International Journal of Bifurcation and Chaos, Vol. 14, No. 2 (2004) 427–451

IMPLEMENTING THE MULTILAYER RETINAL MODEL ON THE COMPLEX-CELL CNN-UM CHIP PROTOTYPE

DÁVID BÁLYA, ISTVÁN PETRÁS and TAMÁS ROSKA
Analogical and Neural Computing Laboratory,
Hungarian Academy of Sciences Computer and Automation Research Institute,
Lágymányosi u. 13, Budapest, H-1111, Hungary
Jedlik Laboratories, Department of Information Technology,
Pazmany Peter Catholic University, Budapest, Hungary

RICARDO CARMONA and ANGEL RODRÍGUEZ VÁZQUEZ
Instituto de Microelectronica de Sevilla – CNM-CSIC,
Edificio CICA-CNM, C/Tarifa s/n, 41012-Sevilla, Spain

Received January 27, 2003; Revised March 10, 2003

There is hardly a more important sensory modality for humans and other mammals than vision. The first and best-known part of the visual system is the retina, which is not a mere photoreceptor or static camera but a sophisticated feature preprocessor with continuous input and several parallel output channels. These interacting channels represent the visual scene. Never before has it been known in neuroscience how these channels build up a "visual language". Our mammalian retina model can generate the elements of this visual language. In the present paper the design steps of the implementation of the multilayer CNN retinal model is shown. It is rare that an analogic CNN algorithm has such a sophisticated series of different complex dynamics, meanwhile it is feasible on a recently fabricated complex cell CNN-UM chip. The mammalian retina model is decomposed into a full-custom mixed-signal chip that embeds digitally programmable analog parallel processing and distributed image memory on a common silicon substrate. The chip was designed and manufactured in a standard 0.5 μm CMOS technology and contains approximately 500,000 transistors. It consists of 1024 processing units arranged into a 32×32 grid. The functional features of the chip are in accordance with the second-order complex cell CNN-UM architecture: two CNN layers with programmable inter- and intra-layer connections between cells as well as programmable layer time constants. The uniqueness of this approach, among others, lies in the reprogrammability, i.e. the openness to any new discovery, even after a possible retinal implementation.

Keywords: Rabbit retina; multi-channel inner retina; model decomposition; silicon retina; cellular neural network; CACE1k.

1. Introduction

This paper presents an analogic CNN algorithm, i.e. analog and logic transient array computation model, in order to mimic the mammalian retina. The modeling approach is neuromorphic in its spirit, relying on both morphological and electrophysiological information. The fine-tuning of the model is based on the flashed square response in the rabbit retina. Such an algorithm can be useful for many purposes, e.g. to improve and develop more efficient algorithms for object classification, recognition and tracking. Due to the computational

complexity of the algorithm the model can also help to create sensing aids, like retina prosthesis or retina-chips [Wyatt & Rizzo, 1996; Dagnelie & Massof, 1996; Boahen, 2002; Hesse *et al.*, 2000].

The cellular neural/nonlinear network (CNN) is a regular, single or multilayer, parallel processing structure with analog nonlinear computing base cells [Chua & Yang, 1988]. The state value of each individual processor is continuous in time and their connectivity is local in space. The function of the network is completely determined by the pattern of the local interactions, the so-called template [Chua & Roska, 1993]. The time-evolution of the analog array transient, given by the template operator and the processor dynamics, represents the computation in CNN [Chua & Roska, 2002]. The result of the computation can be defined in equilibrium points for image processing tasks or nonequilibrium states of the network for transient computation or retina modeling [Chua, 1998]. Completing the base cells of CNN with local sensors, local data memories, arithmetical, and logical units, plus global program memories and control units have resulted in the CNN Universal Machine (CNN-UM) architecture [Roska & Chua, 1993].

The Complex Cell CNN programmable array computer is an extension of the CNN Universal Machine with two or more CNN core layers. Its second-order 3-layer version provides complex spatial-temporal dynamics with a few programmable parameters [Rekeczky *et al.*, 2000]. The elementary cells of the specific chip (CACE1k) are organized into a 32×32 square grid. Each cell has second order dynamics and local interconnections to its neighbors [Carmona *et al.*, 2002]. This structure is especially suitable for computing a certain complex set of ordinary differential equations. Simple PDEs can be transformed so that they can be easily programmed on the array computer. With the programmable Complex Cell CNN Universal Machine scientists have been provided with a unique opportunity to study some active-wave propagation and other reaction–diffusion differential equations based phenomena in a programmable manner, in real-time.

A realistic mammalian retina model, abbreviated MRM, should reproduce the measured 2D spatial-temporal patterns described in [Roska & Werblin, 2001]. An MRM is not necessarily simple nor does it operate in a huge parameter space [Koch & Segev, 1989] thus a well-designed MRM can be implemented on silicon. It can reproduce the full range of the known retinal phenomena, not just one general effect or some selected features but each and every measured retina effect. Its outputs are the responses of several different ganglion cell types or parallel retina channels [Roska & Werblin, 2001]. It needs qualitatively more sophisticated constructions than the spatial "resistive grid" idea, i.e. coupled diffusion layers with different time-constants and not necessarily symmetric weight templates. An MRM can be tuned to different species.

The presented mammalian retina model reproduces several 2D space-time patterns, embedded in a multi-layer CNN framework [Balya *et al.*, 2002]. The algorithmic approach enables us to adapt the model to different measured retina phenomena. The steps of the algorithm can also be implemented by hardware. By *combining a few simple CNN layers* copying receptive field dynamics, and *embedding these into a stored programmable CNN-UM* we can generate not just simple, but complex spatial-temporal patterns.

This is the first time that a *programmable, realistic*, and *implemented* complete mammalian retina model is presented. In Sec. 2, we show the CNN computational framework as a formal introduction. Section 3 contains the outer retina model as a case study. In Sec. 4 the structure of the complete retina model is described. Section 5 deals with the decomposition of the multilayer model to simple uniform elements that we call Retinal Units. Section 6 gives a concise description of the recently fabricated complex-cell CNN-UM chip. We present our first experiments with this CACE1k chip and describe the measurements of the retina blocks in Sec. 7.

2. Wave Computing — Formal Introduction

(1) Data is defined as a continuous image flow $\mathbf{\Phi}(t)$

$$\mathbf{\Phi}(t) : \{\varphi_{ij}(t),\, i=1,\, 2,\ldots,\, n;\, j=1,\, 2,\ldots,\, m\} \in \mathbf{R}^2 \quad t \in T=[0,\, t^*]$$

A frame is obtained by setting the time variable in a given finite time instance t^*, i.e. $P = \mathbf{\Phi}(t^*)$. Without loss of generality, we may assume that in a grayscale image the black and white levels are represented by $+1$ and -1, the gray levels are in between. We will display sample frames from the image flow at different time-instances using pseudo-coloring ($+1$ dark red and -1 dark blue) to improve the visualization, see Fig. 3.

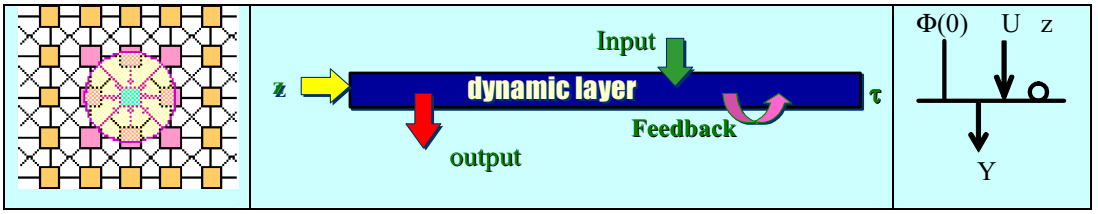


Fig. 1. The graphical representation of the elementary wave processing. The individual $\varphi_{ij}(t)$ base units and their connections are shown on the left. On the right the horizontal bar represents the processing structure as a two-dimensional layer and the arrows show the external connections to each cell. The color of the arrow refers to the sign of the connection i.e. red: positive, green: negative, yellow: zero, and pink: any. At the far right a simple flow-chart notation is shown [Roska, 2003].

(2) Elementary instructions Ψ are the basic wave instructions:

$$\mathbf{\Phi}_{\text{output}}(t) = \Psi(\mathbf{\Phi}_{\text{input}})$$

Input : $\mathbf{U}(t) \equiv \mathbf{\Phi}_{\text{input}} : uij(t),\ t \in T$

State : $\mathbf{X}(t) \equiv \mathbf{\Phi}(t) : x_{ij}(t),\ t \in T$ Initial state :$X(0)$

Output : $\mathbf{Y}(t) \equiv \mathbf{\Phi}_{\text{output}} : y_{ij}(t),\ t \in T$

Bias : $Z = z_{ij}$ time invariant and space variant map

Operator : the solution of the two-dimensional spatial-temporal state equation/output equation:

$$\tau \frac{dx_{ij}(t)}{dt} = -x_{ij} + z_{ij} + \sum_{kl \in S_r(ij)} A_{ijkl} y_{kl}(t) + \sum_{kl \in S_r(ij)} B_{ijkl} u_{kl}(t) \tag{1}$$

$S_r(\cdot)$: sphere of influences: $S_r(ij) = \{C(kl) : \max\{|k-i|,\ |l-j|\} = r\}$ $y_{ij}(t) = \sigma(x_{ij}(t))$; σ: a nonlinear, usually sigmoid function

(3) This simplest wave instruction is equivalent to the CNN dynamics:

$$\tau \frac{dx_{ij}(t)}{dt} = -x_{ij} + I_{ij}(t) \qquad y_{ij}(t) = \sigma(x_{ij}(t)) \tag{2}$$

and the local cell interaction "pattern" is defined by

$$I_{ij}(t) = \sum_{kl \in S_r(ij)} A_{ijkl} y_{kl}(t) + \sum_{kl \in S_r(ij)} B_{ijkl} u_{kl}(t) + z_{ij} \tag{3}$$

The $\mathbf{C} : \{\mathbf{A}, \mathbf{B}, \mathbf{z}\}$ is the cloning template [Chua, 1998].

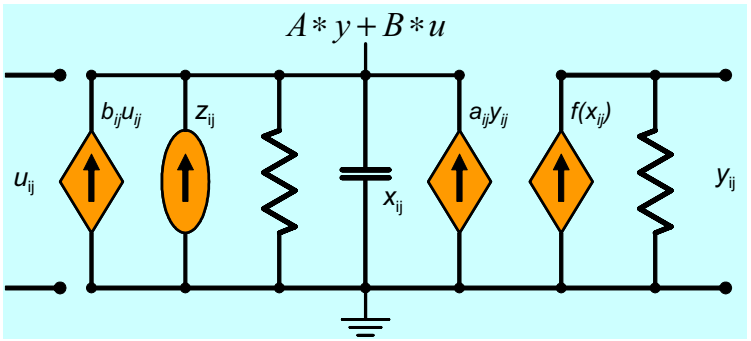


Fig. 2. The circuit diagram of a single CNN base cell with the important signals and interactions.

Useful notations:

$$\dot{x} \equiv \frac{dx_{ij}(t)}{dt}; \quad A * y \equiv \sum_{kl \in S_r(ij)} A_{ijkl} y_{kl}(t);$$

$$\text{if } (S_r = 0) \sum_{kl \in S_r(ij)} B_{ijkl} u_{kl}(t) = b_{ij} u_{ij}(t) \tag{4}$$

(4) A single cell can be implemented by a circuit and the interactions can be added, as shown in Fig. 2.

3. Case Study: Outer Retina Model

This section shows some common partial retina models starting with the simplest one, while our complete retina model is presented in the next section. Each step is described in several ways: differential equations, cloning template, graphic representation of the processing structure, and the simulated output of the system for a common input image flow or stimulus. The input is a static image which disappears suddenly, after a certain time t_{cut}, but the state of the system is not altered externally. The state evolution starts from a zero initial state $X(0) = 0$ and evolves continuously in time and value until t_{end}. The formal definition of the input flow is given in Eq. (5) and the stimulus is shown in Fig. 3. The graphical representation of the single, continuous flow contains two rows, the upper one displays the frames came from $t \leq t_{\text{cut}}$ and the second one from $t > t_{\text{cut}}$.

$$U(t) = \begin{cases} \text{Static image} & 0 \leq t \leq t_{\text{cut}} \\ \text{Gray field} & t_{\text{cut}} < t \leq t_{\text{end}} \end{cases} \tag{5}$$

3.1. *Simple naïve outer retina model: Diffusion effect*

The simplest dynamic retina effect is the diffusion or blurring [Gál *et al.*, 2004]. It incorporates one dynamic layer with simple nearest neighborhood feedback connections. The diffusion is originally defined

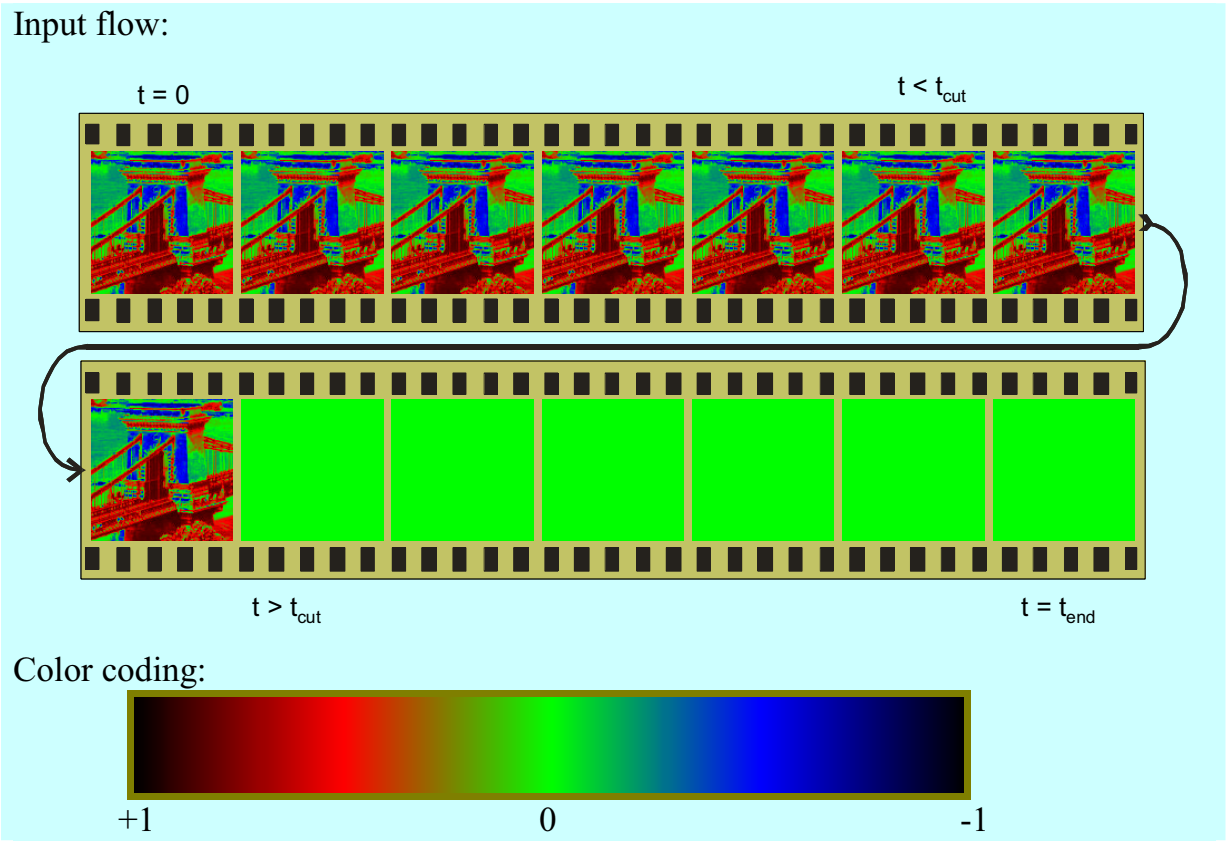


Fig. 3. The test input image flow of the outer retina, the stimulus. The continuous input flow is sampled at certain timepoints, these are the pseudo-colored frames. The two rows are the input of the system before the sudden change (static image) and after the change (gray field).

