Event-based optical flow on neuromorphic hardware

Tobias Brosch
Institute of Neural Information Processing
Ulm University
Ulm, Germany
tobias.brosch@uni-ulm.de

Heiko Neumann
Institute of Neural Information Processing
Ulm University
Ulm, Germany
heiko.neumann@uni-ulm.de

ABSTRACT

Event-based sensing, i.e. the asynchronous detection of luminance changes, promises low-energy, high dynamic range, and sparse sensing. This stands in contrast to whole image frame-wise acquisition using standard cameras. Recently, we proposed a novel biologically inspired efficient motion detector for such event-based input streams and demonstrated how a canonical neural circuit can improve such representations using normalization and feedback. In this contribution, we suggest how such a motion detection scheme is defined by utilizing a canonical neural circuit corresponding to the resolution of cortical columns. In addition, we develop a mapping of key computational elements of this circuit model onto neuromorphic hardware. In particular, we focus on the recently developed TrueNorth chip architecture by IBM to realize a real-time, energy-efficient and adjustable neuromorphic optical flow detector. We demonstrate the function of the computations of the canonical model and its approximate neuromorphic realization.

Keywords

Optical flow, event-based sensing, address-event representation, real-time vision, neuromorphic computing

1. INTRODUCTION

The initial stages of visual processing extract a vocabulary of relevant feature items related to a visual scene. Rays of light reach the observer's eye and are transformed to internal representations. Conventional frame-based cameras read out measurements of all light-sensitive pixels at a fixed rate. Since the temporal sampling rate is limited through reading all pixel values in a fixed time interval, fast local luminance changes are integrated over time and cannot be differentiated in the further processing. When no changes occur in the intensity function, redundant information is generated that is carried to the subsequent processing steps. Address-event representations (AER), on the other hand, originate from image sensors in which pixels operate at individual rates generating events based on local decisions to generate an output response, like in the mammalian retina [23, 19]. We will focus on silicon retinas that generate an AER, namely the dynamic vision sensor (DVS; [15]). Whenever the change in the log-luminance function exceeds a predefined threshold ϑ, either through an increase or decrease in the luminance, events $e_k \in \{-1, 1\}$ are generated at times t_k. Such events emulate spike sequences of retinal ganglion cells sensitive to on- and off-contrast cells, respectively (Fig. 1).

We have proposed a canonical circuit architecture to define an abstract definition of cortical processing at a mesoscopic level of description [8]. Neurons organized into distinct layers in cortical areas show different clusterings, lateral intra-cortical connectivities [2], and different major termination zones. Different cell types are distributed and excitatory pyramidal cells are accompanied by a myriad of inhibitory cells of different subtypes [18]. In [4] we analyzed an abstract model of such cortical processing. The canonical circuit consists of an initial stage of filtering the input signals and lateral interaction followed by a stage of modulation that allows to enhance the activity generated by the first stage through contextual information [26], and a final stage of competitive interaction between cells in a pool defined over a space-feature domain that leads to activity normalization in a group of cells [9]. This model framework has already been successfully applied to account for various neural and perceptual phenomena in boundary grouping, texture boundary detection, figure-ground segregation, or motion detection and integration [31, 1, 28, 33, 6]. The theoretical investigation of the dynamical properties of the architecture in [4] has demonstrated the stable and inhibition stabilized regimes and has shown the influencing model components to force occurrences of oscillatory behavior.

In this contribution, we will investigate such canonical circuit architecture and its impact for detecting initial motion. The initial filtering is, thus, specialized to spatio-temporal input detection as an initial stage. In [7], we discussed what kind of information is accessible from the initial stages of event-based visual sensing and compared different approaches to estimate optical flow from the spatio-temporal stream of on- and off-events visualized in Fig. 1. In response to this analysis we proposed a novel biologically inspired motion detector and conducted experiments to validate the theoretical predictions of flow estimation. In addition, we have shown that the interactions of feedback and normalization in a canonical neural circuit can enhance the representation [32, 7]. Here,

