TU Delft

Delft University of Technology

Unsupervised Learning of a Hierarchical Spiking Neural Network for Optical Flow Estimation

From Events to Global Motion Perception

Paredes Valles, Fede; Scheper, Kirk; de Croon, Guido

DOI
10.1109/TPAMI.2019.2903179

Publication date
2019

Document Version
Accepted author manuscript

Published in
IEEE Transactions on Pattern Analysis and Machine Intelligence

Citation (APA)
Paredes Valles, F., Scheper, K., & de Croon, G. (2019). Unsupervised Learning of a Hierarchical Spiking Neural Network for Optical Flow Estimation: From Events to Global Motion Perception. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *42 (2020)*(8), 2051-2064. Article 8660483. https://doi.org/10.1109/TPAMI.2019.2903179

Important note
To cite this publication, please use the final published version (if applicable).
Please check the document version above.

Copyright
Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

Takedown policy
Please contact us and provide details if you believe this document breaches copyrights.
We will remove access to the work immediately and investigate your claim.

This work is downloaded from Delft University of Technology.
For technical reasons the number of authors shown on this cover page is limited to a maximum of 10.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2019.2903179, IEEE Transactions on Pattern Analysis and Machine Intelligence

Unsupervised Learning of a Hierarchical Spiking Neural Network for Optical Flow Estimation: From Events to Global Motion Perception

Federico Paredes-Vallés, Kirk Y. W. Scheper, and Guido C. H. E. de Croon, *Member, IEEE*

Abstract—The combination of spiking neural networks and event-based vision sensors holds the potential of highly efficient and high-bandwidth optical flow estimation. This paper presents the first hierarchical spiking architecture in which motion (direction and speed) selectivity emerges in an unsupervised fashion from the raw stimuli generated with an event-based camera. A novel adaptive neuron model and stable spike-timing-dependent plasticity formulation are at the core of this neural network governing its spike-based processing and learning, respectively. After convergence, the neural architecture exhibits the main properties of biological visual motion systems, namely feature extraction and local and global motion perception. Convolutional layers with input synapses characterized by single and multiple transmission delays are employed for feature and local motion perception, respectively; while global motion selectivity emerges in a final fully-connected layer. The proposed solution is validated using synthetic and real event sequences. Along with this paper, we provide the *cuSNN* library, a framework that enables GPU-accelerated simulations of large-scale spiking neural networks. Source code and samples are available at https://github.com/tudelft/cuSNN.

Index Terms—Event-based vision, feature extraction, motion detection, neural nets, neuromorphic computing, unsupervised learning

1 Introduction

Whenever an animal endowed with a visual system navigates through an environment, turns its gaze, or simply observes a moving object from a resting state, motion patterns are perceivable at the retina level as spatiotemporal variations of brightness [1]. These patterns of apparent motion, formally referred to as *optical flow* [2], are a crucial source of information for these animals to estimate their ego-motion and to have a better understanding of the visual scene. A great example of the efficacy of these cues in nature is in flying insects [1], [3], which are believed to heavily rely on these visual cues to perform high-speed maneuvers such as horizontal translation or landing [4].

Considering their size and weight limitations, insects are a clear indicator of the efficiency, robustness, and low latency of the optical flow estimation conducted by biological systems. The ability to reliably mimic this procedure would have a significant impact on the field of micro-robotics due to the limited computational capacity of their onboard processors. As an example, Micro Air Vehicles (MAVs), such as the DelFly Explorer [5] or the DelFly Nimble [6], could benefit from a bio-inspired visual motion estimation for high-speed autonomous navigation in cluttered environments.

Biological visual systems receive their input from photoreceptors in the retina. These light-sensitive neurons absorb and convert incoming light into electrical signals which serve as input to the so-called ganglion cells. The activity of these neurons consists of temporal sequences of discrete *spikes* (voltage pulses) that are sent to large networks of interconnected cells for motion estimation, among other tasks. Since it is spike-driven, these biological architectures are characterized by a sparse, asynchronous, and massively parallelized computation. Further, they are seen to adapt their topology, i.e. connectivity pattern, in response to visual experience [7], [8]. This adaptation, or *learning* mechanism, allows them to operate robustly in different environments under a wide range of lighting conditions.

In contrast, the working principle of the majority of cameras used for artificial visual perception is categorized as *frame-based*: data is obtained by measuring the brightness levels of a pixel array at fixed time intervals. Although convenient for some computer vision applications, these sensors are inefficient for the task of motion estimation as the frame rate is independent of the dynamics of the visual scene. Additionally, due to the limited temporal resolution of these sensors, rapidly moving objects may introduce motion blur, limiting the accuracy of optical flow estimation.

However, not all artificial systems rely on conventional frame-based cameras for visual motion estimation. Inspired by biological retinas, several *event-based* vision sensors have recently been presented [9], [10], [11], [12]. Similar to ganglion cells, each of the elements of the pixel array reacts asynchronously to brightness changes in its corresponding receptive field by generating *events*. A microsecond temporal resolution, latencies in this order of magnitude, a wide dynamic range, and a low power consumption make these sensors an ideal choice for visual perception.

Regardless of the vision sensor, the estimation of optical flow by artificial systems is normally performed algorithmically, with solutions that are built on simplifying assumptions that make this problem tractable [13], [14]. In spite of this, the recent progress in parallel computing hardware has

- *The authors are with the Department of Control and Simulation (Micro Air Vehicle Laboratory), Faculty of Aerospace Engineering, Delft University of Technology, Kluyverweg 1, 2629 HS Delft, The Netherlands. E-mail: {f.paredesvalles, k.y.w.scheper, g.c.h.e.decroon}@tudelft.nl*

0162-8828 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2019.2903179, IEEE Transactions on Pattern Analysis and Machine Intelligence

enabled artificial motion perception to be addressed from a more bio-inspired perspective: Artificial Neural Networks (ANNs). Similar to biological architectures, ANNs consist of large sets of artificial neurons whose interconnections can be optimized for the task at hand. However, despite the high accuracy reported with both frame- [15] and event-based sensors [16], [17], there is still a fundamental difference: the underlying communication protocol in ANNs relies on synchronous packages of floating-point numbers, rather than on trains of asynchronous discrete spikes. As a consequence, these architectures are often computationally expensive.

Taking further inspiration from nature, Spiking Neural Networks (SNNs) have been proposed as a new generation of ANNs [18]. As the name suggests, the computation carried out by these bio-realistic neural models is asynchronous and spike-based, which makes them a suitable processing framework for the sparse data generated by event-based sensors [19]. Moreover, SNNs can benefit from an efficient real-time implementation in *neuromorphic hardware*, such as IBM's TrueNorth chip [20] or Intel's Loihi processor [21]. Despite these promising characteristics, the spiking nature of these networks limits the application of the successful gradient-based optimization algorithms normally employed in ANNs. Instead, learning in SNNs is dominated by Spike-Timing-Dependent Plasticity (STDP) [22], a biologically plausible protocol that adapts the strength of a connection between two neurons based on their correlated activity. STDP has been successfully applied to relatively simple image classification tasks [23], [24], [25], [26], [27]. However, until now, no study has discussed the use of this learning rule for the estimation of event-based optical flow.

This paper contains *three main contributions*. First, a novel adaptive mechanism for the Leaky Integrate-and-Fire (LIF) spiking neuron model [28] is introduced. This adaptation extends the applicability of this model to the rapidly varying input statistics of a moving event-based vision sensor. Second, a novel, inherently-stable STDP implementation is proposed. With this learning rule, the strength of neural connections naturally converges to an equilibrium distribution without the need for the ad-hoc mechanisms used by most of the existing formulations. Third, the proposed neuron model and STDP rule are combined in a hierarchical SNN architecture that, after learning, resembles the main functionalities of biological visual systems: feature extraction and local and global motion perception. To the best of the authors' knowledge, this paper shows, for the first time, that neural selectivity to the local and global motion of input stimuli can emerge from visual experience in a biologically plausible unsupervised fashion.

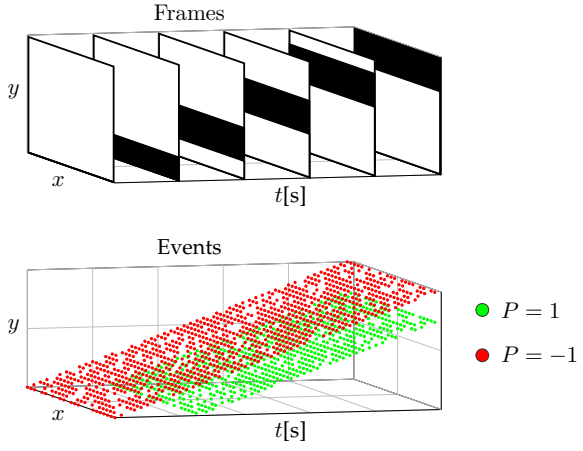


Fig. 1: Comparison of the output of frame- and event-based vision sensors under the stimulus of a black horizontal bar moving upward over a homogeneous white background. While frames are basically two-dimensional snapshots of the visual scene, events are spatiotemporal sparse points tracking the leading and trailing edges of the bar.

The rest of the paper is structured as follows. Section 2 provides background information concerning event-based vision, SNNs, and optical flow estimation. The foundations of the spike-based processing and learning of the proposed SNN are detailed in Sections 3 and 4, respectively. Thereafter, the network architecture is described in Section 5, and empirically evaluated in Section 6.

2 Background Information

2.1 Event-Based Vision Sensors

Inspired by biological retinas, each of the pixels of an event-based vision sensor reacts asynchronously to local changes in brightness by generating discrete temporal events. Specifically, the generation of an event is triggered whenever the logarithmic change of the image intensity $I(x, y, t)$ exceeds a predefined threshold C such that $\left|\Delta \log\big(I(x, y, t)\big)\right| > C$ [9]. This variation is computed with respect to a reference brightness level set by the last occurring event at that pixel.

Each event encodes information about its timestamp t, its corresponding (x, y) location in the pixel array, and the polarity $P \in \{-1, 1\}$ of the intensity change. This communication protocol is referred to as Address-Event Representation (AER), and any camera that makes use of it is categorized as Dynamic Vision Sensor (DVS). A visual comparison of the output of frame- and event-based sensors under the same stimulus is illustrated in Fig. 1.

2.2 Spiking Neural Networks

Models of spiking neurons: In biological networks, neural communication consists in the exchange of voltage pulses [18]. For the reproduction of this asynchronous and spike-based mechanism in SNNs, multiple models of spiking neurons have been presented at various levels of abstraction. Biophysical formulations lead to accurate representations of neural dynamics [29], however, their complexity limits their use in large-scale networks. Alternatively, phenomenological formulations offer a compromise between computational load and biological realism. The most used models are the aforementioned LIF [28], the Izhikevich [30], and the Spike Response Model [31].

