Neuromorphic Spiking Neural Network With Synaptic Normalization Mechanisms For Edge Orientation Classification In Artificial Bio-Inspired Skin

Ali Dabbous[1,2*], Michele Mastella[3,4], Elisabetta Chicca[3,4], Maurizio Valle[1], and Chiara Bartolozzi[2]

[1]Connected Objects Sensing Materials Integrated Circuits (COSMIC) lab. University of Genova-DITEN, Genova, Italy
[2]Event-Driven Perception for Robotics (EDPR) lab. Istituto Italiano Di Tecnologia (IIT), Genova, Italy
[3]Bio-Inspired Circuits and Systems (BICS) Lab. Zernike Institute for Advanced Materials (Zernike Inst Adv Mat), University of Groningen (Univ Groningen), Nijenborgh 4, NL-9747 AG Groningen, Netherlands
[4]CogniGron (Groningen Cognitive Systems and Materials Center), University of Groningen (Univ Groningen), Nijenborgh 4, NL-9747 AG Groningen, Netherlands
[*]ali.dabbous@edu.unige.it

ABSTRACT

Touch sensing plays an important role in exploring and distinguishing objects. As such, robots and users of hand prostheses could easily manipulate objects if endowed with the sense of touch. Towards this goal, information about touched objects and surfaces has to be inferred from raw data coming from the sensors. The orientation of edges, which is employed as a pre-processing stage in both artificial vision and touch, is a key indication for object discrimination. Inspired on the encoding of edges in human first-order tactile afferents, we present a biologically inspired, spiking models architecture that mimics human tactile perception with computational primitives that are implementable on low-power subthreshold neuromorphic hardware. The network uses three layers of Leaky Integrate and Fire neurons to distinguish different edge orientations of a bar pressed on the artificial skin of the iCub robot. We demonstrate that the network can learn the appropriate connectivity through unsupervised spike-based learning, by implementing a random structure of overlapping receptive fields. The unconstrained and random structure of the connectivity among layers can produce unbalanced activity in the output neurons, which are driven by a variable amount of synaptic inputs. We explored two different mechanisms of synaptic normalization (weights normalization and homeostasis), defining how this can be useful during the learning phase and inference phase. The architecture is successfully able to discriminate between 35 orientations of 36 (0^o to 180^o with 5^o step increments) with homeostasis and weights normalization mechanisms.

Glossary

CDC Capacitance to Digital Converter Integrated Circuits. 11

L1 Layer-One. 2–4, 6, 9, 11, 12

L2 Layer-Two. 2–4, 6–9, 11–14

L3 Layer-Three. 2–4, 6–9, 11–14

LIF Leaky Integrate and Fire. 2–4, 11, 12

RF Receptive Field. 2–4, 7, 9–13

SDSP Spike Driven Synaptic Plasticity. 4–8, 11–13

STDP Spike Timing Dependent Plasticity. 5

VLSI Very large-scale integration. 15

WTA Winner-Take-All. 2–4, 6, 11, 14

YARP Yet Another Robot Platform. 11

1 Introduction

Touch sensing plays an important role in humans daily life. Tasks like exploring, grasping and manipulating objects[1] deeply rely on it. As such, hand prosthesis and robots endowed with the sense of touch can more easily interact with things and other agents physically[1]. In robotics, in order to mimic the tactile capability of biological touch, several types of tactile sensors have been developed based on various techniques (e.g., capacitive, piezoresistive, optical, magnetic, binary, and piezoelectric)[2–4]. Most of these sensors rely on circuits that sample the environment at a fixed rate and generate discrete data, even when the device is not in contact with a stimulus, and introducing a latency that can be critical in dangerous situations, when a fast reaction time is needed as escape reflex from dangerous contacts. Biological sensors, on the other hand, react to changes in the sensory signal, encoding information in form of sequences of digital pulses (or spikes) only upon the detection of contact. During contact, multiple types of sensors on the skin (called mechanoreceptors) are stimulated. This stimulation activates peripheral neurons, communicating with the brain details about the object‘s texture, shape, and size[5].

Inspired by the computational efficiency of biology, neuromorphic tactile sensing has been gaining an increasing appeal, using spiking neurons and artificial sensors to mimic the mechanoreceptor’s response and the processing happening in the somatosensory nervous system[6,7]. So far, neuromorphic tactile sensing has shown its potential in upper limb neuroprosthesis, where neural interfaces provide tactile feedback to amputees[8,9]. It has also been employed for classifying different textured surfaces, differentiating various indentation conditions, and modulating prosthesis grip control, in neurorobotic applications[10–12].

In most robotic and prosthetic applications, edge orientation detection is often used as pre-processing stage, crucial for contour following and object manipulation[13,14]. As such, we aim at implementing edge orientation selectivity in tactile neuromorphic systems, taking inspiration on the human glabrous skin.

Edge orientation-selective neurons have been observed in the first-order tactile afferents of the human fingertip[13,15]. This selectivity can be modelled with a network of spiking neurons[15]: the first layer consists of neurons with overlapping RFs (each neuron possesses a distinct distribution of highly sensitive zones, the mechanoreceptors, on the skin[16]); the second layer decodes the correct orientation of the stimulus, by detecting temporal coincident activation within the first layer. This structure has been presented in Parvizi et al. (2021)[17], where 3 layers of spiking neural models recreate the somatosensory path, from the afferents to the cortex. In[18] authors proposed a model-based spatiotemporal correlation matching method to estimate the orientation of the boundary edge while a piezoresistive tactile sensor array attached to robotic arm palpates over the object. In similar experiment, authors in[14] contacted static edge perception experiments by tapping the stimuli with tactile sensor to demonstrate passive tactile perception for contour following. The edge detection was done only on right angles. Other studies tried to mimic the behaviour of the somatosensory path using more traditional methods, e.g. Wong et al. (2013)[19] where the authors used AI-based vector regression and Rongala et al. (2019)[20] where the authors used differential delay lines and spike-based classification. However, such solutions lack the possibility to be embedded on robots or prosthetic devices, due to the need for offline learning and the presence of structures not easily transferable in silicon. In such situations, where energy and space are major constraints, a hardware implementation with online learning and low power devices is usually preferred. Such a methodology enables the system to perform end-to-end computation from the sensors to the processing and classification, consuming low power. In this paper, we present a neuromorphic model for edge orientation selectivity that relies on event-driven acquisition and unsupervised spike-driven learning, based on computational primitives that are implementable on low-power mixed-mode sub-threshold neuromorphic hardware. In this work, we used the capacitive skin of the iCub robot[21] as front-end sensors, connected to a three-layers network of Leaky Integrate and Fire (LIF) neurons, as visible in figure 1A. Layer-One (L1) converts the analog values measured by the pressure-sensitive capacitors into spike trains. Specifically, the analog pressure value is used as a current injected in the neuron‘s membrane. The neuron is then spiking and its output firing rate is proportional to the pressure, mimicking the sustained response of SA-I types of mechanoreceptors. Layer-Two (L2) is the Receptive Field (RF) layer, where the activity of multiple mechanoreceptors from L1 converges to it. As shown in[13,22], the connectivity between these layers performs better when the synapses are sparse and not clustered depending on spatial area on the skin. For this reason we implemented the structure of RFs based on a matrix of connectivity between L1 and L2 neurons generated randomly, such that, each neuron in L1 is connected to only one neuron in L2 as shown in figure 1B. Layer-Three (L3) decodes different orientations using the temporal coincidence activation of L2 neurons. The connection between these two layers is all to all and the correct connections are learnt during operation using local unsupervised learning as shown in figure 1C. For better selectivity, L3 employs Winner-Take-All (WTA) structure, composed of a global inhibitory neuron[23]. The