BICT 2015, December 03-05, New York City, United States

DOI 10.4108/eai.3-12-2015.2262447

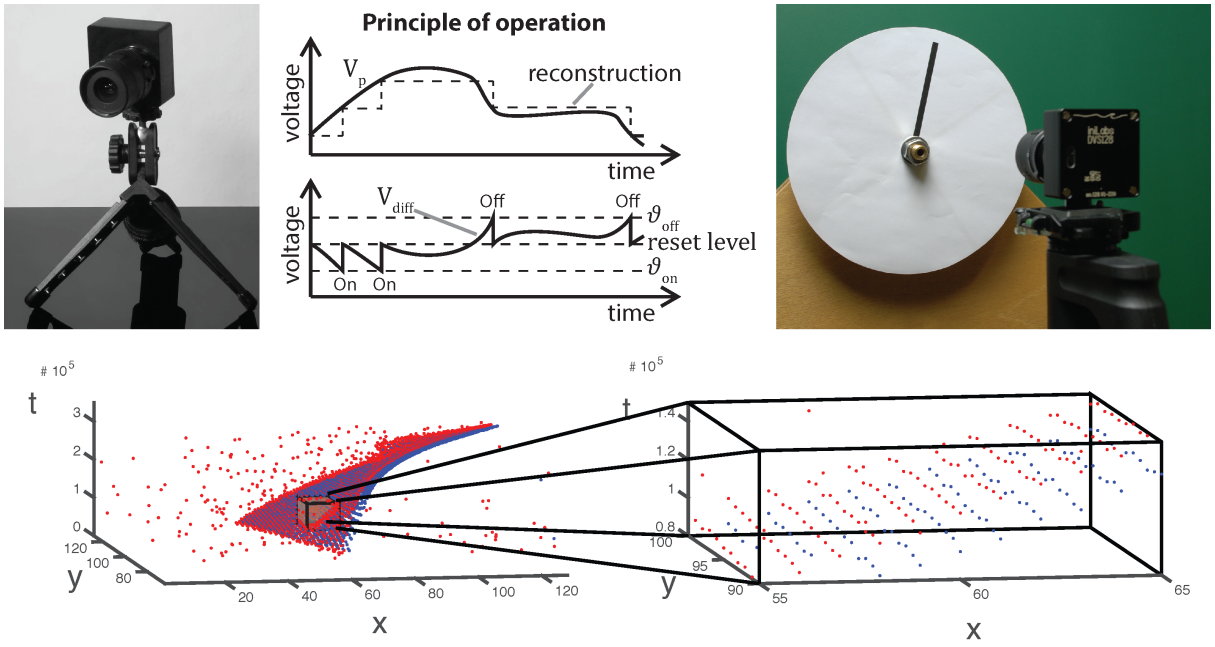

Figure 1: Top (left to right): Image, operating principle, and stimulus generation. Luminance changes exceeding a threshold evoke ON/OFF events. The very low latency of the DVS ($15\mu s$) requires analog stimulus generation (right). Bottom: Spatio-temporal event cloud generated by the rotating stimulus in the upper right. Only few events have been generated at a single location (reproduced from [7]).

we now show how the proposed event-based optical flow detector and the canonical neural circuit operations can be mapped to neuromorphic hardware to realize a real-time, energy-efficient and adjustable neuromorphic optical flow detector.

2. NOMENCLATURE, CANONICAL NEURAL CIRCUIT, AND FILTER DESIGN

In this section we recapitulate the overall processing scheme previously described in [7].

2.1 Event-stream generation

The input data is generated by an event-driven spatial sensor that approximates a retinal image. We describe the stream of events by the function (c.f. [7])

$$e : \mathbb{R}^2 \times \mathbb{R} \to \{-1, 0, 1\} \tag{1}$$

which is always zero except for tuples $(x_k, y_k; t_k) = (\boldsymbol{p}_k; t_k)$ which define the location and time of an event k generated when the luminance function increases or decreases by a significant amount. In other words, the function that defines the event generation $e(\boldsymbol{p}_k; t_k) = e_k$, generates 1 if the log-luminance level changed more than a threshold ϑ, i.e. an ON event, and -1 if it changed more than $-\vartheta$, i.e. an OFF event. This sampling of the lightfield essentially represents the temporal derivative of the luminance function g, i.e.

$$\frac{d}{dt} g(\boldsymbol{p}; t) = g_t(\boldsymbol{p}; t) \approx \frac{\vartheta}{\Delta t} \sum_{k:\, t_k \in (t-\Delta t, t]} e_k \,, \tag{2}$$

with ϑ denoting the sensitivity threshold of the DVS.

Below, we will recapitulate the definition of spatio-temporal filters that we have defined as a model of the initial cortical input filtering at the stage of area V1 [7]. Since this filtering stage defines the initial stage of the canonical circuit architecture, we briefly summarize its definition and its key functionality further down below.

2.2 Canonical neural circuit model

Based on a vast number of investigations we have defined a mesoscopic level description of a cortical column that serves as a core computational unit (see [17, 21, 30, 29] for overviews on different details about structural and functional principles of cortical processing). The proposed architecture as described in [4], and a further detailed investigation in [5], has been demonstrated to account for a multitude of experimental findings. Also, detailed mathematical investigations of the dynamic properties (based in a reduced dimensional model definition) have revealed stable regimes of the columnar circuit as well as conditions in which inhibition stabilization properties or qualitative changes in the overall dynamics may occur (bifurcations; for details, see [4]).

The simplified dynamics of the canonical circuit are given by

$$\begin{aligned} \tau \dot{r}_{i\theta} &= -\alpha r_{i\theta} + (\beta - r_{i\theta}) g^{ex}_{i\theta} - (\delta + r_{i\theta}) I^{in}_{i\theta} - \gamma \cdot r_{i\theta} \cdot g_p(p_i) \,, \\ \tau_p \dot{p}_i &= -\alpha_p p_i + \beta_p \cdot \left\{ \sum_{k,\theta} g_r(r_{k,\theta}) \cdot \Lambda^-_{i,k,\theta} \right\} + (I_c)_i \,, \end{aligned} \tag{3}$$

with excitatory net input defined by

$$g_{i\theta}^{ex} = \left(I_{i\theta}^{ex} + \gamma_{lat} \cdot \left\{ \sum_k g_r(r_k) \cdot \Lambda_{i,k,\theta}^{+} \right\}\right) \left(1 + \lambda \cdot net_{i\theta}^{FB}\right), \tag{4}$$

where $g_{p,r}(\cdot)$ denote gain functions that map the potentials $r_{i\theta}$ and p_i, respectively, to output firing-rates. The constants τ, $\tau_p > 0$ define the membrane constants for an excitatory unit and the inhibitory pool, respectively. $\Lambda^{+/-}$ denote the input weights to define the lateral interaction and the extent of the pool (e.g. with Gaussian weights). An additional input I_c for the pool inhibition can account for tonically active pool neurons and additional surround effects. The parameters $\alpha, \alpha_p, \beta, \delta, \beta_p, \eta, \eta_p, \gamma, \lambda \geq 0$ are constants to denote the activity decay, the excitatory and inhibitory saturation levels, the relative strength of the pool inhibition and the feedback strength.[1]

The net input defined by $g_{i\theta}^{ex}$ summarizes the feedforward signal generated by the stage of input filtering, denoted by $I_{i\theta}^{ex}$ and the lateral intra-cortical integration of linked activations, as denoted by $\sum_k g_r(r_k) \cdot \Lambda_{i,k,\theta}^{+}$. Such activities are modulated by re-entrant signals $net_{i\theta}^{FB} \geq 0$, calculated in different areas, and that generate the signal enhancement in case of combined correlated feedforward and reentrant signals. Here, we consider such reentrances generated by feedback from areas at higher stages in the hierarchy of processing stages. Since such coincident feedforward and reentrant feedback signal activation can only lead to increases in the overall columnar activation, the pool normalization leads to a down-modulation of those activities which have not received any reentrant up-modulation.

