Retina model with real time implementation

David Bálya
Faculty of Information Technology
Pazmany University and MTA-SzTAKI
Budapest, Hungary
balya@sztaki.hu

Botond Roska
Harvard Society of Fellows
Harvard University
Cambridge, MA, U.S.A.

Abstract— **This paper describes a novel retina model and its implementation. Our modeling approach is neuromorphic. The primary motivation is to deduce an algorithmic skeleton from the measurements and to make possible the on-line parameter changing. The model is tuned for half a dozen different channels with less than half percent spatial-temporal error. An of-the-shelf stand-alone system, the Bi-I, is used to real-time implement this mammalian retina model. The system is capable to compute four retina channels in video real time. The easy on-line control is solved with faders and buttons.**

I. INTRODUCTION

Several different retina models are developed through the years. Some are low-complexity black-box models [1]; others are high complexity detailed neuromorphic models. Most of them tries to find answers for specific questions thus these are not complete retina models.

The common neuronal modeling packages simulate networks of synaptically connected neurons e.g. Genesis and Saber. The structural size of the model depends on the task of the network and a retina model easily can reach fifty thousand neurons. Each neuron is modeled by compartments (state equations describing the behavior of the different parts of the neuron represented by the attached electrical circuits) and sometimes the models use a dozen compartments for one neuron. Simulating these systems require powerful workstations because of the huge number of variables that need to be involved [2].

The mammalian retina consists of a dozen different ganglion cell types [3]. They form parallel output channels to the brain. These parallel operating ganglion cell populations are feature detectors that are built from a series of space-time transformations generated at different retinal levels [3]. The presented model is tuned to emulate these channels.

The cellular nonlinear network (CNN [4]) based models are neuromorphic and copy the essence of the receptive field organization [5]. The latest CNN mammalian retina model [6] is simplified, compared to the previous CNN studies [7] and models the important cell-types with CNN layers. Through the years several simulators have been developed to model different sensory systems, the latest one is called RefineC.

Supported by the Human Frontier Science Program young investigator grant and by the Office of Naval Research N00014-02-1-0884 grant, as well as, the Hungarian Academy of Sciences

These models, both multi-layer CNN and others, are not designed for real-time environment. Therefore, their implementations are hard or impossible with the current technology. This paper proposes a channel-based retina model computable real-time with an of-the-shelf system.

Here, a CNN-UM array computer is used as a fast processor to video real-time implement the mammalian retina model. Our modeling approach is neuromorphic in its spirit relying on all three: morphological, electro-physiological, and pharmacological information [8]. The primary motivation is to develop an algorithmic skeleton where the parameters are derived from the spatial-temporal data recorded from ganglion cells. The secondary motivation is to create a user-friendly system, where the parameters can be changed on-line, to enable the creative experimentation with the real-time retina model.

The structure of the paper is the following. The next section mentions a few special requirements. The third section outlines the model. The fourth section describes the hardware-software environment and the last section displays the system in action.

II. REQUIREMENTS

First of all, it is a hard real-time task. It is 'hard', because the device should operate with a given stimulation-rate and if the input frame is not filtered within the prescribed time, the result is irrelevant. Contrarily, the simulation is a soft real-time task, because those results are analyzed after the experiment. The computations should not depend on the content of the input image-flow or on the selected retina channel, i.e. on the parameters of the algorithm.

Secondly, it is a portable device so the size, weight and power consumption is not negligible [9]. The prototype model is implemented on a stand-alone system in order to show the feasibility of the suggested solution. The performance can be improved by dedicated hardware systems.

Thirdly, the controlling of the device is crucial to enable the creative experimentation with the real-time retina model. The space-time filters of the retina model emulation can be examined in two ways: selecting a retina channel and fine-tune its transfer functions. We propose two unique solutions

for the interface: a midi controller with buttons and faders and an accelerometer sensed stick.

