Available online at www.sciencedirect.com
SCIENCE DIRECT®
Neurocomputing 64 (2005) 301–317
NEUROCOMPUTING
www.elsevier.com/locate/neucom

A neural architecture for the symmetric-axis transform

C. Rasche[a,b,*]

[a]California Institute of Technology, Division of Biology, Pasadena, CA, USA
[b]Department of Psychology, University of California Santa Barbara, Santa Barbara, CA, USA

Received 9 January 2004
Available online 19 January 2005

Abstract

The symmetric-axis transform is a process that dynamically encodes the space of a visual shape through self-interaction of its contours. It is generally simulated using computer algorithms. A neural architecture of this transformation is presented that is conceptually simple enough for a hardware implementation. Its architecture consists of a wave-propagating map, orientation-selective columns detecting wave pieces of specific orientation, and a coincidence map detecting the clash of two wave fronts. We illustrate its operation on partial contours extracted from gray-scale images.

Keywords: Symmetric-axis transform; Wave propagation; Orientation columns; Neural architecture

1. Introduction

Visual shape is generally described by its contours (or boundaries), for example a rectangle is expressed by its four connected L features [26]. Underlying this idea is the mathematics of Euclidean geometry, in which shape is described by symbols and numbers, and which is also the prevailing view of dealing with shapes in mathematics

*Corresponding author. Dept. of Psychology, The Pennsylvania State University, 417 Moore Building, University Park, PA 16802, USA. Tel.: +1 814 865 1930; fax: +1 814 867 1107.
E-mail address: cur12@psu.edu (C. Rasche).

0925-2312/$ - see front matter © 2004 Elsevier B.V. All rights reserved.
doi:10.1016/j.neucom.2004.11.024

[3]. There exist a few alternative mathematical concepts that attempt to encode not only the contours of an object but also its space (or region) [14]. Blum has proposed such a space geometry, the symmetric-axis transform (SAT), which he specifically designed to perform as a biologically plausible process [3]. Its idea is simple: a shape is dipped into an excitable neuronal map, igniting a 'grass-fire' process. Wherever these grass-fire fronts meet, they cancel each other out, evolving a geometrical pattern in time called the symmetric axis (sym-ax). The sym-ax defines the contour geometry of the shape and its including space completely. Fig. 1 shows the SAT for a rectangle. The sym-ax consists of the loci of symmetric points (sym-points) which represent discrete coincidences during the evolvement of the sym-ax. The advantage of encoding space using this transform is that two-dimensional space is encoded into vectors, sym-ax pieces, of one spatial dimension, which is more convenient for object representation than using the entire region. Numerous algorithms have been developed that evolve the SAT using different techniques (see [25] for a review) and have been applied to problems like general shape encoding [21,27], letter recognition [15], medical image analysis [8] and motion coding [17], to mention only a few examples. Many of these algorithms are concerned with an exact reconstruction of the shape and often work on shapes that possess or even require closed and complete contours. Some of them iterate until the exact sym-ax has been evolved. We here report about a simulation of the SAT using a neural architecture that is conceptually simple enough for an approach in analog hardware for example. For the creation of the present architecture, two design goals were regarded as important. (1) The system should generate the sym-ax from a contour picture that is *not* necessarily complete. (2) The system should employ simplest types of neural units, which have the potential for an implementation into analog Very-Large-Scale-Integrated (VLSI) circuits. We demonstrate the operation of our neural SAT on simple shapes as well as on complex shapes obtained from objects depicted in photos. For a broader motivation of the purpose of this study, the reader may refer to [30].

Blum originally envisioned that a grass-fire process can be implemented by a plane with the following properties for each point: '(1) it has value either one or zero (all or none excitation); (2) it may be excited by an adjacent point or from an external

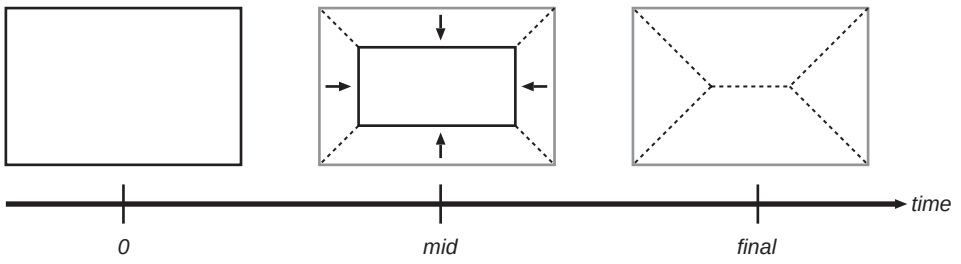

Fig. 1. The SAT illustrated on the rectangle. The shape is dipped into the excitable map at time $t = 0$. $t =$ mid shows the waves (solid lines) at a progressed state. The four dotted lines represent the loci of symmetric points (sym-points) that continuously appear by the coincidence of two merging waves. At the conclusion of the process ($t =$ final) the shape has been completely transformed into its sym-ax.

source; (3) an excited point cannot be re-excited for some interval of time (refractory property)'. The list of properties very much implies that the grass-fire process is carried out by a sheet of neurons. Yet, there is a difficulty with implementing this grass-fire process: it is tedious to distinguish whether two wave fronts coincide or whether there is only a single wave traveling across a given point (neuron) in the map. We imagine that this is easiest solved by a selective mechanism, that is able to