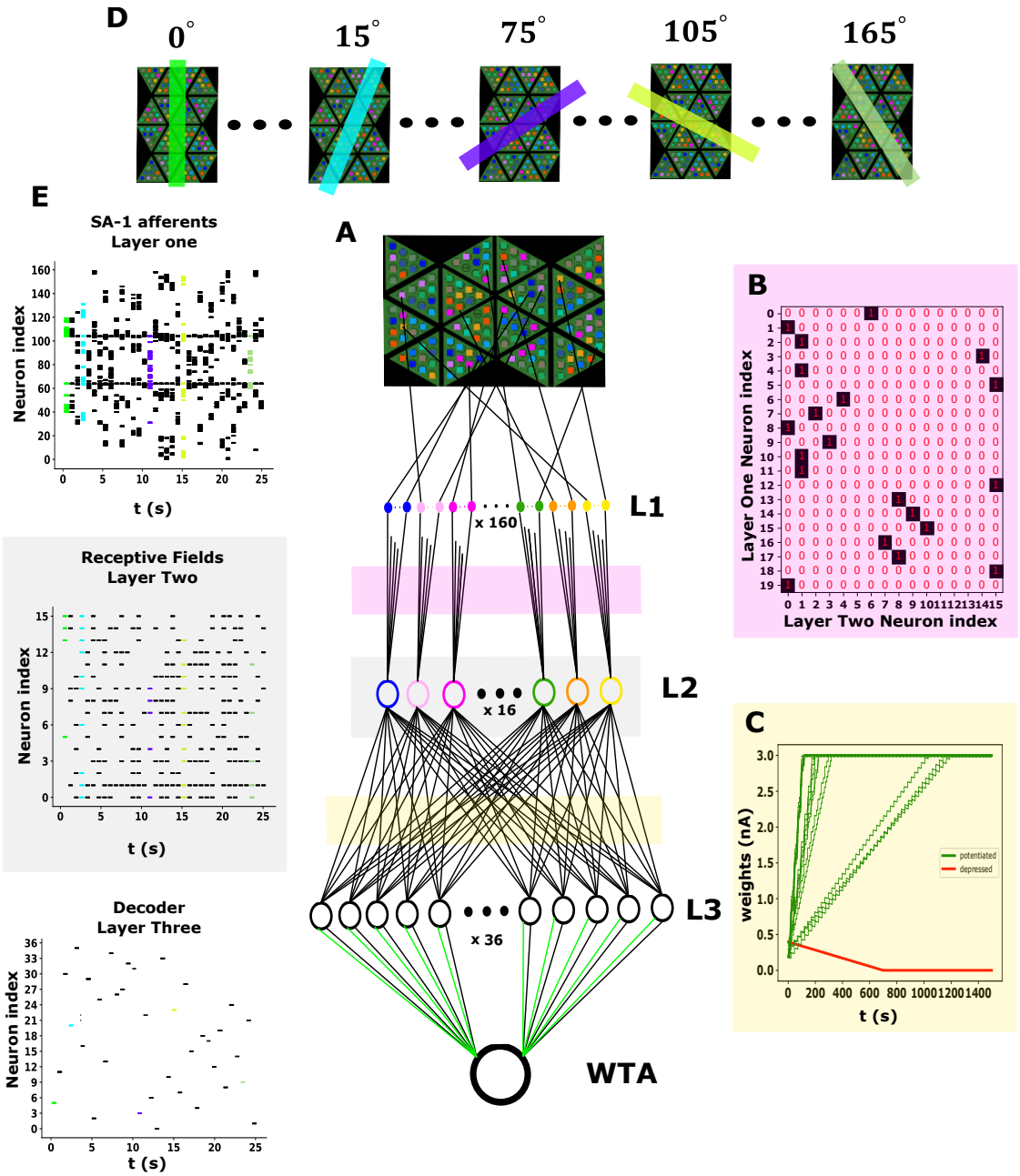

Figure 1. Spiking neural network for edge orientation selectivity. (A) Network architecture comprising skin patch and three layers of LIF neurons. L1 neurons convert the measured capacitance value into spike trains (SA-I afferents); L2 gathers input from multiple L1 neurons and having spatially distributed, overlapping, RFs; L3 neurons receive input from L2, each neuron of L3 is selective to a specific input orientation. Recurrent connectivity by means of a global inhibitory neuron ensures that a single L3 neuron is active, implementing a form of WTA competition. (B) Randomly generated connectivity matrix between L1 and L2 (showing only the first 20 L1 neurons) ; each neuron in L1 (rows) is connected to only one neuron in L2 (cols), each L2 neuron can receive input from a random number of L1 neurons. (C) Variation of synaptic weights connecting all neurons in L2 to one neuron in L3 during learning. Synaptic weights bistable and after learning can be either potentiated (green) reaching w_{max} or depressed (red) reaching w_{min}. (D) Examples of different bars pressed on the skin at different orientations (0^o, 15^o, 75^o, 105^o, 165^o). (E) Raster plots of SA-1 afferents (L1), RFs (L2), and decoder (L3) as a function of time for 36 orientations (0^o to 180^o with 5^o step increments), the colored spike trains match the oriented bar in part (D), and the black spike trains represent the remaining orientations.

unconstrained and random structure of the connectivity among layers can produce unbalanced activity in the output neurons, that are driven by a different number of synaptic inputs. Two different forms of activity balancing (weights normalization and homeostasis) can be implemented to create an homogeneous network that enables the system to discriminate between 35 orientations of 36 (0^o to 180^o with 5^o step increment).

Results

Touch Afferents: Layer-One Neurons

The constant pressing of a bar over the skin generates a sustained constant in the sensors. Depending on the applied stimulus, different sensors are activated at the same time. For example, a vertical bar placed in the middle of the skin results in a response only in central sensors. Each stimulus generates different analog signals, proportional to the applied force, that are fed as a current into L1 neurons, modelled as Leaky Integrate and Fire (LIF). This process is similar to what happens in the human skin, where the Merkel disks[24] convert stable pressure into analog ion currents that stimulate afferent neurons that produce tonic spiking, mimicking the response of SA-I afferents[25].

Receptive fields: Layer-Two Neurons

Once the tactile stimulus has been converted into spikes, the signal needs to be funneled into a further layer: the Layer-Two (L2). In this part the dimensionality of the stimulus gets compressed from 160 to 16 elements. This allows efficient transmission across the acquisition path, with limited information loss.

Receptive Field Structure Differently from the case of visual RFs[26], a common agreement on tactile RFs is still missing. Some early studies suggested that the tactile RFs are similar to the visual ones but shaped differently due to the different stimuli they interface to. Vision seems handling more natural scenes with several objects, and a heterogeneous background, while touch is mainly related with surface textures and regular repetitions. Touch stimuli seem in general sparser than visual stimuli[27]. Some more recent studies[13,28] suggest that this sparsity greatly influences the mechanism behind RFs in touch. The sparsity of the signal could have led, according to the studies, to a semi-random distribution of the RF. The authors also suggest that an explanation behind semi-random distributions in RF comes from compressive sampling theory, in which high sampling accuracy can be obtained using random bases[13]. A previous analysis was conducted to propose a RF structure on the iCub skin to maximize the mutual information transmitted between L1 and L2[22]. The outcome of the study was that, in presence of a noiseless stimulus, higher information is delivered by the random shaped structure, while a more structured RF performed increasingly worse when reducing the length of the pressed bar. In this structure the different sensors present on the skin patch are assigned randomly to 16 different RF. The activation of one of the afferent in the RF activates the whole RF. More the active afferents in the RF, higher the RF activity. In the following parts, for reference, a random distribution has been generated and kept identical for all the simulations.

Decoder Layer: Layer-Three Neurons

L3 neurons decode the orientation of a bar pressed on the skin patch by detecting the temporal coincident activation of several L2 neurons. For a specific orientation, multiple L3 neurons could detect coincident activation in L2 neurons, as their RFs overlap. However the L3 neuron that corresponds to the correct orientation will be more active than those that are not properly tuned to the input orientation. A WTA network ensured that the activity of weakly active neurons was suppressed and only the most active neuron kept firing.

Winner-Take-All The WTA competition is implemented by a global inhibitory neuron, that receives excitatory inputs from all 36 neurons of L3 (black synapses in figure 3-A), and in turn, stimulates the 36 inhibitory synapses of the 36 excitatory neurons of L3 (green synapses in figure 3-A), such that only the neuron receiving the highest input can be active.

Projections between L2 and L3 neurons The aim of this study was two folds: (1) assessing the system's ability to build the right connectivity between L2 and L3 neurons through local unsupervised learning and (2) finding the optimal structure of connectivity for orientation selectivity. To achieve these goals, first we connected L2 and L3 neurons with random all-to-all weights, after which we endowed the network with WTA for self supervision, and then we endowed the network with local unsupervised synaptic learning rule between L2 and L3. In this work, the importance of the WTA competition is to have a single active neuron per orientation, in order to run the learning and potentiate the correct synapses, which connects the activated L2 neurons to that neuron.

Spike Driven Synaptic Plasticity Initially proposed by Brader et al. in 2007[29], the Spike Driven Synaptic Plasticity (SDSP) is counted as one of the most biologically plausible learning rule which it is implementable on low power neuromorphic hardware. Learning rules are algorithms performed in the synapses: their transmission capability (or weight) depends on the activities of the neurons around them. More specifically, the way the weight changes depends on the behaviour of the neuron