The circuit model incorporates several canonical operations which implement the key arithmetic operations to generate the first-order changes in the potential, or activity state. Regarding a mapping of the circuit dynamics onto neuromorphic hardware the different arithmetic operations need to be carefully considered. Briefly, the operations used in eqn. (3) are summation, subtraction, and multiplication. The firing-rate functions can, for example, be implemented as rectifications. In the calculation of weighted sums, task-specific filtering function are utilized. In addition, activities are multiplied by terms that account for the membrane-specific mass-action [16] as well as the reentrant modulation that is generated by feedback signals. Such a multiplicative enhancement needs to be generated as well by a neuromorphic circuit that implements the core operations. These issues will be investigated in the following.

2.3 Stage of input filtering

We will now recapitulate the definition of the spatio-temporal filters [7] that are fitted to the physiological findings from [13]. We do so since the initial filtering stage of the motion detection model utilizes several important observations from experimental investigations [13]. These observations lead to distinct design rules which, in turn, are important when we map spatio-temporal filters for initial motion detection onto neuromorphic hardware.

The filter proposed in [7] essentially reverses the decomposition of neural responses conducted by [13] (also c.f. [32]). Based on physiological findings first described by DeAngelis [14], De Valois suggested that inseparable filters stem from a combination of various separable components [13]. In [13] cortical V1 cells were tested and strong evidence for the coexistence of two distinct types of populations of cells emerged: One population showed spatio-temporally separable weight functions of either even or odd spatial symmetry. These have either temporally mono- or bi-phasic response characteristics which were mainly determined by a single principal component in 2D (of a singular value decomposition). The other population of cells was spatio-temporally inseparable showing a receptive field distribution of selectivity that were slanted with respect to the time axis, i.e. motion sensitive (c.f. Fig. 2, right; c.f. also [12]). Response characteristics of these cells were determined by *two* strong principal components in 2D. These two components of the second group were itself spatio-temporally separable with spatially out-of-phase components and always composed of pairs of mono- and bi-phasic distributions. This main observation lead us to propose a family of spatio-temporally direction selective filters as illustrated in Fig. 2, that are generated by superposed separable filters with quadrature pairs of spatial weighting profiles (G_{odd} and G_{even}) and mono-/bi-phasic temporal profiles (T_{mono} and T_{bi}). The details of the construction process are outlined in the following sections.

2.3.1 Separation of spatial Gabor filters and mono- and bi-phasic temporal filters

To construct the *spatial* component of the spatio-temporal filters illustrated in Fig. 2 we define Gabor filters that are fitted to the experimental results of [13]. To construct multiple spatio-temporally tuned filters of different spatial orientation selectivity, we employ a filter-bank of kernels as illustrated in [7], Fig. 5. More precisely, we employ Gabor filters maximally selective for the spatial frequency (f_x^0, f_y^0) (with a standard deviation σ in local space) defined by

$$G(x,y) = \frac{2\pi}{\sigma^2} \exp\left[2\pi j \left(f_x^0 x + f_y^0 y\right)\right] \exp\left[\frac{-2\pi^2\left(x^2+y^2\right)}{\sigma^2}\right], \tag{5}$$

in local space. This defines the two components $G_{odd} = \Im(G_{\sigma,f_x^0,f_y^0})$ and $G_{even} = \Re(G_{\sigma,f_x^0,f_y^0})$ to construct the spatial filters shown on top of Fig. 2 (compare with e.g. [22]).

The second component required in the spatio-temporal filter generation process illustrated in Fig. 2 is the definition of the *mono- and bi-phasic temporal* filters, T_{mono} and T_{bi}. To fit the experimental data of [13], we define (c.f. Fig. 2)

$$T_{mono}(t) = G_{\sigma_{mono},\mu_{mono}}(t), \tag{6}$$

$$T_{bi}(t) = -s_1 \cdot G_{\sigma_{bi1},\mu_{bi1}}(t) + s_2 \cdot G_{\sigma_{bi2},\mu_{bi2}}(t), \tag{7}$$

with the unnormalized Gaussian function

$$G_{\sigma,\mu}(t) = \exp\left(-\frac{(t-\mu)^2}{2\sigma^2}\right). \tag{8}$$

[1] Note, that here we simplified the definition of the action of the inhibitory pool regarding the definition given in [4] by omitting the subtractive inhibition component in the center-surround pool interaction.