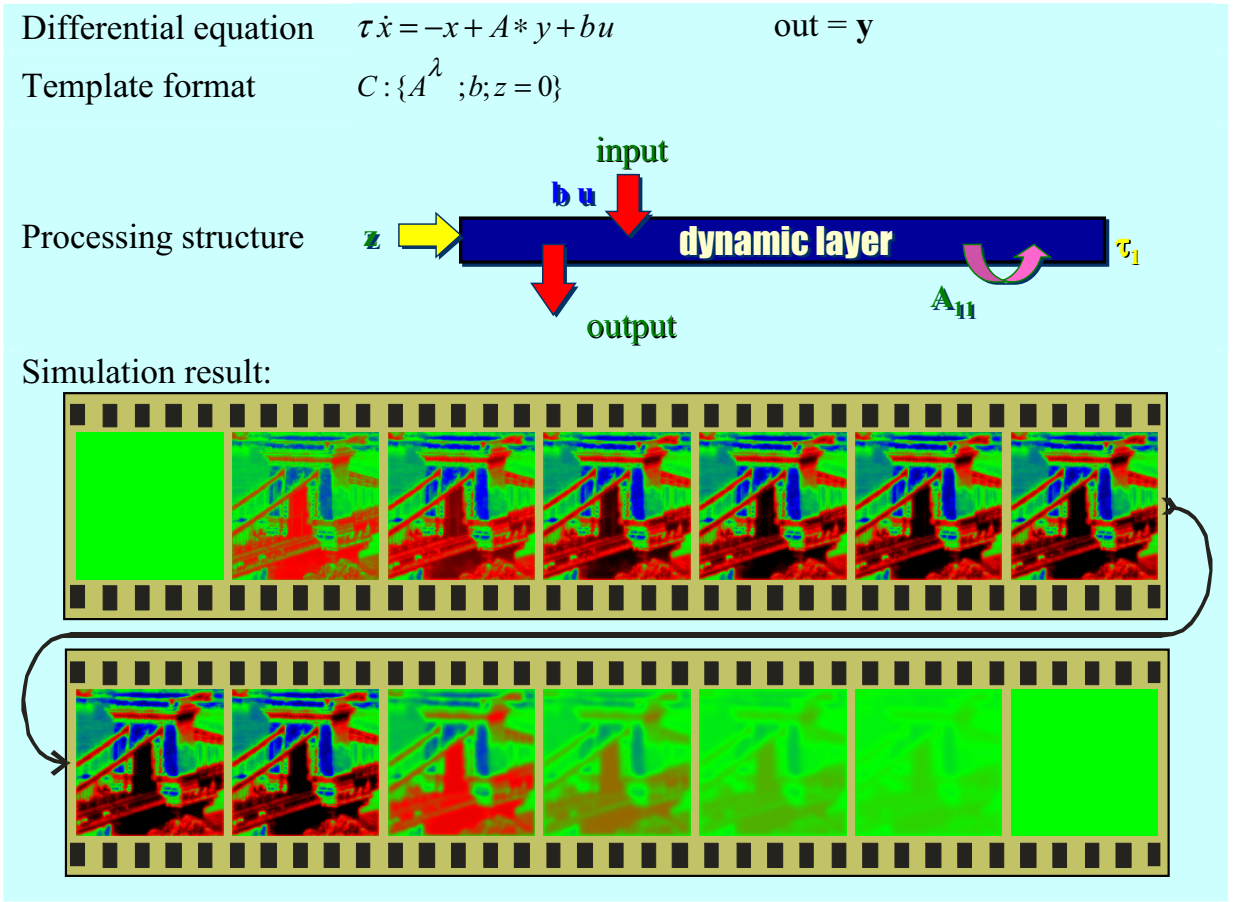


Fig. 4. The CNN description of the simple naïve outer retina model. The first row shows the differential equation, the second row displays the CNN template structure, the third row is a graphic representation of the processing structure, and the last row shows the simulated output (**out**) of the system for the continuous input flow given in Fig. 3.

as a differential equation operating on a spatially continuous input (analog input: $F_{xy}(t)$). It is discretized in space in order to be implemented as a basic wave instruction on CNN with a low-complexity template structure [Chua, 1997], given in Eq. (6). The space constant λ is a free parameter, which controls the strength of the diffusion. Figure 4 shows the processing structure. The arrows are colored using the conventions given in Fig. 1. The first frame shows the output of the system at time zero, clearly it displays the initial state of the system: $X(0) = 0$.

$$F_{xy}(t) \leftarrow U(t)\,; \quad Y(t) \leftarrow F_{xy}(t)$$

$$\frac{dF_{xy}(t)}{dt} = -\lambda^2 \frac{d^2 F_{xy}(t)}{dxdy} \Rightarrow \dot{x} = -x + A * y + u$$

$$\Rightarrow C : \left\{ A^{\lambda} = \lambda^2 \begin{bmatrix} 1 & 2 & 1 \\ 2 & -12 & 2 \\ 1 & 2 & 1 \end{bmatrix}, \; B = 1, \; z = 0 \right\} \tag{6}$$

3.2. *Simple "silicon retina"*

Let us combine the previous simple diffusion in Sec. 3.1 with a static layer to compute spatial-temporal edges. The hardware implementation of this structure is straightforward: a two-dimensional photoreceptor field is coupled by a resistive grid and the output is the difference between the primary image and the blurred one [Werblin & Jacobs, 1994; Cauwenberghs & Waskiewicz, 1999]. Results are illustrated in Fig. 5.

3.3. *Difference of Gaussians principle — DOG*

Difference of Gaussians is an image enhancement method that involves the subtraction of one blurred version of a grayscale image from another, less blurred version of the original input. The blurred images are obtained by convolving the original grayscale image with two Gaussian kernels having

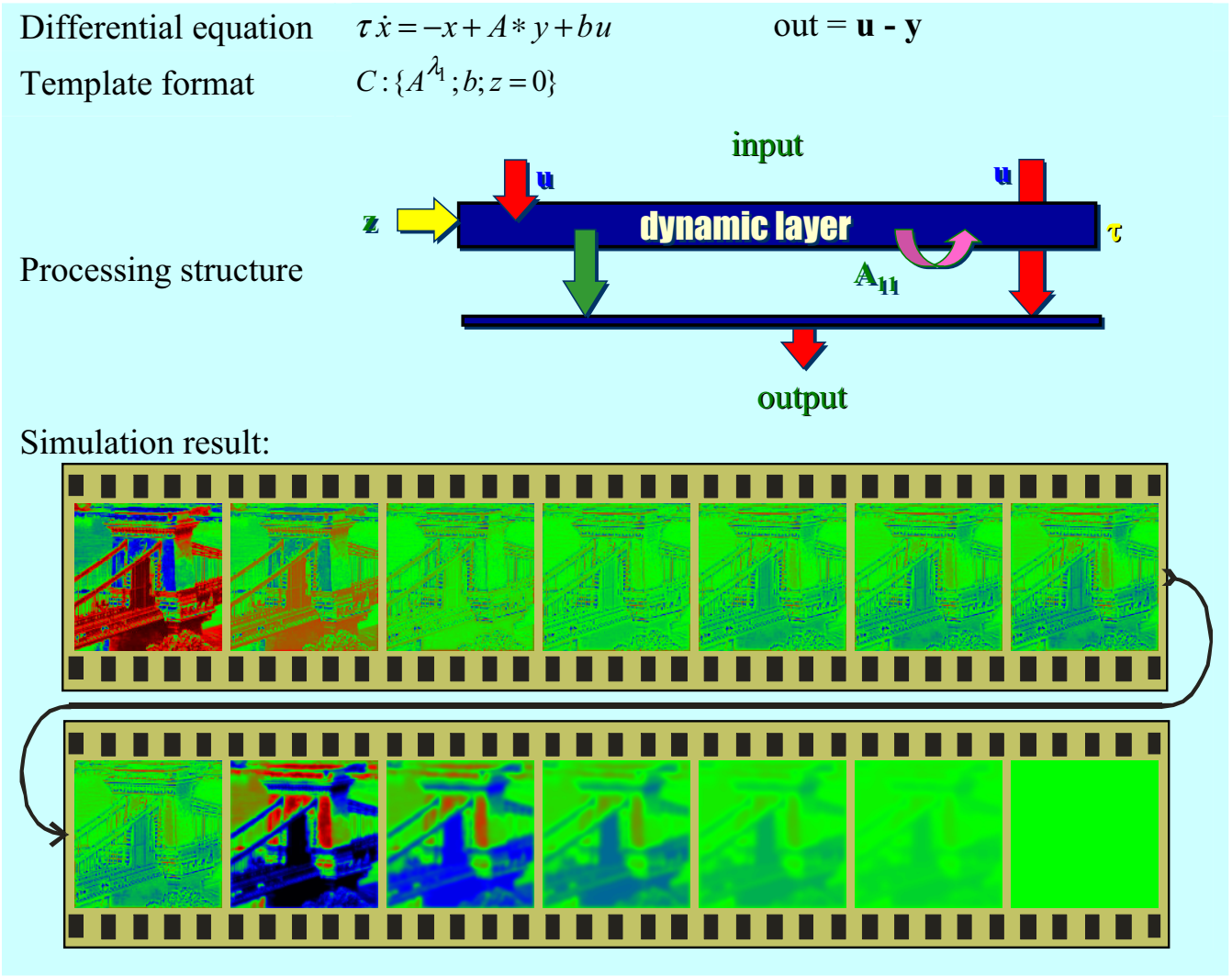


Fig. 5. The CNN description of the simple "silicon retina". The upper part of the figure shows the differential equation and the CNN template and processing structure, while the last row shows the simulated output (**out**) of the system for the continuous input flow illustrated in Fig. 3.

different standard deviations. The CNN implementation of the Gaussian blurrings are done by two diffusion templates, with two different λ's. It is equivalent to a bandpass filter that discards all unwanted spatial frequencies present in the original input. In its operation, the difference-of-Gaussians algorithm is believed to mimic how neural processing in the retina extracts details from images destined for transmission to the brain [Linsenmeier *et al.*, 1982]. The latest findings about the retina, e.g. the parallel channels [Werblin & Roska, 2004], provides a more appropriate view of the process, therefore the model of the complete retina is based on those results.

3.4. *2-layer wave instruction: Dynamic outer retina models*

This section presents the basic behavior of two mutually coupled CNN layers. The first emphasizes the second-order temporal dynamics of the retina and the second structure emphasizes the edge enhancement property of the retina. These effects should be combined to make a realistic outer retina model. The previous models in Secs. 3.1–3.3 can be implemented as analogic algorithm using one dynamic layer so they are feasible on the ACE4k CNN-UM chip [Espejo *et al.*, 1996a; Liñan *et al.*, 2000]. The structure in this section has two strongly