From a conceptual perspective, the majority of these models share some fundamental principles and definitions. The junction of two neurons is called *synapse*; and relative to these cells, the transmitting neuron is labeled as *presynaptic*, while the receiving as *postsynaptic*. Each spiking neuron, as processing unit, is characterized by an internal state variable, known as *membrane potential* $v_i(t)$, which temporally integrates presynaptic spikes over time. If the arrival of a spike leads to an increase (decrease) in $v_i(t)$, then the

0162-8828 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2019.2903179, IEEE Transactions on Pattern Analysis and Machine Intelligence

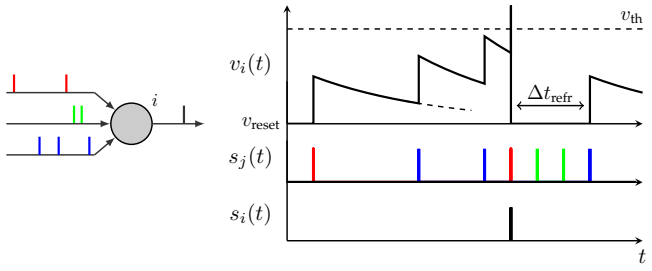


Fig. 2: A model of a LIF neuron. The graphic (right) shows the temporal course of the membrane potential $v_i(t)$ of the ith neuron (left), driven by a sample presynaptic spike train $s_j(t)$ from three input neurons $j = \{1, 2, 3\}$. Spikes are depicted as vertical bars at the time at which they are received (if presynaptic) or emitted (if postsynaptic). Here, the reset v_{reset} and resting v_{rest} potentials are equal in magnitude.

spike is said to have an *excitatory* (*inhibitory*) effect on the neuron. $v_i(t)$ decays to a resting potential v_{rest} in case no input is received. Lastly, a postsynaptic spike is triggered whenever $v_i(t)$ crosses the *firing threshold* v_{th}. Afterwards, the neuron resets its membrane potential to v_{reset}, and enters in a *refractory period* Δt_{refr} during which new incoming spikes have negligible effect on $v_i(t)$. Fig. 2 illustrates these concepts for the case of a LIF neuron [28].

Synaptic plasticity: Defined as the ability to modify the *efficacy* (weight) of neural connections, *synaptic plasticity* is the basic mechanism underlying learning in biological networks [32]. These architectures are seen to rely on different learning paradigms depending on their duty [33]. For instance, information encoding in biological visual systems is established in an unsupervised fashion, while reinforcement and supervised learning are employed for tasks such as decision making and motor control. Accordingly, various forms of synaptic plasticity have been proposed for SNNs.

In the context of SNNs, unsupervised learning is generally referred to as Hebbian learning, since plasticity rules from this paradigm are based on Hebb's postulate: "*cells that fire together, wire together*" [34]. In essence, these methods adapt the efficacy of a connection based on the correlated activity of pre- and postsynaptic cells. Among others, the biologically plausible STDP protocol is, by far, the most popular Hebbian rule for SNNs [22]. With STDP, the repeated arrival of presynaptic spikes to a neuron shortly before it fires leads to synaptic strengthening, also known as Long-Term Potentiation (LTP); whereas if the arrival occurs shortly after the postsynaptic spike, synapses are weakened through Long-Term Depression (LTD). Therefore, the change of efficacy ΔW is normally expressed as a function of the relative timing between these two events. STDP formulations exclusively dependent on this parameter are referred to as *additive* rules [35], are inherently unstable, and require the use of constraints for the weights, thus resulting in bimodal distributions [22]. On the other hand, *multiplicative* STDP rules incorporate the current efficacy value in the computation of ΔW. The formulations proposed in [23], [24], [25] incorporate the weights in a proportional fashion, and represent the current sate-of-the-art in pattern recognition with SNNs. However, they still lead to bimodal distributions. Contrarily, [26], [27] claim that, by incorporating the weight dependency in an inversely proportional manner, stable unimodal distributions are obtained. Nevertheless, their stability results from a complex temporal LTP-LTD balance, and it is not theoretically guaranteed.

Several lines of research can be distinguished regarding the use of supervised learning in SNNs, with the most promising based on the well-known error backpropagation algorithm [36]. Firstly, numerous adaptations to the discontinuous dynamics of SNNs have recently been proposed for learning temporally precise spike patterns [37], [38], [39], [40]. Alternatively, due to the popularity of this method in ANNs, SNNs commonly rely on transferring optimization results from their non-spiking counterparts [41], [42], [43]. In both cases, high accuracy levels are reported in image classification tasks, but still far below those obtained with conventional ANNs.

With respect to reinforcement learning in SNNs, various models have been presented, the majority of which consist in the modulation of STDP with a reward function [44], [45]. However, applications of this paradigm are mainly focused on neuroscience research [46], [47], besides several goal-directed navigation problems [48], [49] and the digit-recognition application from [50].

2.3 Event-based Optical Flow Estimation

The recent introduction of the DVS and other retinomorphic vision sensors has lead to the development of several novel approaches to event-based optical flow estimation. Depending on their working principle, these solutions are divided into algorithmic and neural methods.

Gradient-, plane-fitting-, frequency-, and correlation-based approaches set the basis of the algorithmic state-of-the-art. These techniques compute sparse optical flow estimates for each newly detected event (or group of events) based on its spatiotemporal polarity-specific neighborhood. Firstly, adaptations of the gradient-based Lucas-Kanade algorithm [51] were presented in [52], [53]. Secondly, the methods proposed in [13], [54], [55] extract optical flow by computing the gradients of a local plane fitted to a spatiotemporal surface of events. Thirdly, multiple adaptations of the bio-inspired frequency-based methods have been introduced [53], [56], [57], which allow the implementation in neuromorphic hardware [58]. Lastly, the recent correlation-based approaches presented in [59], [60], [61], [62] employ convex optimization algorithms to associate groups of events over time, and report the highest algorithmic accuracy to date. Part of this category is also the block-matching method recently proposed in [63], which employs conventional search techniques to find the best matching group of events in previous temporal slices of the input.

The estimation of event-based optical flow with neural models is dominated by SNNs. However, there are a couple of ANN-based approaches worth remarking. In [16], a self-supervised learning scheme was employed to train a convolutional ANN to estimate dense image flow. The input to the network consists of the per-pixel last timestamp and count of events over a specific time window. Using the average timestamp instead, in [17], the authors presented the first neural model to approach the full structure-from-motion problem using event-based input. In [17], two ANNs are employed for depth and dense optical flow estimation. Regarding the latter task, accuracy levels considerably higher than those from [16] are reported.

0162-8828 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2019.2903179, IEEE Transactions on Pattern Analysis and Machine Intelligence

Though the main goal of [64] is for predicting future input activations, this work presented the first neural architecture capable of learning spatiotemporal features from raw event data. For this purpose, multiple recurrent ANNs were employed in combination with a single layer of spiking neurons. A self-supervised learning scheme, based on a recursive least-squares algorithm, was proposed for training the ANNs to capture the spatiotemporal information. Note that, for compatibility reasons, this approach requires the conversion of the event data into analog signals.

With respect to pure SNN-based approaches, in [65], [66], the authors propose an architecture in which motion selectivity results from the synaptic connections of a bursting neuron to two neighboring photoreceptors, one excitatory and the other inhibitory. If the edge is detected first by the excitatory cell, spikes are emitted at a fixed rate until the inhibitory pulse is received. Otherwise, the neuron remains inactive. Optical flow is consequently encoded in the burst length and in the relative orientation of the photoreceptors.

In contrast, the SNNs presented in [67], [68] extract motion information through synaptic delays and spiking neurons acting as coincidence detectors. A simple spike-based adaptation of the Reichardt model [69] is introduced in [67] to show the potential of this approach. This idea is explored in more detail in [68], in which the authors propose the convolution of event sequences with a bank of spatiotemporally-oriented filters, each of which is comprised of non-plastic synapses with equal efficacies, but with delays tuned to capture a particular direction and speed. Similarly to frequency-based methods [70], these filters compute a confidence measure, encoded in the neural activity, rather than the optical flow components. Additionally, this solution employs a second spike-based pooling layer for mitigating the effect of the aperture problem [71].

Whether, and how, direction and speed selectivity emerge in biological networks from visual experience still remains an open question. Some initial work by [72], [73], [74] shows that robust local direction selectivity arises in neural maps through STDP if, apart from presynaptic feedforward connections, neurons receive spikes from cells in their spatial neighborhood through plastic synapses with distance-dependent transmission delays. However, no study has assessed the speed selectivity of these cells, which is crucial for optical flow estimation.

3 Adaptive Spiking Neuron Model

Let $j = 1, 2, \ldots, n^{l-1}$ denote a group of presynaptic neurons, from layer $l-1$, fully connected in a feedforward fashion to a set of postsynaptic cells $i = 1, 2, \ldots, n^{l}$, from layer l. As depicted in Fig. 3, these neural connections can be considered as *multisynaptic*, i.e. the link between two cells is not restricted to a single synapse, but several can coexist. In this work, the number of multisynaptic connections m is layer-specific, and each synapse has its own transmission delay as given by $\boldsymbol{\tau} \in \mathbb{R}^m$. In addition to this delay vector, layer connectivity is also characterized by a weight matrix $\boldsymbol{W} \in \mathbb{R}^{n^l \times n^{l-1} \times m}$, which determines the synaptic efficacy of the connections.

Apart from $\boldsymbol{W}$ and $\boldsymbol{\tau}$, each synapse keeps track of an additional parameter that captures the recent history of spikes transmitted. Referred to as the *presynaptic trace* [75], and defined as $\boldsymbol{X} \in \mathbb{R}^{n^l \times n^{l-1} \times m}$, its dynamics is given by:

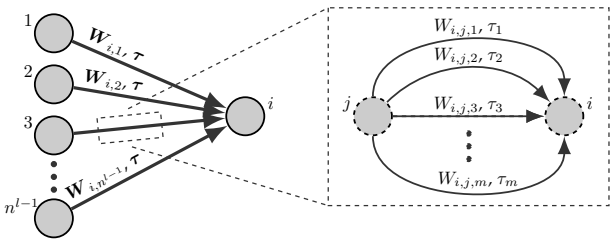


Fig. 3: Schematic of the feedforward connectivity between neurons from two adjacent layers (left). These connections can be considered as being multisynaptic (right), each one having its own efficacy, transmission delay, and trace.

$$\lambda_X \frac{dX_{i,j,d}(t)}{dt} = -X_{i,j,d}(t) + \alpha s_j^{l-1}(t - \tau_d) \tag{1}$$

where λ_X is the time constant of the system, α is a scaling factor, and $\boldsymbol{s}^l(t) \in \mathbb{R}^{n^l}$ denotes the (binary) record of neural activity, or *spike train*, of cells from layer l. Note that $d = 1, 2, \ldots, m$ serves to refer both to connections within a multisynaptic group and their corresponding delays.

From Eq. (1), whenever a spike arrives at a postsynaptic neuron i via a synapse with transmission delay τ_d, the corresponding presynaptic trace $X_{i,j,d}(t)$ increases by a factor of α. In case no spike is received, the trace decays exponentially towards zero according to λ_X.

The LIF model [28] is the most widely used spiking neuron model in literature. This is due to its main assumption that in SNNs, information is not encoded in the spike amplitude, but rather in the firing time. Consequently, neural activity is reduced to discrete and binary temporal events, thus ensuring computational tractability. The neuron model used in this paper is a modified LIF formulation, defined as:

$$\lambda_v \frac{dv_i(t)}{dt} = -\big(v_i(t) - v_{\text{rest}}\big) + i_i(t) \tag{2}$$

$$i_i(t) = \sum_{j=1}^{n^{l-1}} \sum_{d=1}^{m} \big(W_{i,j,d} s_j^{l-1}(t - \tau_d) - X_{i,j,d}(t)\big) \tag{3}$$

where λ_v denotes the time constant of the membrane potential, and $\boldsymbol{i}(t)$ is the so-called *forcing function* of the system.