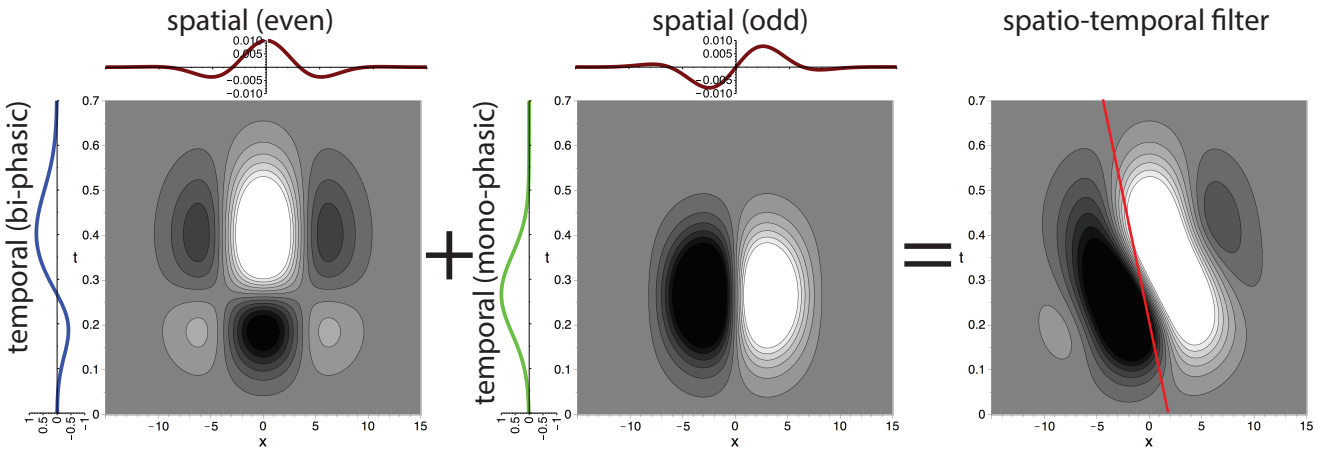

Figure 2: Superposition of spatio-temporally separable filters creates motion-direction sensitive filter. The spatio-temporal filter proposed in [7] (right) is constructed by using the results of the singular value decomposition of the receptive field of motion directional cells by [13] (c.f. [14], their Fig. 3). The superposition of two separable filters (pairs of a bi-phasic temporal and an even spatial or a mono-phasic temporal and an odd spatial filter; line profile plots) creates the spatio-temporal selective filter. The red line indicates the preferred speed selectivity identified by a Fourier analysis of the filter function (reproduced from [7]).

When the experimental findings are incorporated, it is only necessary to choose a value for μ_{bi1}. All other parameters can be inferred according to the experimental data from [13]:

- The bi-phasic scaling factors s_1 and s_2 are adapted to the minimum and maximum values of the experimental data relative to the maximum value of the monophasic kernel (which is one), i.e. $s_1 = 1/2$ and $s_2 = 3/4$.
- A good fit with the experimental data reported in [13] is achieved by setting the relation between the mean values to $\mu_{bi2} = 2\mu_{bi1}$.
- The standard deviations σ_{mono} and σ_{bi1} are chosen such that the Gaussians are almost zero for $t = 0$, i.e. $\sigma_{mono} = \mu_{mono}/3$, $\sigma_{bi1} = \mu_{bi1}/3$ (3σ–rule; 99.7% of the values lie within three standard deviations of the mean in a normal distribution).
- The standard deviation of the second Gaussian of the bi-phasic kernel is about 3/2 of that of the first, i.e. $\sigma_{bi2} = \frac{3}{2} \cdot \sigma_{bi1} = \frac{1}{2} \cdot \mu_{bi1}$.
- The mean of the mono-phasic kernel μ_{mono} is given by the zero-crossing of the biphasic kernel, i.e. $\mu_{mono} = \frac{1}{5} \cdot \mu_{bi1} \cdot \left(1 + \sqrt{36 + 10 \cdot \ln(s_1/s_2)}\right)$.

Figure 6 of [7] illustrates that these settings result in a good fit of the temporal filters with the experimental data reported in [13]. Based on these definitions of separated one-dimensional filters we will now construct the full spatio-temporal selective filters as outlined in Fig. 2.

2.3.2 *Combined spatio-temporal filters*

The inseparable spatio-temporal filter F is defined in accordance to the scheme of Fig. 2, i.e. by the sum of two products consisting of the odd spatial $G_{odd} = \Im(G_{\sigma,f_x^0,f_y^0})$, the monophasic temporal T_{mono}, the even spatial $G_{even} = \Re(G_{\sigma,f_x^0,f_y^0})$, and the biphasic temporal filter T_{bi} (c.f. Fig. 2):

$$F(x,y,t) = \Im(G_{\sigma,f_x^0,f_y^0}(x,y)) \cdot T_{mono}(t) + \Re(G_{\sigma,f_x^0,f_y^0}(x,y)) \cdot T_{bi}(t)\,. \quad (9)$$

The preferred speed of the filter can be determined by an analysis of the Fourier transform $\hat{F}(f_x, f_y, f_t)$ of the filter function $F(x,y,t)$. From the location $(f_t^{max}, f_x^{max}, f_y^{max})$ where $\hat{F}$ is maximal one can infer the filter's preferred normal velocity, i.e. the velocity parallel to the gradient of the luminance edge

$$u_\perp = -\frac{f_t^{max}}{(f_x^{max})^2 + (f_y^{max})^2} \cdot f_x^{max}\,, \quad (10)$$

$$v_\perp = -\frac{f_t^{max}}{(f_x^{max})^2 + (f_y^{max})^2} \cdot f_y^{max}\,, \quad (11)$$

in $pixel/s$ (c.f. [7]; eqns. (26–27)). For the parameter values that fit the experimental data from [13], i.e. $\sigma = 25$, $f_x^0 = f_y^0 \approx 0.057$ ($f_0 = \sqrt{(f_x^0)^2 + (f_y^0)^2} = 0.08$) and $\mu_{bi1} = 0.2$, we numerically determined the values as $f_t^{max} = 0.974$, $f_x^{max} = f_y^{max} = 0.057$ which maximize $|\hat{F}|$. Thus, the fitted spatio-temporal selective filter F is maximally selective for the velocity $(u_\perp, v_\perp) = (8.61, 8.61) pixel/s$, i.e. a speed of $12.2 pixel/s$. See Table 1 of [7] for possible parameterizations.