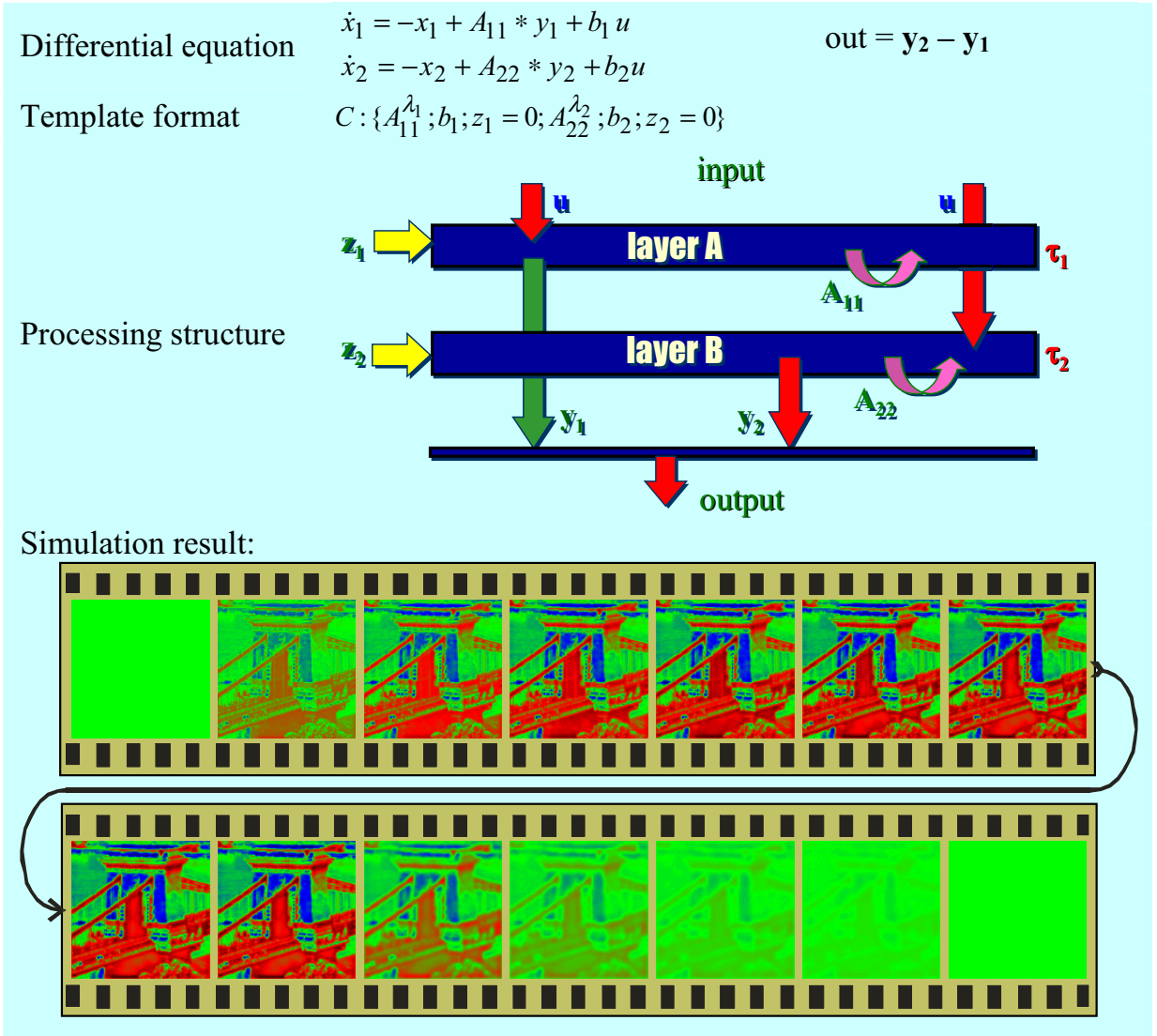


Fig. 6. The CNN description of the difference of Gaussians method. The upper part of the figure shows the differential equations and the CNN template and processing structure, while the last row shows frames from the continuous output (**out**) of the system, the difference between the two dynamic layers.

coupled dynamic layers, therefore their implementation needs second-order CNN base cells, which is currently available only in the CACE1k complex-cell CNN-UM chip [Carmona *et al.*, 2003].

The difference between the two dynamics can be seen by comparing the two simulated outputs. The input is the same continuous flow and color coding as in Fig. 3. Explore the second-order effect in Fig. 7(a): the color of the arch changes from blue to navy and back to cyan or below the arches from red to maroon and back to tomato red. Moreover a strong negative scene, slate blue color between the arches, can be seen after the static image disappeared — it is a well-known retina effect. Figure 7(b) shows the other structure, which enhances the edges.

3.5. *3-layer Outer Plexiform Model (OPL)*

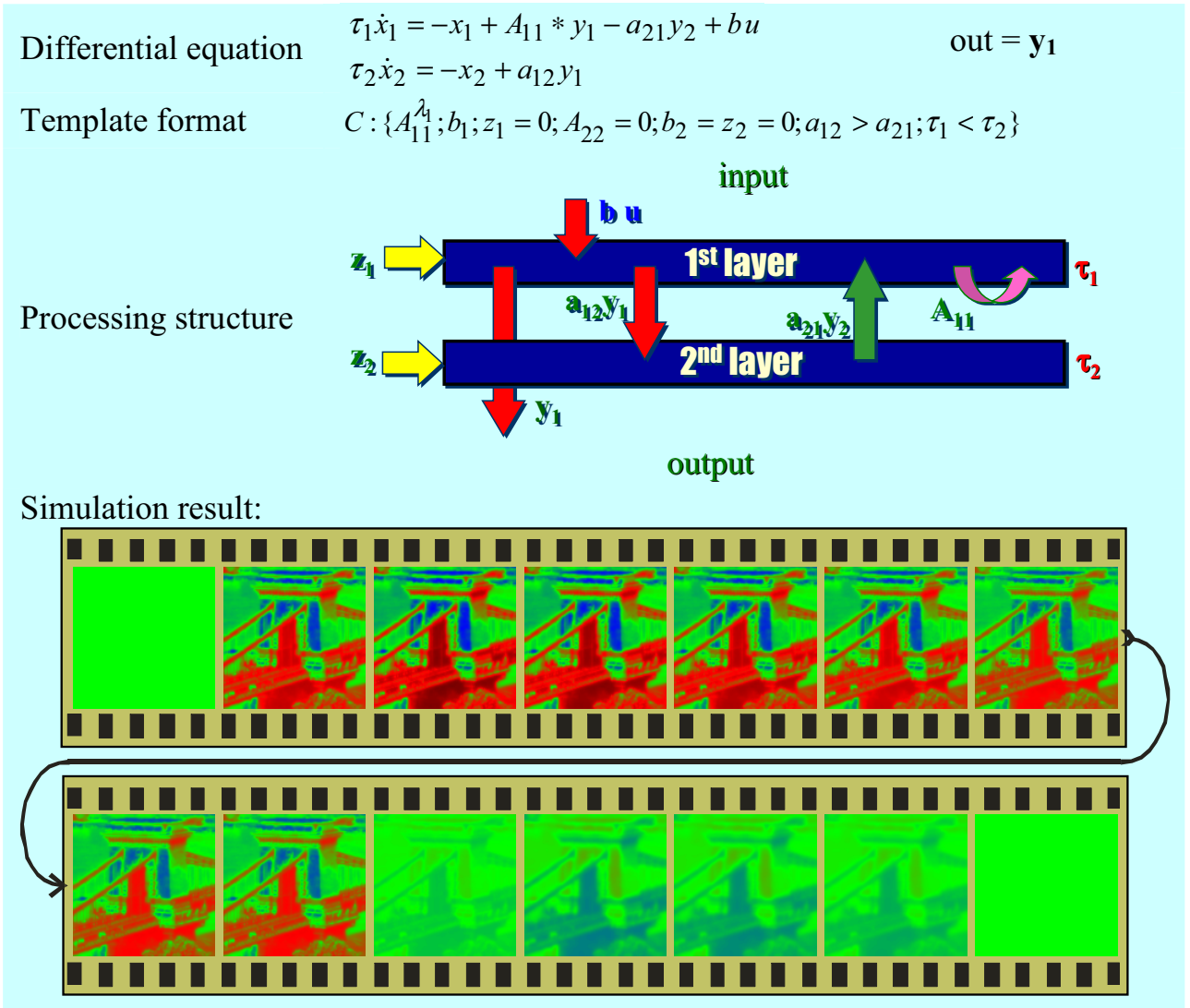
The second-order structures in Sec. 3.4 are integrated into one system to create a more realistic outer retina model. This model can reproduce both the spatial and the temporal properties of the required retina qualities [Balya *et al.*, 2002], therefore it may be considered as the best neuromorphic model of the outer retina among the presented ones. The hardware implementation of the structure

involves two locally coupled two-dimensional layers and a third capacitor to each complex base cell. The modest complexity of the system makes it hardware feasible.

The integration of the effects can be investigated by comparing Figs. 7 and 8. In Fig. 8 the color of the space below the arch changes in time from crimson to black and back to coral as in Fig. 7(a) but the last frame is sharper and in Fig. 7(b). Moreover a strong negative image in dark blue color can be seen after the static image disappeared, like in Fig. 7(a). The color coding and the input can be seen in Fig. 3.

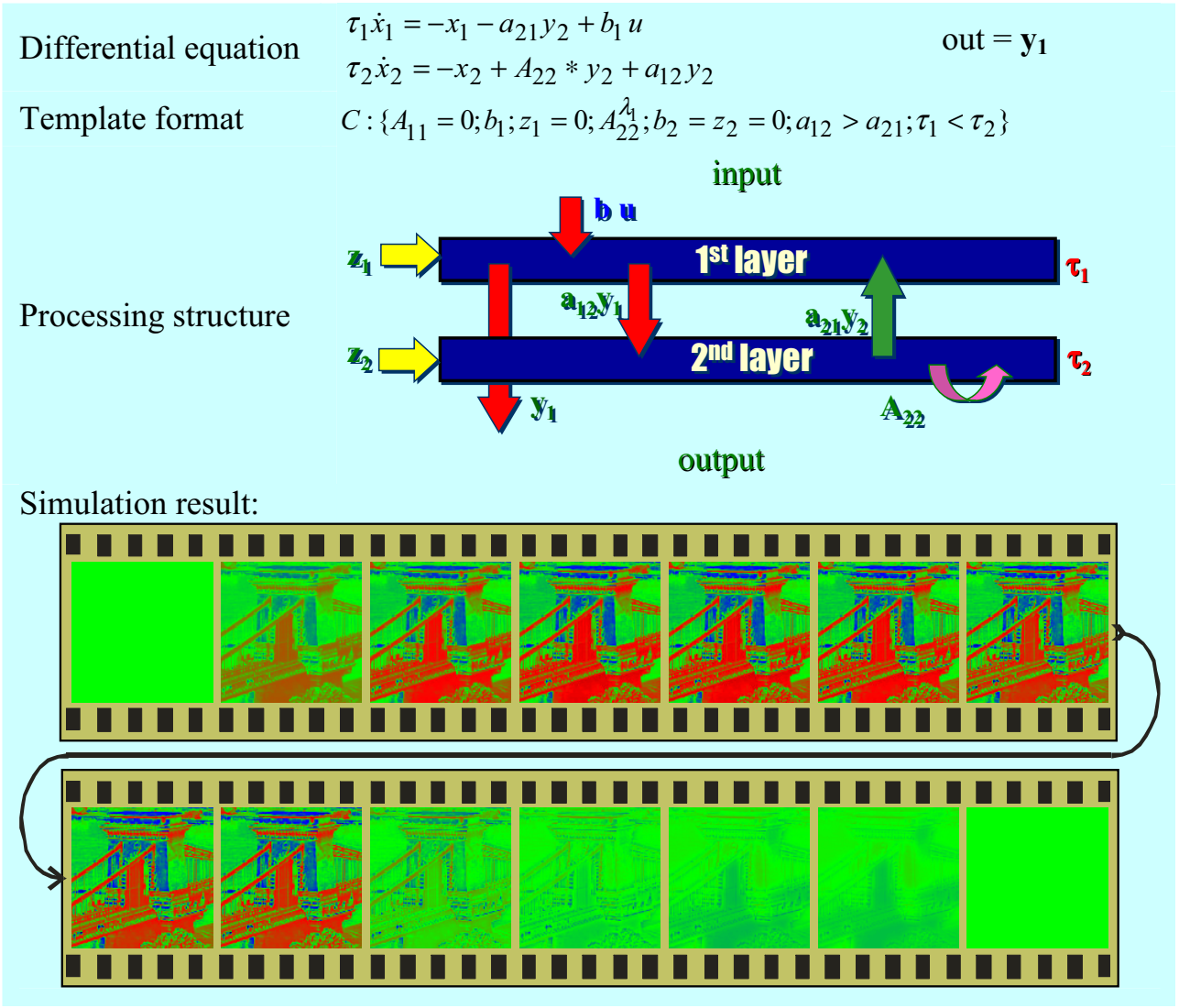
4. The Main Goal: A Complete Retina Model

A good outer retina model is just the beginning to establish a neurobiologically relevant retina model. Our main goal is to design a realistic, qualitatively correct complete mammalian retina model and to implement it on silicon. The neurobiological experiments provide clues to our work. The anatomy provides basic information about the structure of the circuitry. The electro-physiological measurements of the neurons enable us to identify the most important couplings and their strength [Rekeczky *et al.*, 2001; Balya *et al.*, 2002b]. Pharmacology can serve



(a)

Fig. 7. The CNN descriptions of the dynamic outer retina models. (a) Emphasizes the temporal dynamics of the retina, while (b) shows a better edge enhancement. Look at the difference between the frames around the arches.



(b)

Fig. 7. (*Continued*)

as a verification tool by virtually modifying the internal structure of the measured retina [Werblin *et al.*, 1995]. The basic structure of our model is derived from textbooks on the retina morphology [Kandel *et al.*, 2001] and from the latest results related to the structure of the inner retina [Werblin & Roska, 2004]. The model parameters can be estimated from the measured space-time patterns using spatial-temporal analysis and they can be finely tuned by comparing the simulation and measurement data [Roska & Werblin, 2001; Rekeczky *et al.*, 2001].

Following our previous paper [Balya *et al.*, 2002], the retina contains hierarchically organized neuron layers. From a neurobiological point of view several types of neurons are in the retina, but fortunately, they can be categorized into five main groups or classes: photoreceptors (cones, rods), horizontal, bipolar, amacrine, and ganglion cells. The ganglion cells form the optical nerve [Dowling, 1987].