III. THE MODEL

A. *The model framework*

The steps of the computation are as follows, see Fig.1. First, the input image is convolved with a spatial-temporal kernel. Second, the result is divided into two parts. If it is an On channel, the positive part is the excitation and the negative part is inverted and processed further. Third, the inhibition is computed by low-pass filtering the previous signal. Finally, the excitation minus the inhibition gives the predicted output voltage. This voltage is thresholded to determine the output, i.e. if the output voltage is above a certain level in a given position, the corresponding output pixel will be colored otherwise white. The spiking frequency is not predicted.

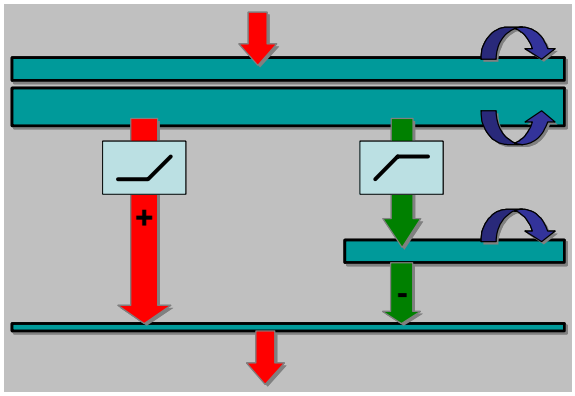

Figure 1. The schema of the processing. The horizontal lines represent processing steps and the vertical arrows show the data-flow.

It is assumed that the first spatio-temporal kernel is separable in time and space. Therefore the computation of the first spatio-temporal filter means two convolutions. The two convolution kernels are derived from the measurements. The spatial one is a linear combination of two low-pass filters. The temporal one is a FIR (final impulse response) filter: linear combination of a few previous frames.

The temporal convolution (1) determined as follows. Measured responses are known for white square stimuli. The measurement data from the cells below the middle of the square give essentially the unit step response of the system $\boldsymbol{H}(t)$. The data series are approximated with an analytic function (2). Therefore the derivative of the function gives the weight function of the filter. The main parameters are the time constants $\boldsymbol{\tau}_{1,2}$ and the sustained component $\boldsymbol{\sigma}$.

$$x_{ij}(t) = \int \boldsymbol{h}(\tau)\, u_{ij}(t-\tau)\, d\tau \qquad h(t) = dH(t)/dt \tag{1}$$

$$\boldsymbol{H}(t) = (1 - \mathrm{Exp}[-(t/\boldsymbol{\tau}_1)^{\mathbf{p}}])\,(\mathrm{Exp}[-(t/\boldsymbol{\tau}_2)^{\mathbf{q}}] + \boldsymbol{\sigma}) \tag{2}$$

The spatial filter, kernel in 1D, is essentially a scaled difference of Gaussian blurring (3). The parameters, space constants $\boldsymbol{\lambda}_{1,2}$ and weight $\mathbf{r}$, are fitted using only the maximum response at the central cell for five differently sized white squares. For practical reasons the blurring is computed using an equivalent 3-by-3 feedback or iterative convolution kernel.

$$\boldsymbol{k}(x) = \mathrm{Exp}[-(x/\boldsymbol{\lambda}_1)^2] - \mathbf{r}\,\mathrm{Exp}[-(x/\boldsymbol{\lambda}_2)^2] \tag{3}$$

The non-linear part of the model is based on the morphology and measurements. For example, the Alpha and Delta channels lack of inhibition, while the rest has. The spatial filter on the inhibition is a blurring proportional to the size of its amacrine cell. The final threshold is selected to create a sparse spiking pattern.

B. *The model parameters*

The parameters of the algorithmic framework are computed as described before. The mean square errors in the different channels between the measured and emulated cases are computed. The spatial and temporal errors are shown in Table 1. The numbers are remarkable compared to the accuracy of the measured biological system.

The tuning was verified using a new white square stimulus size and analyzing the spatial temporal properties of the computed result. The results for the OnBeta cell, as an example, are shown in Fig.2 and 3. Beta cells are the most frequently occurring ganglion cell type in the primate's fovea and are the homologues of midget ganglion P-cells [10].