From Eqs. (2) and (3), the membrane potential $v_i(t)$ of a neuron evolves over time by integrating scaled presynaptic spikes from its input synapses, similarly to the conventional LIF model [28]. Whenever $v_i(t)$ reaches (or surpasses) the firing threshold v_{th}, a postsynaptic spike is generated, i.e. $s_i^l(t) = 1$, and $v_i(t)$ is reset to v_{reset}. In addition, the neuron enters in a refractory period Δt_{refr} during which presynaptic spikes have no effect on $v_i(t)$ to ensure the temporal separation of postsynaptic pulses. In case no spike is fired at time t, this is reflected in the neuron's spike train as $s_i^l(t) = 0$.

Unlike traditional LIF [28], the forcing function $\boldsymbol{i}(t)$ of our neuron model includes an additional term, further referred to as the *homeostasis* parameter, which is inspired by the internal regulatory mechanisms of biological organisms [76]. This is used to adapt the neural response to the varying input statistics—in particular, to the per-pixel firing rate—using the presynaptic trace $\boldsymbol{X}$ as an *excitability* indicator.

0162-8828 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2019.2903179, IEEE Transactions on Pattern Analysis and Machine Intelligence

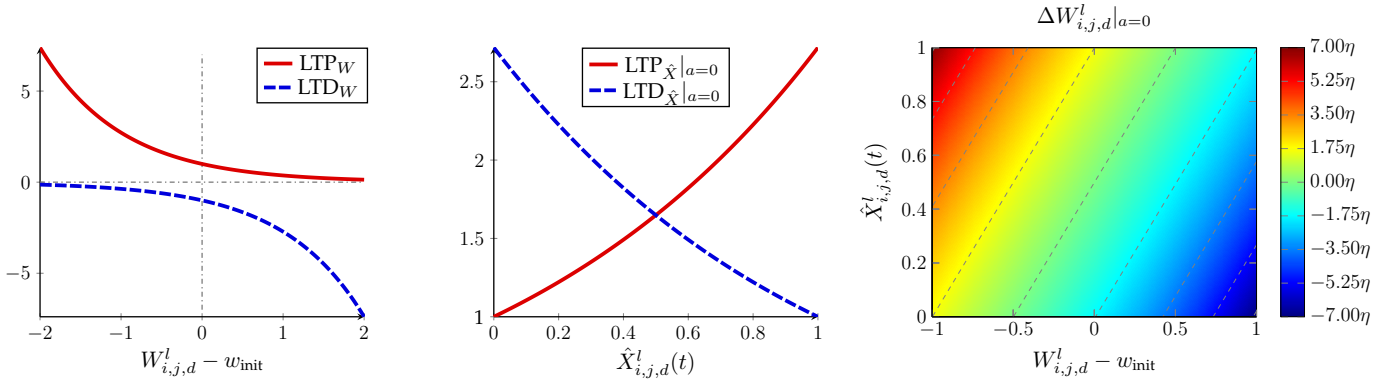


Fig. 4: Illustration of the novel multiplicative STDP rule proposed in this work. The weight update (right) results from the linear combination of the non-exclusive LTP and LTD processes. These, in turn, are characterized by symmetrical dependencies on the synaptic weights (left) and normalized presynaptic traces (center). Note that, in the schematic of the weight update (right), the weight axis is limited to the $[-1, 1]$ range only for the purpose of a better visualization of the equilibrium weights (dashed thick line) for $a = 0$.

Inferring from Eq. (3), this parameter acts, in essence, as an inhibitory penalty in the update rule of $\boldsymbol{v}(t)$. A postsynaptic neuron connected to a group of highly-active presynaptic cells is said to have low excitability due to its relatively high $\boldsymbol{X}$. For this neuron to fire, it needs to receive a large number of presynaptic spikes shortly separated in time. Conversely, the same cell connected to poorly-active neurons is highly excitable; and thus, the firing threshold v_{th} can still be reached despite the considerably larger time difference between input spikes. Note that, to get the desired neural adaptation, the scaling factor α, from Eq. (1), needs to be selected in accordance with the neural parameters, mainly v_{th} and the range of possible $\boldsymbol{W}$ values. The effect of this parameter on the neural response is explored in [77]. When dealing with an event-based camera as source of input spikes, the firing rate of the sensor is not only correlated to the appearance of features from the visual scene, but also to their optical flow and the sensitivity settings of the camera. Slow apparent motion leads to successive events being more distant in time than those captured from fast motion. Consequently, if these events are to be processed with a network of spiking neurons, a homeostasis mechanism is required to ensure that similar features are detected regardless of the encoding spike rate.

Other approaches to homeostasis have been presented, such as threshold balancing [78] or weight scaling [79]. However, these methods use postsynaptic spikes to adjust the homeostatic inhibition through an adaptive mechanism. With such neural feedback, there is a delay in adjusting the excitability of the neurons. These approaches are therefore less suitable for the rapidly varying statistics of the data generated by a moving event-based vision sensor.

4 Stable STDP Learning Rule

In this work, we propose a novel multiplicative STDP implementation that, contrary to the state-of-the-art of this learning protocol, is inherently stable by combining the weight-dependent exponential rule from [27] with presynaptic trace information. Hereafter, we will simply refer to it as STDP.

Whenever a neuron i fires a spike, the efficacy of its presynaptic connections is updated as follows:

$$\Delta W_{i,j,d} = \eta(\text{LTP} + \text{LTD}) \tag{4}$$

$$\begin{aligned} \text{LTP} &= \text{LTP}_W \cdot \text{LTP}_{\hat{X}}, & \text{LTD} &= \text{LTD}_W \cdot \text{LTD}_{\hat{X}} \\ \text{LTP}_W &= e^{-(W_{i,j,d} - w_{\text{init}})}, & \text{LTD}_W &= -e^{(W_{i,j,d} - w_{\text{init}})} \\ \text{LTP}_{\hat{X}} &= e^{\hat{X}_{i,j,d}(t)} - a, & \text{LTD}_{\hat{X}} &= e^{(1-\hat{X}_{i,j,d}(t))} - a \end{aligned} \tag{5}$$

where η is the learning rate of the rule, w_{init} refers to the initialization weight of all synapses at the beginning of the learning process, and $\hat{\boldsymbol{X}}_i \in [0, 1]$ denotes the presynaptic traces of neuron i normalized to the current maximum at the moment of firing. Further, for stability, $\eta > 0$ and $a < 1$ regardless of the value of w_{init} (see Appendix A).

From Eqs. (4) and (5), the weight update $\Delta \boldsymbol{W}_i$ results from the linear combination of the output of two non-mutually exclusive processes: LTP, for strengthening, and LTD, for weakening synaptic connections. Both of these processes are dependent on the weights (LTP$_W$, LTD$_W$) and normalized traces (LTP$_{\hat{X}}$, LTD$_{\hat{X}}$) of the synapses under analysis. On the one hand, the weight dependency of our learning rule takes inspiration from the STDP formulation presented in [27]. LTP$_W$ and LTD$_W$ are inversely proportional to $\boldsymbol{W}_i$ in an exponential fashion, and are centered around w_{init} (see Fig. 4, left). Consequently, the effect of LTP$_W$ decreases (increases) the larger (smaller) a synaptic weight is in comparison to w_{init}. The opposite relation holds true for LTD$_W$. On the other hand, rather than relying on the precise spike timing [27], our rule employs normalized presynaptic trace information as a measure of the relevance of a particular connection to the postsynaptic spike triggering the update. The higher (lower) the value of $\hat{X}_{i,j,d}(t)$, the larger (smaller) the effect of LTP$_{\hat{X}}$, and vice versa for LTD$_{\hat{X}}$ (see Fig. 4, center).

With this formulation, a weight is established for each value of $\hat{X}_{i,j,d}(t)$ through a stable equilibrium of LTP-LTD contributions on $\Delta \boldsymbol{W}_i$ (see Fig. 4, right). The parameter a has control over this non-linear mapping through the steepness of LTP$_{\hat{X}}$ and LTD$_{\hat{X}}$ in $\hat{\boldsymbol{X}}_i \in [0, 1]$. The higher (lower) the value of a—below the stability limit—, the wider (narrower) the distribution of synaptic weights after convergence. As such, no additional mechanism is required for preventing weights from vanishing or exploding. Synapses characterized by weights that are higher (lower) than their corresponding equilibrium state are consistently depressed (potentiated) until synapse-specific stability is achieved.

0162-8828 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2019.2903179, IEEE Transactions on Pattern Analysis and Machine Intelligence

To track the convergence of the learning process, we propose the use of the following mean square error criterion, where $\hat{\boldsymbol{W}}_i \in [0,1]$ denotes the presynaptic weights of neuron i after an update, normalized to the current maximum:

$$\mathcal{L}_i = \frac{1}{n^{l-1}m} \sum_{j=1}^{n^{l-1}} \sum_{d=1}^{m} \left(\hat{X}_{i,j,d}(t) - \hat{W}_{i,j,d}\right)^2 \tag{6}$$

As the learning progresses, the moving average of $\mathcal{L}_i$ converges to a (close-to-zero) equilibrium state. In this work, we stop synaptic plasticity using a fixed threshold on this parameter, denoted by $\mathcal{L}_{\text{th}}$.

4.1 Local inter-lateral competition

For neurons to learn distinct features from the input data through STDP, this learning rule needs to be combined with what is known as a Winner-Take-All (WTA) mechanism [80]. This form of competition implies that, when a neuron fires a spike and updates its presynaptic weights according to Eqs. (4) and (5), the rest of postsynaptic cells (from the same layer) locally connected to the same input neurons get inhibited. As a result, these cells are prevented from triggering STDP while the neuron that fired first, i.e. the *winner*, remains in the refractory period.

Instead of relying on non-plastic synapses transmitting inhibitory spikes with a certain delay, our implementation assumes that the internal dynamics of these neurons are intercorrelated. Whenever the winner resets its membrane potential and enters in the refractory period, neurons affected by the WTA mechanism do the same immediately afterwards. In case multiple neurons fire simultaneously, the cell with the highest membrane potential has preference for triggering the weight update. Further, the postsynaptic spikes from the other firing neurons are not considered. To ensure coherence between the training and inference phases of our proposed SNN, layers trained with STDP maintain the WTA mechanism after the learning process.

5 Spiking Neural Network Architecture for Motion Perception

To extract a robust measure of motion from the raw camera input, we propose the multi-layer SNN illustrated in Fig. 5. This section highlights the unique goal of each of the layers comprising this architecture, together with the variations of the proposed neuron model and learning rule that are required depending on their connectivity scheme.

5.1 Input Layer

Being the first stage of the network, the Input layer encodes the event-based sensor data in a compatible format for the rest of the architecture. This layer can be understood as to be comprised of spiking neurons with no internal dynamics, whose neural activity is determined by event arrival. Neurons are arranged in two-dimensional *neural maps*, one per polarity, resembling the grid-like topology of the vision sensor. Depending on the spatial resolution of these maps, each neuron is assigned with the polarity-specific events of one or multiple pixels with no overlap.

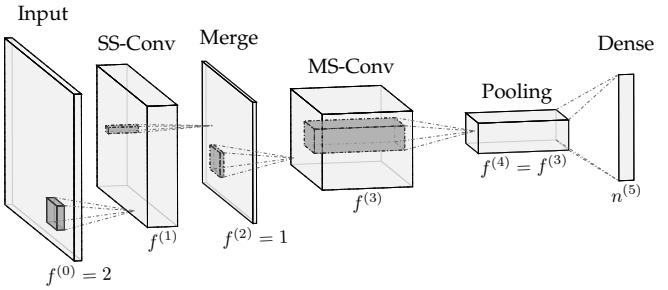


Fig. 5: Overview of the feedforward SNN architecture.