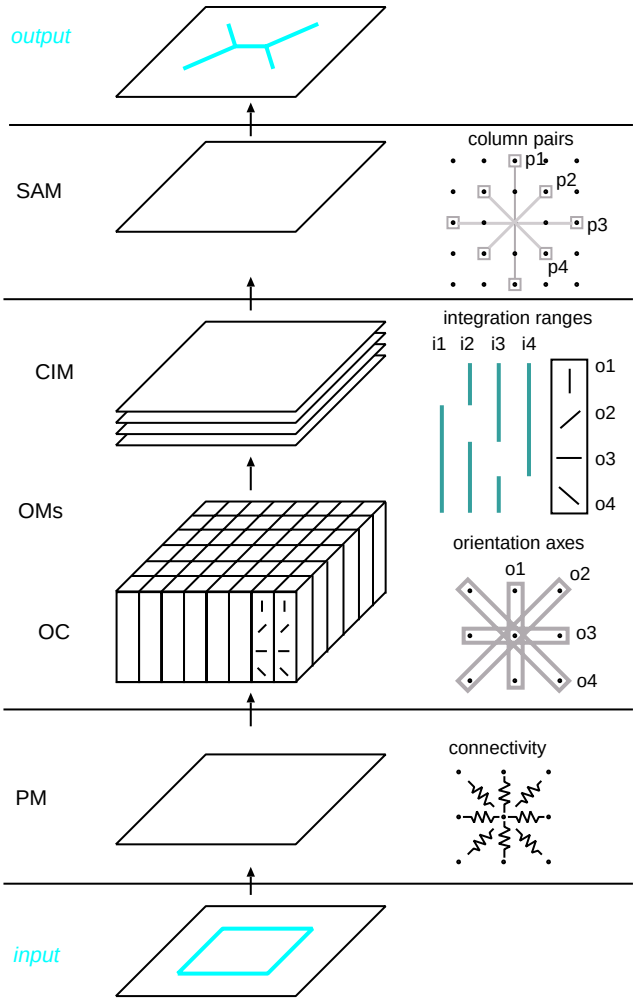

Fig. 2. Architecture of the SAT simulation. Processing occurs from bottom to top. PM: propagating map. It consists of a sheet of integrate-and-fire neurons propagating the activity waves. OMs: orientation maps. OC: orientation columns. A column possesses a set of neurons selective to different orientations of propagating waves. CIM: column integrator maps, adding input from different ranges of orientation-selective neurons. SAM: symmetric map, consisting of a sheet of coincidence detector neurons.

clearly sense when two waves clash (and to ignore when a single wave travels across the map). We have chosen orientation-selective and coincidence detector units to solve the problem. Orientation-selective units signal the orientation of a wave piece. Coincidence-detector units signal when two proximal orientation-selective units fire simultaneously, thus indicating that two wave fronts are about to clash. The units are implemented using integrate-and-fire dynamics.

Fig. 2 shows the architecture: Processing occurs from bottom to top. The shape is dipped into a propagating map (PM), an excitable map that simulates the wave propagation. Its generated spikes are fed into the cells of their spatially corresponding orientation columns (OCs). An OC consists of units sensitive to different spatial orientations for that local patch of the PM. The spikes generated in the OCs are range-wise integrated in column-integrator maps (CIM), whose activity in turn is sensed by coincidence detector neurons in a subsequent symmetric-axis map (SAM). The SAM neurons fire when two waves are about to collide and thus evolve the sym-ax. The coarse outline of this architecture and simulation results have already been reported somewhere else. Here the specifics of the simulation are discussed [29,30].

2. Methods

2.1. Propagating map (PM)

The PM is a sheet of neurons laid out in a two-dimensional array. Each PM neuron connects with its eight neighbors (see Fig. 2, PM right side, connectivity). A PM neuron is modeled as a variation of the integrate-and-fire (I&F) unit. The basic I&F unit, the perfect (or non-leaky) integrator, consists of a single capacitance integrating the charge delivered by its synaptic connections, and a fixed voltage threshold triggering a spike [16]. The I&F neuron of the PM possesses a refractory property in addition, in order to avoid bounce back of activity (property three of Blum's list). Instead of simulating the current and voltage dynamics explicitly, we merely model one variable corresponding to the membrane potential, the voltage V, of the neuron. When synaptic input drives V above the spiking threshold, V is set to a maximum value, E_{Na}—interpreted as the reversal potential for sodium—and stays up there for a short while. Then a refractory period is simulated by setting V to a minimum value, E_{K}—interpreted as the reversal potential for potassium—and stays down there for a short while. The synaptic input from a neighboring neuron is proportional to the voltage difference and can be interpreted as a resistor or conductance. In a one-dimensional array of such neurons, the wave propagation corresponds to the process of axonal spike propagation.

More formally, the neuronal voltage V_{PM} at location (x, y), at its next step, $t+1$, is given by its present potential plus the input of its neighboring neurons, $I_{\mathrm{n}}(t)$, and initial external input I_{e}, the visual shape or other stimulation:

$$V_{\mathrm{PM}}(x, y, t+1) = V_{\mathrm{PM}}(x, y, t) + I_{\mathrm{n}}(t) + I_{\mathrm{e}}. \tag{1}$$

I_n is the sum of positive membrane differences multiplied by the conductance, g_a, for each of its 8 neighboring neurons:

$$I_n(t) = \sum_{k=1}^{8} \max[g_a(V_k(t) - V(t)), 0]. \quad (2)$$

When V_{PM} exceeds the spiking threshold, V_{thres}, a spike of duration d_s is triggered, followed by the refractory period of duration d_r. The binarized spiking output of each PM neuron, $S_{PM}(x, y, t) = [1 \text{ if } V_{PM}(x, y, t) > V_{thres}, 0 \text{ otherwise}]$, is fed into the neurons of the OCs, meaning the synapses of the orientation-selective neurons only sense the presence of PM spikes.

We have chosen some parameter values in accordance to voltage values in analog VLSI circuits for reason of familiarity [9]. E_{Na} and E_K are set to 5 and 0, respectively. The spiking threshold is set to be 2.0. The simulation time step is 0.2 and can be interpreted as seconds. d_s and d_r are set to 0.6 and 1.2 s correspondingly. g_a is set to 0.11.

The connectivity of the PM—8 neighboring neurons—is somewhat coarse for some of our input shapes. Because our shape contours are generally one neuron thick, lines are aliased due to the discrete nature of the grid, which will cause little insufficiencies in our SAT. We hereafter refer to this as the resolutional insufficiency. Due to the aliasing problem, the map also propagates contours asynchronously.