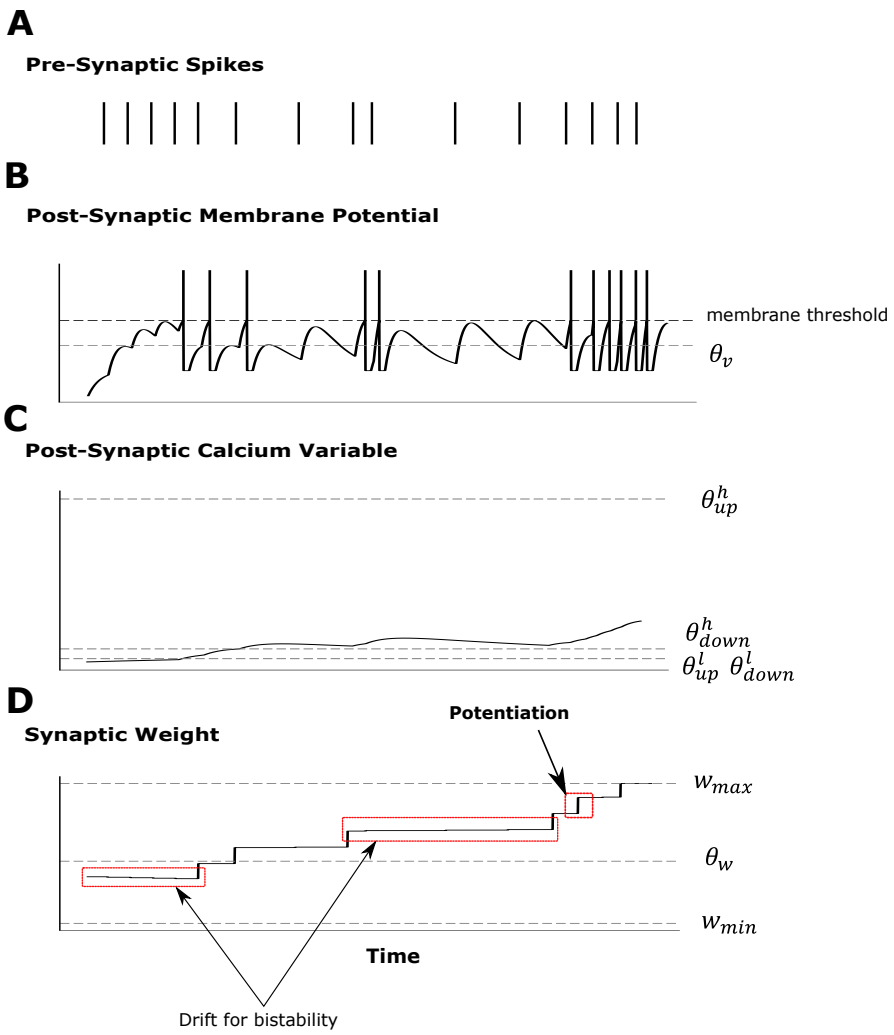

Figure 2. Toy example of two neurons connected via SDSP synapse. (A) Pre-synaptic neuron spikes. (B) Post-synaptic membrane potential. (C) Post-synaptic calcium variable. (D) Synaptic weights variation as a function of time.

before the synapse (called PRE) and the neuron after the synapse (called POST). The synaptic weights in SDSP are dependent on two variables, as shown in figure 2. In the first part it's shown the first variable: the state of the postsynaptic membrane potential at the time of presynaptic spike (see figure 2B), while in the second part the second variable: the concentration of the slow postsynaptic calcium (see figure 2C). This rule has been already tested by Brader et al.[29] who demonstrated the ability of 2000 input neurons in classifying thousands of overlapping patterns from the MNIST dataset. In the same work, they also demonstrated the ability of the network to generalize to other classification tasks with a similar or better performance than artificial neural networks (ANN). Finally, they compared between the long term performance of Spike Timing Dependent Plasticity (STDP) and SDSP and they found that they were similar, but SDSP presented better generalization properties and biophysical accuracy. In addition,[30] proposed an online learning digital spiking neuromorphic processor that utilized on-chip learning for the classification of 16x16 MNIST images. These findings establish that the SDSP learning rule is not only biologically realistic, but also can be implemented on low power mixed mode sub-threshold neuromorphic hardware.

Figure 2 shows the results of a simulation where two neurons are connected in a PRE-POST link using SDSP-equipped synapses for a 500 ms stimulation period. Figure 2-A represents the presynaptic spike train. When input spikes arrive from the presynaptic neuron, they increase the membrane potential of the postsynaptic neuron above the SDSP membrane threshold θ_v. This behaviour goes on until it reaches the neural membrane threshold, this triggers a spike in the post-synaptic neuron, as shown in figure 2-B. According to the SDSP rule, every time the post-synaptic neuron emits a spike, the post-synaptic calcium variable C(t), which represents calcium concentration, increases with a fixed value J_C. This variable is therefore proportional to the instantaneous neuron's spiking activity and it decays with a fixed time constant, as shown in figure 2-C. Under specific conditions related with the calcium concentration and POST's membrane potential (see methods sections), the presynaptic

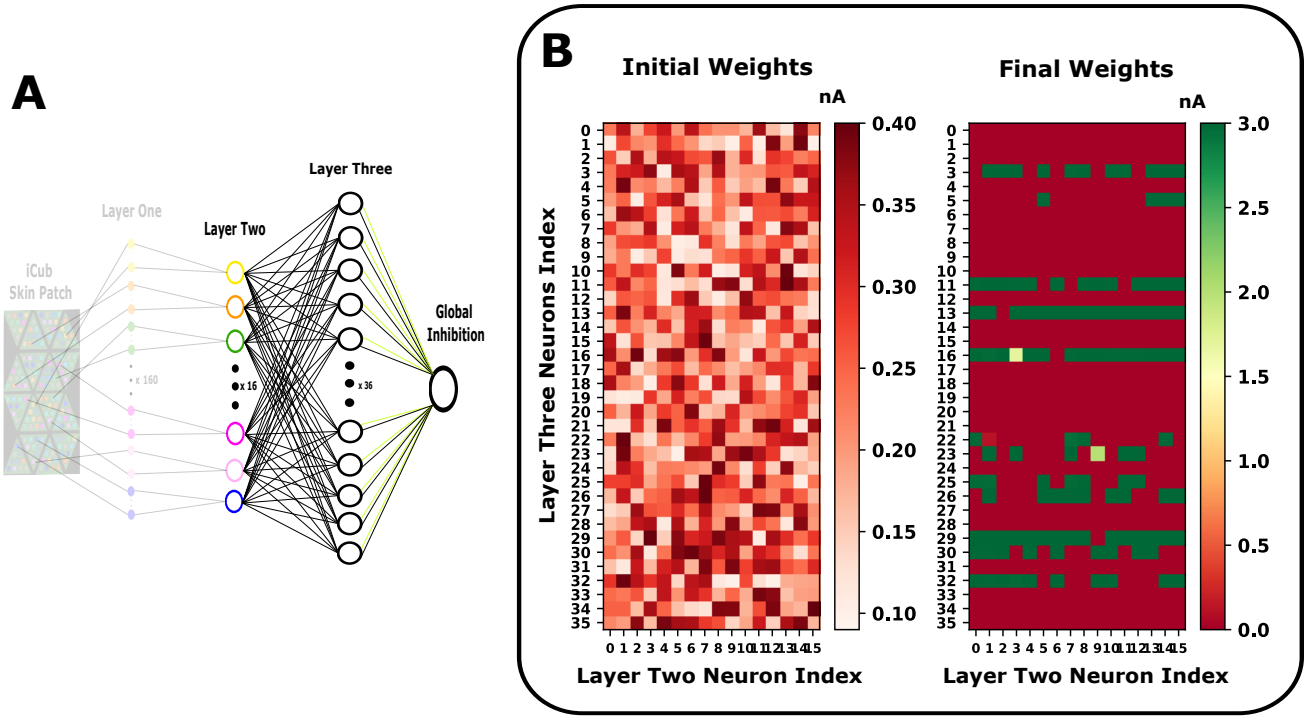

Figure 3. SDSP Learning rule with Baseline Network. (A) Baseline network with all to all connection between L2 neurons and L3 neurons endowed with SDSP synaptic learning rule to build the required connectivity through potentiation or depression of the synaptic weights and to learn an appropriate connectivity patterns for the classification. (B) Synaptic weights at the initial and final state of the simulation; (left) The synaptic weights between L2 and L3 were initialized using uniform random distribution such that all the weights were depressed (red boxes); (right) The synaptic weights at the end of the learning, each small box represents the synaptic weights of one neuron in L2 (x-axis) to one neuron in L3 (y-axis), where most L2 neurons to L3 neurons connecting weights were still depressed (red) and some potentiated (green).

spike triggers a variation in the internal weight of the synapse (W), increasing it by a value A_+, as shown in figure 2-D. Despite the instantaneous adaptation, W is converging toward one of two stable states (W_{max} or W_{min}). The direction of the convergence depends on whether the current value is above or below a given threshold θ_w, as well as, the synapses has efficacy J_+ if (W) is greater than θ_w, otherwise the synapse has efficacy J_- (see method sections), as shown in figure 2-D.

Learning Orientations with Random Receptive Fields

The goal of this work was to make the network learn the different orientations using temporal coincidence. In this work, three different networks are studied. The three networks share the same common structure: three layers of neurons, a random connection between L1 and L2, and a learning connection between L2 and L3. The difference between these three networks is that while the first network is equipped only with a WTA (Baseline Network), in the other two there is weight normalization (Weight Normalization Network) and homeostasis (Final Network), respectively. These two techniques are used to balance the activity of the output neurons.

Baseline Network The baseline network was trained with multiple consecutive stimuli presentations, at randomly selected orientations in the range 0^o - 180^o (5^o resolution). During learning, the synapses connecting L2 to L3 were potentiated or depressed depending on the reciprocal activation of L2 neurons.

Figure 3-B shows the synaptic weights before and after learning. The synaptic weights were initialized using an uniform random distribution (figure 3-B left) such that all the weights were depressed. After learning, only a small percentage of weights were potentiated (green squares in the L2 and L3 connectivity matrix of figure 3-B right) and, correspondingly, several L3 neurons did not learn any of the orientations presented during the learning phase, in which all the synapse weights connecting to them were still depressed (red squares in the L2 and L3 connectivity matrix of figure 3-B right). Nevertheless, most likely a subgroup of output neurons learnt several orientations, leaving some output neurons non-selective to any orientation.

To illustrate the spiking response of L3 for different edge orientations, we fed the network with 36 orientations ranging from 0^o to 180^o with step 5^o. Left part of Figure 5-B represents the spiking responses of L3 neurons with baseline network. In

this case it's visible that while some neurons responded to several patterns some other neurons were unresponsive.