5.2 SS-Conv Layer: Feature Extraction

The goal of the single-synaptic convolutional layer, or SS-Conv, is to extract visual features from the input, and by doing so, to filter out the input events that may otherwise corrupt the learning process, and hence the performance, of subsequent layers in the architecture.

Neurons in this layer are retinotopically arranged in $k = 1, 2, \ldots, f^{(1)}$ two-dimensional maps. Each of these neurons receives spikes from presynaptic cells within a specific spatial receptive field, of size r, in all maps of the previous layer. This sparse connectivity is characterized by a set of excitatory synaptic weights, formally referred to as a *convolutional kernel* $\boldsymbol{W}_k \in \mathbb{R}^{r \times f^{(0)}}$, that is equal for all neurons belonging to the same map. This form of *weight sharing* ensures that, within a map, neurons are selective to the same feature but at different spatial locations.

Let the input connectivity of neuron i from the map k be characterized by the aforementioned convolutional kernel $\boldsymbol{W}_k$, the presynaptic trace $\boldsymbol{X}_i \in \mathbb{R}^{r \times f^{(0)}}$, and the spike train $s^{(0)}_{i,k}(t)$. Further, let $\boldsymbol{N}_{i,k}$ refer to the map-specific direct neural neighborhood of the cell, including itself. Then, considering neural connections as single-synaptic with transmission delay τ, the forcing function driving the internal dynamics of neurons in this layer is defined as follows:

$$\begin{aligned} i_{i,k}(t) = &\sum_{j=1}^{r} \sum_{ch=1}^{f^{(0)}} W_{j,ch,k} s^{(0)}_{j,ch}(t-\tau) \\ &- \max_{\forall b \in \boldsymbol{N}_{i,k}} \sum_{j=1}^{r} \sum_{ch=1}^{f^{(0)}} X_{b,j,ch}(t) \end{aligned} \tag{7}$$

Apart from the sparse connectivity, the only difference between this expression and the fully-connected formulation, i.e. Eq. (2), is in the homeostasis parameter. When arranged retinotopically, the neurons' dynamics do not only depend on their own presynaptic trace $\boldsymbol{X}_i$, but also on the synaptic traces characterizing their direct spatial neural neighborhood $\boldsymbol{N}_{i,k}$. By using the maximum trace, neurons are prevented from specializing to the leading edge of moving visual features, rather than to the features themselves (see Appendix D.1).

An augmentation of the proposed STDP rule is also required to handle the fact that multiple updates can be generated simultaneously in different spatial locations of the same map. Since these neurons share convolutional kernel, $\Delta\boldsymbol{W}_k$ is computed through synapse-specific averages of the local contributions. Additionally, due to the high overlap of presynaptic receptive fields, the WTA inhibitory mechanism

0162-8828 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2019.2903179, IEEE Transactions on Pattern Analysis and Machine Intelligence

described in Section 4.1 is expanded to cells within a small neighborhood of the firing neurons, regardless of the neural map. Note that, after learning, only the neuron-specific competition is maintained.

5.3 Merge Layer: Feature Aggregation

Due to the aperture problem [71], the different types of local motion that can be perceived at this stage of the architecture are exclusively dependent on the spatial configuration of input features, i.e. their appearance, and not on their polarity. Consequently, the $f^{(1)}$ neural maps of the SS-Conv layer can be merged into a single combined map without losing useful information for motion perception. The Merge layer is used for this purpose. Compared to when local motion is to be perceived directly from the SS-Conv output, this operation results in a decrease of both the number of convolutional kernels required in the subsequent layer, and the amount of per-kernel trainable parameters.

Similarly to SS-Conv, the Merge layer is convolutional and single-synaptic. The internal dynamics of its neurons is driven by Eq. (7) (with $l = 2$ in this case), but without the need for $\boldsymbol{N}_{i,k}$ since presynaptic connections are not plastic. Because of the latter, the application of the WTA mechanism is also neglected. Instead, this layer is characterized by a single 1×1 convolutional kernel with unitary connections to each of the neural maps of the previous layer.

5.4 MS-Conv Layer: Local Motion Perception

MS-Conv is presented as a variation of the SS-Conv layer whose role is to provide local motion estimates of the features extracted in the previous layers, by means of velocity-selective neurons. Similarly to feature identification, this selectivity emerges from visual experience through STDP.

For the purpose of local motion perception, we propose an augmentation of Eq. (7) based on the foundations of frequency-based optical flow methods [70] and bio-inspired motion detectors [69], [81]. Firstly, motion is to be extracted as orientation in the spatiotemporal domain. Therefore, neural connections in the MS-Conv layer are considered multisynaptic with different constant transmission delays as given by $\boldsymbol{\tau} \in \mathbb{R}^m$. Secondly, since these delays (and the rest of neural parameters) are equal for all (spatiotemporal) convolutional kernels, inhibitory synapses are required to prevent the firing of erroneous postsynaptic spikes when the input trace only fits part of the excitatory component of the kernels. To account for this, each MS-Conv kernel is defined by a pair of excitatory and inhibitory plastic weight matrices, denoted by $\boldsymbol{W}_k^{\text{exc}} \in \mathbb{R}^{r \times m}$ and $\boldsymbol{W}_k^{\text{inh}} \in \mathbb{R}^{r \times m}$, respectively. According to these additions, the forcing function of cells in this layer is expressed as:

$$\begin{aligned} i_{i,k}(t) = &\sum_{j=1}^{r}\sum_{d=1}^{m}(W_{j,d,k}^{\text{exc}} + \beta W_{j,d,k}^{\text{inh}})s_j^{(2)}(t-\tau_d) \\ &- \max_{\forall b \in \boldsymbol{N}_{i,k}} \sum_{j=1}^{r}\sum_{d=1}^{m} X_{b,j,d}(t) \end{aligned} \tag{8}$$

where $\beta \in [0,1]$ scales the impact of inhibitory synapses, and the presynaptic trace is defined as $\boldsymbol{X}_i \in \mathbb{R}^{r \times m}$.

Due to the neural spatial disposition, the implementation of STDP in this layer is, in essence, identical to the one employed for SS-Conv. The only difference comes from the fact that, for inhibitory synapses, the weights are initialized at 0, and $w_{\text{init}}^{\text{inh}}$ is set to $-w_{\text{init}}^{\text{exc}}$. This discrepancy between $w_{\text{init}}^{\text{inh}}$ and the initialization weight enables neurons in this layer to be reactive to different input features until specialization.

5.5 Pooling Layer: From Local to Global

As an intermediate stage between the MS-Conv and Dense layers, the Pooling layer is employed in the SNN architecture as a means to reduce the spatial dimensionality of the former, and hence to facilitate the learning process of the latter. The intuition of this layer is that, by pooling local motion estimates over large portions of the visual scene, a more accurate measure of the global motion in each of these regions can be obtained, thus mitigating the effect of the aperture problem [71].

Similarly to the Merge layer, the Pooling layer is convolutional and single-synaptic, and its presynaptic connections are not plastic. This layer is characterized by the same number of neural maps as the MS-Conv, each one assigned with an excitatory kernel $\boldsymbol{W}_k$ that has unitary weights with its presynaptic counterpart and null with the rest. In addition, there is no overlap between receptive fields.

5.6 Dense Layer: Global Motion Perception

The Dense layer, as the final stage of the SNN architecture, is comprised of individual neurons fully connected to cells in the Pooling layer via single-synaptic plastic connections. Similarly to final regions of biological visual motion systems [1], [3], neurons in this layer develop selectivity to the global motion of the scene from visual experience through STDP.

With respect to implementation details, synaptic plasticity is conducted as described in Section 4, and the forcing function of Dense neurons resembles Eq. (3), but referring to the convolutional presynaptic layer to which these cells are connected. This expression is then defined as:

$$i_i(t) = \sum_{j=1}^{n^{(4)}}\sum_{ch=1}^{f^{(4)}} \left(W_{i,j,ch}\, s_{j,ch}^{(4)}(t-\tau) - X_{i,j,ch}(t)\right) \tag{9}$$

where the weights and trace of input connections are defined as $\boldsymbol{W}_i \in \mathbb{R}^{n^{(4)} \times f^{(4)}}$ and $\boldsymbol{X}_i \in \mathbb{R}^{n^{(4)} \times f^{(4)}}$, respectively.

6 Experimental Results

In this section, we evaluate the performance of our SNN on synthetic and real event sequences. Appendix B includes illustrations of the textures and natural scenes employed for generating these sequences, together with other implementation details, such as network parameters, sensor characteristics, training settings, and data augmentation mechanisms.

6.1 Synthetic Data Experiment

Firstly, we assess our motion-selective architecture on several noise-free sequences restricted to the pure vertical and horizontal image motion of a checkerboard pattern. This very structured texture and motion facilitate the understanding of the behavior and main properties of the network. Visual stimuli and ground truth were generated with

0162-8828 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2019.2903179, IEEE Transactions on Pattern Analysis and Machine Intelligence

the DVS simulator [82], and this analysis is based on the planar optical flow formulation from [55] (see Appendix C).

Starting with the SS-Conv layer, Fig. 6 shows the four convolutional kernels learned from these sequences. With this kernel scale, our learning rule leads to the successful identification of edges at the different spatial orientations present in the input data, and with the two combinations of event polarity. Using these kernels for feature extraction, and aggregating their spiking activity in the Merge layer, an MS-Conv layer consisting of sixteen spatiotemporal kernels was trained thereafter. Fig. 7 shows the appearance of these kernels after convergence, and the response of their corresponding neural maps as a function of the ventral flow components (ω_x, ω_y).

This figure confirms that, with the connectivity pattern of the MS-Conv layer, STDP leads to the successful identification of the spatiotemporally-oriented traces of input features, and hence their local motion. Out of the sixteen kernels trained, seven specialized to pure horizontal motion, and the remaining nine to pure vertical. Each direction of motion (up, down, left, right) was captured by at least four kernels, which, in turn, were selective to a particular stimulus speed. For instance, upward motion was identified by kernels $k = \{13, 14, 15, 16\}$, from slow to fast tuning speed. Therefore, kernels in this layer can be understood as local velocity-tuned filters that resemble those employed in frequency-based optical flow methods [53], [56], [68], [70]. However, instead of being manually designed, these filters emerge from visual experience in an unsupervised fashion. A three-dimensional illustration of two MS-Conv kernels can be found in Appendix D.2.

In addition, remarkable is the fact that two of the (generally) four kernels that specialized to each of the aforementioned motion directions have overlapping neural responses despite the WTA mechanism described in Section 4.1. This is indicative of the relatively weak speed selectivity of MS-Conv neurons in comparison to their strong direction selectivity. Appendix D.3 confirms these results through an evaluation of both selectivities as a function of β.

Lastly, selectivity to global motion emerges in neurons from a Dense layer trained as the final stage of the SNN, using the low-dimensional activity of the Pooling layer. Fig. 8 shows the neural response (after convergence) of the sixteen cells in this layer as a function of (ω_x, ω_y). From this figure, it can be seen that neurons are successful at capturing the dominant global motion pattern from the spatial distribution of local motion estimates from previous layers. Out of the neurons trained, groups of four specialized to each motion direction, with different tuning speeds. Note that the velocity-selective properties of these neurons are exclusively dependent on those of the MS-Conv kernels. Appendix D.4 includes an evaluation of the temporal activity of these neurons in response to speed profiles that differ from the constant-speed sequences employed for learning.