2.2. Orientation maps (OMs)

The OCs can be regarded as a three-dimensional cube of neurons, with two dimensions corresponding to the spatial dimensions of the PM, and one-dimension marking the different orientations (Fig. 2, OC). A single OC contains a set of different orientation-selective neurons covering a half-circle, see Fig. 2, OC, right side, orientation axis. The example shows four orientations, $o_1 \ldots o_4$, with an orientation length (receptive field diameter) of three PM neurons. A column is sensitive to its spatially corresponding local patch in the PM. An orientation-selective neuron (or OC neuron) is modeled as a leaky (or forgetful) I&F unit [16]. A leakage current is necessary in order to avoid the summation of subsequent waves (which does not occur yet in our present simulations though). The voltage of a OC neuron at location (x, y) and orientation selectivity o is described as

$$V_{OC}(x, y, o, t + 1) = V_{OC}(x, y, o, t) + I_{ori}(t) - L - I_{inh}(t), \quad (3)$$

where L is a constant amount of leakage and $I_{inh}(t)$ is an optional inhibitory input from the orthogonal OC neurons, which helps to balance the resolutional insufficiency. $I_{ori}(t)$ is the input from PM neurons (S_{PM}) along an orientation-specific axis, three in our case, multiplied by a constant synaptic amplitude A:

$$I_{ori}(t) = \sum_{k=1}^{3} S_{PM}(i_k, j_k, t) * A, \quad (4)$$

with i_k, j_k corresponding to the indices along an orientation-specific axis, centered around the index (x, y) of the receiving OC neuron. When V_{OC} exceeds the spiking

threshold, a spike of duration d_s is triggered, followed by resetting V_{OC} to 0 (E_K). No refractory period is modeled in this unit. The voltage of OC neurons is binarized, $S_{OC} = bin(V_{OC})$, where *bin* is the threshold function as described for S_{PM} previously. S_{OC} is fed into the column integrator maps.

In our simulations, we use 12 orientations, four of which are as shown in Fig. 2, OMs, right side. The other eight are the 'aliased' orientations fitting between the shown ones. The 12 orientations cover all possible orientations in a 3×3 block. I_{inh} is chosen to be the input from the exact orthogonal neuron (S_{OC}) and its two neighboring ones (range of three) and is large enough to prevent the OC neuron from firing. It can be interpreted as shunting inhibition. The parameter values for the OC neuron are chosen such, that it fires when its input neurons show spike appearances for six time units: The synaptic amplitude, A, is set to be 0.85, the amount of leakage, L, is set to be 0.8 (in our simulations the leakage is subtracted before checking V_{OC} for its spiking threshold level). The neuron therefore behaves as a coincidence detector. All other parameter values are the same as for the PM neurons.

Neurons in the SAM should fire when two proximal waves of different orientation are about to meet. One could selectively pair close OC neurons of different orientations and read their synchronized spiking. Yet, the SAM neuron does not need to know about the exact orientation of its input. We therefore integrate the column activity over a certain range, thereby excluding a small neighborhood of orientations to avoid the signaling of equal or similar orientations. Four overlapping ranges, $i_1 \ldots i_4$, are created and added into column integrator maps (Fig. 2, CIM, right side, integration ranges). For only four orientations, the ranges would be [2 3 4], [3 4 1], [4 1 3] and [1 2 3] whereby only one orientation is excluded. The subsequent SAM neurons will then selectively combine the spiking output from these column integrator maps (CIM). A CIM neuron of layer r is modeled as a forgetful I&F unit:

$$V_{CIM}(x, y, r, t+1) = V_{CIM}(x, y, r, t) + \sum_{k=1}^{K} S_{OC}(x, y, o, t) * A - L, \tag{5}$$

where K is the range of neighboring orientations, S_{OC} is the spiking activity of the corresponding OC neuron, A is a constant synaptic amplitude and L is a constant leakage term. When V_{CIM} crosses the spiking threshold, it is set to E_{Na} for a single time step only, and is set to E_K after that.

The parameter values are chosen such that a single OC spike will raise V_{CIM} above spiking threshold: A is set to 2.01, L is set to 2.0, representing thus sharpest coincidence dynamics, as the OC neuron does. An integration range of 7 orientations (out of 12) was chosen in our simulations.

2.3. Sym-ax map (SAM)

The SAM is a two-dimensional sheet of neurons with spatial dimensions corresponding to the previous maps. Each SAM neuron receives input from four pairs of CIM neurons, $p_1 \ldots p_4$, as shown in Fig. 2, SAM right side, column pairs.

The pairs lie on an axis on opposite sides of the receiving neuron and are taken from the layer whose range center is orthogonal to the axis. For the four orientations (shown in Fig. 2, CIM, right side), the column pairs with axis number 1 would receive input from the integration range number 1, which ignores the OC orientation axis number 1.

A SAM neuron is also modeled as a leaky integrate-and-fire unit:

$$V_{\mathrm{SAM}}(x, y, t+1) = V_{\mathrm{SAM}}(x, y, t) + I_{\mathrm{p}}(t) - L - I_{\mathrm{e}}, \tag{6}$$

where L is a constant amount of leakage, I_{e} is inhibitory input that suppresses the generation of sym-points along the shape contour (the external input). That suppression is necessary because when the shape contour triggers the two waves—departing in opposite directions—it is initially 'sandwiched' by them and as a result, all the OC neurons along the contour fire because they receive full input from the contour itself and the two adjacent waves, which in turn would cause the SAM neurons along the contour to fire unnecessarily.

$I_{\mathrm{p}}(t)$ is the sum of the pair of CIM neurons:

$$I_{\mathrm{p}}(t) = [S_{\mathrm{CIM}}(i, j, r, t) + S_{\mathrm{CIM}}(m, n, r, t)] * A, \tag{7}$$

where the indices i, j, m, n correspond to the pair of OC neurons centered around the receiving SAM neuron. A is a constant synaptic amplitude. The parameter values are chosen such, that when both OC neurons fire simultaneously, the SAM neuron will fire as a result of the rapid integration of their input: A is set to be 1.05, L is the same as for the CIM neuron (2.0). Again, these parameter values represent sharp coincidence detector dynamics. I_{e} is sufficiently large to prevent the SAM neuron from firing at all. It is not necessary to model any specific spike duration nor refractory period because we are merely interested in the onset of a spike in time, which signals a sym-point.

