experimental or neuromorphic cameras are especially prone to this problem. Cameras such as the event based DVS camera [23] can suffer from faulty pixels which generate noisy streams of pulses where there should be no activation and this can have a detrimental effect on upstream recognition systems. Such faults can require on-going examination of the camera by an expert user in order to detect and remove such noise corrupted pixels. Here a synaptic SNR encoding neuron is particularly useful in being able to simultaneously perform both the noise removal and the online unsupervised feature learning task.

To demonstrate the concurrent learning of common features in a sequence of spatio-temporal patterns and block noisy inputs, the neuron was presented with a subset of the zero images in the MNIST handwritten digit dataset which is commonly used for training image processing systems [24]. The pixels in the dataset images were directly mapped to the synaptic inputs of the neuron. However since the neuron receives spikes as inputs, the pixel values needed to be converted to spikes. The simplest approach, which was used here, involved mapping intensity to spike latency, with the brightest pixels arriving first and the darkest arriving last. This transformation was performed by a simple one to one mapping, however, neuromorphic approaches such as use of distributed integrate and fire neurons can also be used to convert real value signals to spike times [25].

To simulate the faulty camera a group of pixel in the central region of the image were corrupted with random levels of noise ($1:1 <$ SNR $< 1:3$) as shown in Fig. 11(a). Here the design choice regarding zero weights referred to in Section III-B was implemented, where once a synaptic weight reached zero value, the synapse, and therefore the pixel, was disabled. This resulted in a system where the noise corrupted pixels were all disabled after at most 189 training images as shown in Fig. 11(d), leaving only the noise free channels for the kernels to train on and generating the receptive field of the neuron shown in Fig. 11(b).

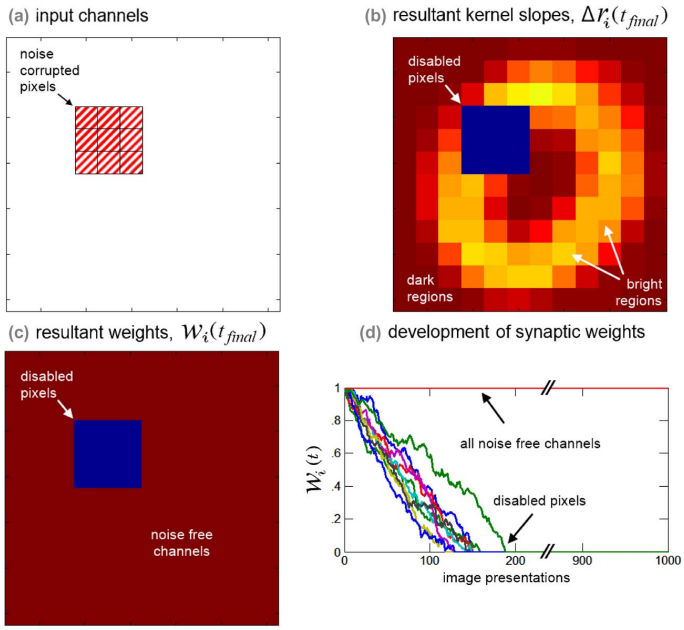

Fig. 11. Unsupervised learning of common features concurrent with SNR encoding synaptic weights on noise corrupted spatio-temporal patterns encoding handwritten zero digits from the MNIST dataset using a single neuron. (a) The input space with corrupted pixels highlighted. (b) The final receptive field of the neuron after exposure to the MNIST zeros. Pixels with higher probability of being dark are more likely to generate late spikes which a correctly trained neuron should encode in the form of narrow kernels or high kernel slopes $\Delta r_i(t)$. Conversely, pixels more likely to be bright should be encoded by lower kernel slopes, as is seen. As a result, the further an input image is from this 'model' of a zero, the weaker the response of the neuron to the image. (c) The final synaptic weights of the neuron showing the disabled pixels. (d) The evolution of the synaptic weights over time. The neuron correctly weighted all the equally noiseless pixels equally high while weakening the weight of the noisy channels until they reached their minimum value at which point they are disabled.