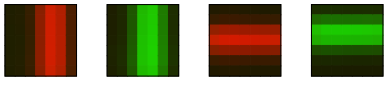
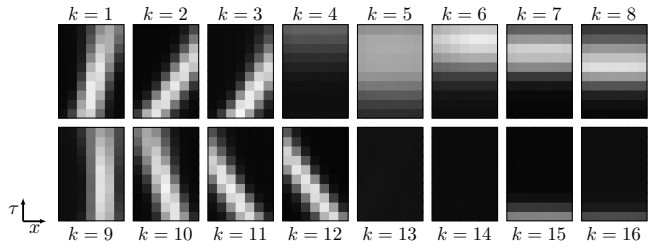
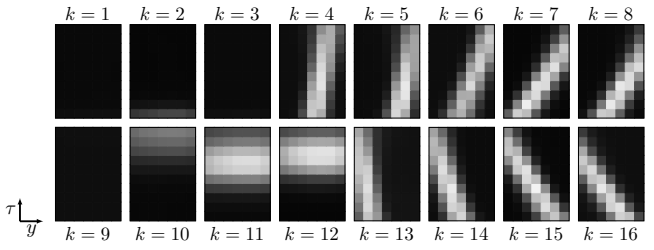


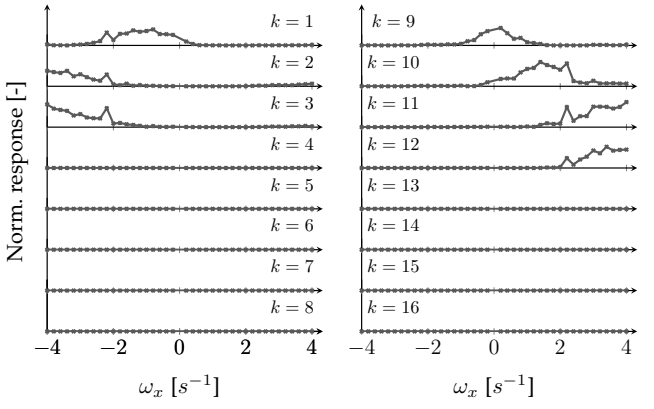
Fig. 6: SS-Conv kernels learned from the checkerboard texture. Synaptic strength is encoded in color brightness: green for input neurons with positive (event) polarity, and red for negative.



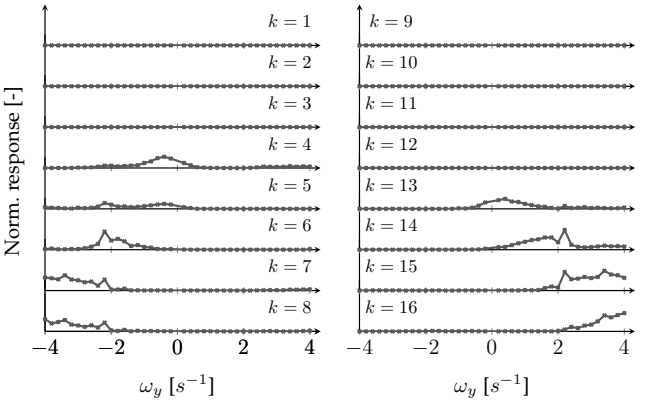
(a) x-τ representation



(b) y-τ representation



(c) Pure horizontal motion

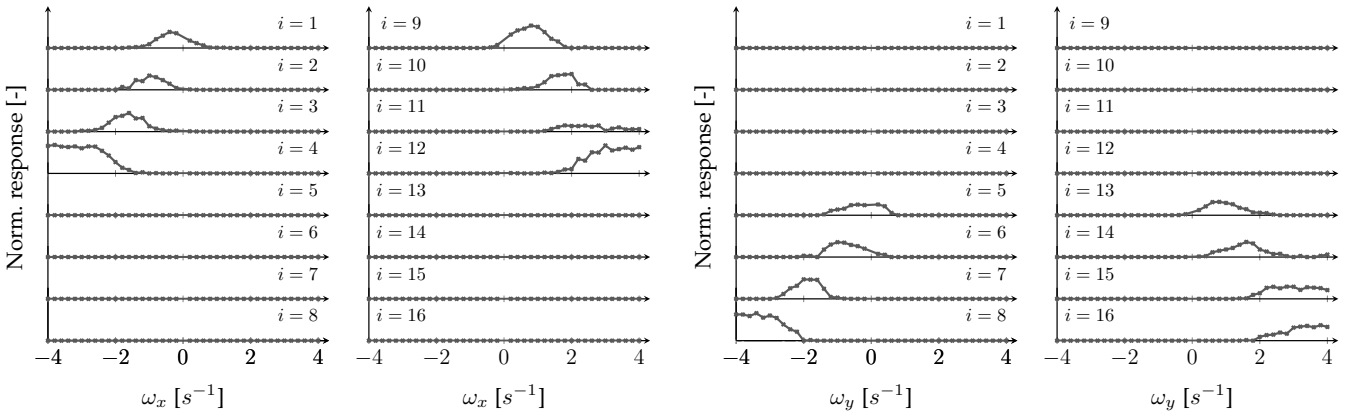


(d) Pure vertical motion

Fig. 7: Appearance (top) and neural response (bottom) of the sixteen spatiotemporal kernels learned from the checkerboard texture in the MS-Conv layer. Response plots are normalized by the maximum kernel response on the stimuli evaluated: 8.2763 spikes/ms by k =11 for ω_x = 4.0 s^{-1}. Synaptic strength is encoded with brightness using the kernel formulation from Eq. (8), i.e. $W^{\text{exc}} + \beta W^{\text{inh}}$.

0162-8828 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2019.2903179, IEEE Transactions on Pattern Analysis and Machine Intelligence



(a) Pure horizontal global motion (b) Pure vertical global motion

Fig. 8: Neural response of the sixteen individual neurons from the Dense layer trained in the checkerboard texture. Response plots are normalized by the maximum neural response on the stimuli evaluated: 0.3 spikes/ms by $i = 4$ for $\omega_x = -3.8\ s^{-1}$.

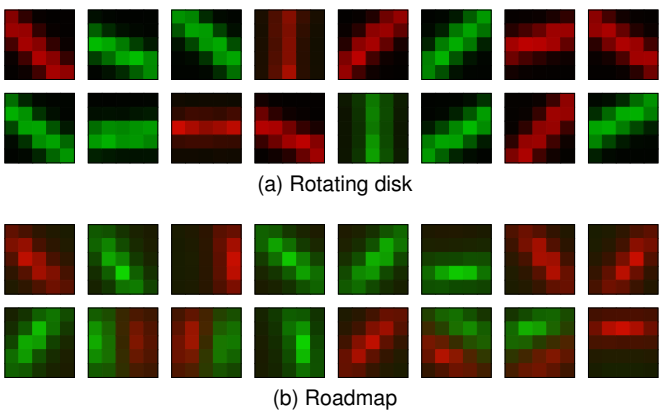
6.2 Real Data Experiments

For the experiments with real data, we use samples from different sources. In a first evaluation, we employ the rotating-disk sequence from [83], which provides input events corresponding to a disk slowly turning at a constant speed. Furthermore, several unconstrained recordings of a roadmap pattern are used in a second experiment characterized by more unstructured and noisy visual stimuli. For this, we also use natural scene sequences from the Event Camera Dataset [82] for validation. The DAVIS [11] and SEES1 [12] are the DVS sensors with which this data was generated.

6.2.1 Rotating-Disk Sequence

Fig. 9a shows the appearance of the SS-Conv kernels trained on the rotating-disk sequence. Similarly to the checkerboard case, neurons in this layer become selective to the most frequent input features, which are edges at different spatial orientations, and of different event polarity.

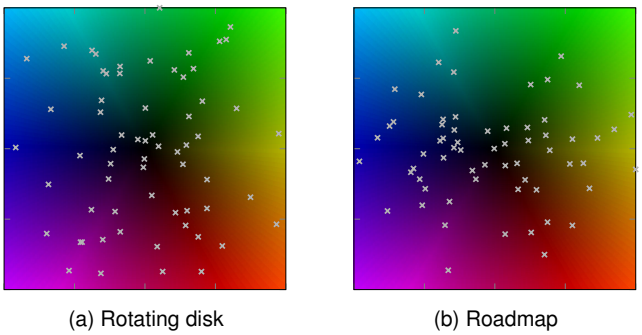
With respect to the MS-Conv layer of this architecture, Fig. 10a shows its 64 kernels in the (normalized) optical flow space, according to the method explained in Appendices D.5 and D.6. From this figure, we observe that, through our STDP rule, these MS-Conv kernels learn to identify a wide variety of optical flow vectors, including diagonal motion at different speeds. The performance of this layer in local motion perception can be assessed from the qualitative results in Fig. 12 (first two rows). Here, we compare the response of the network at this stage to the output of EV-FlowNet [16], which represents the state-of-the-art of conventional ANNs in event-based optical flow estimation. From these results, in both the clockwise and counterclockwise sequences, the response of the MS-Conv layer resembles that of EV-FlowNet, thus confirming the validity of our SNN in local motion perception. Additional qualitative results are provided in the supplementary video (see Appendix D.9).



(a) Rotating disk

(b) Roadmap

Fig. 9: SS-Conv kernels learned from real sequences. Synaptic strength is encoded in color brightness.



(a) Rotating disk (b) Roadmap

Fig. 10: MS-Conv kernels learned from real sequences in the (normalized) optical flow space.

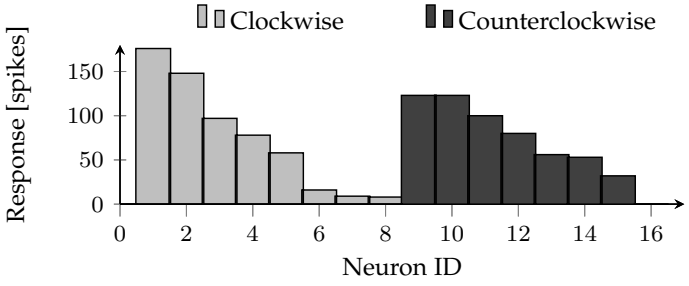


Fig. 11: Neural response of the individual neurons from the Dense layer trained in the rotating-disk sequence.