For quantitative assessment, we divided the learning simulation into 10 epochs. In each epoch, we extracted the weights and tested the network on two different datasets. The first dataset was composed of 36 orientations ranging from 0^o to 180^o with 5^o step increments, randomly repeated for many times, where in each dataset the pressure is constant. The second dataset was similar to the first except that the pressure was varying. We used mutual information to compute the number of orientations and the information the network decodes at L3 (see method section). Figure 6 shows the mutual information as a function of number of epochs. In each epoch, we computed the number of orientations detected using the mutual information (number of orientations = 2^{MI}). As such, using the first dataset, the baseline network was able to detect only six orientations at the end of learning, whereas, using the second data set, the network was able to detect only three orientations. Figure 6, shows that the baseline network doesn't recognize all the possible input orientations. This behaviour is given by the presence of different amount of excitatory currents arriving from L2 to L3 neurons at each orientation. In fact, due to the non-uniform distribution of taxels in the L2 RF, it can happen than an output neuron encoding a specific orientation which has just three L2 neurons connected to it will always spike weaker that an output neuron that has six L2 neurons connected to it (middle part of figure 1-E). This becomes even more clear if we consider that several neurons at the L2 are encoding in their spiking different orientations, making it difficult for the network to differentiate between those.

Weights Normalization Network Given the problem highlighted in the baseline example, a way to reduce the unbalanced input current from L2 to L3 was tackled. Specifically, to assure that the response of different neurons competing with each other was the same, regardless of the quantity of input synapses connected to them, weight normalization was used. The latter is a technique where synapses' input is reduced according to some algorithm in the network. In this work, we changed the current magnitude of each synapse depending on how many active synapses were connected to a L3 neuron (i.e. how many weights were potentiated).

Early studies have suggested that weights normalization could be one of the key mechanisms for classification problems in SNNs. A previous work from Mostafa et al.[31] utilized weight normalization in a MNIST classification task, setting the normalization weights as the following: the neuron spikes when its membrane potential crosses a firing threshold which was set to 1, where the firing threshold was dynamically changed using spiking activity. In another work from Soures et al.[32], the variation in synaptic weights was considered as a function of the activity of a larger set of neurons. As they state, the normalization of synaptic weights enables the network to achieve better classification accuracy through adding global constraints to the synapse‘s strength. When the sum of the currents arriving from pre-synaptic neurons to post-synaptic neurons exceeds a certain threshold, the synaptic weight decreases, such that the current remains within the identified threshold.

In the weights normalization network, we introduced a component that reduces the current coming from the synapses looking only at the weights of the connection to a specific neuron. This way of implementing synapse normalization has the advantage of implementing local changes without the need of global variables. Using the same dataset as the previous example, we studied how the weights normalization network was able to discriminate between different orientations. The weight normalization was especially useful during learning: synaptic weights, driven by SDSP rule, were used to define dynamically the current that each synapse was able to conduct. When a weight between one L2 neuron and one L3 neuron exceeded a normalization threshold, then all the synapses connected to that L3 neuron were reduced in magnitude, as shown in figure 4-A. This resulted in a balance of activity between different L3 neurons, helping the learning procedure. As in the baseline network, the synaptic weights between the two layers were initialized using uniform random distribution (figure 4-B to the left) with all the weights depressed at the initial state. After learning, synaptic weights progressively change to converge to stable connectivity patterns, as shown in figure 4-B to the right. Thanks to SDSP learning rule and weights normalization mechanism, each neuron in L3 learned a specific orientation. To study the accuracy performance of the network with different initialization of the weights, we repeated the simulation for 10 different trials, in each trials the initial weights randomized in different distributions. Figure 4-C shows the number of detected orientations at each trial, where the number of detected orientations looks similar for all trials.

Middle part of figure 5-B shows the raster plot of L3 neurons with weights normalization network, where for each orientation only one neuron is firing.

Figure 6-A shows the improved performance of the weight normalization over the baseline network. Whereas Figure 6-B shows that the weight normalization was not able to infer all the orientations when pressure variation was present. The pressure heterogeneity is an important feature in a network that wants to detect real-world stimuli.

Homeostasis Network As an alternative to weight normalization to balance the neurons‘ synaptic drive and allow the network to discriminate between different orientations, we explained the use of homeostasis between L2 neurons and L3 neurons. Synaptic homeostasis is a mechanism observed in biological neural systems that maintains a homogeneous response from neurons within a given operating range[33]: when neuron‘s spiking activity stably leaves given boundaries, their synaptic drive is scaled to restore the activity within their functional operating range. The process is usually very slow and accounts for

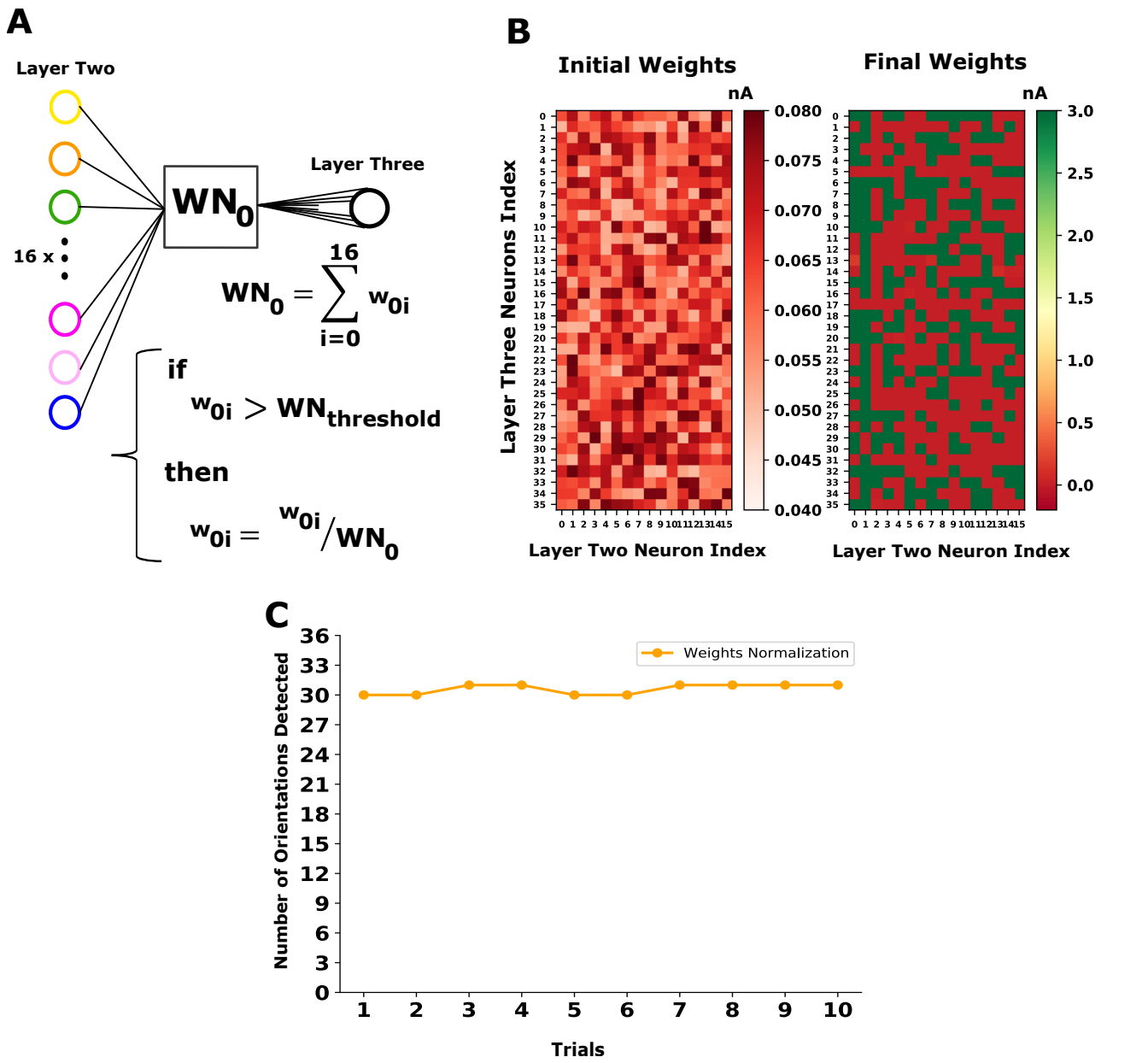

Figure 4. SDSP learning rule with weights normalization network. (A) Schematic representation of the weights normalization mechanism between L2 neurons and L3 neurons; synaptic weights connecting all L2 neurons to a single neuron in L3 were normalized by dividing the weights by the summation of the activated synapses (i.e., how many weights exceeds the normalization threshold). (B) Synaptic weights at the initial and final state of the simulation; (left) The synaptic weights between the two layers were initialized using same synapses random distribution as the previous example; (right) The synaptic weights at the end of the learning, each row represents the synaptic weights connecting all neurons in L2 to one neuron in layer-three state: depressed (red) or potentiated (green), where the weights normalization mechanism help the learning procedure in a balance of activity between different neurons and converging the synaptic weights to stable connectivity patterns. (C) Number of orientation detected by weights normalization network for 10 different trials, by changing the weights initialization random values in each trial.

dioramic drifts of activity, without interfering with signal processing.