3. Results

3.1. PM behavior

A block of four neurons, whose V_{PM} is raised above the spiking threshold for one time step, triggers an expanding annular wave growing outward as shown in Fig. 3a. Grayish dots represent neuronal subthreshold activity, black dots represent spiking activity ($V_{\mathrm{PM}} = E_{\mathrm{Na}}$). There is no subthreshold activity behind the wave front, because the refractory period clamps the voltage to 0 (for duration d_{r}). The propagation speed is proportional to the conductance value determining the connecting strength between neurons. A higher conductance value increases propagation speed and also increases the wave width. In the present simulations, the values are set to obtain a narrow wave width. A larger wave width would cause several neighboring OC neurons to fire, which leads to worthless sym-points along the shape contour and which can generate accidental sym-points for curved waves (later discussed again). Raising V_{PM} of a single neuron above threshold (for one time

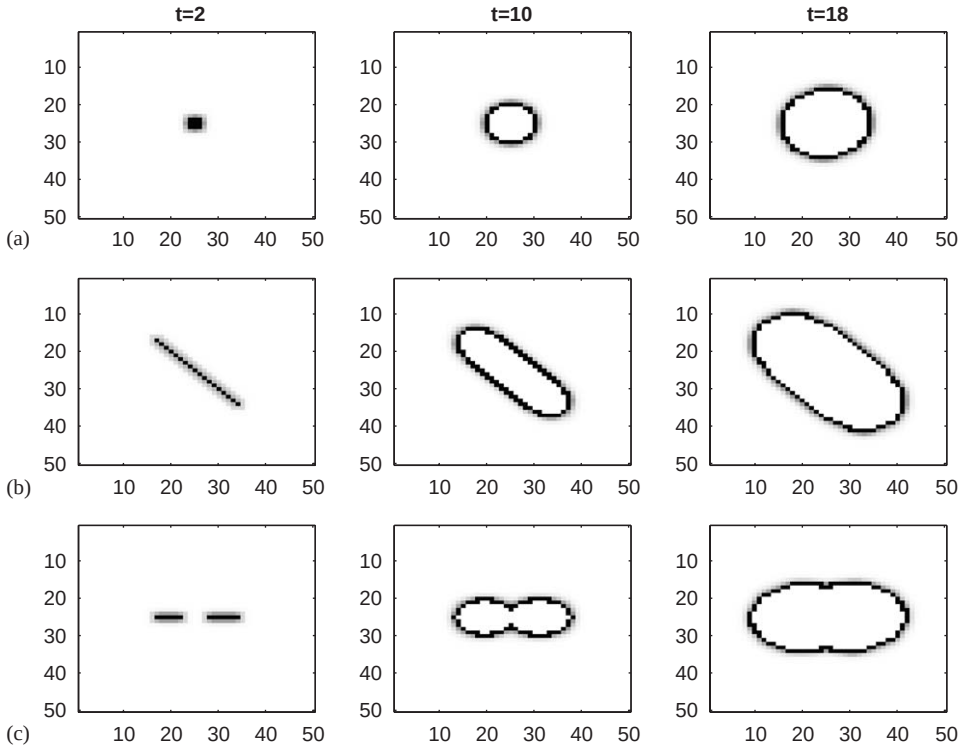

Fig. 3. Behavior of the PM in response to simple stimuli (Matlab Simulation). (a) Block source. (b) Diagonal line. (c) Dashed horizontal line. The snapshots are taken at times $t = 2$, $t = 10$ and $t = 18$, with time equal the number of time steps. Gray: subthreshold voltages ($V_{PM} < V_{thres}$). Black: Spike voltages ($V_{PM} = E_{NA}$). Reprinted with permission by Springer (Rasche 05a).

step) does not trigger a wave because it does not spread enough activity to ignite its neighboring neurons.

Fig. 3b shows the response of the PM to a line: it triggers an oval-shaped wave. The map's status at $t = 10$ shows that the wave width is occasionally two spikes (black dots only). This transient wave enlargement can not be resolved due to the resolutional insufficiency.

Fig. 3c shows the response of the PM to a horizontal line with a gap: the two small lines trigger two growing oval-shaped wave fronts, which, when they grow into each other, cancel each other out. The two merging waves eventually evolve into a single oval-shaped wave: the contour gap is thus smoothened out during wave propagation, which is the crucial property that allows fragmented contours as input to the SAT.

In the following figures only the spiking activity of a propagating wave is shown, meaning the subthreshold values are omitted.

3.2. Sym-ax evolvement of a L feature

Fig. 4 shows the evolvement of three varying L features, one with solid lines (a)—solid L—one with dashed lines (b)—dashed L—and one with curved lines (c)—curved L.

Until $t = 4$, the solid L has generated two sym-points shown as black dots. The wave spikes are shown in gray (no subthreshold values shown in this and the following figures). The dashed L has generated only a single sym-point because not enough OC neurons have fired due to the short (dashed) line pieces. The curved L has no sym-points until this point in time due to its curvature, which in turn is correctly ignored by the SAM neurons.

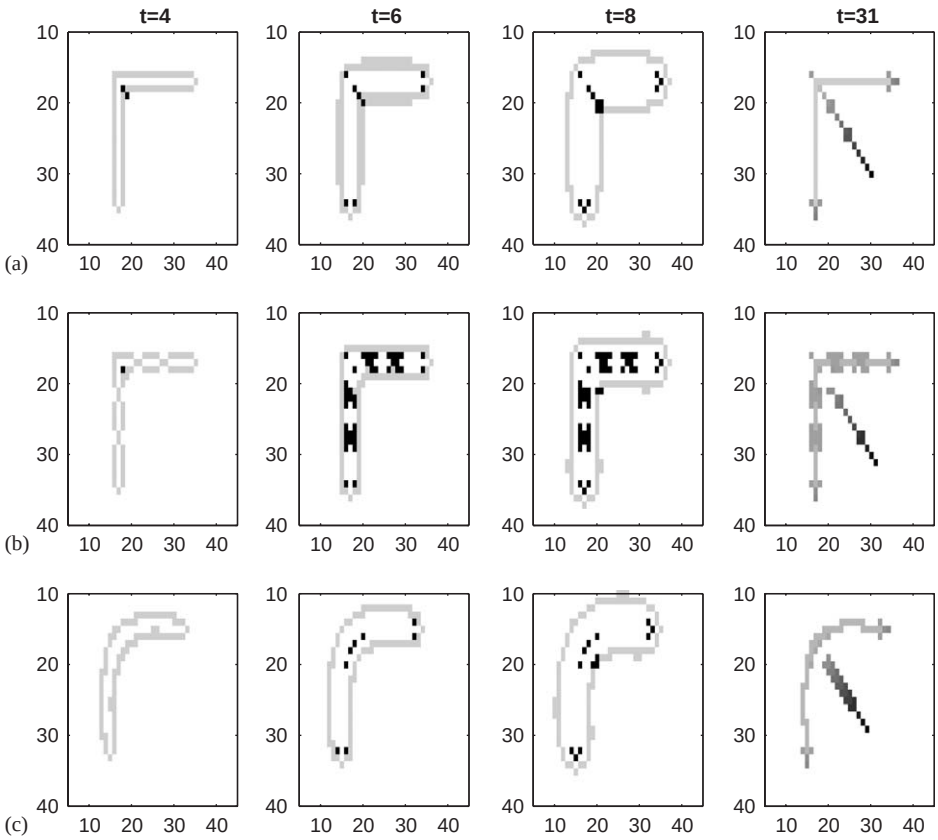

Fig. 4. Sym-ax evolvement of varying L features. (a) solidn L. (b) dashed L. (c) curved L. The snapshots are taken at times $t = 4$, $t = 6$, $t = 8$ and $t = 31$, with time being the number of time steps. For $t = 4, 6, 8$: gray dots or lines represents spiking activity in the PM (S_{PM}); black represents sym-points as evolved in the SAM (no subthreshold activity shown in this and the following figures). For $t = 31$: lightest gray: original L shape; gray to black: sym-ax evolved in time. Reprinted with permission by Springer (Rasche 05a).