0162-8828 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2019.2903179, IEEE Transactions on Pattern Analysis and Machine Intelligence

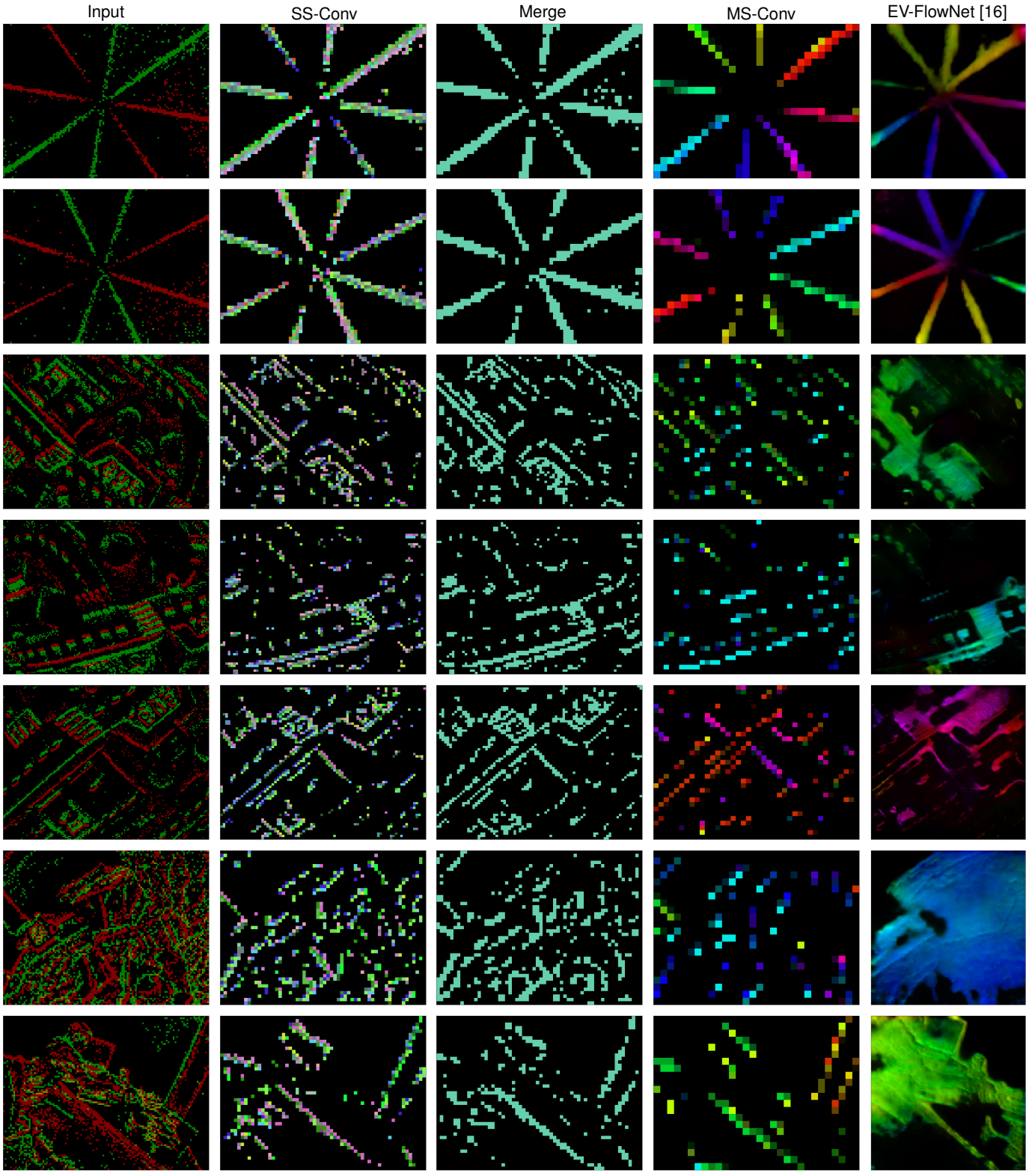


Fig. 12: Qualitative results from the evaluation on real event sequences. From left to right, the first column corresponds to the input events, the following three to the spiking response of the SS-Conv, Merge, and MS-Conv layers, respectively; and the last column to the optical flow estimation of EV-FlowNet [16]. A color is assigned to each of the kernels comprising the SS-Conv and MS-Conv layers. MS-Conv color reference shown in Fig. 10, and computed as in Appendices D.5 and D.6. SS-Conv reference not shown in this paper.

Lastly, a Dense layer comprised of sixteen neurons was trained, and the response of its cells is shown in Fig. 11. As expected, the two global motion patterns present in the data are successfully captured: half of the neurons react to clockwise rotation, and the rest to counterclockwise. Besides competition, the different response levels are due to distinct distributions of local motion estimates in the Pooling layer leading to the same global motion pattern.

0162-8828 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2019.2903179, IEEE Transactions on Pattern Analysis and Machine Intelligence

6.2.2 Roadmap Texture and Natural Scenes

Fig. 9b shows the appearance of the SS-Conv kernels from the SNN trained on roadmap recordings. Similarly to those obtained with the rotating disk, these kernels learned edges (and combinations thereof) at several orientations, and of different polarities. However, note that kernel appearance is significantly less smooth due to the unstructured and low-contrast features of this texture, besides the sensor noise.

Regarding the MS-Conv layer, Fig. 10b shows its 64 spatiotemporal kernels in the (normalized) optical flow space (according to Appendices D.5 and D.6). In this figure, we observe that despite the wide variety of vectors learned, these are not as uniformly distributed as for the rotating-disk case. One can see that, first, horizontal motion is the most frequent local image motion type in the roadmap recordings; and second, the unsupervised nature of STDP prioritizes frequent features over others, less frequent, that may be more distant in this two-dimensional space.

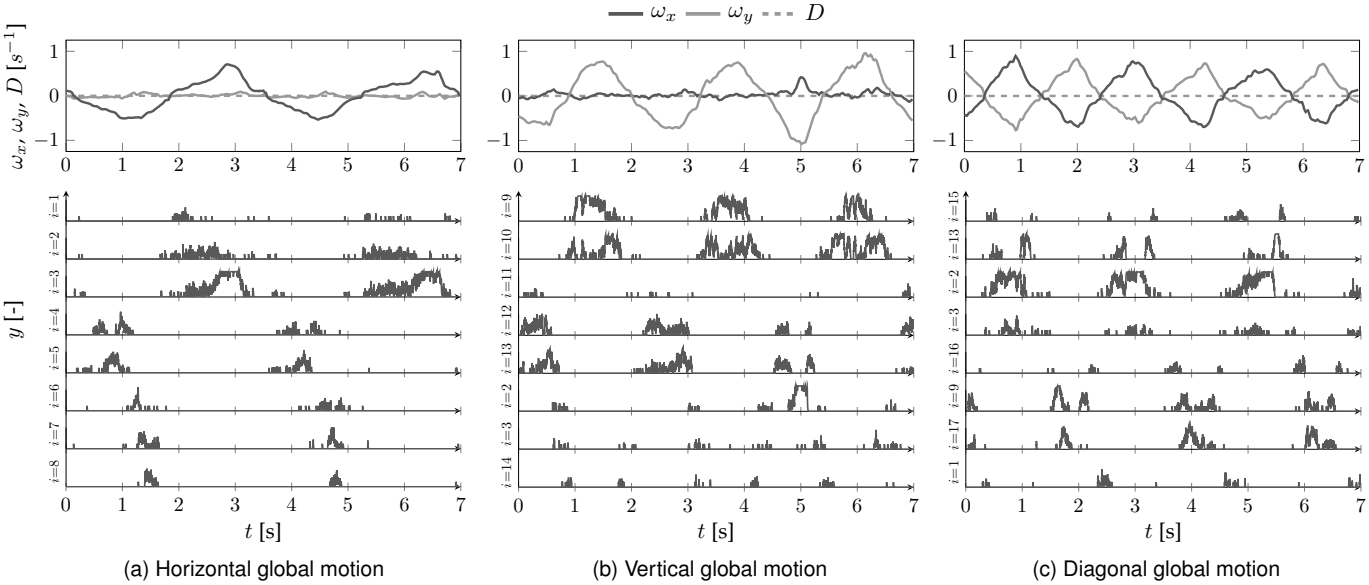
Qualitative results of the network performance up to this layer are shown in Fig. 12 for roadmap and natural scene recordings (last five rows). We draw several conclusions from these results. Firstly, the SS-Conv layer is a key component of the architecture, since it successfully filters out inconsistent local events sequences, which benefits the learning and performance of subsequent layers. Secondly, the optical flow estimation of EV-FlowNet [16] validates our MS-Conv layer, since it estimates highly similar optical flow vectors. However, there is a significant difference between the estimates of these two approaches, besides resolution (i.e. detail level). EV-FlowNet [16] performs best in high texture regions, providing a semi-dense estimate of the local motion. On the other hand, our network only provides local motion estimates whenever and wherever it discerns features whose spatiotemporal trace fits one of the MS-Conv kernels. Due to trace overlap, no estimation is provided for image regions with high feature density. This limitation comes from the working principle of this layer, which takes inspiration from frequency-based optical flow methods [70] and bio-inspired motion detectors [69], [81], and for which these regions are also problematic. Additional qualitative results are provided in the supplementary video (see Appendix D.9).

Lastly, Fig. 13 shows the temporal activity of some of the 32 neurons comprising the Dense layer of this architecture, in response to several global planar motion patterns. These results confirm the validity of this layer, and hence of the entire SNN, in becoming selective to this motion information through STDP. Moreover, similarly to the rotating-disk case, these results reinforce that, since notably different distributions of local motion estimates may correspond to the same global motion type, multiple Dense neurons can specialize to the same motion pattern without overlapping responses. This is further explained and illustrated in Appendix D.7 for neurons $i = \{4, \ldots, 8\}$ from Fig. 13a.

6.3 STDP Evaluation

The final experiment of this work consists in an evaluation of several STDP formulations in the task of learning the kernels of an SS-Conv layer from the recordings of the roadmap texture. Specifically, we compare our rule, as in Section 4, to those proposed by Kheradpisheh *et al.* [25], and Shrestha *et al.* [27]; two of the most recent multiplicative formulations that have successfully been used for image classification with SNNs. Fig. 14 shows the weight distribution evolution of the SS-Conv kernels throughout the learning process, using each of the aforementioned formulations. Kernel appearance after learning is shown in Appendix D.8.

The working principle of all STDP formulations is essentially the same. Whenever a neuron fires, the presynaptic connections that transferred the input spikes causing the firing are potentiated, while those that did not are

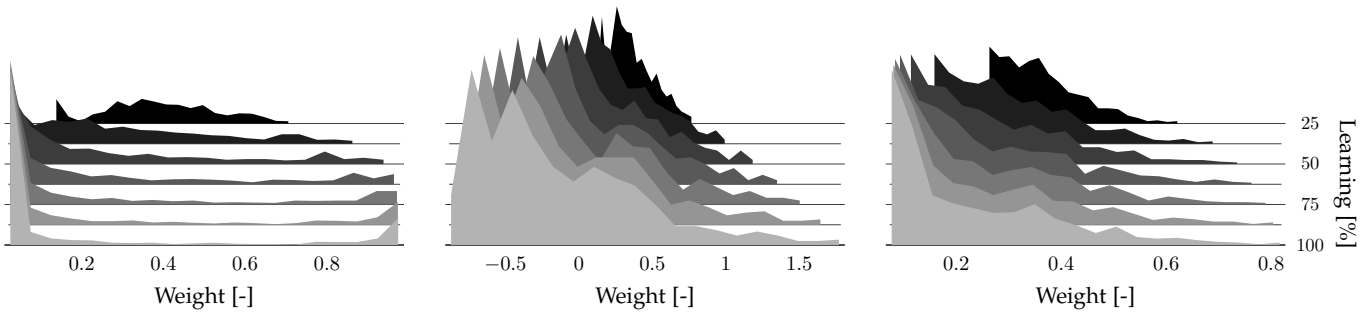


(a) Horizontal global motion (b) Vertical global motion (c) Diagonal global motion

Fig. 13: Temporal course of the postsynaptic trace (as in Appendix D.4) of the eight most-active neurons (for each case) from the Dense layer learned from the roadmap texture (bottom), in response to different global planar motion patterns (top). Plots are normalized by the maximum trace on the stimuli evaluated: 1.0 by i = 3 at t = 3.0 s for the horizontal motion case. Optical flow visual observables (ω_x, ω_y, D) computed from the event sequences with the planar optical flow formulation from [55] (see Appendix C).