Note that the aggressiveness of pixel removal system can be controlled via the w_{rise}/w_{fall} ratio.

The disabled synapses shown in Fig. 11 can potentially be reused, making SNR encoding synapses not only useful in terms of enhancing the performance of downstream signal processing systems as demonstrated in Fig. 10, but in reconfigurable systems could also enable the potential reallocation of hardware resources to other tasks. This would allow more efficient hardware use in the context of the noise present in the sensors and in the environment. Future work will focus on the use of the neuron model as noise robust hardware implemented spike based unsupervised feature extractor in an unsupervised-supervised recognition system.

V. Conclusion

An extended model of the synapto-dendritic kernel adapting neuron, with a simplified STDP synaptic weight update rule was presented and shown to perform concurrent unsupervised learning of commonly presented spatio-temporal patterns and synaptic encoding of afferent signal to noise ratio. In addition a novel shift based digital normalization algorithm was introduced which allowed synaptic homeostasis or weight normalization without the need for a fixed-point division operation. While the neuron model is a simplified abstraction of highly complex synaptic, dendritic and somatic processes, its adaptive kernels permit the efficient functional modeling of neurons learning of spatio-temporal spike patterns in the presence of varying levels of noise. The implementation of the neuron model in digital hardware shows that neuron's synapses have hardware usage requirements comparable to a single digital multiplier while being able to generate complex computationally useful behaviors. The signal to noise encoding synapses were shown to compensate for afferents corrupted with noise spikes resulting in improved learning and recognition performance across a range of noise environments with relevance to neuromorphic engineering applications such as bio-inspired visual processors.

References

[1] J. Schemmel, D. Briiderle, A. Griibl, M. Hock, K. Meier, and S. Millner, "A wafer-scale neuromorphic hardware system for large-scale neural modeling," in *Proc. IEEE Int. Symp. Circuits and Systems*, 2010, pp. 1947–1950.

[2] S. B. Furber, F. Galluppi, S. Temple, and L. A. Plana, "The SpiNNaker Project," *Proc. IEEE*, vol. 102, no. 5, pp. 652–665, May 2014.

[3] R. Wang, T. J. Hamilton, J. Tapson, and A. van Schaik, "A compact reconfigurable mixed-signal implementation of synaptic plasticity in spiking neurons," in *Proc. IEEE Int. Symp. Circuits and Systems*, 2014, pp. 862–865.

[4] T. Pfeil, T. C. Potjans, S. Schrader, W. Potjans, J. Schemmel, M. Diesmann, and K. Meier, "Is a 4-bit synaptic weight resolution enough?—Constraints on enabling spike-timing dependent plasticity in neuromorphic hardware," *Front. Neurosci.*, vol. 6, p. 90, Jan. 2012.

[5] P. P. San, S. Hussain, and A. Basu, "Spike-timing dependent morphological learning for a neuron with nonlinear active dendrites," in *Proc. Int. Joint Conf. Neural Networks*, 2014, pp. 3192–3196.

[6] 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," *IEEE Trans. Neural Netw. Learn. Syst.*, 2014, IEEE Xplore, Early Access.

[7] S. Mitra, S. Fusi, and G. Indiveri, "Real-time classification of complex patterns using spike-based learning in neuromorphic VLSI," *IEEE Trans. Biomed. Circuits Syst.*, vol. 3, no. 1, pp. 32–42, Feb. 2009.

[8] S. Afshar, L. George, J. Tapson, A. van Schaik, and T. J. Hamilton, "Racing to learn: Statistical inference and learning in a single spiking neuron with adaptive kernels," *Front. Neurosci.*, vol. 8, p. 377, 2014.