3. NEUROMORPHIC COMPUTING

Inspired by the brain's function and efficiency, IBM recently introduced a novel computing chip which emulates a million simple, digital, reconfigurable, versatile spiking neurons which is one result of the DARPA SyNAPSE initiative that aims at developing neuromorphic computing technology that scales to form and function similar as mammalian brains [11, 24]. IBM's approach of investigating a digital brain-like chip architecture utilizing event-based processing capabilities lead to the development of the TrueNorth chip and its Compass

software simulation environment. The chip architecture is highly efficient, operates in real time and can natively interact with spike based sensors like the DVS described in the introduction. Interaction is organized in short-range connections via an intra-core crossbar memory and long-range connections through an inter-core spike-based message-passing network (c.f. Fig. 3). Since the architectural design of the TrueNorth chip matches key elements of the proposed columnar circuit model outlined above, we suggest to consider how a specific target functionality might be mapped onto such a neuromorphic scheme. We therefore asked whether the proposed motion detector can be implemented on such hardware and how such a mapping can lead to improvements in efficiency for the basic calculations.

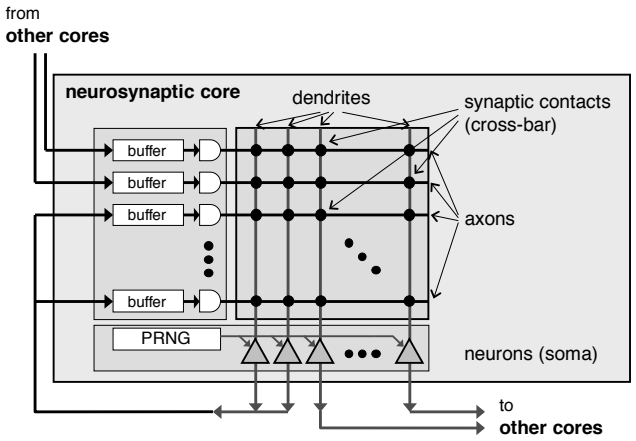

Figure 3: TrueNorth is a chip inspired by the brain and built from an interconnected network of lightweight neurosynaptic cores (one shown) operating in discrete time steps [25, 27, 11]. An intra-core crossbar memory realizes short-range connections (shown) while long-range connections are realized through an inter-core spike-based message-passing network (from/to other cores). The neurosynaptic core has 256 axons, a 256×256 binary crossbar, and 256 neurons. In a time step, if the synapse value for a particular axon-neuron pair is non-zero and the axon is active, then the neuron updates its state by the synaptic weight. Next, each neuron applies a leak, and fires a spike if its state exceeds its threshold. A PRNG (pseudorandom number generator) can add noise to the spike thresholds and stochastically gate synaptic and leak updates. The buffers hold incoming spikes for delayed delivery (redrawn from [11]).

The model described in [24] (without stochastic binary synapses, without leak reversal, without stochastic leak, with fixed reset value, and fixed negative reset $-\delta$) reads

$$V_j(t) \mapsto \max\Big(-\delta, -\alpha_j + V_j(t-1) + \underbrace{\sum_{i=0}^{255} A_i(t)\omega_{ij}s_j^{G_i}}_{=:I_j(t)}\Big)\,, \tag{12}$$

$$\text{if } V_j(t) \geq \theta_j + \eta_j, \text{ spike } \& \; V_j(t) \mapsto R_j\,, \tag{13}$$

with membrane potential V_j, input spikes on the i^{th} axon A_i, synapse between i^{th} and j^{th} unit ω_{ij} (binary), synaptic weight $s_j^{G_i}$, input I_j, threshold θ_j and η denoting a uniformly drawn random number from $[0, M_j - 1]$, with M_j determining the range [11].

3.1 Motion detector on neuromorphic architecture

The construction of the proposed motion sensitive filter illustrated in Fig. 2 enables a separable filter design of the two subcomponents that are added to gain the spatio-temporal filter. Thus, it is possible to first realize a temporal followed by a spatial filter stage. The spatial Gabor filter stage can be realized by weighted summation according to the filter weights $\omega_{ij} \cdot s_j^{G_i}$ stored in the synaptic crossbar (composed of the binary crossbar value ω_{ij} and the synaptic weight $s_j^{G_i}$). The temporal filtering in turn can be realized by *delayed* spike propagation on a neuromorphic computing chip. In a nutshell, every spike emitted from an asynchronous event based sensor is multiplexed to multiple neurons with delayed axon lines that in turn target the temporal filter unit (c.f. Fig. 4). The resulting activity represents the temporally

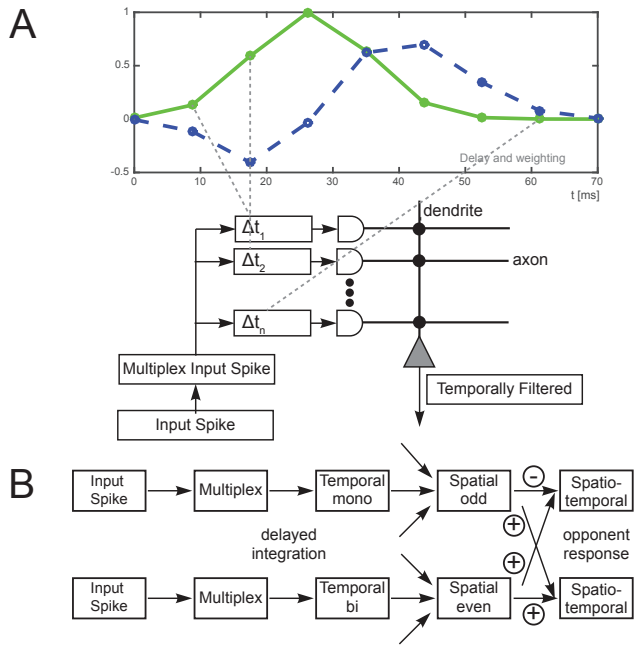

Figure 4: A, Realization of temporal filters on neuromorphic hardware at a *single* location. An input spike is multiplexed to multiple neurons (not shown) with delayed weighted axon-lines (Δt_i). Temporal filtering is realized by a weighted summation at the neuron's dendrite. The result is delivered to the spatial filtering stage using a rate code. B, Entire processing hierarchy for one orientation.

filtered response to the input spikes (note that the response to multiple spikes at a single location are integrated by the neuron's dendrite). These responses have to be determined at all locations and serve as input to another population of neurons spatially integrating these temporally filtered responses according to the scheme shown in Fig. 2. Thus, to realize a

temporal filter selective for a fixed speed, there have to be two populations of temporally selective neurons (one for the mono- and one for the bi-phasic response) each followed by a population of oriented spatial filters (odd and even) which are combined afterwards into a spatio-temporally selective population of units.