4.1. *One realistic retina channel: The twin wave processing principle*

The light-adapted mammalian retina model consists of the following parts: the outer retina model and the multichannel feature extracting inner retina

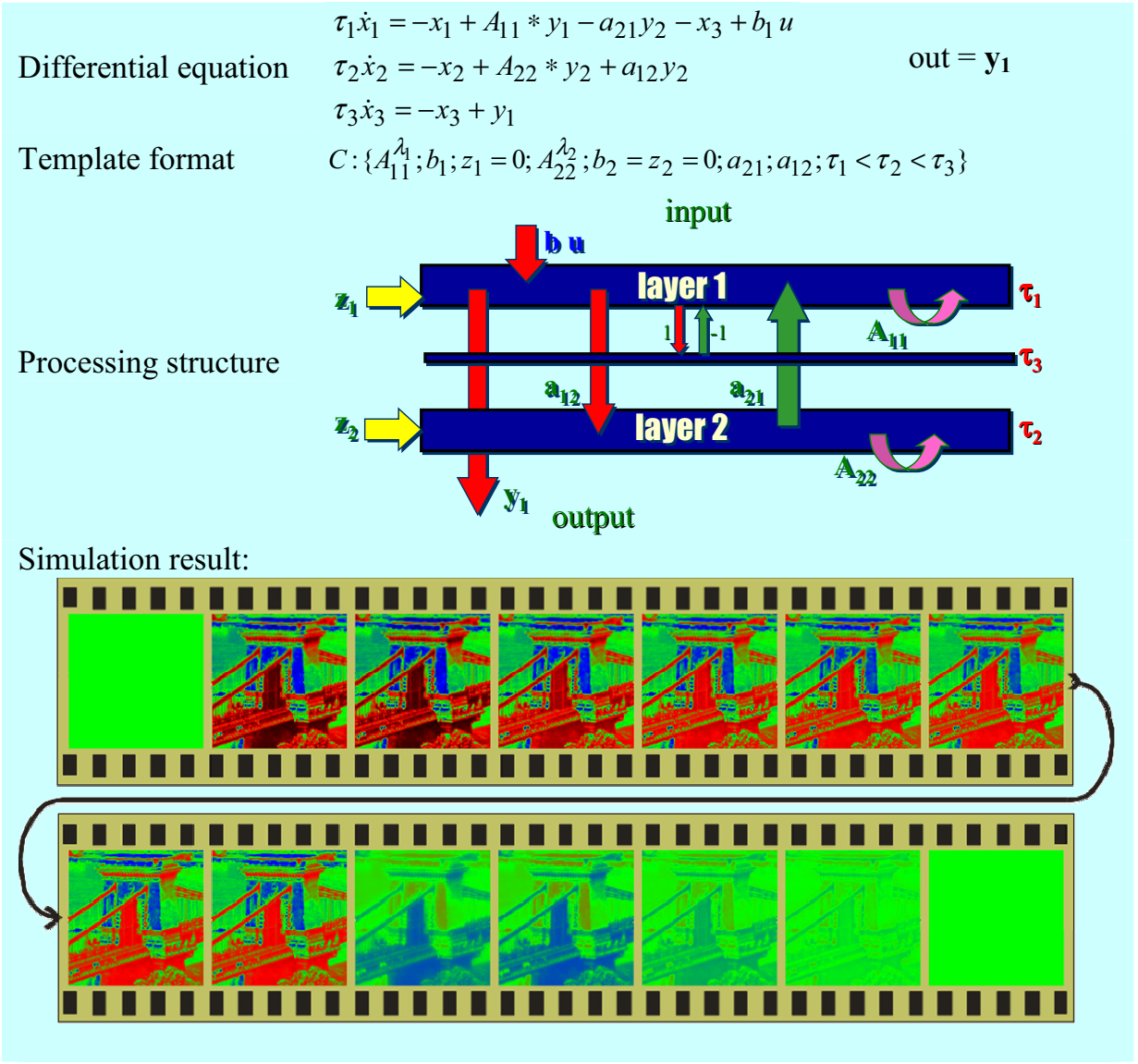


Fig. 8. The CNN description of the outer plexiform layer, which is an outer retina model. The upper part of the figure shows the differential equations and the CNN template and processing structure, while the last row shows the simulated output (**out**) of the system for the continuous input flow illustrated in Fig. 3. Observe the temporal dynamics and the edge enhancement of the model.

model. The basic operation scheme is presented in this issue [Werblin & Roska, 2004]. Each channel consists of three functional blocks: the excitation, the inhibitory subsystem and the combining ganglion cell model. Each of these blocks can be partially or completely implemented on the CACE1k chip [Carmona *et al.*, 2003] as a multilayer template, see Sec. 7 where these templates are given.

The input image flow is separated into two parts, these are processed simultaneously to form the excitation and the inhibition flows, respectively, and then combined together giving the result of the computation. The processing can be seen as a wave computation in the CNN terminology [Chua & Roska, 2002]. The abstract description of the retina channel processing is the following. The first wave instruction divides the continuous input into two flows. The other wave instructions operate on one of these flows to enhance some predefined features. These two qualitatively different processing flows are combined into the last stage: the inhibition flow blocks out some part of the excitation flow to form the output of the system. This is the twin wave processing principle [Roska, 2002], see Fig. 9.

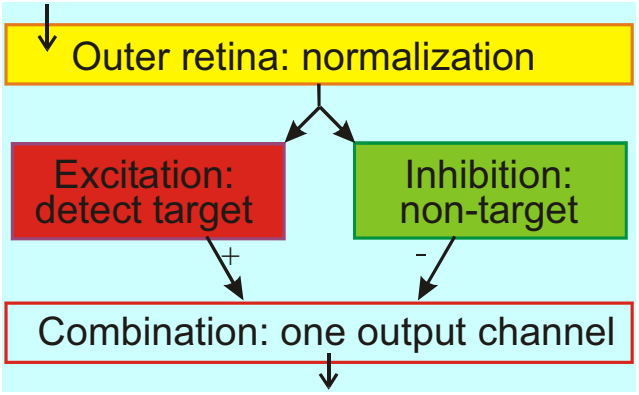


Fig. 9. The twin wave processing principle as the fundamental base of our retina model. The names of the stages refer to a retina model block and to its function.

The general minimal model of one retina channel contains a second-order photoreceptor and horizontal layer to model the outer retina. The names of the CNN layers refer to the name of the modeled neuron class. The main role of the outer retina may be the normalization through local adaptation. During the twin wave processing the output of the outer retina is processed in a nonlinear fashion to form the two different inputs to the ganglion cell layer. A bipolar and amacrine feedback layer generates the excitation, a bipolar and an amacrine feedforward layer forms the inhibitory subsystem. The signs of the inputs of the bipolar layers are different and the outputs of the bipolar layers are nonlinear. The modeled ganglion cell layer subtracts the inhibition from the excitation and computes the dynamic, continuous output of the system.

4.2. *The complete mammalian retina model*

The mammalian retina sends a parallel set of about a dozen different space-time representations of the visual world to the brain. These are the retina channels [Roska & Werblin, 2001]. Each output channel corresponds to one ganglion cell type. It is important that there is no discrete time image — snapshot or frame — in the retina. Each of these representations is generated by the twin wave processing by feature detecting transformations. The identifications of the model elements and the implemented functions in the parallel processing channels are based on retinal anatomy and electro-physiology measurements for a flashed square stimulus. Our complete mammalian retina model reproduces qual-

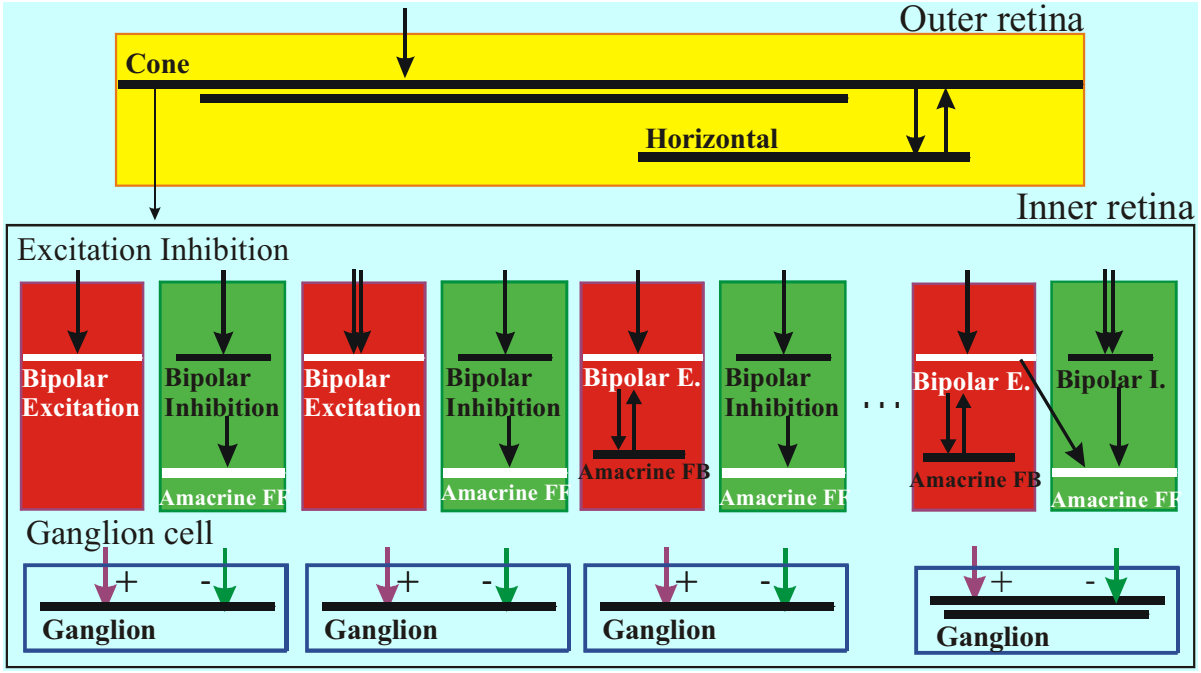


Fig. 10. The processing structure of the complete CNN retina model. The different neuron types in the retina are organized into two-dimensional strata modeled with CNN layers (horizontal lines). A neuron in a given layer affects other neurons through nonlinear synapses (arrows). The layers have different time and space constants. The channels comply with the twin wave principle, compare with Fig. 9.

itatively the same outputs for the flashed square stimulus [Balya *et al.*, 2002].

Each parallel pathway in the retina consists of interacting, hierarchically organized and locally connected diffusion layers with different space and time constants. The modeled parallel pathways differ only in time constants and parameters of the diffusion. The details of the processing structure is given by Werblin and Roska [2003]. Figure 10 gives an overview about our mammalian retina model showing the common outer retina and the parallel twin wave computing blocks, called channels. The output of the complete mammalian retina is not a single image but it consists of several parallel, continuous image flows. Each block of the inner retina may have different complexity. The simplest blocks are depicted on the left-hand side of Fig. 10 and some more complex blocks are shown on the right side. The latter ones incorporate second-order cell models (double horizontal line) or desensitizing synapses (double vertical arrow) [Balya *et al.*, 2002]. Desensitization makes a sensitized or hypersensitive model cell insensitive (synaptically not sensitive) to the sensitizing agent, in this case to the input layer, for details see Sec. 7.4.

5. Model Decomposition to Retinal Units

The developed multilayer CNN mammalian retina model should be decomposed to make it suitable for existing chip technologies. The decomposition takes place in three different domains:

- Temporal decomposition to compute the "frames" iteratively;
- Spatial decomposition of the whole visual field to map a higher resolution scene on the chip; and
- Structural decomposition of the multilayer model into complex-cell blocks.

The target platform of our decomposition is the 3-layer "Retinal Unit" [Werblin *et al.*, 2001]. Its differential equations are Eqs. (7) and (8) and the structure of the unit is shown in Fig. 11. Each layer of the unit consists of first-order RC cells having adjustable time constants (multiple time-scale property). The output characteristics are sigmoid-type functions $\sigma(\cdot)$ and the boundary condition can be constant or zero flux. The neighborhood radii are one for intra-layer connections (e.g. A_{11}) and zero, cell-to-cell links, for inter-layer connections ($A_{12} \rightarrow a_{12}$, $A_{21} \rightarrow a_{21}$) in our present model framework. The inter-layer feedback connections (e.g. a_{21}) can be nonlinear functions $f(\cdot)$ e.g. rectifier.

$$\begin{aligned}
\tau_1 \frac{dx_{1,ij}(t)}{dt} &= -x_{1,ij}(t) + \sum_{kl \in S_1} A_{11,ijkl}\sigma(x_{1,kl}(t)) \\
&\quad + \sum_{kl \in S_1} A_{21,ijkl} f(x_{1,kl}(t)) \\
&\quad - a_{31}x_{3,ij}(t) + b_1 u_{ij} + z_{1,ij} \\
\tau_2 \frac{dx_{2,ij}(t)}{dt} &= -x_{2,ij}(t) + \sum_{kl \in S_1} A_{22,ijkl}\sigma(x_{2,kl}(t)) \\
&\quad + \sum_{kl \in S_1} A_{12,ijkl} f(x_{1,kl}(t)) + b_2 u_{ij} + z_{2,ij} \\
\tau_3 \frac{dx_{3,ij}(t)}{dt} &= -x_{3,ij}(t) + x_{1,ij}(t) + z_{3,ij}
\end{aligned} \tag{7}$$

where,

$$\begin{aligned}
\text{typically} \quad \sigma(x) = f(x) &= \frac{|x+1| + |x-1|}{2} \\
A_{ij} &= \begin{vmatrix} a & 2a & a \\ 2a & c & 2a \\ a & 2a & a \end{vmatrix}, \\
&\text{if (intra-layer: } i = j) \; c = 1 - 12a - b_i
\end{aligned} \tag{8}$$

5.1. *Structural decomposition*

The developed complete mammalian retina model is a multilayer CNN model [Balya *et al.*, 2002]. It consists of several coupled CNN layers that can fortunately be decomposed into smaller blocks. The first step of the decomposition is to determine the tightly coupled blocks. They have both feed-forward and feedback connections but between two such blocks only feed-forward links exist and these connections can be cut off. The results are Retinal Units, simple complex-cell blocks, as shown in Fig. 11. The blocks of the decomposed model are the outer retina, the excitation blocks and the ganglion-complex blocks. The latter is the combination of the inhibitory layer and the ganglion layer. The excitation block can be either second-order or desensitized, see Secs. 7.3 and 7.4, respectively. The parameters of the blocks can be straightforwardly derived from the existing model. The pseudo-code of the decomposed retina model algorithm and the interfaces between the blocks are presented in Table 1. The blocks should be computed parallel (e.g. as threads). The operators are spatial-temporal wave instructions on the

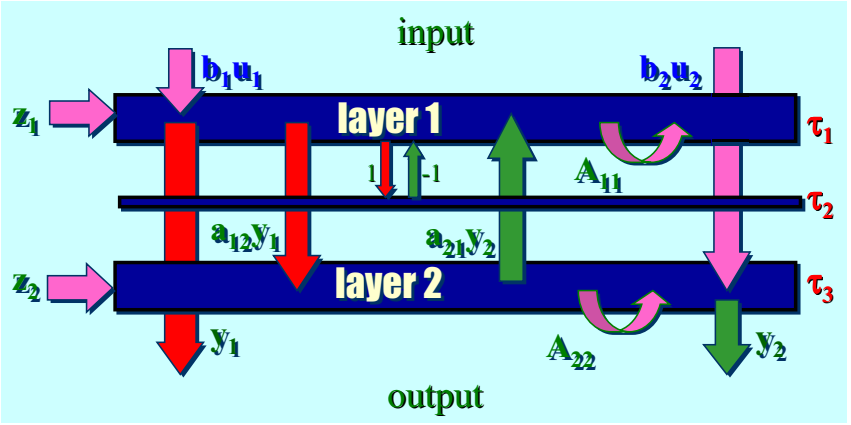


Fig. 11. The structure of the Retinal Unit. The horizontal bars represent two-dimensional CNN layers and the arrows represent the effects between the locally connected cells.