Until $t = 6$, the solid L has evoked one more sym-point adding to the previous two. Another one popped up in the outer corner of the expanding wave (at approximately $x = 15$, $y = 15$) due to the still sharp angle between the two wave fronts (caused by the resolutional insufficiency). It also appeared for the dashed L. Two more pairs exist at each line end for the same reason and are also existent for the dashed and curved L. There is no sym-point in the exact corner of the solid L, its intersection, due to the suppression by the input shape (term $I_p(t)$ in Eq. (6)). The dashed L has many more sym-points due to the many line endings. The curved L has evoked four sym-points along its contour due to the resolutional insufficiency. More specifically, at the breakpoints of the aliased (and curved) contour, an OC neuron—whose receptive field crosses the breakpoint—receives initially full input from the two adjacent waves, thus causing the creation of sym-points around the breakpoints. One could easily suppress them by widening the inhibitory influence of I_e (Eq. (6)), for example I_e would not only suppress the SAM neurons along the shape contour but also its immediate neighbors. Such a widening in turn would also inhibit the sym-points in the corner of the L features (its initial sym-points at $t = 4$) thus loosing accuracy in the exact spatial encoding of the L feature.

Until $t = 8$, one more sym-point was added at each line end. No more sym-points are created at the outer corner of the propagating wave. From this point in time on, the inner corner of the dashed and curved L starts to continuously evolve the sym-ax.

Until $t = 31$, the SAT is complete. The original shape is shown in light gray, the sym-points are shown in gray-scale, according to their time of appearance (light gray dots earliest, black dots latest).

The sym-ax of the solid L starts exactly in the corner, the other two start somewhat later, but have all the same direction and endpoint. If we had chosen no inhibition from orthogonal neurons (term I_{inh} in Eq. (3)), there would be more sym-points around the initial sym-points of the sym-ax (at $t = 4$). The rest of the sym-ax remains the same irrespective of the inhibitory range size. The width of the sym-ax is sometimes two (or three) pixels wide thick due to the resolutional insufficiency.

3.3. Sym-ax of simple shapes

The rectangle leaves behind a sym-ax consisting of five straight lines (Fig. 5a). The four lines starting in each corner are equivalent to the sym-ax as evolved for a L feature. The horizontal middle-piece is evolved at last and at once. The SAT of an ellipse generates a straight sym-ax aligned with the ellipse's elongation (Fig. 5b). Its evolvement starts at both its (spatial) endpoints simultaneously—with some delay as expected from the example of the curved L feature—then growing towards each other increasingly faster until they meet at the center of the ellipse. The circle leaves behind a X-shaped sym-ax due to the resolutional insufficiency (Fig. 5c). The worm shape is transformed into a sym-ax outlining the general curvature of the shape (Fig. 5d). It starts to grow from its spatial endpoints, similar to the sym-ax evolvement of the ellipse, and then soon after that the entire middle section is formed at once, similar to the middle line of the rectangle's sym-ax. There are accidental sym-points along the contours of the ellipse, circle and worm shape. These are generated by the

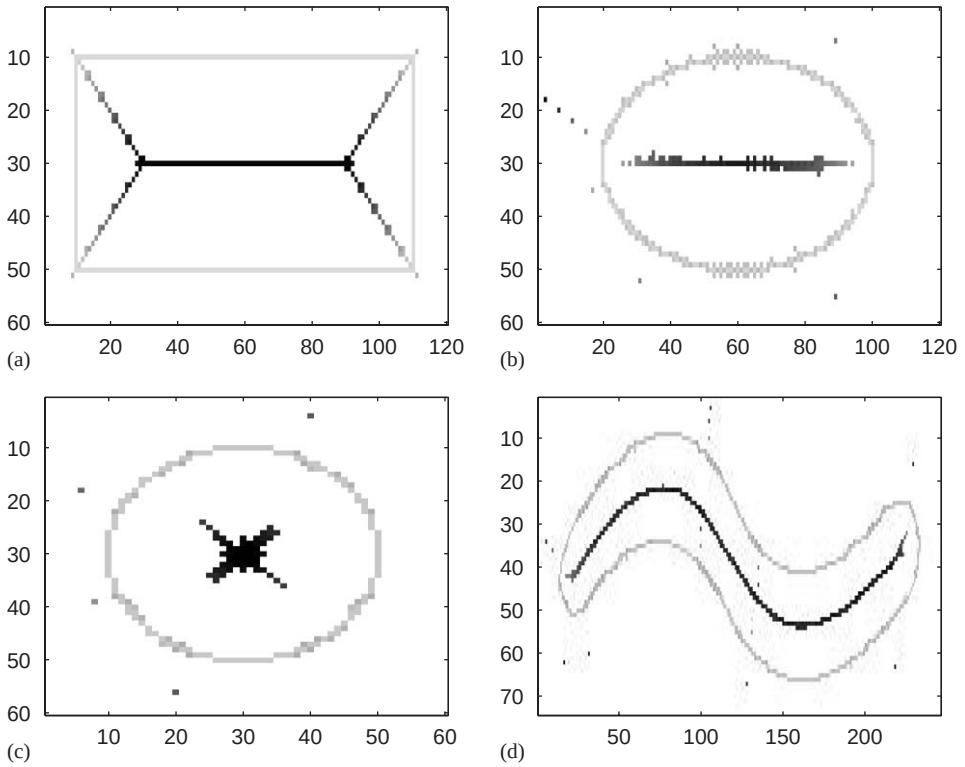

Fig. 5. Sym-ax of simple shapes. (a) Rectangle. (b) Ellipse. (c) Circle. (d) Worm shape. Light gray: shape. Gray to black: evolved sym-ax. Dots inside and outside the shape are accidentally generated sym-points due to the resolutional insufficiency. The sym-points generated along the contour could be suppressed by inhibitory input from the shape contour.

aliasing of the contours, as discussed for the curved L case. There are also accidental sym-points in the outside space of the shapes, as well as in the inside space for the worm shape. These occur by a combination of the transient wave enlargements as well as the occasional sharp contour pieces that are generated for curved propagating waves.