TABLE I. ON CHANNELS' MODELLING ERRORS (MSE)

Channel's name	**Mean square error**	
	Temporal kernel	***Spatial kernel***
Beta	0.00019	0.00050
Transient	0.00046	0.00895
Alpha (Brisk)	0.00266	0.00074
Bistratified	0.00435	0.00273
Edge (LED)	0.00309	0.00114
Delta (Sluggish)	0.00599	0.00616
Polar	0.00037	0.01038

The parameters for the spatial and temporal kernel are computed using the method described above. An independent measurement with a 600µm square is used for verification. The spatial profiles are compared at the time instances of maximum On and Off responses. The non-symmetric emulation results come from the analog CNN computation.

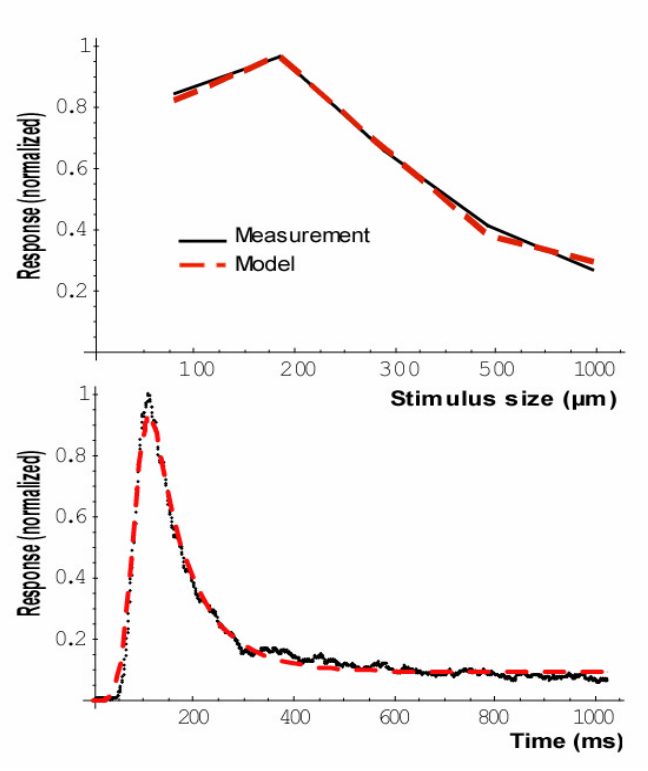

Figure 2. Model tuning result for the OnBeta channel. The black solid lines show the (normalized) measured responses and the red dashed lines are the model predictions. The errors are in Table 1.

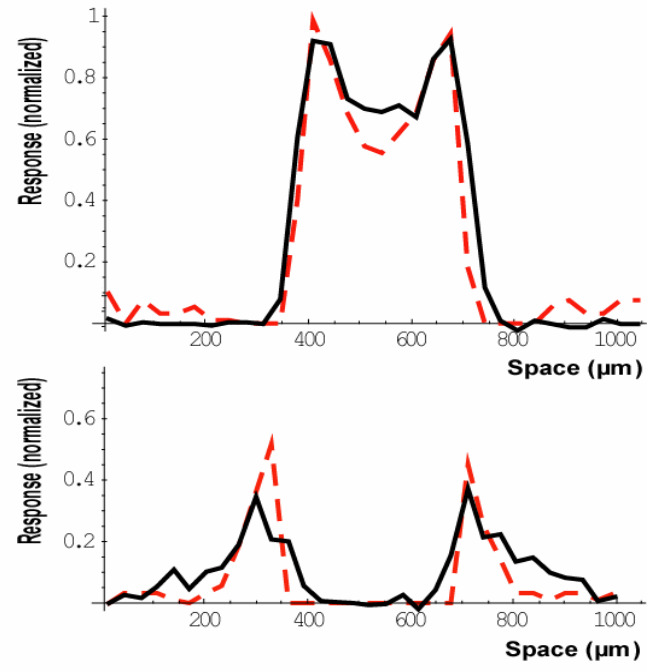

Figure 3. Model verification result for the Beta channel. The black solid lines show the (normalized) measured responses and the red dashed lines are the model predictions. The first is the spatial profile at the maximum On response time, the second is the same for the Off.