Several different types of homeostatic mechanisms have been observed in biological neural networks, comprising both forms of synaptic plasticity mechanism and modification of intrinsic properties of the excitability of the neuron[34–36]. Global homeostatic synaptic scaling allows for the control of the network's overall stability, while complying to the need for change

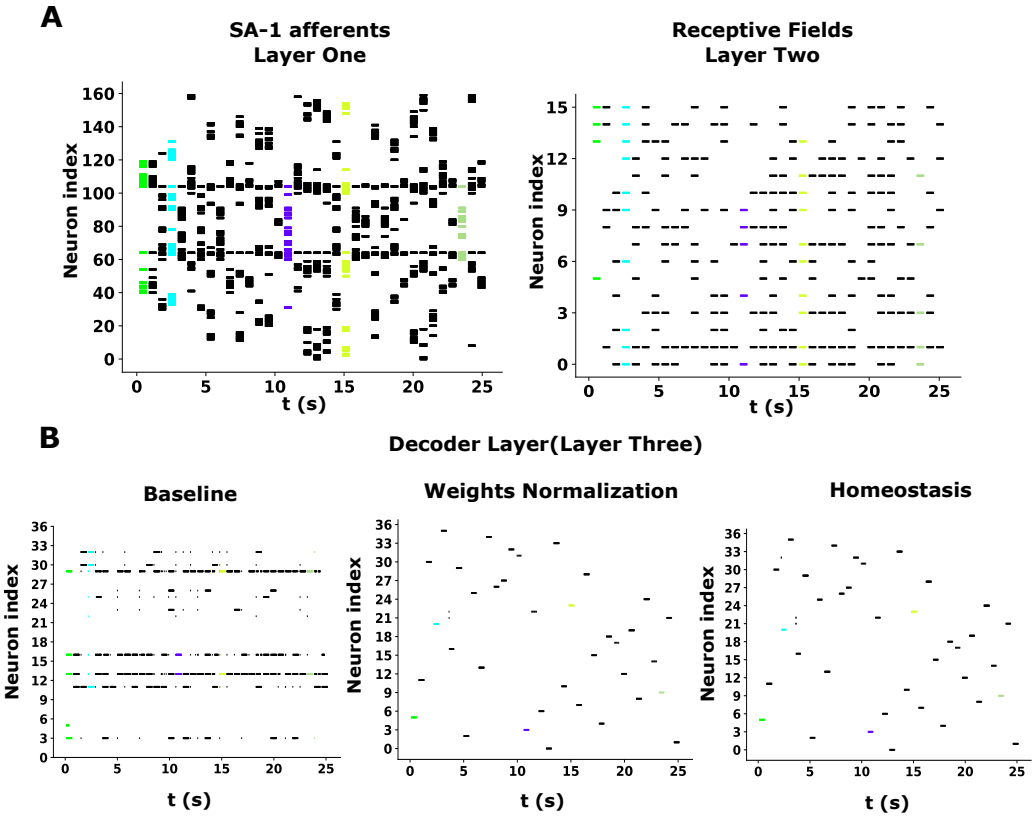

Figure 5. Spiking responses of the three layer of the neuromorphic network architecture. (A) Raster plot of L1 neurons (SA-1 afferents) as a function of time for 36 orientations (0^o to 180^o with step 5^o increments) (left), raster plot of L2 neuron (RFs) (right). (B) L3 spiking responses; (Left) raster plots of L3 neurons as a function of time for 36 orientations (0^o to 180^o with step 5^o increments) with baseline network, (middle) raster plot of L3 neurons with weights normalization network, such that for each orientation only one neuron fire a spikes, (right) raster plot of L3 neurons with homeostasis network.

(or learning), to adapt to the statistic of the input signals[37]. In[38] authors present a synaptic circuit that supports both spike-based learning and global synaptic scaling homeostatic mechanism.

In our implementation, based on[38], the synaptic excitatory current, triggered by L2 neurons and fed into L3 neurons, is compared to a target current. If the value is higher the homeostatic plasticity changes the normalization of the weights. The network with homeostasis has the advantage to adapt when the output pressure of the artificial tactile sensors steadily changes during pressing.

As in the weights normalization network, Figure 5 to the right shows that with homeostasis network, each neuron in L3 is selective to a specific orientation and firing along with respect to the input orientation.

The main advantage of homeostatic network over the weights normalization mechanism is that homeostatic performs better with stimuli with varying pressure. The results highlight that in the constant dataset both the weights normalization and the homeostasis networks act the same both detecting thirty five orientations as shown in figure 6-A. However, in the varying pressure dataset, the homeostasis network outperformed the weights normalization by detecting 35 orientations compared to 31 orientations detected by the weights normalization network. Thanks to homeostasis control, the network was able to discriminate between 35 orientations as shown in figure 6.

Network Latency In order to demonstrate that this architecture can be used in real-time applications, we measured the accuracy of the three networks when stimuli are presented in different interval of times. The duration of stimuli presentation changes from $2ms$ to $20ms$ with a step size of $2ms$. Figure 7 shows that the number of detected orientations increases as a function of the duration of the stimulus presentation, the system then reaches a maximum at $10ms$, recognising all between the 36 orientations both with the weights normalization and homeostasis network.

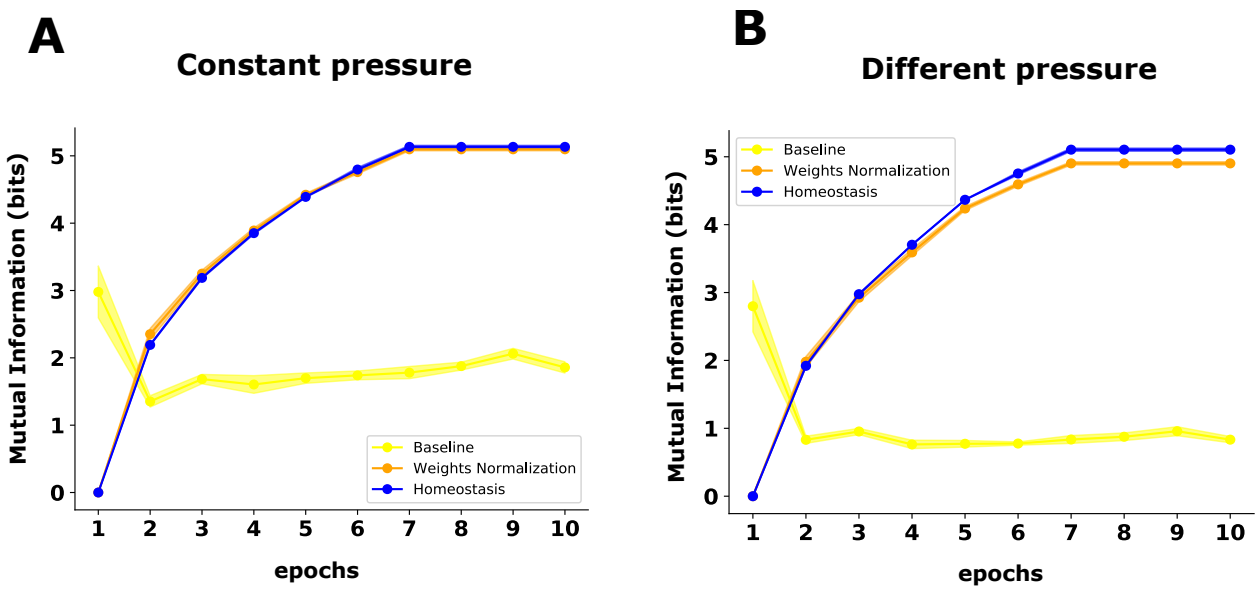

Figure 6. Network performance comparison for the three different networks (Baseline, Weights Normalization, and Homeostasis) based on mutual information. (A) Mutual information as a function of epochs with constant pressure dataset (mean and std shaded); in each epoch 36 orientations ranging from 0^o to 180^o with step 5^o increments, randomly repeated, and all with constant pressure. (B) The benchmark of the three networks when varying pressure stimuli dataset were used; the homeostasis Network outperformed the weights normalization and baseline network by detecting more stimulus orientations.

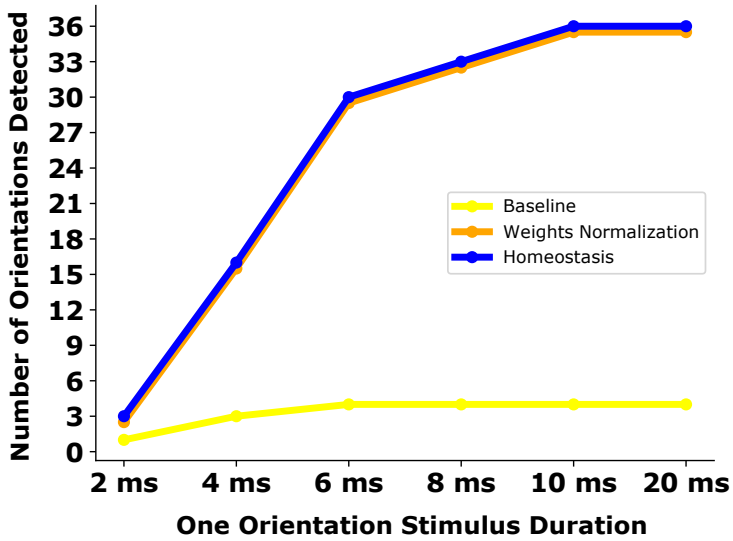

Figure 7. Accuracy of the three networks as a function of stimuli with different intervals of times; the duration of stimuli changes from $2ms$ to $20ms$ with a step increment of $2ms$, both Homeostasis and weights normalization networks outperformed the baseline network in detecting a different number of orientations with different time interval.