3.2 Canonical neural circuit implementation on neuromorphic architecture

As outlined above the mesoscopic level definition of the canonical circuit of an abstract cortical column serves as a framework for filtering, the description of intra-cortical interactions as well as the modulatory reentrant interaction of feedback signals. In the specific context of event-based optical flow we have demonstrated that feedback and normalization can help to disambiguate responses [3, 32, 7]. Here, we demonstrate that such interaction can be implemented on neuromorphic hardware.

Weighted sums can be directly implemented in the model using eqn. (12) (with restrictions regarding the number of weights). There is, however, the requirement to realize products in order to realize the canonical neural circuit model (3) (e.g. the product of $r_{i\theta} \cdot g_p(p_i)$ to realize a shunting pool inhibition). To realize a unit calculating the product of two input signals (also c.f. [11], their Fig. 8) one can use the relation of the joint probability of two events A and B

$$P(A \cap B) = P(A) \cdot P(B)\,. \tag{14}$$

Let us consider a unit for which only coincident spikes can evoke a spike. The probability of coincident spikes in turn is given by $P(A \cap B) = P(A) \cdot P(B)$. Thus, the mean spike rate of the unit will be the product of the mean spike rates of its inputs. A unit which spikes whenever coincident spikes arrive at its dendrite can be constructed as follows (considering an operating range with an upper bound u_b and discrete time steps): The lower bound $-\delta$ and the reset potential R are set to 0, the leakage term α is set to u_b, the threshold θ to some strictly positive value smaller than u_B, the noise range to some arbitrary non-negative value (e.g. $M_j = 0$), and the input connection weights are set to u_b. For products of n variables the leakage must be set to $(n-1) \cdot u_b$.

Thus, such product units can be used to calculate each of the products of the dynamic variables like $r_{i\theta} \cdot g_p(p_i)$ such that the processing steps of feedback and normalization shown in [32, 7] can also be implemented on neuromorphic hardware.

3.3 Experimental results and cores estimate

While being energy efficient, neuromorphic hardware comes with certain restrictions like a limited amount of different weights per neuron. Here, we try to give an estimate on the number of cores/neurons required in the case of TrueNorth. At the time of writing, we did not have access to TrueNorth or its simulator, thus our estimate must be considered as preliminary. A principal question is the choice of signal encoding. In general, the value of a weighted sum cannot be transmitted in a single spike. Thus, we employ rate codes in this investigation. Due to the redundancy in the input and the temporal oversampling (typical filters operate within the range of about 70ms whereas a TrueNorth tick requires 1ms), we expect that a leaking stochastic spiking neuron should result in a good approximation of an optimal rate code, something that needs further investigation for a particular hardware realization like TrueNorth.

Our investigations show that a sampling of the temporal function with only 3 units already provides a good approximation (well within the weight constraints of TrueNorth; c.f. Fig. 5). Thus, the mono-/bi-phasic filtering stages each require 1 axon (connected to the input, that provides a spike if at least one spike occurred within the last tick; should be implemented in the sensor) and 3 neurons to multiplex/delay the signal[2], 3 axons and 3 neurons to delay the signal, and 3 axons and 1 neuron to perform the filtering. Thus, the temporal filter stage at each location requires a total of 7 axons and 7 neurons. Consequently, each TrueNorth core could calculate the temporal filter responses for 36 locations for the mono- or bi-phasic filter. Thus, the temporal filtering stage requires a total of $\lceil 128^2/36 \rceil \cdot 2 = 912$ TrueNorth cores.

The output of one of the temporal filters feeds into the spatial filter stage. We assume a separable filter layout that can be realized for horizontally or vertically oriented Gabor filters. For the DVS128, each TrueNorth core can realize the spatial filtering for an entire row (TrueNorth's crossbar has 256 axons and neurons, enough for 128 required for the first separable filtering and 128 for the second; assuming that the filter mask can be realized using 4 different values which was shown to reasonably approximate the true Gabor filter). Thus, the spatial filtering stage requires a total of $2 \cdot 128 = 256$ TrueNorth cores to process the output of the DVS128.

The final stage requires one neuron per opponent response, thus a total of $128^2 \cdot 2/256 = 128$ cores is required. Consequently, the entire hierarchy as illustrated in Fig. 4B requires $912 + 256 + 128 = 1296$ TrueNorth cores for a single orientation. Processing two orientations only requires duplicating the last two stages and splitting or replicating the neurons of the temporal filtering stage such that a single TrueNorth chip with 4096 cores should be capable to calculate the optical flow in real time. It needs, however, to be investigated, whether the leakage can be adjusted to sufficiently approximate the rate code required for the computations.

For the sake of simplicity, we only employed a feedforward scheme of motion detection at the moment to demonstrate the mapping of the filtering stage onto the neuromorphic architecture. We have already demonstrated that reentrant modulating feedback signals generated by the model area MT (delivering signals back to model area V1) significantly improve the event-based signal representation [32].

4. CONCLUSIONS

This contribution investigates the mapping of an event-based motion estimation algorithm as part of a canonical neural circuit model onto neuromorphic hardware. The dynamics of this circuit consider the compartment structure of cortical columns with its input as well as the supra- and infra-granular levels [29]. At the functional level, the circuit model suggests

[2]On TrueNorth delays are limited to at most 15 ticks [10] which does not increase the number of neurons required as long as the delay from one to the next is smaller than 15ms since we can create a chain of delayed neurons.