0162-8828 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2019.2903179, IEEE Transactions on Pattern Analysis and Machine Intelligence



(a) Kheradpisheh *et al.* [25] (b) Shrestha *et al.* [27] (c) Ours

Fig. 14: Evolution of the weight distribution of sixteen SS-Conv kernels throughout the learning process, using Kheradpisheh's [25], Shrestha's [27], and our STDP formulation. Results obtained with the roadmap texture, the same learning rate, and the same budget of training sequences.

depressed. The differences are in how the relevance of a connection is determined, and in how it is taken into account to compute the weight update ΔW. Both Kheradpisheh's [25] and Shrestha's [27] formulations use temporal windows of fixed length to determine whether an input spike, and so its corresponding synapse, had an influence on the postsynaptic firing. However, this information is only employed to determine whether a synapse is potentiated or depressed, and not in the computation of ΔW. On the one hand, Kheradpisheh's weight update is proportional to the current weight: $\Delta W \propto W_{i,j,d}(1 - W_{i,j,d})$. Results show that this rule leads to the learning of ambiguous features that fail to capture the spatiotemporal properties of the input, since all the weights become either null or unitary (see Fig. 14a). On the other hand, Shrestha's rule incorporates the weight dependency in an inversely proportional manner: $\Delta W \propto e^{-W_{i,j,d}}$ for potentiation, and $\Delta W \propto -e^{W_{i,j,d}}$ for depression. As shown, even though the ΔW for potentiation (depression) diminishes as the weights increase (decrease), weights keep increasing (decreasing) throughout the learning process (see Fig. 14b), and hence constraints to prevent them from exploding (vanishing) are required. The use of these constraints would, in turn, result in a bimodal weight distribution similar to that of Kheradpisheh's rule, with the aforementioned drawbacks.

As explained in Section 4, and to the best of the authors' knowledge, our STDP implementation is the first multiplicative formulation in incorporating synaptic relevance in the computation of ΔW, resulting in an update rule whose LTP and LTD processes are not mutually exclusive. We combine (normalized) presynaptic trace information as a measure of synaptic relevance, with the inversely proportional weight dependency from [27]. Results, and the stability proof included in Appendix A, confirm that with our novel STDP formulation, an equilibrium weight is established for each synapse, towards which the weights converge throughout the learning process (see Fig. 14c). Since the equilibrium state depends on synaptic relevance, the features learned are successful at capturing the spatiotemporal properties of the input.

7 Conclusion

In this paper, we have presented the first SNN in which selectivity to the local and global motion of the visual scene emerges through STDP from event-based stimuli. The success of this emergence depends on three contributions. First, an adaptive spiking neuron model is necessary to handle the rapidly varying input statistics of event-based sensors, and we present a novel suitable formulation for this purpose. Second, we introduce a novel STDP implementation that, contrary to the current state-of-the-art of this learning protocol, is inherently stable. Third, we propose an SNN architecture that learns to perform a hierarchical feature extraction, effectively capturing geometric features, identifying the local motion of these features, and integrating this information into a global ego-motion estimate. We hope that this work, and the framework published alongside it, will provide the first step of many towards highly efficient artificial motion perception.

References

[1] A. Borst and M. Helmstaedter, "Common circuit design in fly and mammalian motion vision," *Nature Neuroscience*, vol. 18, no. 8, pp. 1067–1076, 2015.
[2] J. J. Gibson, "The perception of the visual world." 1950.
[3] A. Borst, J. Haag, and D. F. Reiff, "Fly motion vision," *Annual Review of Neuroscience*, vol. 33, pp. 49–70, 2010.
[4] M. V. Srinivasan, S. Zhang, M. Lehrer, and T. Collett, "Honeybee navigation en route to the goal: Visual flight control and odometry," *Journal of Experimental Biology*, vol. 199, no. 1, pp. 237–244, 1996.
[5] C. De Wagter, S. Tijmons, B. D. W. Remes, and G. C. H. E. de Croon, "Autonomous flight of a 20-gram flapping wing MAV with a 4-gram onboard stereo vision system," in *Proceedings of the 2014 IEEE International Conference on Robotics and Automation*, 2014, pp. 4982–4987.
[6] M. Karásek, F. T. Muijres, C. De Wagter, B. D. W. Remes, and G. C. H. E. de Croon, "A tailless aerial robotic flapper reveals that flies use torque coupling in rapid banked turns," *Science*, vol. 361, no. 6407, pp. 1089–1094, 2018.
[7] A. Kirkwood and M. F. Bear, "Hebbian synapses in visual cortex," *Journal of Neuroscience*, vol. 14, no. 3, pp. 1634–1645, 1994.
[8] L. C. Katz and C. J. Shatz, "Synaptic activity and the construction of cortical circuits," *Science*, vol. 274, no. 5290, pp. 1133–1138, 1996.
[9] P. Lichtsteiner, C. Posch, and T. Delbruck, "A 128x128 120 dB 15 μs latency asynchronous temporal contrast vision sensor," *IEEE Journal of Solid-State Circuits*, vol. 43, no. 2, pp. 566–576, 2008.
[10] C. Posch, D. Matolin, and R. Wohlgenannt, "A QVGA 143 dB dynamic range frame-free PWM image sensor with lossless pixel-level video compression and time-domain CDS," *IEEE Journal of Solid-State Circuits*, vol. 46, no. 1, pp. 259–275, 2011.

0162-8828 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2019.2903179, IEEE Transactions on Pattern Analysis and Machine Intelligence

[11] C. Brandli, R. Berner, M. Yang, S. Liu, and T. Delbruck, "A 240x180 130 dB 3 µs latency global shutter spatiotemporal vision sensor," *IEEE Journal of Solid-State Circuits*, vol. 49, no. 10, pp. 2333–2341, 2014.

[12] C. Brandli, R. Berner, M. Osswald, and N. Baumli, "Silicon eye event sensor SEES1," *Insightness AG*, 2018. [Online]. Available: https://www.insightness.com

[13] R. Benosman, C. Clercq, X. Lagorce, S. Ieng, and C. Bartolozzi, "Event-based visual flow," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 25, no. 2, pp. 407–417, 2014.

[14] D. Fortun, P. Bouthemy, and C. Kervrann, "Optical flow modeling and computation: A survey," *Computer Vision and Image Understanding*, vol. 134, pp. 1–21, 2015.

[15] E. Ilg, N. Mayer, T. Saikia, M. Keuper, A. Dosovitskiy, and T. Brox, "Flownet 2.0: Evolution of optical flow estimation with deep networks," in *Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition*, vol. 2, 2017, pp. 1647–1655.

[16] A. Z. Zhu, L. Yuan, K. Chaney, and K. Daniilidis, "EV-FlowNet: Self-supervised optical flow estimation for event-based cameras," *Robotics: Science and Systems*, 2018.

[17] C. Ye, A. Mitrokhin, C. Parameshwara, C. Fermüller, J. A. Yorke, and Y. Aloimonos, "Unsupervised learning of dense optical flow and depth from sparse event data," 2018. [Online]. Available: https://arxiv.org/abs/1809.08625

[18] W. Maass, "Networks of spiking neurons: The third generation of neural network models," *Neural Networks*, vol. 10, no. 9, pp. 1659–1671, 1997.

[19] G. Orchard and R. Etienne-Cummings, "Bioinspired visual motion estimation," *Proceedings of the IEEE*, vol. 102, no. 10, pp. 1520–1536, 2014.

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

[21] M. Davies, N. Srinivasa, T.-H. Lin, G. Chinya, Y. Cao, S. H. Choday, G. Dimou, P. Joshi, N. Imam, S. Jain *et al.*, "Loihi: A neuromorphic manycore processor with on-chip learning," *IEEE Micro*, vol. 38, no. 1, pp. 82–99, 2018.

[22] N. Caporale and Y. Dan, "Spike timing–dependent plasticity: A Hebbian learning rule," *Annual Review of Neuroscience*, vol. 31, pp. 25–46, 2008.

[23] T. Masquelier and S. J. Thorpe, "Unsupervised learning of visual features through spike timing dependent plasticity," *Public Library of Science: Computational Biology*, vol. 3, no. 2, pp. 247–257, 2007.

[24] P. U. Diehl and M. Cook, "Unsupervised learning of digit recognition using spike-timing-dependent plasticity," *Frontiers in Computational Neuroscience*, vol. 9, pp. 1–9, 2015.

[25] S. R. Kheradpisheh, M. Ganjtabesh, S. J. Thorpe, and T. Masquelier, "STDP-based spiking deep convolutional neural networks for object recognition," *Neural Networks*, vol. 99, pp. 56–67, 2018.

[26] A. Tavanaei and A. S. Maida, "Multi-layer unsupervised learning in a spiking convolutional neural network," in *Proceedings of the 2017 IEEE International Joint Conference on Neural Networks*, 2017, pp. 2023–2030.

[27] A. Shrestha, K. Ahmed, Y. Wang, and Q. Qiu, "Stable spike-timing dependent plasticity rule for multilayer unsupervised and supervised learning," in *Proceedings of the 2017 International Joint Conference on Neural Networks*. IEEE, 2017, pp. 1999–2006.

[28] R. B. Stein, "A theoretical analysis of neuronal variability," *Biophysical Journal*, vol. 5, no. 2, pp. 173–194, 1965.

[29] A. L. Hodgkin and A. F. Huxley, "A quantitative description of membrane current and its application to conduction and excitation in nerve," *The Journal of Physiology*, vol. 117, no. 4, pp. 25–71, 1952.

[30] E. M. Izhikevich, "Simple model of spiking neurons," *IEEE Transactions on Neural Networks*, vol. 14, no. 6, pp. 1569–1572, 2003.

[31] W. M. Kistler, W. Gerstner, and J. L. van Hemmen, "Reduction of the Hodgkin-Huxley equations to a single-variable threshold model," *Neural Computation*, vol. 9, no. 5, pp. 1015–1045, 1997.

[32] M. Baudry, "Synaptic plasticity and learning and memory: 15 years of progress," *Neurobiology of Learning and Memory*, vol. 70, no. 1, pp. 113–118, 1998.

[33] K. Doya, "What are the computations of the cerebellum, the basal ganglia and the cerebral cortex?" *Neural Networks*, vol. 12, no. 7-8, pp. 961–974, 1999.

[34] D. O. Hebb, *The organisation of behaviour: A neuropsychological theory*. Wiley, 1952.

[35] W. Gerstner and W. M. Kistler, *Spiking neuron models: Single neurons, populations, plasticity*. Cambridge University Press, 2002.

[36] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Cognitive Modeling*, vol. 5, no. 3, pp. 533–536, 1988.

[37] J. H. Lee, T. Delbruck, and M. Pfeiffer, "Training deep spiking neural networks using backpropagation," *Frontiers in Neuroscience*, vol. 10, pp. 1–13, 2016.

[38] Y. Wu, L. Deng, G. Li, J. Zhu, and L. Shi, "Spatio-temporal backpropagation for training high-performance spiking neural networks," *Frontiers in Neuroscience*, vol. 12, pp. 1–12, 2018.

[39] A. Taherkhani, A. Belatreche, Y. Li, and L. P. Maguire, "A supervised learning algorithm for learning precise timing of multiple spikes in multilayer spiking neural networks," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–14, 2018.

[40] S. B. Shrestha and G. Orchard, "Slayer: Spike layer error reassignment in time," in *Advances in Neural Information Processing Systems*, 2018, pp. 1417–1426.

[41] J. A. Pérez-Carrasco, B. Zhao, C. Serrano, B. Acha, T. Serrano-Gotarredona, S. Chen, and B. Linares-Barranco, "Mapping from frame-driven to frame-free event-driven vision systems by low-rate rate coding and coincidence processing–application to feed-forward convnets," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 11, pp. 2706–2719, 2013.