IV. System Architecture

A. *Hardware setup*

The hardware setup is on Fig.4. The proposed model is implemented on the stand-alone Bi-I computing device [11] developed by the Analogic Computers Ltd to show the feasibility of the prototype algorithm. It contains a CNN-UM type sensor-processor array: Ace16k and a Texas Instruments signal processor: TMS320C6202. The image acquisition is done by the CNN-UM chip. The typical integration time is 3-4ms in sunny outdoor environment.

We intend to compare the implementations in different machine vision systems (either fully analog, mixed mode, DSP based or custom digital) including the EyeRes chip from the AnaFocus Ltd. and FPGA solutions. The speeds of the current software simulations are not satisfactory on a high end PC, e.g. the optimized Matlab© implementation of the presented algorithm is far from the required processing speed mostly because of the spatial computations.

The controlling tasks are done with a Midi controller. Each button selects a given retina channel. Individual parameter tuning is possible through the faders and the internal parameters can be tuned with the rotary controllers. A special accelerometer sensed stick is applied to tune a set of parameters at once. As the stick is rotated in the air, the spatial-temporal kernel is changed.

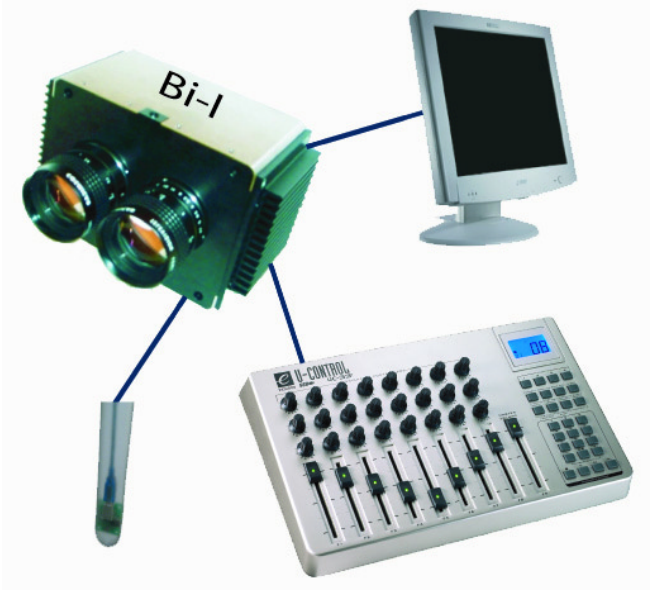

Figure 4. The prototyping setup. The Bi-I stand-alone vision computer is applied for all tasks: its CNN-UM image sensor gives the input and makes all the computations, the parameters can be tuned through a Midi controller and/or an accelerometer sensed stick (bottom-left).

B. *Software architecture*

The software architecture is in the simplified multi-adaptive framework [12]. The incoming image flow is processed frame-by-frame. The source can be an external video, frame-grabber or the CNN-UM visual microprocessor image sensor. A simple global adaptive integration time is maintained. Temporal filtering follows the spatial filter. Its output is non-linearly separated and further processed as described before. The final output is the thresholded linear combination of the excitation and inhibition, see Fig. 2.

The model parameters can be changed in asynchronous fashion. The result of any stage can be read out as a video-flow. A selected pixel can be read-out also to solve the transfer bottleneck. Moreover this makes possible to perform "virtual measurements".

V. RESULTS

Figure 5 shows the system in action. It contains three columns during the OnBeta channel computation. The first displays the spatial and temporal kernels, where the blue curve stands for the inhibition spatial kernel. The middle one shows the input and output frames at a give time-instant, the red blobs are the calculated spiking spots. The right hand side displays the central row spatial profile and the central cell temporal response. The green curve is the input and the red curve corresponds to the output voltage.