Conclusion

In this work a full neuromorphic spiking neural network has been implemented on a neuron simulator to show how, using only neurons, synapses and local learning, complex patterns can be learnt.

In the first part of the work the response of neurons to pressure signals have been studied. It was shown that, by using a spiking model, neurons are able to transmit faithful information about the stimulus activity on the skin. The response from several neurons connected to different sensors have then been collected into RFs to compress the information. The connection

between L1 and L2 (i.e. the neurons connected to sensors and the neurons representing the RFs) has been studied to assess maximum information transmission, ending up in favor of a fully random connectivity between layers.

In the second part L3 has been introduced to decode the coincident activation of RFs. Such coincident activation is in fact the code that neurons use for encoding different orientations. The neuron at the L3 decode which orientation has been applied on the skin. To create this behaviour a popular combination has been initially pursued: unsupervised spike dependant local learning (SDSP) with a WTA, in which only the neuron with higher input (more coincident activation) in the RFs allowed to spike. The spiking neuron was the one more likely to streghten its connection with the previous layer through learning.

This mechanism proved to work and decode orientations, but only when the orientations were activating the same number of RFs. If this condition was not true, the orientation with more active RFs was dominating the ones with less connected neurons, ending up with unbalanced neurons learning several orientations. This problem was solved using synaptic normalization: the number of active synapses connected to each output neuron was defining the degree of normalization each synapse was getting. The higher the number of active synapses connected to the output neuron, the lower the strength of each synapse. This method made possible to reach high precision, despite heterogeneous inputs.

Another problem that was tackled in this work has been the different pressure levels that a pressed bar can exert on the skin. The network ability to infer can be in fact worsened by heterogeneous pressure levels. This issue was solved using homeostasis, where the activity of the neurons trigger a normalization of the synapses. The mechanism can be also used to compensate for drifts in sensor behaviours, given by fatigue or aging.

The network has been designed based on computational primitives that are implementable on low-power sub-threshold neuromorphic hardware, going towards the design of artificial skin with pre-processing embedded capabilities. The future goal is to integrate neuromorphic event-driven readout of the analog values measured by the physical transducers, with the aim of designing compact and efficient sensing devices that can locally preprocess the tactile signal.

Methods

System setup In this work, the system is composed of a skin patch from the iCub robot, a microcontroller, a ZynQ7000 board, and a laptop. The skin patch, with dimensions (11 $\times$ 7.5 cm), comprises 160 capacitive tactile sensing elements (taxels) distributed along 16 triangles. The triangle is the base unit of the tactile sensor. It consists of an off-the-shelf Capacitance to Digital Converter Integrated Circuits (CDC) that provides twelve 16 bits measurements of capacitance. Furthermore the sensor has three communications ports placed along its sides: one for the input from an adjacent triangle, and the others as outputs toward adjacent triangles. The support of the sensor is a two side flex PCB. On one side are etched 12 circular pads, forming the taxels, acting as armatures of capacitors. On the other side of the PCB is mounted the CDC chip and are etched the three communication ports. The triangles are interconnected throw the communication ports on the PCB, without using cables. A single cable goes from the triangles mesh to the microcontroller. The microcontroller unit is used for programming the CDC chips driving the taxels and for reading the pressure measurements[39]. The analog pressure measured by each capacitor sensor (taxel) are then sent to a ZynQ7000 board via I2C bus that uses programmable logic to acquire, deserialise, and associate a timestamp to the samples. The data stream is then written to memory through DMA and made available to any process for computation. In our setup, this is realised by a software grabber module that reads data from DMA and sends to Yet Another Robot Platform (YARP) middleware[40] through Ethernet, where the stream of samples is available for high level software processing. All the experiments performed in this work were implemented and simulated using BRIAN2 simulator[41].

Orientation Detection Dataset The dataset collected for the experiment was obtained by manually pressing the bar (11cm x 0.9cm) on the iCub's skin at 8 different orientations (0^o, $\pm 22.5^o$, $\pm 45^o$, $\pm 67.5^o$, 90^o). We augmented the dataset, increasing the number of possible orientations ranging from (0^o) to (180^o) with (5^o) step increments , by mean of an ad-hoc Python library that computes the superposition of the bar with the taxels for a given orientation and outputs the corresponding measured activation of the taxels. The user can generate all kinds of stimuli, changing bar width, length and orientation. For a simulated press happened, a different approach has been followed: given that the real dataset didn't have any information about the relation between the sensors output and the pressure applied we assumed in the network that the relation between the pressure of the bar and the sensor's response is linear, applying at the latter a multiplicative coefficient.

Neuron Model For L1 neurons presented in this work, LIF neuron model was used with analog data coming from the Bar Pressing Generator algorithm as synaptic current. The variation of membrane potential v_{mem} for L1 neurons in LIF model at time t can be written as:

$$\frac{dv_{mem_i}}{dt} = \frac{v_{rest} - v_{mem_i}}{\tau_{mem_1}} + \frac{I_i(t)}{C_{mem_1}}$$

$$\text{if} \quad v_{mem_i}(t) > v_{th}. \quad \text{then} \quad S_{1_i}(t) \leftarrow 1; \quad v_{mem_i}(t) \leftarrow v_{reset}$$

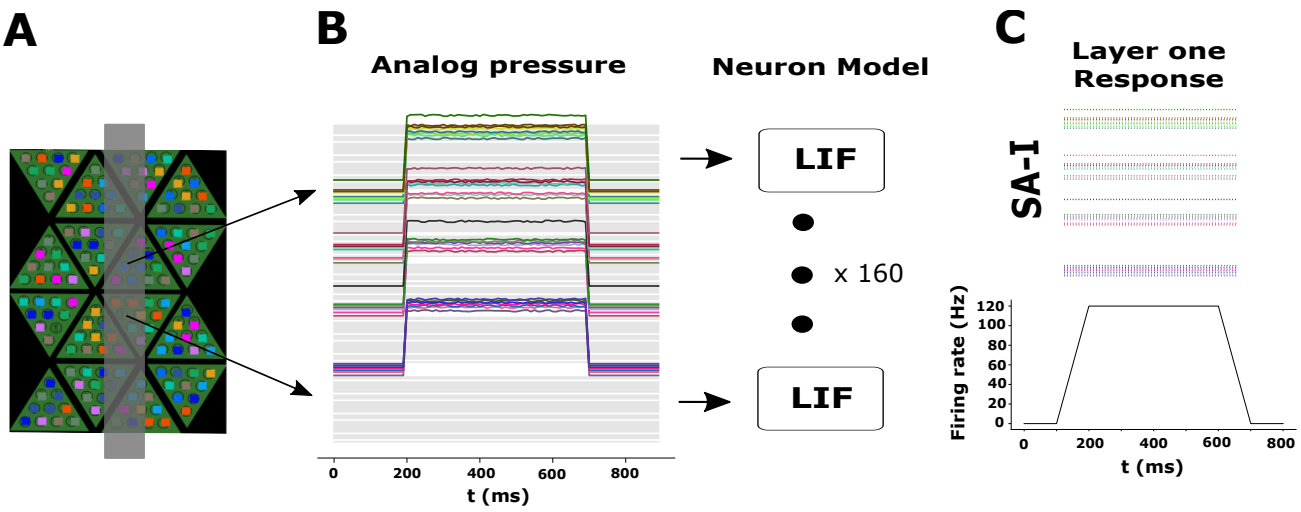

Figure 8. SA-I afferents: Layer-One of the network (L1) encodes analog pressure in spike trains and consists of 160 LIF neurons; (A) An example of a vertical bar pressed on the skin patch that consists of 160 capacitive sensitive elements (Taxels) distributed over 16 triangles. (B) Analog output of the 160 taxels when pressed with the gray vertical (0^o) oriented bar, where the gray shadow represents the non pressed taxels. (C) The output pressure of the taxels is used as input to LIF neurons; (top) Output spikes of neurons after pressing the skin with (0^o) bar; (bottom) firing rate of one neuron.

In this equation, $I_i(t)$ represents the input analog pressure produced by the sensors when pressing the bar on the artificial skin of the iCub robot and C_{m_1} is the membrane capacitance. When applying an input current, the membrane potential v_{mem_i} of the L1 neurons starts to increase until a fixed threshold v_{th}. At this point a spike occurs and the membrane voltage is reset to its resting value v_{reset}. S_{1_i} represents the spike train of L1 neurons. It is a function whose value is 1 when the neuron fires a spike at time t and 0 otherwise. the neuron is leaky since the summed contributions to the membrane potential decays with a characteristic time constant τ_{mem_1}. 8 represents the conversion of analog signals generated by the sensors into neuromorphic spikes when a vertical bar placed on the artificial skin patch. Such as, the firing rate of the output response of layer one is proportion to the output pressure.