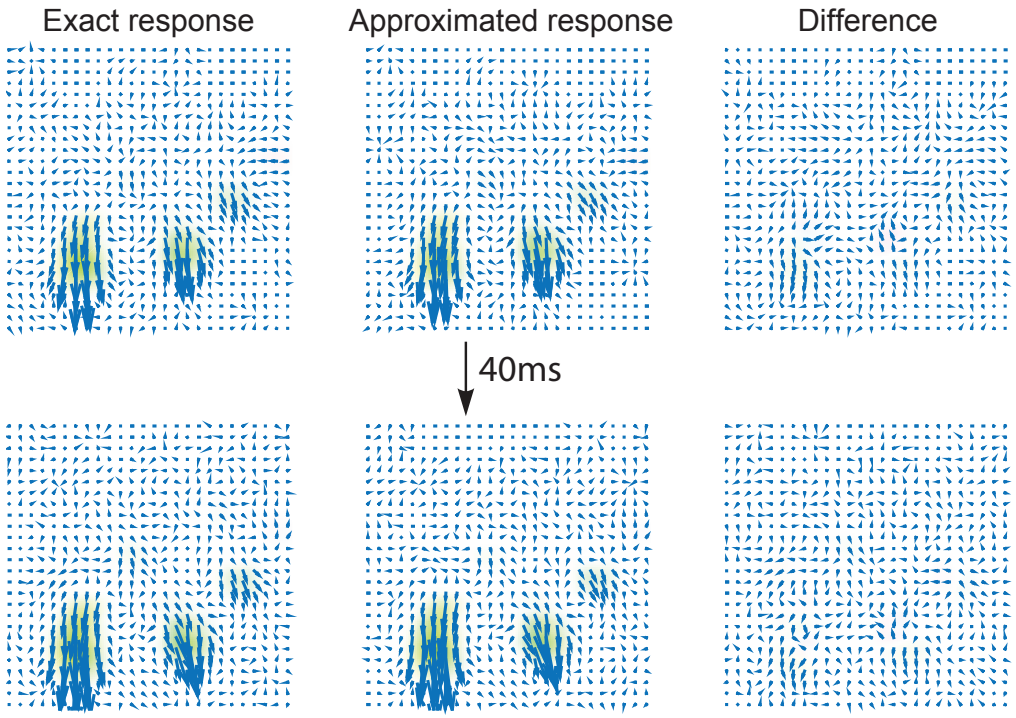

Figure 5: Sampling rate of temporal functions can be as low as 3 sampling points to get a good approximation. Comparison of motion fields of a scene of cars passing under the bridge over the 210 freeway in Pasadena [20]. Optical flow has been determined for horizontal and vertical orientation.

how signals generated by the cortical filtering, modulatory reentry or feedback, and center-surround competition for pool normalization are combined to define a basic computational building block.

The key contribution of this work is in the mapping of the computational circuit mechanisms and the motion detection stage onto neuromorphic architecture. Here, we focused on one such model framework that has recently been developed and made available to a broader scientific community, namely the TrueNorth chip architecture of IBM [24]. Its computational principles are designed in the spirit of event-based processing such that the frameless sensing using a DVS neuromorphic retina can be neatlessly integrated into the architecture. We have reviewed our proposal of a common framework of event-based motion detection and its integration as part of the canonical circuit model of a cortical column. The structure of the filtering allows to define a space-time inseparable motion selective filter mechanism on the basis of the superposition of space-time separable filters using different modulation functions. Our detailed core estimate indicates that this architecture should enable real-time optical flow computation using a DVS128 and a single TrueNorth chip.

Based on the decomposition of the circuit dynamics and the arithmetic operations being involved in their calculation, we derived a specification of the operations that need to be mapped onto the neuromorphic chip architecture. The filtering, particularly the extended temporal filtering, as well as the multiplicative combination of different signal streams define the challenges for such a mapped computational algorithm that is executed on the neuromorphic platform. In a nutshell, the temporal filtering can be realized by weighted delays of local connections on the neurosynaptic core. The modulatory signal interaction for feedforward and reentrant feedback interaction can be realized by an approximation via the coincident spike generation in a population of cells. Numerical simulations of a flow estimation task for estimating the movement of cars in a traffic scenario demonstrate that the approximate solutions of simulations using the mapped algorithm leads to robust and almost indistinguishable results in comparison to the original continuous simulation.

5. ACKNOWLEDGMENTS

This work has been supported by grants from the Transregional Collaborative Research Center "A Companion-Technology for Cognitive Technical Systems" SFB/TRR62 funded by the German Research Foundation (DFG) and BMBF project "SenseEmotion".

6. REFERENCES

[1] P. Bayerl and H. Neumann. Disambiguating Visual Motion Through Contextual Feedback Modulation. *Neural Computation*, 16(10):2041–66, 2004.

[2] W. H. Bosking, Y. Zhang, B. Schofield, and D. Fitzpatrick. Orientation Selectivity and the Arrangement of Horizontal Connections in Tree Shrew Striate Cortex. *Journal of Neuroscience*, 17(6):2112–27, 1997.

[3] T. Brosch and H. Neumann. The Brain's Sequential

Parallelism: Perceptual Decision–Making and Early Sensory Responses. In *ICONIP (Part II)*, volume 7664 of *LNCS*, pages 41–50, 2012.

[4] T. Brosch and H. Neumann. Computing with a Canonical Neural Circuits Model with Pool Normalization and Modulating Feedback. *Neural Computation*, 26(12):2735–89, 2014.

[5] T. Brosch and H. Neumann. Interaction of Feedforward and Feedback Streams in Visual Cortex in a Firing–Rate Model of Columnar Computations. *Neural Networks*, 54:11–6, 2014.

[6] T. Brosch, H. Neumann, and P. R. Roelfsema. Reinforcement Learning of Linking and Tracing Contours in Recurrent Neural Networks. *PLoS Computational Biology*, 11(10):e1004489, 2015.