3.4. Sym-ax of complex shapes

Fig. 6 shows the extraction of the sym-axes of an object depicted in a photograph. Contours are obtained using the Canny algorithm run at the finest scale (= 1), with a high threshold of 0.8 and a low threshold of 0.5 for both pictures. The contours are fragmented having gaps of different sizes. The major regions have been encoded. Most of them are a fraction or variant of the rectangles sym-ax. One can divide the regions and their corresponding sym-axes into three basic types: (1) early evolved

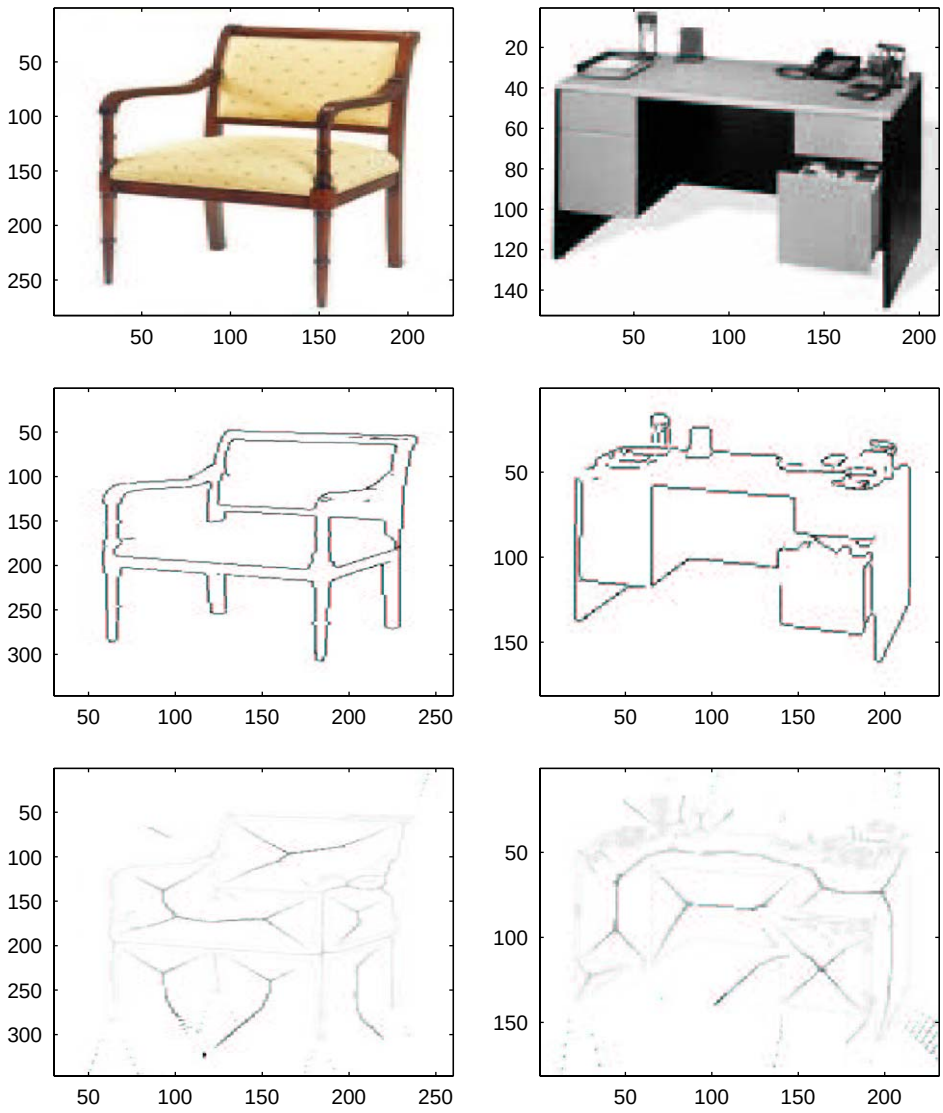

Fig. 6. Sym-ax of complex objects. Top row: Photos of a chair and a desktop. Middle row: Contours extracted with the Canny algorithm at finest scale. Bottom row: Evolved sym-ax from obtained contours. The major spaces (regions) have been encoded. The object contours are shown in lightest gray as before.

sym-axes of object parts, for example the chair's legs; (2) sym-axes outlining general regions, for example the space between the chair's legs; (3) sym-axes representing surfaces as for example the drawers' rectangle. Such sym-axes should be sufficient for generating object hypotheses.

4. Discussion

4.1. Space as a structurally robust feature

The example with varying L features demonstrates best the usefulness of encoding space (Fig. 4). Although the various L features have structural differences, their space between the two lines is similar for all of them. The sym-ax for the dashed and curved L starts with a delay compared to the sym-ax of the solid L, but their direction and endpoints are the same indicating thus the similar space but also the similar contour geometry of the feature. In a pure contour description approach in contrast, the dashed L had to be described by integrating the individual line pieces, which is an expensive undertaking. Using the present SAT architecture, the space between the legs is exploited and thus provides a large amount of robustness to incomplete contours and structural variability. At the core of this robustness is the smearing property of the PM. Fig. 3c illustrates this property explicitly. These properties translate well to contours extracted from a gray-scale image (Fig. 6). Even though fragmented and containing many gaps, the rough structure of the object is captured and can well serve as a starting point for object reconstruction.

It is noteworthy, that the SAT encodes any space, not only the space of object parts. This is important, because general space between object parts is useful structural information too [30].