Random Receptive fields In this work, based on the two biological constraints we build the matrix of connectivity (W2) that connects L1 with L2 as shown in figure1-B. The first constraint is the non-negative regularization in (W2) that simulates the fact that first-order neurons can only be excited when their transduction sites are stimulated. The second constraint is the convergence from L1 to L2 that simulates the many-to-one convergence from mechanoreceptors in the skin to first-order tactile neurons traveling in the nerve. Additional constraint added to the matrix of connectivity is that the number of mechanoreceptors connected to each RF should be equal. The matrix of connectivity generated randomly, where the rows represent the number of mechanoreceptors and columns represent the RF (figure 1-B), at the end, in each row, there is a single one that defined the mechanoreceptor to which RF is connected. By generating the matrix of connectivity randomly, in this way we have implemented the random structure of RF.

SDSP Learning The network depends on the activity of the presynaptic layer only, therefore the SDSP is used for learning the synaptic weights connecting L2 neurons (presynaptic layer) and L3 neurons (post-synaptic layer). SDSP updates occur every time there is a presynaptic spike. Moreover, the synaptic weights in SDSP are dependent on the variation of the slow post-synaptic calcium variable and an effective neuronal model of the postsynaptic layer. L3 neurons also modeled using LIF model represented in the following equation:

$$\frac{dv_{3_j}}{dt} = \frac{v_{rest} - v_{3_j}}{\tau_3} + \frac{I_{ex}(t) + I_{inh}(t)}{C_{m_3}}$$

In this equation, v_{3_j} represents the membrane potential of L3 neurons (post-synaptic layer), τ_{m_3} is the membrane time constant and C_{m_3} is the membrane capacitance. I_{inh} represents the inhibitory current arriving from the global inhibitory neuron, and I_{ex} is the current supplied from L2 neurons which defined as:

$$I_{ex}(t) = \sum_{i=1}^{n} \left(w_{3_{ji}} \sum_{k} S_{2_i}(t - t_k) \right)$$

S_{2_i} represents the spatio-temporal output spikes of the i_{th} neuron in L2. t_k is the time in which the neurons in L2 fire a spike. The change in SDSP dynamic w_3 between the i_{th} neuron in L2 and the j_{th} neuron in L3 depends on the depolarization of the postsynaptic membrane potential and the slow postsynaptic calcium variable, $C(t)$ defined as:

$$\frac{dC}{dt} = \frac{-C}{\tau_C} + J_C \sum_j \delta(t - t_j)$$

When a post-synaptic neuron fires a spike at time t_j the calcium variable $C(t)$ is incremented by value J_C, and then decays exponentially with a time constant τ_c. When a pre-synaptic spike arrives, the instantaneous values of $v(t_{pre})$ and $C(t_{post})$ indicates the change of the synaptic variable w_3 according to the following conditions:

$$\begin{cases} w_3 \rightarrow w_3 + A_+ & \text{if} \quad v(t_{pre}) > \theta_v \quad \text{and} \quad \theta^l_{up} < C(t_{pre}) < \theta^h_{up} \\ w_3 \rightarrow w_3 + A_- & \text{if} \quad v(t_{pre}) < \theta_v \quad \text{and} \quad \theta^l_{down} < C(t_{pre}) < \theta^h_{down} \end{cases}$$

In this conditions, A_+ and A_- are the learning jumps, θ_v is the voltage threshold and θ^l_{up}, θ^h_{up}, θ^l_{down} and θ^h_{down} indicates the thresholds for the calcium variable. When there is no presynaptic activity, or if the conditions above are not satisfied, then the synaptic variable w drifts toward one of the two stable values α or β depending on the θ_{w_3} as shown in the following conditions:

$$\begin{cases} \frac{dw_3}{dt} = \alpha & \text{if} \quad w_3 > \theta_{w_3} \\ \frac{dw_3}{dt} = -\beta & \text{if} \quad w_3 < \theta_{w_3} \end{cases}$$

all the parameters values of the SDSP learning rule are represented in table 1.

v_{rest}	$-75mV$	θ^l_{up}	4.4
v_{th}	$-50mV$	θ^h_{up}	40
τ_3	$25ms$	θ^l_{down}	4
C_{m_3}	$25pF$	θ^h_{down}	4.8
θ_v	$-65mV$	θ_{w_3}	$1nA$
τ_C	$50ms$	α	$3.5w_{max}/s$
J_C	0.1	β	$3.5w_{max}/s$

Table 1. SDSP learning rule parameters.

Weights Normalization In this work, we endowed weights normalization mechanism in the network architecture between L2 and L3 during learning. During learning, the weights normalization mechanism adapts the synaptic weights connecting L2 neurons to L3 neurons. when the synaptic weights exceed $0.5nA$, then the synaptic weights normalized based on the rules defined as:

$$WN_j = \sum w_{3_{ji}} \quad if \quad w_{3_{ji}} > 0.5nA$$

$$w_{3_{ji}} = \frac{w_{3_{ji}}}{WN_j}$$

In this equation, WN_j represents the weights normalization factor for each neuron in L3 and $w_{3_{ji}}$ represents the synaptic weights connecting L2 neurons to L3 neurons. After a certain time, the weights are all normalized based on the activated RF for each orientation. After Normalization, the network was fed with additional dataset comprises of 36 orientations,repeated many times randomly, until all the synaptic weights saturates at one of the two bi-stability values w_{max} or w_{min}.

Homeostasis In this work to increase the accuracy performance of the network architecture in classifying different bar orientations, even when varying the pressure of pressing, we endowed the homeostasis mechanism between L2 and L3 . The implementation of the homeostasis mechanism was based on the rules provided in the work of Bartolozzi[38]. The homeostatic control monitors the activity of L2 neurons at each orientation and adapt the weights connecting L2 neurons to L3 neurons globally when the pressure of the pressed bar increased or decreased, so that the excitatory current driving from L2 neuron to L3 neurons should be equal to a target current I_{target}, as in the following equation:

$$\tau_{homeo}\frac{dhomeo_j}{dt} = -homeo + \frac{\sum_{i=1}^{n}(w_{3_{ji}}S_{2_i}(t-t_k))}{I_{target}}$$

In this equation, τ_{homeo} is the homeostasis time constant, w_{ji} represents the synaptic weights connecting L2 neurons to L3 neurons and S_2 represents the spatio-temporal output spikes of the i_{th} neuron in L2. T_k is the time in which the neurons in L2 fire a spike. To stabilize the activity of the L2 neurons we divide the weights connecting all neurons in L2 to neuron in L3 by the homeostatic control value as follow:

$$w_{3_{ji}} = \frac{w_{3_{ji}}}{homeo_j}$$

By adapting the weights accordingly based on the homeostatic control, the weights will increased or decreased whenever the pressure change during pressing on the artificial skin.

Information Theory When neurons respond to an input stimulus, they encode its characteristic in their spiking activity. This means that the input stimulus variables (e.g. like the pressure intensity or the angle of the bar) is converted into the output variable in the neuron (e.g. the spike count, the interspike interval or the time of the first spike). The value of the output variable is influenced up to a certain degree by the input variable. Estimating of how much the output variable is expressing the input variable is the main aim of Information Theory[42].

Specifically, mutual information is the measure of the reduction of uncertainty that the output variable causes on the input one. The uncertainty (or entropy) can be expressed as:

$$H(X) = \sum_{x\in X} p(x) log_2\left(\frac{1}{p(x)}\right)$$

where $p(x)$ is the probability density function of the value x in the variable X. When the output variable Y is measured, the outcome can depend on the input variable X. This relation can be expressed as

$$H(X|Y) = \sum_{x\in X; y\in Y} p(x,y) log_2\left(\frac{1}{p(x,y)}\right)$$

where $p(x,y)$ represents the joint probability density function between the value x of X and y of Y. By computing how much uncertainty the output variable reduces with respect the input variable, the mutual information (or MI) can be estimated. The resulting formula is

$$I(X,Y) = H(X) - H(X|Y) = \sum_{x\in X; y\in Y} p(x,y) log_2\left(\frac{p(x,y)}{p(x)p(y)}\right)$$

This concept has been used in this work for estimating the ability of the network to distinguish the different orientations. When a stimulus (i.e., a bar pressed with a specific orientation) is presented to the network the output layer response is used for estimating the MI. The amount of spikes emitted by the different neurons in the layers is calculated, after a binning in time. The neuron with the highest spike count is considered the winner and the positive number '1' is assigned to it in the vector containing all the output neurons. All the other neurons receive instead a '0'. This particular way of coding the output is necessary to overcome the drawbacks present in Information theory (explained in detail in this section). Furthermore, the usage of WTA network at the output layer reduces the error generated by this simplification.

As a result, a matrix composed by N time bins and M neurons is generated. This matrix is then used to create the joint probability where in each column is encoded the probability that a specific output neuron, given different input stimuli, spikes more than the others. This matrix can be defined as the joint probability between the two variables $p(X,Y)$.

The information obtained by the formula has been then converted in "Number of Orientations Detected" using the concept that

$$\text{Recognised Orientation} = 2^{MI}$$

Information theory, despite being a very useful tool, has some drawbacks: mutual information can be overestimated in case of biased joint probability. This situation is present when the possible space of solutions (i.e., the dimension of the output variable values) is high in comparison to trials. This problem has been extensively discussed by neuroscientists[43] and tools are present to reduce bias in information theory.