[7] T. Brosch, S. Tschechne, and H. Neumann. On Event–Based Optical Flow Detection. *Frontiers in Neuroscience*, 9(137):1–15, 2015.

[8] M. Carandini. From Circuits to Behavior: A Bridge too Far? *Nature Neuroscience*, 15(4):507–9, 2012.

[9] M. Carandini, D. J. Heeger, and J. A. Movshon. Linearity and Normalization in Simple Cells of the Macaque Primary Visual Cortex. *Journal of Neuroscience*, 17(21):8621–44, 1997.

[10] A. S. Cassidy, R. Alvarez-Icaza, F. Akopyan, et al. Real–time Scalable Cortical Computing at 46 Giga–Synaptic OPS/Watt with $\sim 100\times$ Speedup in Time–to–Solution and $\sim 100,000\times$ Reduction in Energy–to–Solution. In *High Performance Computing, Networking, Storage and Analysis, SC14: International Conference for*, pages 27–38. IEEE, 2014.

[11] A. S. Cassidy, P. Merolla, J. V. Arthur, et al. Cognitive Computing Building Block: A Versatile and Efficient Digital Neuron Model for Neurosynaptic Cores. In *IJCNN*, pages 1–10, 2013.

[12] R. L. De Valois and N. P. Cottaris. Inputs to Directionally Selective Simple Cells in Macaque Striate Cortex. *PNAS*, 95(24):14488–93, 1998.

[13] R. L. De Valois, N. P. Cottaris, L. E. Mahon, S. D. Elfar, and J. A. Wilson. Spatial and Temporal Receptive Fields of Geniculate and Cortical Cells and Directional Selectivity. *Vision Research*, 40(27):3685–702, 2000.

[14] G. C. DeAngelis, I. Ohzawa, and R. D. Freeman. Receptive–Field Dynamics in the Central Visual Pathways. *TINS*, 18(10):451–8, 1995.

[15] T. Delbrück and S.-C. Liu. A Silicon Early Visual System as a Model Animal. *Vision Research*, 44(17):2083–9, 2004.

[16] S. Grossberg. Contour Enhancement, Short–Term Memory, and Constancies in Reverberating Neural Networks. *Studies in Applied Mathematics*, 52:213–57, 1973.

[17] M. Larkum. A Cellular Mechanism for Cortical Associations: An Organizing Principle for the Cerebral Cortex. *Trends in Neurosciences*, 36(3):141–51, 2013.

[18] M. E. Larkum. The Yin and Yang of Cortical Layer 1. *Nature Neuroscience*, 16(2):114–5, 2013.

[19] S.-C. Liu and T. Delbruck. Neuromorphic Sensory Systems. *Current Opinion in Neurobiology*, 20(3):288–95, 2010.

[20] L. Longinotti. Cars Passing Under Bridge Over the 210 Freeway in Pasadena, 2015. URL: `http://www.ini.uzh.ch/~tobi/dvs/events20051221T014416%20freeway.mat.dat`, Accessed: June 23, 2015.

[21] H. Markram, M. Toledo-Rodriguez, Y. Wang, et al. Interneurons of the Neocortical Inhibitory System. *Nature Reviews Neuroscience*, 5(10):793–807, 2004.

[22] S. Marčelja. Mathematical Description of the Responses of Simple Cortical Cells. *Journal of the Optical Society of America*, 70(11):1297–1300, 1980.

[23] C. Mead. Neuromorphic Electronic Systems. *Proceedings of the IEEE*, 78(10):1629–36, 1990.

[24] P. A. Merolla, J. V. Arthur, R. Alvarez-Icaza, et al. A Million Spiking–Neuron Integrated Circuit with a Scalable Communication Network and Interface. *Science*, 345(6197):668–73, 2014.

[25] D. S. Modha, R. Ananthanarayanan, S. K. Esser, et al. Cognitive Computing. *Communications of the ACM*, 54(8):62–71, 2011.

[26] W. A. Phillips, A. Clark, and S. M. Silverstein. On the Functions, Mechanisms, and Malfunctions of Intracortial Contextual Modulation. *Neuroscience and Biobehavioral Reviews*, 52:1–20, 2015.

[27] R. Preissl, T. M. Wong, P. Datta, et al. Compass: A Scalable Simulator for an Architecture for Cognitive Computing. In *High Performance Computing, Networking, Storage and Analysis (SC), 2012 International Conference for*, pages 1–12. IEEE, 2012.

[28] F. Raudies, E. Mingolla, and H. Neumann. A Model of Motion Transparency Processing with Local Center–Surround Interactions and Feedback. *Neural Computation*, 23:2868–914, 2011.

[29] M. W. Self, R. N. Kooijmans, H. Supèr, V. A. Lamme, and P. R. Roelfsema. Different Glutamate Receptors Convey Feedforward and Recurrent Processing in Macaque V1. *PNAS*, 109(27):11031–6, 2012.

[30] S. M. Sherman and R. W. Guillery. On the Actions that One Nerve Cell can Have on Another: Distinguishing "Drivers" from "Modulators". *PNAS*, 95(12):7121–6, 1998.

[31] A. Thielscher and H. Neumann. Neural Mechanisms of Cortico–Cortical Interaction in Texture Boundary Detection: A Modeling Approach. *Neuroscience*, 122:921–939, 2003.

[32] S. Tschechne, T. Brosch, R. Sailer, et al. On Event–Based Motion Detection and Integration. In *8th International Conference on Bio–inspired Information and Communications Technologies*, pages 298–305. ACM, 2014.

[33] S. Tschechne and H. Neumann. Hierarchical Representation of Shapes in Visual Cortex–from Localized Features to Figural Shape Segregation. *Frontiers in Computational Neuroscience*, 8(93):1–20, 2014.