Table 1. The result of the structural decomposition of the complete mammalian retina model is an analogic algorithm.

Compute the parallel channels
in: *stimulus*; out: *retina channels*

- **comp_dynamics** outer retina block
  in1: *stimulus*; out: *oplout*
- **parallel for each** bipolar block
  comp_dynamics ith bipolar block in1: *oplout*
  comp_nonlin function out: *excitation*
- **parallel for each** ganglion block
  sum relevant *excitations*
  if inhibition is complex **comp_dynamics** inhibition
  in1: *excitation*; out: *inhibition*
  comp_dynamics ith ganglion block
  in1: ith *excitation* - *inhibition* in2: jth *excitation*
  comp_nonlin function out: ganglion channel

Retinal Unit structure and the outputs are the sampled frames of the transient and *not* the steady state of the system.

The inhibition is complex if it needs more than one dynamic layer. The **comp_nonlin** operation uses the following settings. It implements a rectifier or Heavyside-function: the output **y** is equal to zero under the threshold (**th**). The corresponding CNN template is described in Eq. (9).

$$C: \left\{ \begin{array}{c} A_{11} = A_{22} = 0;\ g_{31} = 0,\ a_{12} = 1,\ a_{21} = 0;\ z_1 = 1 - th;\ z_2 = th - 1 \\ b_1 = 1;\ b_2 = 0;\ \tau_1 = \tau_2 \end{array} \right\} \text{out} = x_2 \tag{9}$$

5.2. *Spatial decomposition*

The resolution of the complete visual field is usually larger than the actual chip, in other words, the input is larger than the processing area. The classical solution says that the input should be split into overlapping areas. The size of the overlapping bands can be computed if the processing lasts for a finite time, because the number of the affected cells depends on the diffusion constant, only. The processing time and the coupling (λ) rule the number of the potentially influenced cells.

Table 2. Algorithm skeleton to compute one channel of the mammalian retina model.

```
Compute a video clip for one channel

 1. Load the next frame
 2. Repeat twice:
 3.   Execute: Outer retina template (input: the frame)
 4.   Repeat twice:
 5.     Execute: Bipolar #1 template (both input: outer retina)
 6.     Execute: Nonlinear template
 7.     Execute: Bipolar #2 template (both input: outer retina)
 8.     Execute: Nonlinear template
 9.     Execute: Output template (inputs: the two bipolar output)
10.     Threshold template gives the result
11. Send back the result and jump to 1.
```

It is worth mentioning that the fovea size of the biological mammalian retina (about 50,000 neurons) will be attainable with the current VLSI technology in the near future, thus a spatial decomposition is not needed.

5.3. *Temporal decomposition*

The real world gives continuous input to the retina, but the artificial test stimulus, a video flow, is frame based. In the latter case the temporal decomposition is straightforward; the consecutive frames can be computed independently if the output sampling is correct. During the video stimulus a video-flow is projected to the retina. A frame is shown for a constant period of time until the next frame appears, therefore the temporal signal is band limited. If the output sampling is twice the input frame rate and the system is linear the Shannon sampling theorem is satisfied. This suggests that the double frame-rate output sampling is sufficient for the transient computation in the Retinal Unit case as well. The used CACE1k prototype complex-cell chip does not have any optical input therefore the input of the algorithm is a frame based video flow, see Table 2.

In the first case, when the system perceives the visual world, the decomposition should externally sample the input. The sampling frequency would be unacceptably high because (i) the bandwidth of the input signal may be enormous and (ii) some special transient cells can react to a fast movement. Two solutions seem reasonable: one is to sample the world with a computed sampling rate depending on the fastest neuron in the mammalian retina and the input signal or to use an adaptive subsampling method. The latter one could be implemented using special purpose LAMs (local analog memories on the chip), where the important events are stored and later the whole transient can be approximated from them. The local reconstruction can be done via adaptive CNN-UM [Roska, 1999].

6. The CACE1k Chip Architecture

The architecture of the chip follows the design of the standard one layer CNN-UM [Chua & Roska, 2002] chip: ACE4k [Espejo *et al.*, 1996a; Liñan *et al.*, 2000], but its first-order cell core is replaced by a second-order one, see Fig. 12.

The dynamic evolution law of the complex cell CNN is the system of differential equations as follows:

$$\begin{aligned}
\tau_1 \frac{dx_{1,ij}(t)}{dt} &= -h(x_{1,ij}(t)) + \sum_{kl \in S_1} A_{11,ijkl} x_{1,kl}(t) \\
&\quad + a_{21} x_{2,ij}(t) + b_1 u_{ij} + z_{1,ij} \\
\tau_2 \frac{dx_{2,ij}(t)}{dt} &= -h(x_{2,ij}(t)) + \sum_{kl \in S_1} A_{22,ijkl} x_{2,kl}(t) \\
&\quad + a_{12} x_{1,ij}(t) + b_2 u_{ij} + z_{2,ij} \\
1 \le i \le M\,, &\quad 1 \le j \le M\,, \quad 0 \le |u_{ij}| \le 1, \\
&\text{FSR} : 0 \le |x_{ij}(t \ge 0)| \le 1
\end{aligned} \tag{10}$$

where,

$$h(x) = \lim_{m \to \infty} \begin{cases} m(x-1) + 1 \approx +\infty, & \text{if } x > 1 \\ x, & \text{if } |x| \le 1 \\ m(x+1) - 1 \approx -\infty, & \text{if } x < -1 \end{cases} \tag{11}$$

The hard-limiter function $h(\cdot)$ regulates the state so that it stays within the $+1 \cdots -1$ interval. Variables

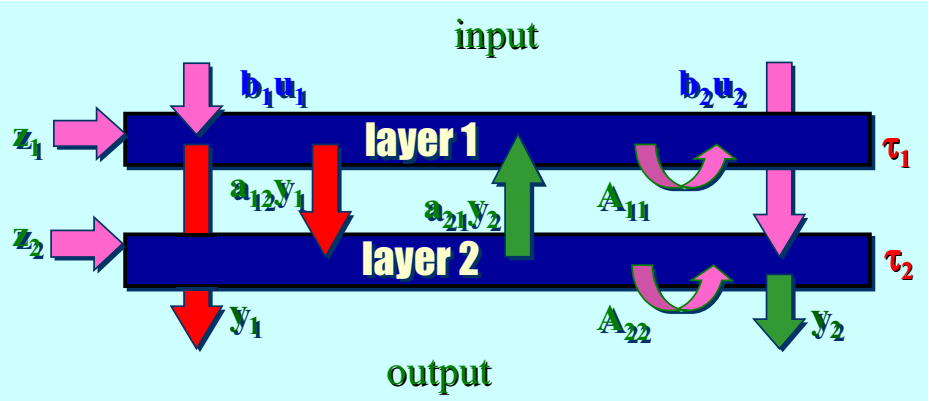


Fig. 12. The processing structure of the complex-cell cellular neural network chip.

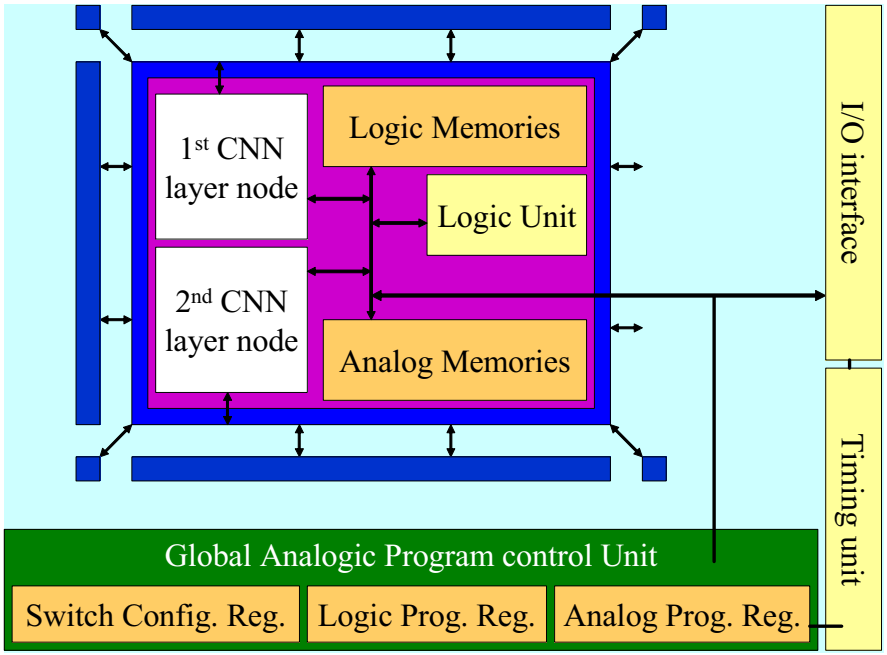


Fig. 13. Conceptual diagram of the complex-cell CNN units and its universal machine environment. The different connecting lines and buses are represented by arrows.

u_1, u_2 are the independent inputs, b_1, b_2 are their weight factors, respectively; $z_{1,ij}$, $z_{2,ij}$ are space variant bias maps. Variables x_1, x_2 denote the state variables of the layers. Each x_{ij} corresponds to one cell; it is one pixel of a M by N image if we consider the system's output as a picture. A_{11}, A_{22} are the weights of the intra-layer couplings, a_{12}, a_{21} are the inter-layer weights. It utilizes the so-called *full signal range* model (FSR), where the voltage of the state variable is always the same as the output $\mathbf{y}$ [Espejo *et al.*, 1996b].

The prototype CACE1k chip [Carmona *et al.*, 2002] consists of an analog programmable array processor of 32×32 identical cells, surrounded by the boundary conditions of the CNN dynamics. There is also an I/O interface, a timing and control unit and a program memory, see Fig. 13. The analog instructions, template values and reference signals need to be transmitted to every cell in the network in the form of analog voltages. Finally, the timing unit is made of an internal clock/counter that generate the internal signals which enable the processes

of image up/downloading and program memory access.

The elementary processor of the complex-cell CNN visual microprocessor chip includes two coupled continuous-time CNN cores belonging to each of the two different layers of the network. The synaptic connections between processing elements of the same or different layers are represented by arrows in Fig. 13. The base unit also contains a programmable local logic unit (LLU) and local analog and logic memories (LAMs and LLMs) to store intermediate results [Roska & Chua, 1993]. All the blocks in the cell communicate via an intra-cell data bus, which is multiplexed to the array I/O interface. Control bits and switch configuration are passed to the cell directly from the global programming unit.

Each CNN node receives contributions from the rest of the processing nodes in the neighborhood which are summed and integrated in the state capacitor. The time constants of the two layers are different. The first layer has a scalable time constant ($\alpha\tau_{\text{cnn}}$, where α is an integer between 1 and 16), controlled by the appropriate binary code, while the second layer has a fixed time constant (τ_{cnn}). The evolution of the state variable is also driven by self-feedback and by the feed-forward action of the stored input and bias patterns. There is a voltage limiter for implementing the full signal range property of the implemented CNN-UM. The state variable is transmitted to the synaptic blocks, in the periphery of the cell, where weighted contributions to the neighbors are generated. Initialization of the state $X(0)$, input U and bias Z voltages are made through a mesh of multiplexing analog switches that connect to the cell's internal data bus.

The prototype chip has been designed and fabricated in a 0.5 μm single-poly triple-metal CMOS technology. Its dimensions are 9.27×8.45 mm^2. The cell density is 29.24 cells/mm^2. The time constant is around $\tau_{\text{cnn}} \approx 100$ ns [Carmona *et al.*, 2002]. The programmable dynamics of the chip permit the observation of different phenomena of the complex wave instructions. Table 3 summarizes the most relevant data of the prototype chip.

A professional development environment, called Aladdin Professional [Zarándy *et al.*, 2003], is built up around the previous single-layer CNN universal machine vision chip (ACE4k). The software environment supports several different methods to reach the potential of the chip from a high-level language to low-level direct commands. The CACE1k chip is integrated into this environment [Petrás *et al.*, 2003]. The hardware interface between the PC and the CACE1k chip is illustrated

Table 3. Specification of the scalable time-constant complex-cell CNN-UM core.

Technology	0.5 µm CMOS 1-P 3-M
Number of cells	32 x 32
Die area (+pads)	8.77+0.5 mm x 7.94+0.5 mm.
Array area	5.98 x 5.93 mm^2.
Power supply voltage	3.3V (Logic "0"/"1" : 0/3.3V)
CNN time constant	< 100ns

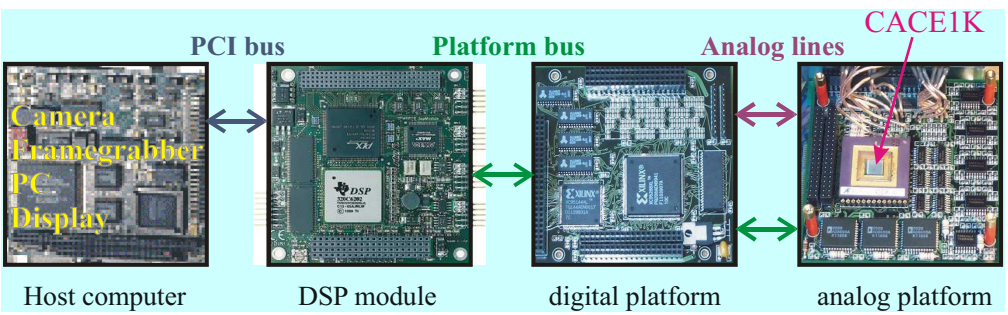


Fig. 14. The complex-cell CNN-UM chip (CACE1k) and its prototyping system.

in Fig. 14. Firstly the analogic program is downloaded to the Texas digital signal processor (DSP module) through the standard PCI bus. The DSP schedules the program and gives the interface between the PC and the platform holding the chip. Secondly the digital platform generates the analog signals to the CACE1k chip and contains the external digital memory. Finally the analog platform receives the digital control signals and the analog data for the direct control of the complex-cell CNN-UM visual microprocessor.