4.2. Simulation limitations

The primary performance limitation is the resolutional insufficiency. L features with an angle greater than approximately 110° do not generate a sym-ax due to the limited column integration range (7 out of 12). If we had chosen a larger integration range to encode large-angle L features, then in turn, the propagating wave of a curve would be signaled as two colliding waves. Thus, there is a tradeoff between encoding large-angle L features and curves. The receptive field size of an OC neuron (diameter of its orientation) also matters. If one chooses a larger size, for example 5 neurons, and a larger diameter for the pairs of CIM neurons $(p_1, \ldots, p_4)$, one would eliminate the accidental sym-points generated inside and outside the shapes (circle, ellipse, worm). Then again, curved contours could not be encoded because the receptive field size is too large to sense subtle curvature. One solution to both of these tradeoffs is to employ receptive fields of different sizes. One layer with small receptive fields and small integration range would be useful for encoding curves and small-angle L features. Another layer with large receptive fields and larger integration range would be useful for encoding large-angle L features. This would be similar to using different resolutional scales to extract contours from a gray-scale image. Another solution to the above-mentioned tradeoff, would be to employ PMs for different orientations. Local contour pieces would firstly be classified according to their orientation, as it presumably occurs in primary visual cortex, and then fed into corresponding PMs that propagate only contour segments of a specific orientation. Curves would not be propagated by such 'oriented' PMs, because they would be broken up across several oriented maps. But

large angle features could be easily detected by finding coincidences across oriented PMs. In such a scheme, parallel lines—which are salient visual features—could be easily detected by sensing colliding waves within the same oriented PM.

4.3. Steps towards analog VLSI circuits

The purpose of the present study aimed at out-lining an architecture that has the potential for an analog hardware implementation. The translation of the present architecture into analog hardware required firstly the proper design and analysis of the individual components, and then another round of software-simulations to determine the fine-tuning between the components. For the development of the suggested system, the first step would be the construction of a propagation map.

The present propagation map, with octagonal connectivity, is actually simulated with an Eq. (2) that emulates unidirectional flow of a resistor, because the octagonal connectivity with bidirectional resistors tends to generate unstable wave propagation [31]. It remains to be seen whether the presently simulated map could by be implemented in analog hardware. An alternative would be to employ a hexagonal grid, for which implementations already exist (e.g. [4,24]). But those would have to be extended to make them functioning with spikes. The use of a hexagonal grid would also require the use of a higher resolution to extract orientations (see [31] for a discussion).

Once such a propagation map has been designed, its exact output would be used to perform a second round of simulations. Those simulations should then be simulated with regard to two major aspects. One is that a partitioning of the network across several chips has to be designed [7]. The other is that the networks have to be simulated electronically more realistically. That would for example include the following items: (a) The network would be simulated at higher temporal time resolution; the present simulation step size of 0.2 is too coarse to guarantee the successful simulation in analog circuits. (b) The network would be simulated with more realistic dynamics of the detailed neuronal units including for example mismatch between neuronal components to mimick chip fabrication vagrancies. Present multi-chip systems, as for example the one emulating orientation-selective units [22] could serve as guidance or standard in establishing such a simulation methodology.

4.4. Comparison to biological hardware

The prevailing view of how the real visual system describes visual shape is that it integrates contour pieces. But there is also psychophysical as well as neurophysiological evidence, that the visual system carries out SAT like processes (e.g. [5,18,20]). In these schemes it is assumed that there is a map of neurons, each receiving input from two edge detector units that are equidistant from the receiving neuron, signaling thus a sym-point. In our terms, these are basically SAM neurons that directly receive input from an array of edge detectors. Such an implementation requires an enormous amount of wiring across space to extract sym-axes. The presented architecture evolves the sym-ax using merely local wave detectors thereby

minimizing the amount of wiring. Moreover, the architecture lends to a possibly different view on the processes taking place in primary visual cortex as outlined next.

One can regard the wave propagation (PM) as a wave traveling across an excitable map. Traveling waves exist in the nervous system of many animals (e.g. [12,28,32,34]). Generally, they were considered as non-functional, accidentally emerging from networks of neurons. Recently, there has been some effort to attribute computational functions to traveling waves. Jacobs and Werblin measured traveling waves in the salamander retina in response to visual stimulation, and speculated that it may have edge enhancing effects [13]. They envision that such waves are possibly involved in different neural computations. Barch and Glaser use excitable membranes to detect motion: different motion signals leave traveling waves of characteristic shape on an excitable membrane [2]. Our PM is closest to their model of an excitable map.

The OCs in our architecture bear obvious similarity to the OCs found in the primary visual cortex of cat and monkey [10,11]. In the present study, the original motivation was to solve the problem of detecting two colliding waves. When we decided to chose local orientation detectors, it was obvious to call these columns and to illustrate them as the OC architecture found in animals. Orientation-selective neurons in animals fire when a (static) edge or line of appropriate orientation appears within their receptive field. Our OC neurons also fire when they are stimulated with a contour stimulus within their receptive field. But that is not their main function: they primarily fire whenever there is a dynamic wave piece of appropriate orientation, propagating through their receptive field—they react to stimuli that have also been placed outside their receptive field. This may not be directly supported by physiological experiments because stimulation outside the receptive field generally causes a suppression in the firing rate (e.g. [6] for a review). But since our system does not deal with actual firing frequencies we point out a different experimental finding that may support this alternate view. If an anesthetized animal is presented with a simple and optimal stimulus, then the orientation-selective cells fire at a high frequency (e.g. [10]). In contrast, if one uses natural stimuli, then very low firing rates are observed [1,23,33]: For example when test animals were presented a video sequence of their typical environment, VI neurons fired at only around 14–50 Hz; In humans (epileptic patients) the firing frequency of enthorhinal and temporal cortical neurons was measured around 4–10 Hz for recognized images [19]. In this low-frequency case, it seems more plausible that an occasional spike may be the result of a propagating wave traveling through the receptive field of that neuron. The high-frequency case can thus be interpreted as the continuous accumulation (or collision) of waves; or it could also be interpreted as an 20 idling of cortical feedback loops due to the 'unnaturally' simple input; or a mixture of both mechanisms.

Acknowledgements

The study was started in Prof. C. Koch's lab at Caltech and was funded by the Engineering Research Centers Program of the National Science Foundation under Award Number EEC-9402726. It has been continued in Prof M. Eckstein's lab at

UCSB, funded by NIH-RO1 53455, NASA NAG 9-1157, NSF 0135118. The author wishes to thank both professors for generous support.