[9] M. Rahimi Azghadi, N. Iannella, S. F. Al-Sarawi, G. Indiveri, and D. Abbott, "Spike-based synaptic plasticity in silicon: Design, implementation, application, challenges," *Proc. IEEE*, vol. 102, pp. 717–737, 2014.

[10] S. Hussain, A. Basu, R. M. Wang, and T. Julia Hamilton, "Delay learning architectures for memory and classification," *Neurocomput.*, vol. 138, pp. 14–26, 2014.

[11] Y. Buskila, J. W. Morley, J. Tapson, and A. van Schaik, "The adaptation of spike backpropagation delays in cortical neurons," *Front. Cell. Neurosci.*, vol. 7, p. 192, Jan. 2013.

[12] R. M. Wang, T. J. Hamilton, J. C. Tapson, and A. van Schaik, "A mixed-signal implementation of a polychronous spiking neural network with delay adaptation," *Front. Neurosci.*, vol. 8, p. 51, Jan. 2014.

[13] T. Dowrick, S. Hall, and L. McDaid, "A simple programmable axonal delay scheme for spiking neural networks," *Neurocomput.*, vol. 108, pp. 79–83, May 2013.

[14] T. J. Hamilton, S. Afshar, A. van Schaik, and J. Tapson, "Stochastic electronics: A neuro-inspired design paradigm for integrated circuits," *Proc. IEEE*, vol. 102, no. 5, pp. 843–859, 2014.

[15] R. J. Sofatzis, S. Afshar, and T. J. Hamilton, "The synaptic kernel adaptation network," in *Proc. IEEE Int. Symp. Circuits and Systems*, 2014.

[16] R. J. Sofatzis, S. Afshar, and T. J. Hamilton, "Rotationally invariant vision recognition with neuromorphic transformation and learning networks," in *Proc. IEEE Int. Symp. Circuits and Systems*, 2014.

[17] S. Afshar, G. K. Cohen, R. M. Wang, A. Van Schaik, J. Tapson, T. Lehmann, and T. J. Hamilton, "The ripple pond: Enabling spiking networks to see," *Front. Neurosci.*, vol. 7, pp. 1–12, 2013.

[18] J. Sjöström and W. Gerstner, "Spike-timing dependent plasticity," *Scholarpedia*, vol. 5, no. 2, p. 1362, Feb. 2010.

[19] H. Markram, W. Gerstner, and P. J. Sjöström, "Spike-timing-dependent plasticity: A comprehensive overview," *Front. Synaptic Neurosci.*, vol. 4, p. 2, Jan. 2012.

[20] B. Nessler, M. Pfeiffer, L. Buesing, and W. Maass, "Bayesian computation emerges in generic cortical microcircuits through spike-timing-dependent plasticity," *PLoS Comput. Biol.*, vol. 9, 2013.
[21] G. G. Turrigiano, "The self-tuning neuron: Synaptic scaling of excitatory synapses," *Cell*, vol. 135, no. 3, pp. 422–435, Oct. 2008.
[22] E. Marder and J.-M. Goaillard, "Variability, compensation and homeostasis in neuron and network function," *Nature Rev. Neurosci.*, vol. 7, no. 7, pp. 563–574, Jul. 2006.
[23] P. Lichtsteiner, C. Posch, and T. Delbruck, "A 128 × 128 120 dB 15 μs latency asynchronous temporal contrast vision sensor," *IEEE J. Solid-State Circuits*, vol. 43, no. 2, pp. 566–576, 2008.
[24] Y. LeCun and C. Cortes, The MNIST Database of Handwritten Digits [Online]. Available: http://yann.lecun.com/exdb/mnist
[25] S. Afshar, O. Kavehei, A. van Schaik, J. Tapson, S. Skafidas, and T. J. Hamilton, "Emergence of competitive control in a memristor-based neuromorphic circuit," in *Proc. Int. Joint Conf. Neural Networks*, 2012, pp. 1–8.