7. Measurements of the Decomposed Retina Model

The mammalian retina model has been presented in Sec. 4, the decomposition principles in Sec. 5. This section deals with the implementation of the decomposed model on the complex-cell CNN-UM, which is outlined in the previous section. The decomposed retina model is implemented as an analogic CNN algorithm given in Table 2. The steps of the algorithm are complex-cell template executions, each step implements a single typical retina model block. These are the outer retina, different bipolar units and the output blocks. Two typical bipolar blocks are examined: a second-order in Sec. 7.3 and a desensitizing one in Sec. 7.4. The processing structure of one retina channel is displayed in Fig. 15. It is worth mentioning that the inhibitory subsystem is indicated as one block in Sec. 4, but during the structural decomposition in Sec. 5.1 it is separated into two parts. The bipolar model forms another bipolar block, while the amacrine feed-forward layer is implemented together with the corresponding ganglion layer.

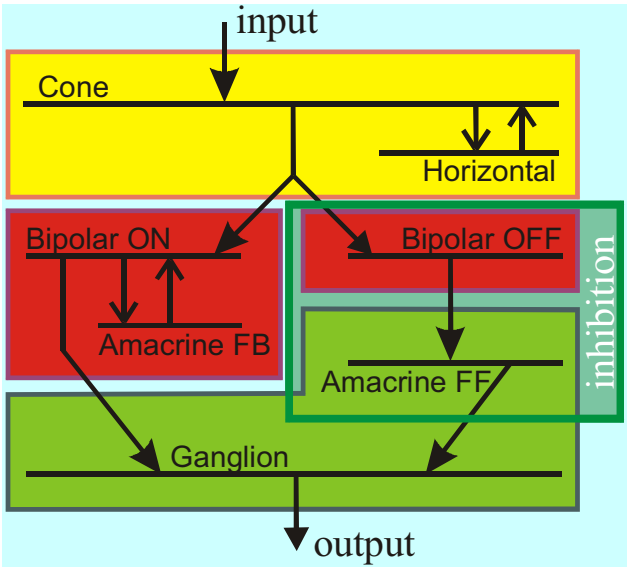


Fig. 15. One decomposed retina channel. The double line represents the second-order layer, the double arrow indicates the desensitized synapse. The result of the decomposition is a series of template operations, the main steps are represented by boxes.

The CACE1k chip can be programmed using an assembler language, which contains additional high-level instructions, including image loading, frame grabber instructions and built-in image processing functions. After the computation is done, the results are read out from the chip's local analog memories and are displayed. The elementary program of the chip is the template. It contains the weight factors of the coupling between the cells and weights for the input and the bias map.

$$\mathbf{A}_{11} = \begin{bmatrix} a^1_{-1,-1} & a^1_{-1,0} & a^1_{-1,1} \\ a^1_{0,-1} & a^1_{0,0} & a^1_{0,1} \\ a^1_{1,-1} & a^1_{1,0} & a^1_{1,1} \end{bmatrix};$$

$$\mathbf{A}_{22} = \begin{bmatrix} a^2_{-1,-1} & a^2_{-1,0} & a^2_{-1,1} \\ a^2_{0,-1} & a^2_{0,0} & a^2_{0,1} \\ a^2_{1,-1} & a^2_{1,0} & a^2_{1,1} \end{bmatrix}; \quad a_{21}; \quad a_{12}$$

$$b_1; \quad b_2; \quad z_1; \quad z_2; \quad \boldsymbol{\tau}_1 : \boldsymbol{\tau}_2 = \tau : 1; \quad \tau = 1 \ldots 16 \tag{12}$$

The operation of the array computer is completely determined by the 25 template values, the initial states and boundary conditions. In Eq. (12) the $\mathbf{A}_{11}$ and $\mathbf{A}_{22}$ matrices include the weights of the intra layer connections of the slower and the faster layer, respectively. The strength of the influence of the second layer on the first is controlled by a_{21}, and a_{12} stands for the reverse case. Symbols b_1, b_2, z_1 and z_2 are the weights of the independent inputs and the space variant bias maps. The ratio of the time constants of the two CNN layers are controlled by τ_1, while τ_2 is fixed. An analogic algorithm is made up of template executions, logic instructions and spatial arithmetic operations.

Only the diffusion template is applied in the computation of each of the retina dynamic blocks, therefore the diffusion constant (λ) completely determines the values of the template A, cf. Eqs. (13) and (6).

$$A = \lambda^2 \begin{bmatrix} 0.25 & 0.50 & 0.25 \\ 0.50 & -3 & 0.50 \\ 0.25 & 0.50 & 0.25 \end{bmatrix}, \quad |\lambda| < 1 \tag{13}$$

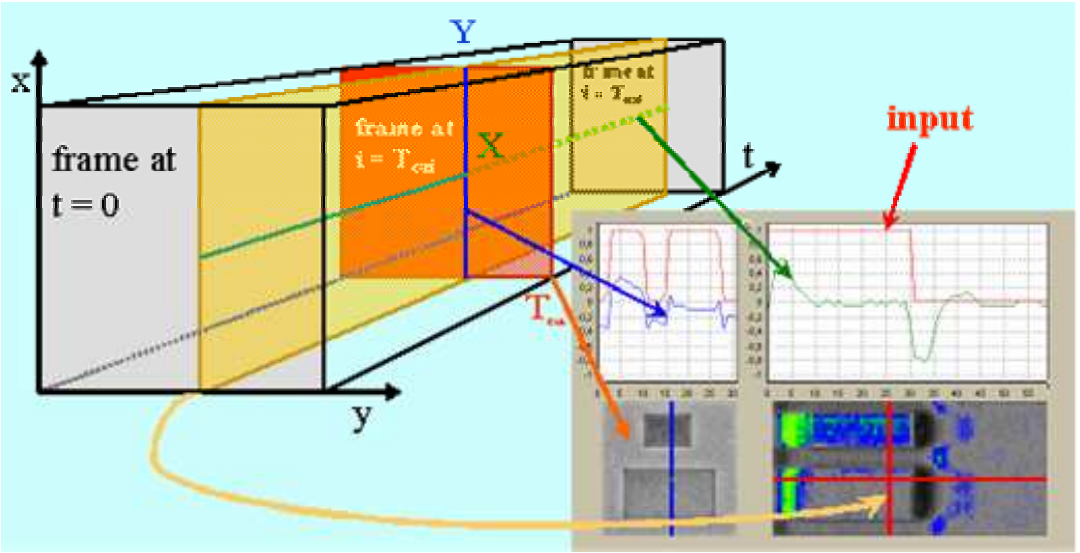


Fig. 16. The derivation of the common diagrams from the space-time measurement flow. The prism on the left-hand side represents the space time flow and the resulting cuts are the different diagrams on the right-hand side.

In Figs. 18(a), 19(b) and 20(b) the display panel is composed of four areas which are organized into rows and columns. The drawing areas contain different views of the outputs of the second, faster layer of the CACE1k chip. Figure 16 explains the extraction method, which produces the output diagrams from the measured space-time flow. The fields in the left column are *amplitude-space* diagrams. The right-hand column of the panel contains *space-time* diagrams, where one column is taken and the samples of that line of cells are depicted spatially at consecutive time instants. The plots in the upper row are cross-sections of the diagrams of the bottom row superposed on the stimulus displayed in red. The horizontal line in the bottom-right and the vertical line in the bottom-left shows the selected spatial position. The vertical line in the bottom-right shows the time instant, at which the images displayed were sampled and displayed on the left-hand side and the vertical line in the bottom-left denotes the position of the sampled column displayed on the right-hand side.

7.1. *Scaling*

The CACE1k chip has some limitations, which is due to the nature of the silicon implementation. Only one of the time-constants can be adjusted. The absolute weight of the connections has an upper limit therefore the spatial-coupling (λ) is limited. In simulation the execution time of the system is uniform: 1 unit simulation time corresponds to 1 ms measurement time. In the implementation the transient is stopped after a precomputed time but this timing is not directly connected to the measurement time. This additional degree of freedom enables us to overcome the limitation of the hardware.

The structural decomposition has to store the suspended transient state. The temporal decomposition demands the higher output-sampling rate. For example, if the input frame rate is 25 frames per second (fps), then the output of the outer retina is 50 fps and the output of the bipolar block should be 100 fps.

The main idea of the scaling is to compensate the changes of the space and time constants with the running time of the template and so different chip transient times can correspond to the same measurement time. The absolute time constants of the layers in a block are not important. Only their ratio plays a role in the processing. If the feasible time-constant is smaller than the prescribed value (this is the typical case) then the execution time should be decreased.

Each layer has a specific space-constant, which is implemented by a diffusion template. This is the only intra-layer coupling in these blocks. The effect of the diffusion depends on both the values of the template, determined by λ, and the execution time. The longer it runs and/or the bigger is λ the more blurry is the output. If the λ parameter defines

Table 4. The steps of the parameter scaling of each block and an example on the right-hand side.

Algorithm	Example
Compute the simulated **execution time** from the output sampling rate	t=20 τ_1=210 λ_1=4 τ_2=30 λ_2=8
Select the bigger space-constant: set it to the maximum **scale down** the other space-constant **increase** the execution time	τ_2=30 λ_2=1 τ_1=210 λ_1=0.5 t = 160
Select the smaller time-constant and link to the fixed-timing layer **scale down** the other time-constant **decrease** the execution time	τ_2=1 λ_2=1 τ_1=7 λ_1=0.5 t = 5

an exceedingly large (not programmable) template value, then an appropriate smaller λ value should be chosen and decrease the input coupling. Correspondingly the execution time must be increased in order to maintain the same effect. However, this affects the other layer as well, therefore the space-constant of the other layers should also be modified keeping the ratio between the λ's constant. To sum up the required modifications, the parameters of each block should be modified as described in Table 4.

7.2. *Outer retina measurement*

In the modeling process we developed different types of ganglion responses using the same outer retina model.

The outer retina is responsible for enhancing the edge information of the image. Figures 18(a) and 18(b) illustrate the CACE1k computation of the outer retina block. The input is a smaller and a larger black rectangle in a gray field, as shown in the last row in Fig. 17. The input frame is

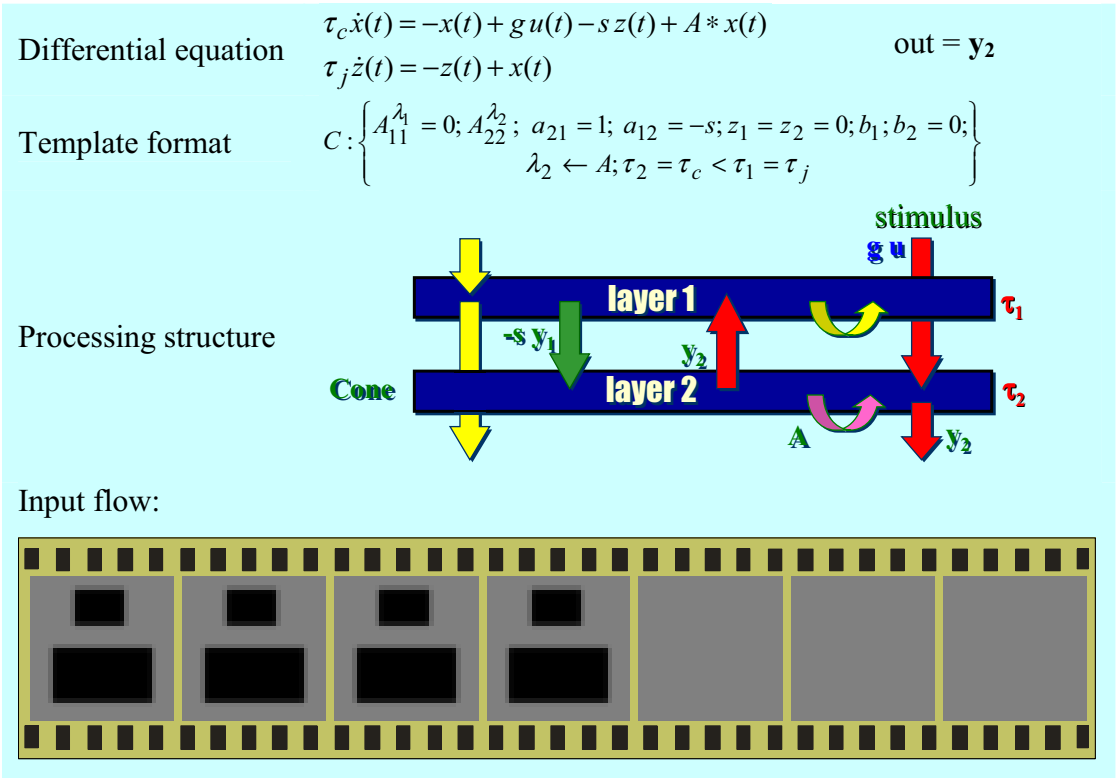
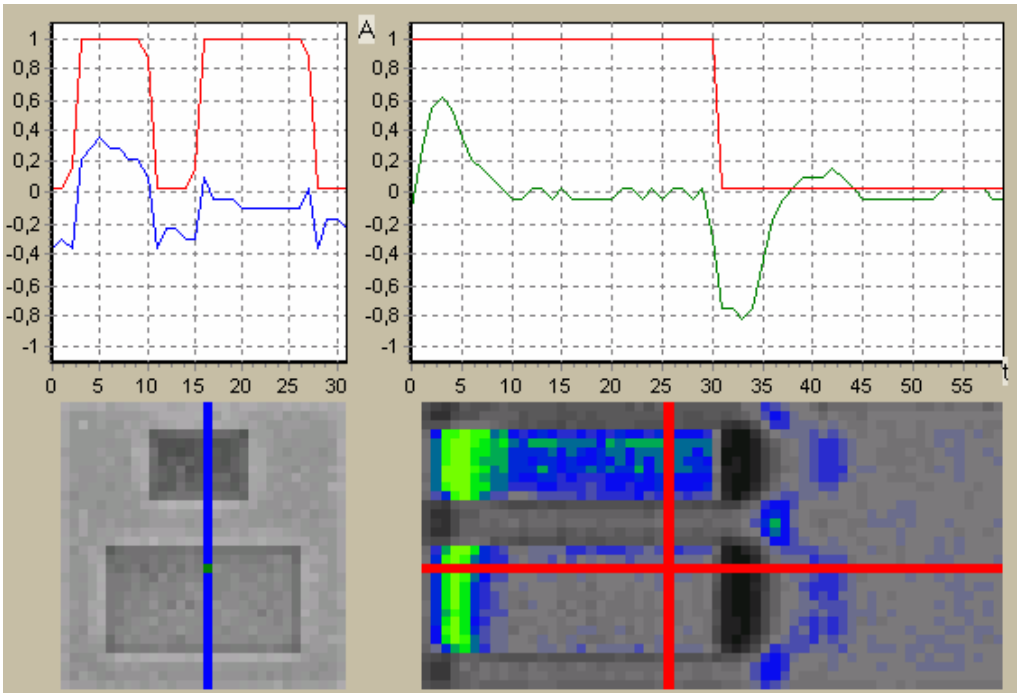
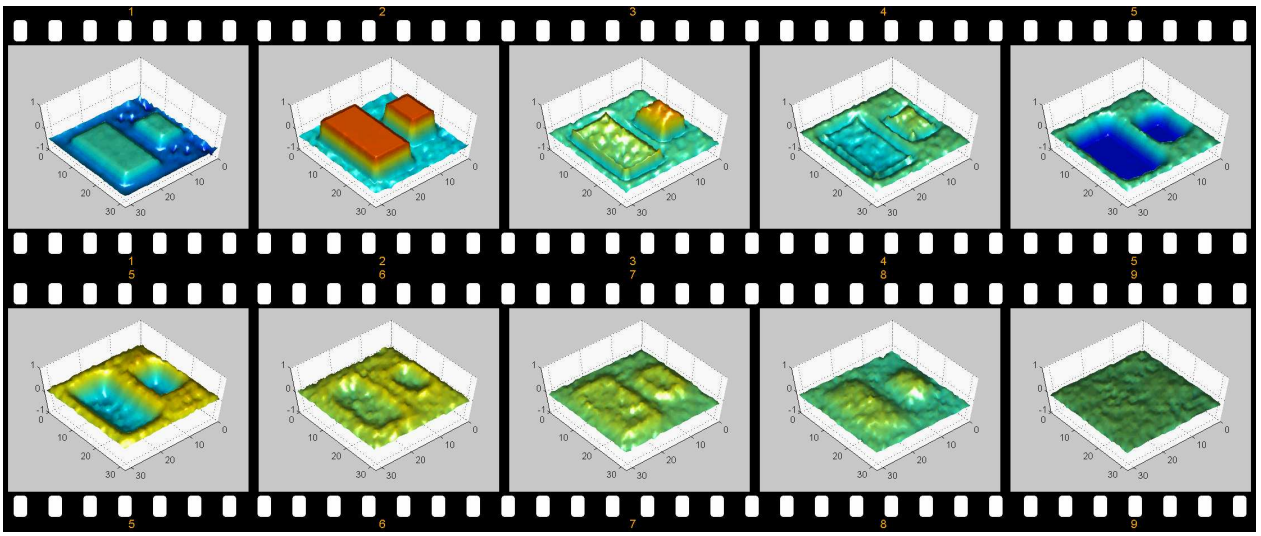