[42] D. Zambrano, R. Nusselder, H. S. Scholte, and S. M. Bohte, "Efficient computation in adaptive artificial spiking neural networks," 2017. [Online]. Available: https://arxiv.org/abs/1710.04838

[43] B. Rueckauer, I.-A. Lungu, Y. Hu, M. Pfeiffer, and S.-C. Liu, "Conversion of continuous-valued deep networks to efficient event-driven networks for image classification," *Frontiers in Neuroscience*, vol. 11, pp. 1–12, 2017.

[44] R. V. Florian, "Reinforcement learning through modulation of spike-timing-dependent synaptic plasticity," *Neural Computation*, vol. 19, no. 6, pp. 1468–1502, 2007.

[45] E. M. Izhikevich, "Solving the distal reward problem through linkage of STDP and dopamine signaling," *Cerebral Cortex*, vol. 17, no. 10, pp. 2443–2452, 2007.

[46] J. O. Rombouts, P. R. Roelfsema, and S. M. Bohte, "Neurally plausible reinforcement learning of working memory tasks," in *Advances in Neural Information Processing Systems*, 2012, pp. 1871–1879.

[47] J. O. Rombouts, A. van Ooyen, P. R. Roelfsema, and S. M. Bohte, "Biologically plausible multi-dimensional reinforcement learning in neural networks," in *International Conference on Artificial Neural Networks*, 2012, pp. 443–450.

[48] J. Friedrich and M. Lengyel, "Goal-directed decision making with spiking neurons," *Journal of Neuroscience*, vol. 36, no. 5, pp. 1529–1546, 2016.

[49] Z. Bing, C. Meschede, K. Huang, G. Chen, F. Rohrbein, M. Akl, and A. Knoll, "End-to-end learning of spiking neural network based on R-STDP for a lane keeping vehicle," in *2018 IEEE International Conference on Robotics and Automation*. IEEE, 2018, pp. 1–8.

[50] M. Mozafari, M. Ganjtabesh, A. Nowzari-Dalini, S. J. Thorpe, and T. Masquelier, "Combining STDP and reward-modulated STDP in deep convolutional spiking neural networks for digit recognition," 2018. [Online]. Available: https://arxiv.org/abs/1804.00227

[51] B. D. Lucas and T. Kanade, "An iterative technique of image registration and its application to stereo," in *Proceedings of the 7th International Joint Conference on Artificial Intelligence*, vol. 2, 1981, pp. 674–679.

[52] R. Benosman, S. Ieng, C. Clercq, C. Bartolozzi, and M. Srinivasan, "Asynchronous frameless event-based optical flow," *Neural Networks*, vol. 27, pp. 32–37, 2012.

[53] T. Brosch, S. Tschechne, and H. Neumann, "On event-based optical flow detection," *Frontiers in Neuroscience*, vol. 9, pp. 1–15, 2015.

[54] M. T. Aung, R. Teo, and G. Orchard, "Event-based plane-fitting optical flow for dynamic vision sensors in FPGA," in *Proceedings of the 2018 IEEE International Symposium on Circuits and Systems*. IEEE, 2018, pp. 1–5.

[55] B. J. P. Hordijk, K. Y. W. Scheper, and G. C. H. E. de Croon, "Vertical landing for micro air vehicles using event-based optical flow," *Journal of Field Robotics*, vol. 35, no. 1, pp. 69–90, 2018.

[56] S. Tschechne, R. Sailer, and H. Neumann, "Bio-inspired optic flow from event-based neuromorphic sensor input." in *Proceedings of the 6th IAPR Workshop on Artificial Neural Networks in Pattern Recogniftion*. Springer, 2014, pp. 171–182.

0162-8828 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2019.2903179, IEEE Transactions on Pattern Analysis and Machine Intelligence

[57] F. Barranco, C. Fermuller, and Y. Aloimonos, "Bio-inspired motion estimation with event-driven sensors," in *International Work-Conference on Artificial Neural Networks*. Springer, 2015, pp. 309–321.

[58] T. Brosch and H. Neumann, "Event-based optical flow on neuromorphic hardware," in *Proceedings of the 9th EAI International Conference on Bio-inspired Information and Communications Technologies*, 2016, pp. 551–558.

[59] A. Z. Zhu, N. Atanasov, and K. Daniilidis, "Event-based feature tracking with probabilistic data association," in *Proceedings of the 2017 IEEE International Conference on Robotics and Automation*. IEEE, 2017, pp. 4465–4470.

[60] G. Gallego and D. Scaramuzza, "Accurate angular velocity estimation with an event camera," *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 632–639, 2017.

[61] A. Mitrokhin, C. Fermuller, C. Parameshwara, and Y. Aloimonos, "Event-based moving object detection and tracking," 2018. [Online]. Available: https://arxiv.org/abs/1803.04523

[62] G. Gallego, H. Rebecq, and D. Scaramuzza, "A unifying contrast maximization framework for event cameras, with applications to motion, depth, and optical flow estimation," in *Proceedings of the 2018 IEEE International Conference on Computer Vision and Pattern Recognition*, vol. 1, 2018.

[63] M. Liu and T. Delbruck, "ABMOF: A novel optical flow algorithm for dynamic vision sensors," 2018. [Online]. Available: https://arxiv.org/abs/1805.03988

[64] X. Lagorce, S.-H. Ieng, X. Clady, M. Pfeiffer, and R. B. Benosman, "Spatiotemporal features for asynchronous event-based data," *Frontiers in Neuroscience*, vol. 9, pp. 1–13, 2015.

[65] M. Giulioni, X. Lagorce, F. Galluppi, and R. B. Benosman, "Event-based computation of motion flow on a neuromorphic analog neural platform," *Frontiers in Neuroscience*, vol. 10, pp. 1–13, 2016.

[66] G. Haessig, A. Cassidy, R. Alvarez, R. Benosman, and G. Orchard, "Spiking optical flow for event-based sensors using IBM's TrueNorth neurosynaptic system," *IEEE Transactions on Biomedical Circuits and Systems*, vol. 12, no. 4, pp. 860–870, 2018.

[67] C. Richter, F. Röhrbein, and J. Conradt, "Bio-inspired optic flow detection using neuromorphic hardware," *Bernstein Conference on Computational Neuroscience*, 2014, poster.

[68] G. Orchard, R. Benosman, R. Etienne-Cummings, and N. V. Thakor, "A spiking neural network architecture for visual motion estimation," in *Proceedings of the 2013 IEEE Biomedical Circuits and Systems Conference*, 2013, pp. 298–301.

[69] W. Reichardt, "Autocorrelation, a principle for the evaluation of sensory information by the central nervous system," *Sensory Communication*, pp. 303–317, 1961.

[70] E. H. Adelson and J. R. Bergen, "Spatiotemporal energy models for the perception of motion," *Journal of the Optical Society of America*, vol. 2, no. 2, pp. 284–299, 1985.

[71] S. Ullman, *The interpretation of visual motion*. MIT Press, 1979.

[72] A. P. Shon, R. P. Rao, and T. J. Sejnowski, "Motion detection and prediction through spike-timing dependent plasticity," *Network: Computation in Neural Systems*, vol. 15, no. 3, pp. 179–198, 2004.

[73] O. G. Wenisch, J. Noll, and J. L. Van Hemmen, "Spontaneously emerging direction selectivity maps in visual cortex through STDP," *Biological Cybernetics*, vol. 93, no. 4, pp. 239–247, 2005.

[74] S. V. Adams and C. M. Harris, "A computational model of innate directional selectivity refined by visual experience," *Scientific Reports*, vol. 5, pp. 1–13, 2015.

[75] A. Morrison, A. Aertsen, and M. Diesmann, "Spike-timing-dependent plasticity in balanced random networks," *Neural Computation*, vol. 19, no. 6, pp. 1437–1467, 2007.

[76] M. Abercrombie, C. J. Hickman, and M. L. Johnson, *A dictionary of biology*. Routledge, 2017.

[77] F. Paredes-Vallés, *Neuromorphic Computing of Event-Based Data for High-Speed Vision-Based Navigation*. M.Sc. Thesis, Faculty of Aerospace Engineering, Delft University of Technology, 2018. [Online]. Available: TU Delft Education Repository.

[78] S. M. Bohte, "Efficient spike-coding with multiplicative adaptation in a spike response model," in *Advances in Neural Information Processing Systems*, 2012, pp. 1835–1843.

[79] M. C. W. Van Rossum, G. Q. Bi, and G. G. Turrigiano, "Stable Hebbian learning from spike timing-dependent plasticity," *Journal of Neuroscience*, vol. 20, no. 23, pp. 8812–8821, 2000.

[80] S. J. Thorpe, "Spike arrival times: A highly efficient coding scheme for neural networks," *Parallel Processing in Neural Systems*, pp. 91–94, 1990.

[81] H. B. Barlow and W. R. Levick, "The mechanism of directionally selective units in rabbit's retina." *The Journal of Physiology*, vol. 178, no. 3, pp. 477–504, 1965.

[82] E. Mueggler, H. Rebecq, G. Gallego, T. Delbruck, and D. Scaramuzza, "The event-camera dataset and simulator: Event-based data for pose estimation, visual odometry, and SLAM," *The International Journal of Robotics Research*, vol. 36, no. 2, pp. 142–149, 2017.

[83] B. Rueckauer and T. Delbruck, "Evaluation of event-based algorithms for optical flow with ground-truth from inertial measurement sensor," *Frontiers in Neuroscience*, vol. 10, pp. 1–17, 2016.

[84] H. C. Longuet-Higgins and K. Prazdny, "The interpretation of a moving retinal image," *Proceedings of the Royal Society of London B: Biological Sciences*, vol. 208, no. 1173, pp. 385–397, 1980.

[85] K. McGuire, G. C. H. E. de Croon, C. De Wagter, B. D. W. Remes, K. Tuyls, and H. Kappen, "Local histogram matching for efficient optical flow computation applied to velocity estimation on pocket drones," in *2016 IEEE International Conference on Robotics and Automation*. IEEE, 2016, pp. 3255–3260.

Federico Paredes-Vallés received his B.Sc. in Aerospace Engineering from the Polytechnic University of Valencia, Spain, in 2015, and his M.Sc. from Delft University of Technology, the Netherlands, in 2018. He is currently a Ph.D. candidate in the Micro Air Vehicle Laboratory at the latter university. His research interest is the intersection of machine learning, neuroscience, computer vision, and robotics.

Kirk Y. W. Scheper received his M.Sc. from the Faculty of Aerospace Engineering at Delft University of Technology, the Netherlands, in 2014. Since then, he has been a Ph.D. candidate in the Micro Air Vehicle Laboratory at the same university. His research focuses on the development of embedded software which facilitates high level autonomy of micro air vehicles. His work is mainly in the fields of evolutionary robotics, embodied cognition, and vision-based navigation.

Guido C. H. E. de Croon received his M.Sc. and Ph.D. in the field of Artificial Intelligence at Maastricht University, the Netherlands. His research interest lies with computationally efficient algorithms for robot autonomy, with an emphasis on computer vision. Since 2008 he has worked on algorithms for achieving autonomous flight with small and light-weight flying robots, such as the DelFly flapping wing MAV. In 2011-2012, he was a research fellow in the Advanced Concepts Team of the European Space Agency, where he studied topics such as optical flow based control algorithms for extraterrestrial landing scenarios. Currently, he is associate professor at Delft University of Technology, the Netherlands, where he is the scientific lead of the Micro Air Vehicle Laboratory.

0162-8828 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.