Saeed Afshar (M'12) received the B.E. degree in electrical engineering from the University of New South Wales, Sydney, NSW, Australia, in 2014.

Currently, he is a Research Assistant in the Bioelectronics and Neuroscience Group within The MARCS Institute at the University of Western Sydney, Sydney, NSW, Australia. His research interests include neuromorphic computational architectures and biologically inspired visual and control systems.

Libin George (S'08) received the B.E. (first class honors) degree in electrical and electronic engineering from the University of Auckland, Auckland, New Zealand, in 2008.

Currently, he is working toward the Ph.D. degree in electrical engineering at the University of New South Wales, Sydney, NSW, Australia. Since 2013, he has been a Research Assistant within the Bioelectronics and Neuroscience Group at The MARCS Institute, University of Western Sydney, Sydney, NSW, Australia. His research interests include analog and mixed-signal circuit design, power management ICs, and integrated DC-DC converters.

Chetan Singh Thakur (S'14) received the M.Tech. degree in biomedical engineering from the Indian Institute of Technology, Bombay, India, in 2007.

He is working toward the Ph.D. degree in the Bioelectronics and Neuroscience Group at The MARCS Institute, University of Western Sydney, Sydney, NSW, Australia. He has worked with Texas Instruments as a Senior Integrated Circuit Design Engineer in the area of mobile processors. His research interests include neuromorphic engineering, stochastic electronics, and computational neuroscience.

Jonathan Tapson (M'05) received the B.Sc. degree in physics, the B.Sc. degree in electrical engineering, and the Ph.D. degree in engineering from the University of Cape Town, Cape Town, South Africa.

Currently, he is Director of The MARCS Institute, University of Western Sydney, Sydney, NSW, Australia, which he joined in June 2011, having previously been Head of Electrical Engineering at the University of Cape Town. His research interests are in bio-inspired sensors and systems. Along with his coauthors, he is working on a major program of analog and mixed signal IC design in the area of stochastic electronics.

Dr. Tapson is a former President of the South African Council on Computation and Automation. He is a Fellow of the South African Academy of Engineering.

André van Schaik (M'00–SM'02–F'14) received the M.Sc. degree in electrical engineering from the University of Twente, Enschede, The Netherlands, and the Ph.D. degree in electrical engineering from the Swiss Federal Institute of Technology (EPFL), Lausanne, Switzerland, in 1990 and 1998, respectively.

Currently, he is a Professor of Bioelectronics and Neuroscience at The MARCS Institute, University of Western Sydney, Sydney, NSW, Australia. His research focuses on neuromorphic engineering, bioelectronics, and neuroscience. He has authored more than 150 papers and is an inventor of more than 30 patents. He is a founder of three start-up companies: VAST Audio, Personal Audio, and Heard Systems.

Philip de Chazal (M'94–SM'14) received the B.E. degree in electronic engineering, and the M.Biomed.E. and Ph.D. degrees in biomedical engineering from the University of New South Wales, Sydney, NSW, Australia, in 1989, 1995, and 1999, respectively.

Currently, he is a Professor and the ResMed Chair of Biomedical Engineering in the School of Electrical and Information Engineering, University of Sydney, Sydney, NSW, Australia. His research interests include signal processing and pattern recognition for biomedical applications. He has authored more than 115 publications and is a named inventor on eight patents and patent applications.

Tara Julia Hamilton (S'97–M'00) received the B.E. degree (first class honors) in electrical engineering and the B.Com. degree from the University of Sydney, Sydney, NSW, Australia, in 2001, the M.Engineering.Sc. degree in biomedical engineering from the University of New South Wales, Sydney, NSW, Australia, in 2003, and the Ph.D. degree from the University of Sydney in 2008.

Currently, she is a Senior Research Lecturer in bioelectronics and neuroscience within The MARCS Institute, University of Western Sydney, Sydney, NSW, Australia. Her current research interests include neuromorphic engineering, mixed-signal integrated circuit design, and biomedical engineering.