Fig. 17. Programming the CACE1k chip for outer retina model computation.



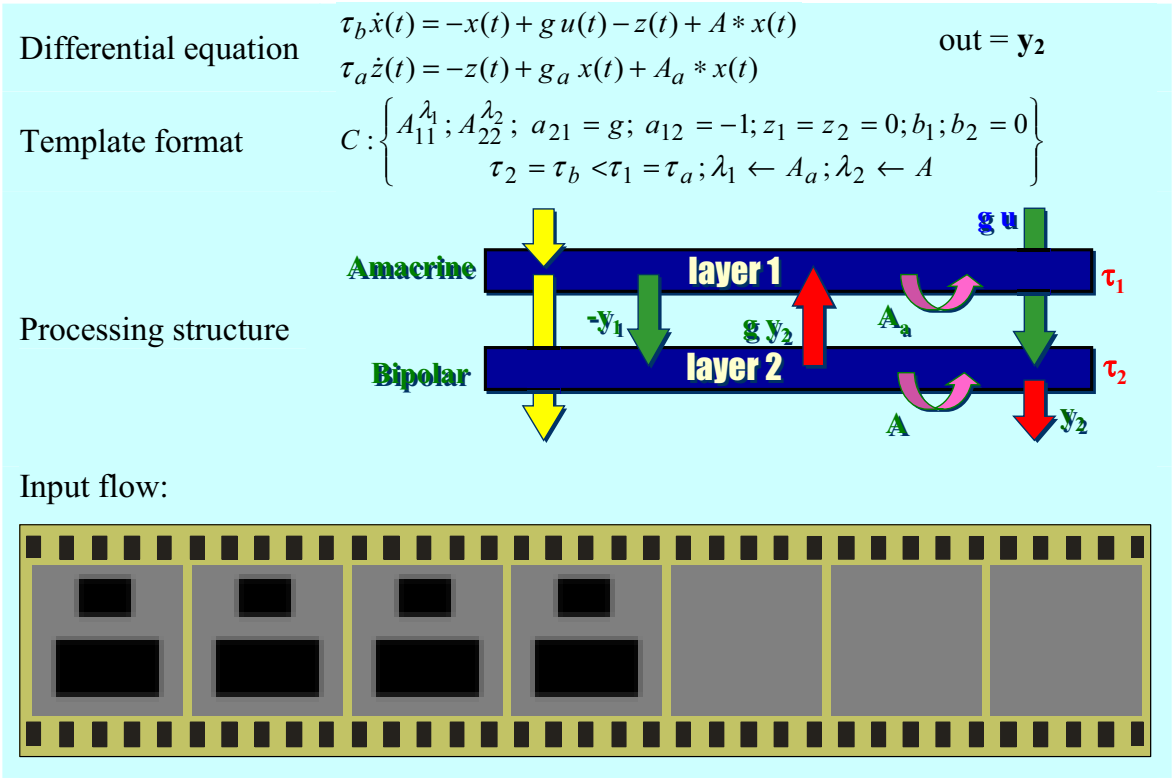
(a)



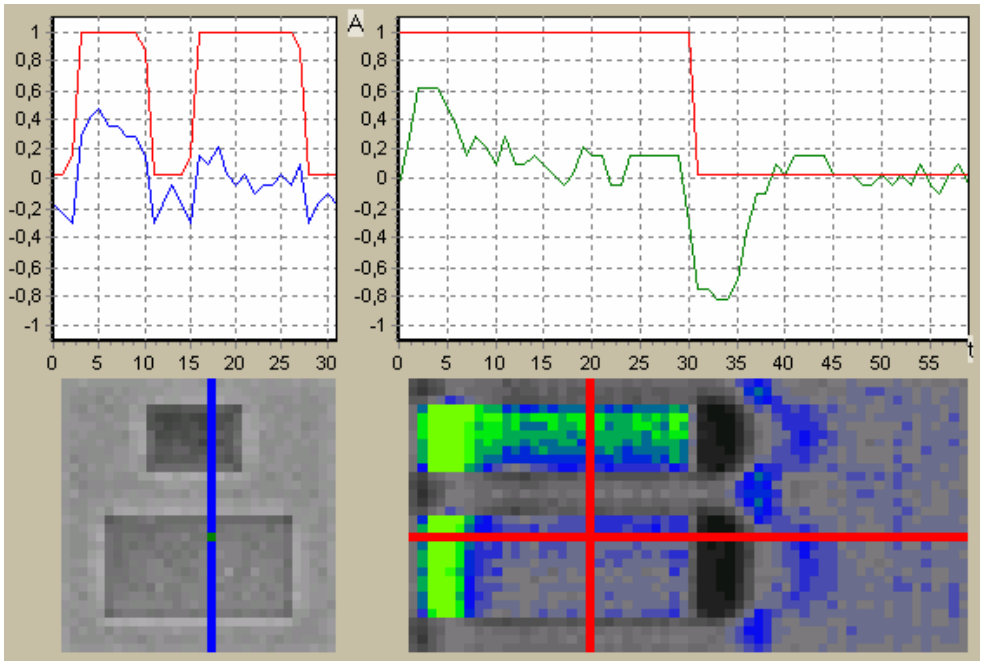
(b)

Fig. 18. (a) CACE1k measurement result of the outer retina block of the retina model in Fig. 17. The transient response of the system is displayed using the space-time representation described in Fig. 16. (b) CACE1k measurement of the outer retina model snapshots of state $x_{ij}(t)$ in two dimensions.

displayed for a certain time, then removed (erased to zero). At the beginning of the flash-up of the input we get an activity peak that decays soon, but the final (steady) output depends on the size of the black square. If the flashed square is small, a certain level of activity is preserved inside the object until the removal of the input. Otherwise, in the case of a larger rectangle, only the edges of the object are preserved. When the input is removed deactivation follows. However, excitation occurs along the edges. These correspond to the mammalian retina measurements [Roska & Werblin, 2001].



(a)



(b)

Fig. 19. (a) Programming the CACE1k chip for second-order bipolar model computation. (b) CACE1k measurement result of a typical second-order bipolar block of the retina model. The transient response of the system is displayed using the space-time representation described in Fig. 16.

7.3. *Second-order bipolar block*

The template in Fig. 19(a) demonstrates the operation of a second-order excitation block. It contains a bipolar layer with one strongly coupled amacrine feedback layer. If different temporal characteristics are needed, e.g. a larger overshoot ratio, the desensitized-block should be used, see Sec. 7.4. At the beginning of the flash-up of the input we get an activity peak that decays.

7.4. *The desensitizing effect*

Some synapses possess a desensitivity feature. This means that the synapse weight is decreasing over time hence the synapse becomes less effective. Even if the input flow is the same, the effect of the input is smaller or even zero as time evolves. The following structure is especially useful to model this phenomenon. The first layer plays the role of the special synapse, which desensitizes.

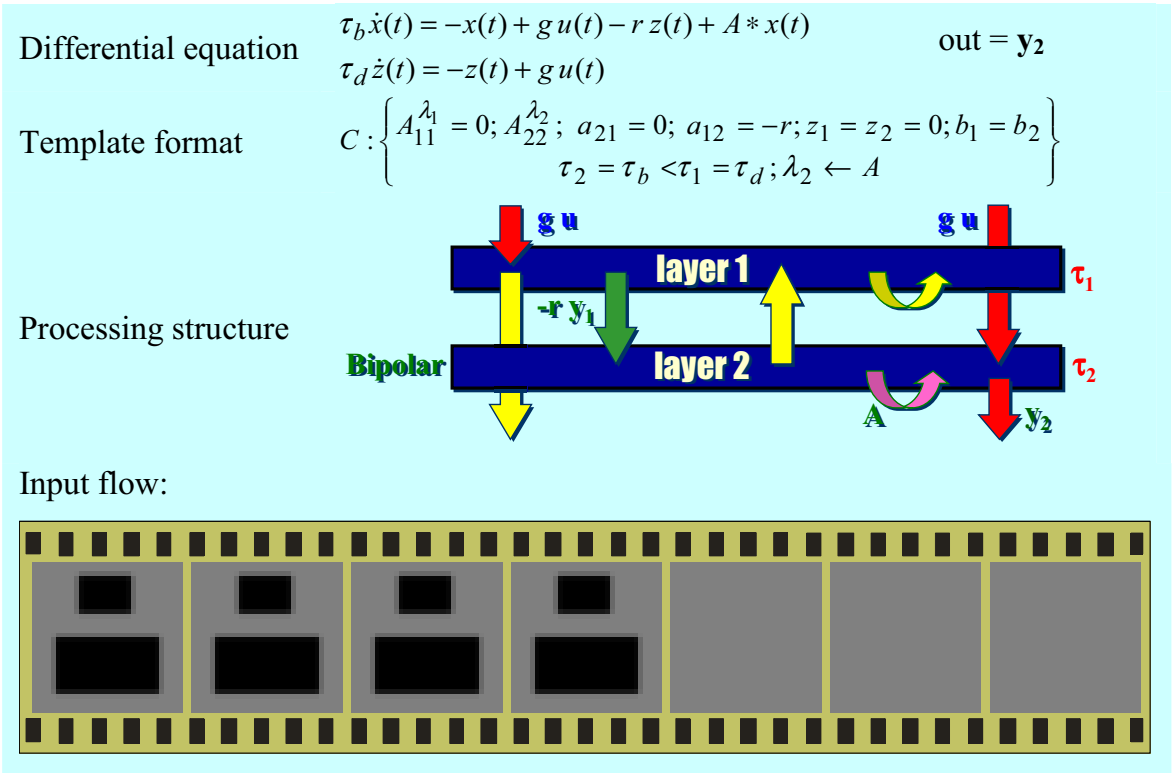
The length of the two active periods is influenced by the time constant of the slower layer τ_1 (see Fig. 20). For bigger τ_1, the longer is the response.

7.5. *Ganglion block*

The ganglion block models a single ganglion cell channel and the simple inhibition computation. This subtracts the inhibitory path from the excitatory path. The first layer represents the inhibitory path, computes the inhibition from a previously computed bipolar block, see Fig. 16. The second layer stands for the ganglion output.