Nevertheless, the presence of bias is undesirable and should be avoided if possible. In this work a different solution for reducing the bias has been used: by defining a solely winner for each trial (assigning to it the '1') the number of possible values of the output is reduced to the number of output neurons. Following the rule of thumb proposed in Panzeri Treves[43], the number of trials was at least 5 times the dimensionality of the space, making the bias negligible.

References

1. Dahiya, R. S. & Valle, M. *Robotic tactile sensing: technologies and system* (Springer Science & Business Media, 2012).
2. Lepora, N. F. & Ward-Cherrier, B. Superresolution with an optical tactile sensor. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2686–2691 (IEEE, 2015).
3. Cotton, D. P., Chappell, P. H., Cranny, A., White, N. M. & Beeby, S. P. A novel thick-film piezoelectric slip sensor for a prosthetic hand. *IEEE sensors journal* **7**, 752–761 (2007).
4. Tiwana, M. I., Shashank, A., Redmond, S. J. & Lovell, N. H. Characterization of a capacitive tactile shear sensor for application in robotic and upper limb prostheses. *Sensors Actuators A: physical* **165**, 164–172 (2011).
5. Vallbo, Å. & Hagbarth, K.-E. Activity from skin mechanoreceptors recorded percutaneously in awake human subjects. *Exp. neurology* **21**, 270–289 (1968).
6. Caviglia, S., Pinna, L., Valle, M. & Bartolozzi, C. Spike-based readout of posfet tactile sensors. *IEEE Transactions on Circuits Syst. I: Regul. Pap.* **64**, 1421–1431 (2016).
7. Rongala, U. B., Mazzoni, A. & Oddo, C. M. Neuromorphic artificial touch for categorization of naturalistic textures. *IEEE transactions on neural networks learning systems* **28**, 819–829 (2015).
8. Osborn, L. E. *et al.* Prosthesis with neuromorphic multilayered e-dermis perceives touch and pain. *Sci. robotics* **3** (2018).
9. Oddo, C. M. *et al.* Intraneural stimulation elicits discrimination of textural features by artificial fingertip in intact and amputee humans. *elife* **5**, e09148 (2016).
10. Sankar, S. *et al.* Texture discrimination using a flexible tactile sensor array on a soft biomimetic finger. In *2019 IEEE SENSORS*, 1–4 (IEEE, 2019).
11. Rasouli, M., Chen, Y., Basu, A., Kukreja, S. L. & Thakor, N. V. An extreme learning machine-based neuromorphic tactile sensing system for texture recognition. *IEEE transactions on biomedical circuits systems* **12**, 313–325 (2018).
12. Lee, W., Cabibihan, J. & Thakor, N. Bio-mimetic strategies for tactile sensing. In *SENSORS, 2013 IEEE*, 1–4 (IEEE, 2013).
13. Pruszynski, J. A., Flanagan, J. R. & Johansson, R. S. Fast and accurate edge orientation processing during object manipulation. *Elife* **7**, e31200 (2018).
14. Martinez-Hernandez, U. *et al.* Active contour following to explore object shape with robot touch. In *2013 World Haptics Conference (WHC)*, 341–346 (IEEE, 2013).
15. Pruszynski, J. A. & Johansson, R. S. Edge-orientation processing in first-order tactile neurons. *Nat. neuroscience* **17**, 1404–1409 (2014).
16. Johansson, R. S. Tactile sensibility in the human hand: receptive field characteristics of mechanoreceptive units in the glabrous skin area. *The J. physiology* **281**, 101–125 (1978).
17. Parvizi-Fard, A., Amiri, M., Kumar, D., Iskarous, M. M. & Thakor, N. V. A functional spiking neuronal network for tactile sensing pathway to process edge orientation. *Sci. reports* **11**, 1–16 (2021).
18. Kumar, D., Ghosh, R., Nakagawa-Silva, A., Soares, A. B. & Thakor, N. V. Neuromorphic approach to tactile edge orientation estimation using spatiotemporal similarity. *Neurocomputing* **407**, 246–258 (2020).
19. Wong, R. D. P., Hellman, R. B. & Santos, V. J. Spatial asymmetry in tactile sensor skin deformation aids perception of edge orientation during haptic exploration. *IEEE transactions on haptics* **7**, 191–202 (2013).
20. Rongala, U. B. *et al.* Tactile decoding of edge orientation with artificial cuneate neurons in dynamic conditions. *Front. neurorobotics* **13**, 44 (2019).
21. Natale, L., Bartolozzi, C., Nori, F., Sandini, G. & Metta, G. icub. *arXiv preprint arXiv:2105.02313* (2021).
22. Dabbous, A. *et al.* Artificial bio-inspired tactile receptive fields for edge orientation classification. In *2021 IEEE International Symposium on Circuits and Systems (ISCAS)*, 1–5 (IEEE, 2021).
23. Chicca, E., Indiveri, G. & Douglas, R. J. An event-based VLSI network of integrate-and-fire neurons. In *2004 IEEE International Symposium on Circuits and Systems (IEEE Cat. No. 04CH37512)*, vol. 5, V–357 (IEEE, 2004).
24. Johnson, K. O., Yoshioka, T. & Vega-Bermudez, F. Tactile functions of mechanoreceptive afferents innervating the hand. *J. Clin. Neurophysiol.* **17**, 539–558 (2000).
25. Abraira, V. & Ginty, D. The sensory neurons of touch. *Neuron* **79**, 618–639, DOI: 10.1016/j.neuron.2013.07.051 (2013).

26. Gauthier, J. L. *et al.* Receptive fields in primate retina are coordinated to sample visual space more uniformly. *PLoS biology* **7**, e1000063 (2009).

27. Park, C., Bai, Y. H. & Choe, Y. Tactile or visual?: Stimulus characteristics determine receptive field type in a self-organizing map model of cortical development. In *2009 IEEE Symposium on Computational Intelligence for Multimedia Signal and Vision Processing*, 6–13, DOI: 10.1109/CIMSVP.2009.4925641 (2009).

28. Zhao, C. W., Daley, M. J. & Pruszynski, J. A. Neural network models of the tactile system develop first-order units with spatially complex receptive fields. *PloS one* **13**, e0199196 (2018).

29. Brader, J. M., Senn, W. & Fusi, S. Learning real-world stimuli in a neural network with spike-driven synaptic dynamics. *Neural computation* **19**, 2881–2912 (2007).

30. Frenkel, C., Indiveri, G., Legat, J.-D. & Bol, D. A fully-synthesized 20-gate digital spike-based synapse with embedded online learning. In *2017 IEEE International Symposium on Circuits and Systems (ISCAS)*, 1–4 (IEEE, 2017).

31. Mostafa, H. Supervised learning based on temporal coding in spiking neural networks. *IEEE transactions on neural networks learning systems* **29**, 3227–3235 (2017).

32. Soures, N., Hays, L., Bohannon, E., Zyarah, A. M. & Kudithipudi, D. On-device stdp and synaptic normalization for neuromemristive spiking neural network. In *2017 IEEE 60th International Midwest Symposium on Circuits and Systems (MWSCAS)*, 1081–1084 (IEEE, 2017).

33. Turrigiano, G. Homeostatic signaling: the positive side of negative feedback. *Curr. opinion neurobiology* **17**, 318–324 (2007).

34. Turrigiano, G. G. Homeostatic plasticity in neuronal networks: the more things change, the more they stay the same. *Trends neurosciences* **22**, 221–227 (1999).

35. Marder, E. & Prinz, A. A. Modeling stability in neuron and network function: the role of activity in homeostasis. *Bioessays* **24**, 1145–1154 (2002).

36. Echegoyen, J., Neu, A., Graber, K. D. & Soltesz, I. Homeostatic plasticity studied using in vivo hippocampal activity-blockade: synaptic scaling, intrinsic plasticity and age-dependence. *PloS one* **2**, e700 (2007).

37. Abbott, L. F. & Nelson, S. B. Synaptic plasticity: taming the beast. *Nat. neuroscience* **3**, 1178–1183 (2000).

38. Qiao, N., Indiveri, G. & Bartolozzi, C. Automatic gain control of ultra-low leakage synaptic scaling homeostatic plasticity circuits. In *2016 IEEE Biomedical Circuits and Systems Conference (BioCAS)*, 156–159 (IEEE, 2016).

39. Cannata, G., Maggiali, M., Metta, G. & Sandini, G. An embedded artificial skin for humanoid robots. In *2008 IEEE International conference on multisensor fusion and integration for intelligent systems*, 434–438 (IEEE, 2008).

40. Natale, L., Paikan, A., Randazzo, M. & Domenichelli, D. E. The icub software architecture: evolution and lessons learned. *Front. Robotics AI* **3**, 24 (2016).

41. Stimberg, M., Brette, R. & Goodman, D. F. Brian 2, an intuitive and efficient neural simulator. *eLife* **8**, e47314, DOI: 10.7554/eLife.47314 (2019).

42. Timme, N. M. & Lapish, C. A tutorial for information theory in neuroscience. *eneuro* **5** (2018).

43. Panzeri, S. & Treves, A. Analytical estimates of limited sampling biases in different information measures. *Network: Comput. neural systems* **7**, 87 (1996).