References

[1] R. Baddeley, L.F. Abbott, M.C. Booth, F. Sengpiel, T. Freeman, E.A. Wakeman, E.T. Rolls, Responses of neurons in primary and inferior temporal visual cortices to natural scenes, P. Roy. Soc. London B Bio. 264 (1389) (1997) 1775–1783.
[2] D. Barch, D.A. Glaser, Slowly moving stimuli induce characteristic periodic activity waves in an excitable membrane model of visual motion processing, Neurocomputing 44 (2002) 43–50.
[3] H. Blum, Biological shape and visual science, J. Theoret. Biol. 38 (2) (1973) 205–287.
[4] K. Boahen, A retinomorphic chip with parallel pathways: Encoding increasing, on, decreasing, and off visual signals, Analog Integr. Circs. 30 (2) (2002) 121–135.
[5] C.A. Burbeck, S.M. Pizer, Object representation by cores—identifying and representing primitive spatial regions, Vision Res. 35 (13) (1995) 1917–1930.
[6] J.R. Cavanaugh, W. Bair, J.A. Movshon, Nature and interaction of signals from the receptive field center and surround in macaque v1 neurons, J. Neurophysiol. 88 (5) (2002) 2530–2546.
[7] S.R. Deiss, R.J. Douglas, A.M. Whatley, A pulse-coded communications infrastructure for neuromorphic systems, in: W. Maass, C.M. Bishop (Eds.), Pulsed Neural Networks, The MIT Press, New York, 1998, in press (Chapter 6).
[8] A.R. Dill, M.D. Levine, Multiple resolution skeletons, IEEE Trans. Pattern Anal. Mach. Intell. 9 (4) (1987) 495–504.
[9] R.J. Douglas, C. Rasche, Silicon neurons, in: M.A. Arbib (Ed.), The Handbook of Brain Theory and Neural Networks, MIT Press, Cambridge, MA, 2002.
[10] D. Hubel, T. Wiesel, Receptive fields, binocular interaction and functional architecture in the cat's visual cortex, J. Physiol. 160 (1962) 106–154.
[11] D.H. Hubel, T.N. Wiesel, Receptive fields and functional architecture of monkey striate cortex, J. Physiol. (London) 195 (1968) 215–243.
[12] J.R. Hughes, The phenomenon of traveling waves—a review, Clin. Electroencephal. 26 (1) (1995) 1–6.
[13] A.L. Jacobs, F.S. Werblin, Spatiotemporal patterns at the retinal output, J. Neurophysiol. 80 (1) (1998) 447–451.
[14] M. Jammer, Concepts of Space, Harvard University Press, Cambridge, MA, 1954.
[15] B. Kegl, A. Krzyak, Piecewise linear skeletonization using principal curves, IEEE Trans. Pattern Anal. Mach. Intell. 24 (1) (2002) 59–74.
[16] C. Koch, Computational Biophysics of Neurons, MIT, Cambridge, MA, 1999.
[17] I. Kovacs, A. Feher, B. Julesz, Medial-point description of shape: a representation for action coding and its psychophysical correlates, Vision Res. 38 (15–16) (1998) 2323–2333.
[18] I. Kovacs, B. Julesz, Perceptual sensitivity maps within globally defined visual shapes, Nature 370 (6491) (1994) 644–646.
[19] G. Kreiman, C. Koch, I. Fried, Category-specific visual responses of single neurons in the human medial temporal lobe see comments, Nat. Neurosci. 3 (9) (2000) 946–953 Comment in: Nat. Neurosci. 3(9) (2000) 855–856.
[20] T.S. Lee, D. Mumford, R. Romero, V.A.F. Lamme, The role of the primary visual cortex in higher level vision, Vision Res. 38 (15–16) (1998) 2429–2454.
[21] F. Leymarie, M.D. Levine, Simulating the grassfire transform using an active contour model, IEEE Trans. Pattern Anal. Mach. Intell. 14 (1) (1992) 56–75.
[22] S.C. Liu, J. Kramer, G. Indiveri, T. Delbruck, T. Burg, R. Douglas, Orientation-selective avlsi spiking neurons, Neural Networks 14 (6–7) (2001) 629–643.
[23] S.J. Luck, L. Chelazzi, S.A. Hillyard, R. Desimone, Neural mechanisms of spatial selective attention in areas v1, v2, and v4 of macaque visual cortex, J. Neurophysiol. 77 (1) (1997) 24–42.
[24] C.A. Mead, Analog VLSI and Neural Systems, Addison-Wesley, Reading, MA, 1989.

[25] R.L. Ogniewicz, O. Kubler, Voronoi tessellation of points with integer coordinates: Time-efficient implementation and online edge-list generation, Pattern Recogn. 28 (12) (1995) 1839–1844.
[26] S.E. Palmer, Vision Science: Photons to Phenomenology, MIT Press, Cambridge, MA, 1999.
[27] S.M. Pizer, W.R. Oliver, S.H. Bloomberg, Hierarchical shape-description via the multiresolution symmetrical axis transform, IEEE Trans. Pattern Anal. Mach. Intell. 9 (4) (1987) 505–511.
[28] J.C. Prechtl, L.B. Cohen, B. Pesaran, P.P. Mitra, D. Kleinfeld, Visual stimuli induce waves of electrical activity in turtle cortex, P. Nat. Acad. Sci. USA 94 (14) (1997) 7621–7626.
[29] C. Rasche, Signaling contours by neuromorphic wave propagation, Biol. Cybernet. 90 (4) (2004) 272–279.
[30] C. Rasche, The Making of a Neuromorphic Visual System, Springer, New York, 2005.
[31] C. Rasche, Excitable maps for visual processing, under review, 2005.
[32] I.A. Shevelev, E.N. Tsicalov, Fast thermal waves spreading over the cerebral cortex, Neuroscience 76 (2) (1997) 531–540.
[33] W.E. Vinje, J.L. Gallant, Sparse coding and decorrelation in primary visual cortex during natural vision, Science 287 (5456) (2000) 1273–1276.
[34] H.R. Wilson, R. Blake, S.H. Lee, Dynamics of travelling waves in visual perception, Nature 412 (6850) (2001) 907–910.

Christoph Rasche received a diploma in theoretical neurobiology at the University of Zürich in 1996. He completed a Ph.D. degree in Computational Neuroscience, with a specialization in Neuromorphic Engineering, at the ETH Zürich in 1999. During a postdoctoral stay at Caltech, Pasadena, he began studying the subject of object and scene recognition. He also studied visual psychophysics (perceptual learning) at UC Santa Barbara and Penn-State. His research interests are neuromorphic engineering of a visual recognition architecture (http://www.springeronline.com/0-387-23468-3) and visual perceptual learning.