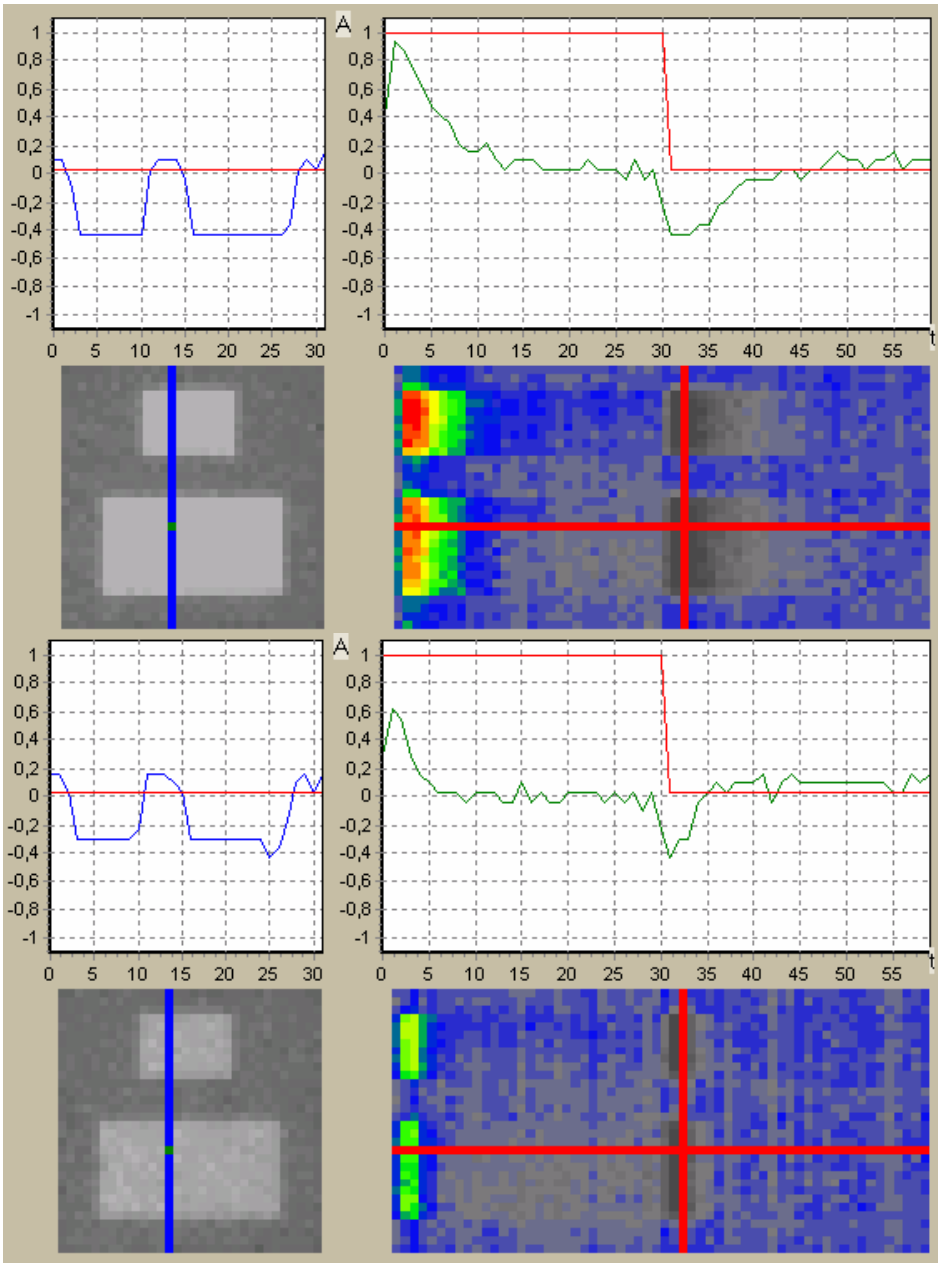
8. Conclusions

The analogic CNN algorithm outlined in the present paper mimics the mammalian retina from the photoreceptors to the ganglion cells (the output of the retina). The structure of the model is



(a)

Fig. 20. (a) Programming the CACE1k chip for desensitizing bipolar model computation. The first layer models the special receptor, which is desensitizing. (b) CACE1k measurement of the desensitizing effect with two different τ_1 parameters: the upper part of image $\tau_1 = 13$ and the lower part of image $\tau_1 = 7$. The transient response of the system is displayed using the space-time representation described in Fig. 16. The output is a transient detection: the black input objects appear and disappear when the scene is changing.



(b)

Fig. 20. (*Continued*)

based on the retinal morphology and the parameters are tuned through comparisons with electrophysiological measurements.

The prototype complex-cell CNN-UM chip is a special tool to reproduce spatial-temporal effects. The effects can be programmed as a CNN template and the sequences of the stored templates define an analogic algorithm. This chip opens new possibilities in high-speed spatial-temporal wave dynamics computations.

We demonstrated that at least second/third-order dynamics is necessary for correct retinal modeling. This paper has demonstrated the application of the prototype complex-cell CNN-UM chip for computing one retina channel on silicon. This is the first step of the hardware implementation toward the stored programmable neuromorphic CNN mammalian retina model. The multilayer CNN model has been decomposed and scaled to meet the specific hardware requirements.

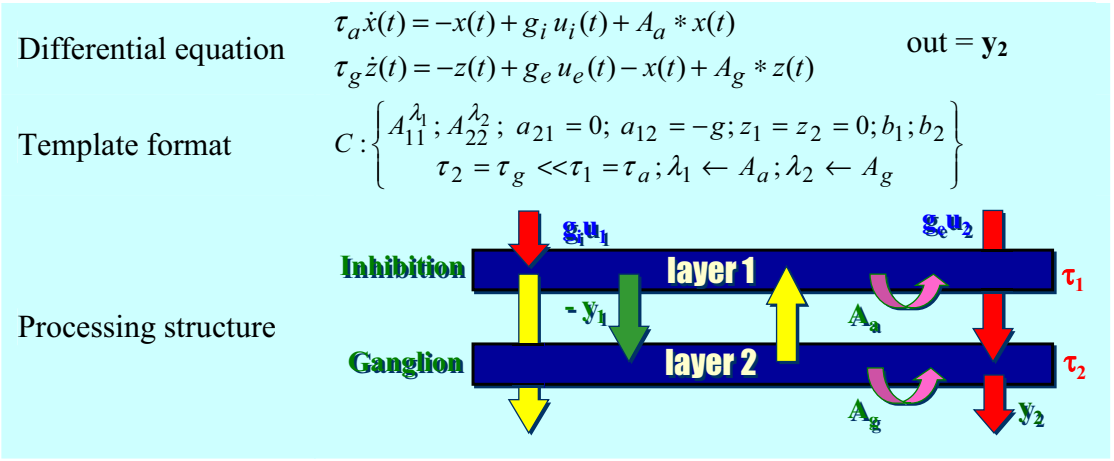


Fig. 21. Programming the CACE1k chip for the ganglion block computation. The first layer represents the inhibitory path and the second layer stands for the ganglion output.

The next generation 3-layer complex-cell chip, under design, will have an optical input and different adaptation effects (e.g. gain control) based on the adaptive CNN-UM architecture. This will simplify the outer retina block calculation and speed-up the whole procedure.

The decomposed model can serve as the algorithmic base for sensing aids, like retina prosthesis and can transform the programmable complex-cell visual microprocessors to retina-chips. These chips can be reprogrammed even after a possible retinal implementation and so open to any new discovery. The retina, as a computational device, is a sophisticated tool for multichannel preprocessing of a video flow. It is anticipated that the results can be embedded into several complex algorithms and applications targeting real-life applications, like object classification, recognition, tracking, and alarming.

Acknowledgments

The support of the Office of Naval Research USA (ONR) grant no. N00014-021-0884 and N00014-00-C-0295, the Hungarian Academy of Sciences and the Jedlik Laboratories of the Pázmány University are acknowledged.

References

Bálya, D., Rekeczky, Cs. & Roska, T. [2002a] "A realistic mammalian retinal model implemented on complex cell CNN universal machine," *Proc. IEEE ISCAS-2002*, IV-161–164.

Bálya, D., Roska, B., Roska, T. & Werblin, F. S. [2002b] "A CNN framework for modeling parallel processing in a mammalian retina," *Int. J. Circuit Th. Appl.* **30**, 363–393.

Bloomfield, S. A. & Miller, R. F. [1986] "A functional organization of ON and OFF pathways in the rabbit retina," *J. Neurosci.* **6**, 1–13.

Boahen, K. A. [2002] "A retinomorphic chip with parallel pathways: Encoding ON, OFF, INCREASING, and DECREASING visual signals," *J. Anal. Integr. Circuits Sign. Process.* **30**, 121–135.

Carmona, R., Jiménez-Garrido, F., Domínguez-Castro, R., Espejo, S., Roska, T., Rekeczky, Cs., Petrás, I. & Rodríguez-Vázquez, A. [2002a] "A bio-inspired two-layer mixed-signal flexible programmable chip for early vision," *IEEE Trans. Neural Networks* **14**, 1313–1336.

Carmona, G. R., Jiménez Garrido, F., Domínguez Castro, R., Espejo Meana, S. & Rodríguez Vázquez, A. [2002b] *CACE1k User's Guide*, Instituto de Microelectrónica de Sevilla.

Cauwenberghs, G. & Waskiewicz, J. [1999] "Focal-plane analog VLSI cellular implementation of the boundary contour system," *IEEE Trans. Circuits Syst.-I*: *Fund. Th. Appl.* **46**, 327–334.

Chua, L. O. & Yang, L. [1988a] "Cellular neural networks: Theory," *IEEE Trans. Circuits Syst.* **35**, 1257–1272.

Chua, L. O. & Yang, L. [1988b] "Cellular neural networks: Applications," *IEEE Trans. Circuits Syst.* **35**, 1273–1290.

Chua, L. O. & Roska, T. [1993] "The CNN paradigm," *IEEE Trans. Circuits Syst.-I* **40**, 147–156.

Chua, L. O. [1998] *CNN*: *A Paradigm for Complexity* (World Scientific, Singapore).

Chua, L. O. & Roska, T. [2002] *Cellular Neural Networks and Visual Computing* (Cambridge University Press, Cambridge, UK).

Dagnelie, G. & Massof, R. [1996] *Toward an Artificial Eye*, Johns Hopkins University School of Medicine.

Dowling, J. E. [1987] *The Retina*: *An Approachable Part of the Brain* (Belknap Press of Harvard University Press, Cambridge).

Eeckman, F. H. (ed.) [1993] *Neural Systems*: *Analysis and Modeling* (Kluwer Academic).

Espejo, S., Carmona, R., Domínguez-Castro, R. & Rodríguez-Vázquez, A. [1996a] "A CNN universal chip in CMOS technology," *Int. J. Circuit Th. Appl.* **24**, 93–110.

Espejo, S., Carmona, R., Domínguez-Castro, R. & Rodríguez-Vázquez, A. [1996b] "A VLSI oriented continuous-time CNN model," *Int. J. Circuit Th. Appl.* **24**, 341–356.

Gál, V. *et al.* [2004] "Receptive field atlas and related CNN models," *Int. J. Bifurcation and Chaos* **14**, 551–584.

Hesse, L., Schanze, T., Wilms, M. & Eger, M. [2000] "Implantation of retina stimulation electrodes and recording of electrical stimulation responses in the visual cortex of the cat, Graefe's Arch. Clin. Exp.," *Ophthalmol.* **238**, 840–845.

Kandel, E. A., Schwartz, J. A. & Jessell, T. M. (eds.) [2000] *Essentials of Neural Science and Behavior* (Prentice Hall International), Chap. 16, pp. 507–522.

Kék, L. (ed.) [1999] *CNN Software Library* (*Templates and Algorithms*) *Version 1.1*, Analogic Computers Ltd., Budapest, Hungary.

Koch, C. & Segev, I. [1989] *Methods in Neuronal Modeling* (MIT Press).

Liñan, G., Espejo, S., Domínguez-Castro, R., Roca, E. & Rodríguez-Vázquez, A. [2000] "The CNNUC3: An analog I/O 64×64 CNN universal machine chip prototype with 7-Bit analog accuracy," *Proc. IEEE Int. Workshop on Cellular Neural Networks and Their Applications* (CNNA'2000), Catania, 0-7803-6344-2, pp. 201–206.

Linsenmeier, R. A., Frishman, L. J., Jakiela, H. G. & Enroth-Cugell, C. [1982] "Receptive field properties of X and Y cells in the cat retina derived from contrast sensitivity measurements," *Vis. Res.* **22**, 1173–1182.

Petrás, I., Rekeczky, Cs., Roska, T., Carmona, R., Jiménez-Garrido, F. & Rodríguez-Vázquez, A. [2003] "Exploration of spatial-temporal dynamic phenomena in a 32×32-cells stored program 2-layer CNN Universal Machine chip prototype," *J. Circuits Syst. Comput.* **12**, in press.

Rekeczky, Cs., Serrano, T., Roska, T. & Rodríguez-Vázquez, A. [2000] "A stored program 2nd order/3-layer complex cell CNN-UM," *Proc. IEEE CNNA*, pp. 213–218.

Rekeczky, Cs., Roska, B., Nemeth, E. & Werblin, F. S. [2001] "The network behind spatio-temporal patterns," *Int. J. Circuit Th. Appl.* **29**, 197–239.

Roska, T. & Chua, L. O. [1993] "The CNN universal machine: An analogic array computer," *IEEE Trans. Circuits Syst.-II* **40**, 163–173.

Roska, T. [1999] "Computer-sensors: Spatial-temporal computers for analog array signals, dynamically integrated with sensors," *J. VLSI Sign. Process.* **23**, 221–238.

Roska, B., Nemeth, E., Orzo, L. & Werblin, F. S. [2000] "Three levels of lateral inhibition: A space-time study of the retina of the tiger salamander," *J. Neurosci.* 1941–1951.

Roska, B. & Werblin, F. S. [2001] "Vertical interactions across ten parallel stacked representations in mammalian retina," *Nature* **410**, 583–587.

Roska, T. [2003] "Computational and computer complexity of analogic cellular wave computers," *J. Circuits Syst. Comput.* **12**, 539–562.

Werblin, F. S. & Jacobs, A. [1994] "Using CNN to unravel space-time processing in the vertebrate retina," *Proc. CNNA-94*, pp. 33–40.

Werblin, F. S., Roska, T. & Chua, L. O. [1995] "The analogic cellular neural network as a bionic eye," *Int. J. CTA* **23**, 541–569.

Werblin, F. S., Roska, B., Bálya, D., Rekeczky, Cs. & Roska, T. [2001] "Implementing a retinal visual language in CNN: A neuromorphic study," *Proc IEEE ISCAS-2001*, Sydney.

Werblin, F. S. & Roska, B. M. [2004] "Multichannel retinal processing: A tutorial of retinal function from photoreceptors to parallel feature abstractions mediated by the analog array processing capabilities of the retina," *Int. J. Bifurcation and Chaos* **14**, 843–852.

Wyatt, J. & Rizzo, J. [1996] *Ocular Implants for the Blind* (Harvard Medical School).

Zarándy, A., Rekeczky, Cs., Földesy, P., Szatmári, I. [2003] "The new framework of applications — The Aladdin system," *J. Circuits Syst. Comput.* **12**, in press.