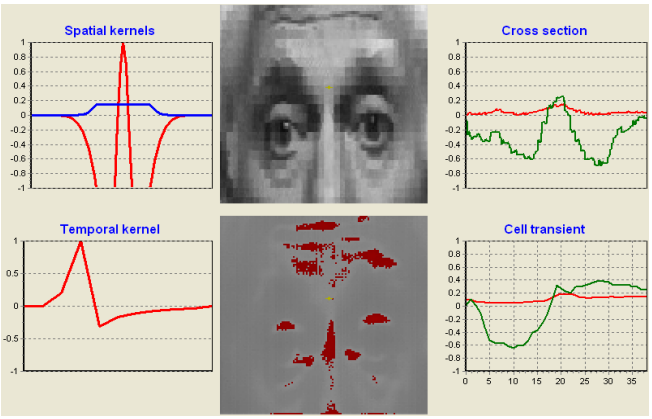

Figure 5. System in action screenshot. The kernels, the input and output voltage and spiking pattern, and the spaial profile and central cell transient are shown. Green lines are the input, red curves are the output.

Figure 6 displays two screenshots, two calculated spiking patterns. On the left hand side the On and Off Delta and Alpha channels are computed simultaneously. On the right hand side the calculated spiking spots for the On and Off Beta and Delta channels are shown. The green/olive blobs indicate the Off channel spikes and the red/blue blobs shows the On channels calculated spiking spots. The system is capable to compute and display all these channels in real time, i.e. within 30ms.

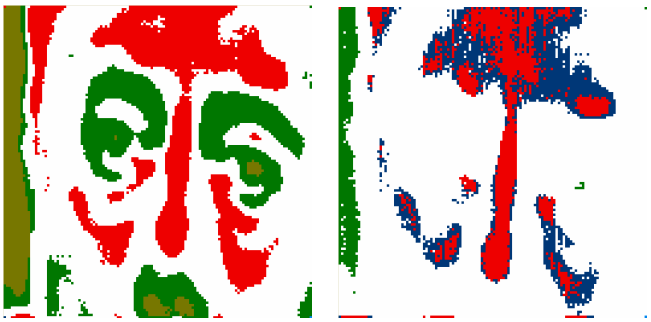

Figure 6. Four different channels are computed real-time and showed with different colors. Left hand side: On and Off Delta and Transient channels, right hand side: On and Off Beta and Alpha channels. Green is the Off channels and red is the On channels spiking predictions.

ACKNOWLEDGMENT

The authors would like to thank Tamas Roska for his advices, furthermore to the Analogic Computers Ltd for the development support to the Bi-I. This work was supported in part by the Analogic Lab at the MTA-SzTAKI and the Jedlik Lab at the Pazmany University.

REFERENCES

[1] M. Meister and M. J. Berry, "The Neural Code of the Retina," Neuron, Vol. 22, pp. 435-450, 1999.

[2] Erik De Schutter:"A consumer guide to neuronal modeling software", Trends in Neurosciences, Vol. 15, pp.462-464,1992

[3] B. Roska and F. S. Werblin, "Vertical interactions across ten parallel, stacked representations in the mammalian retina," Nature, Vol. 410, pp. 583-587, 2001.

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

[5] F. S. Werblin, T. Roska and L. O. Chua, "The analogic cellular neural network as a bionic eye," Intl. J. of Circuit Theory and Applications; Vol. 23, pp. 541-569, 1995

[6] D. Bálya, B. Roska, T. Roska, F. S. Werblin, "A CNN Framework for Modeling Parallel Processing in a Mammalian Retina," Int'l Journal on Circuit Theory and Applications, Vol. 30, pp. 363-393, 2002

[7] V. Gál, J. Hámori, T. Roska, D. Bálya, Zs. Borostyánkői, M. Brendel, K. Lotz, L. Négyessy, L. Orzó, I. Petrás, Cs. Rekeczky, J. Takács, P. Venetiáner, Z. Vidnyánszky, Á Zarándy: "Receptive Field Atlas and Related CNN Models," International Journal of Bifurcation and Chaos, Vol. 14(2), pp. 551-584, 2004.

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

[9] Mark Humayun, D. Yanai, R.J. Greenberg, J. Little, B.V. Mech, M. Mahadevappa, J.D. Weiland, G.Y. Fujii, E. deJuan, Jr., "Clinical Results with the Model 1 IRP implant," International Joint Conference on Neural Networks IJCNN04, Budapest, July 2004

[10] Heinz Wässle, "Parellel processing in the Mammalian Retina," Nature Reviews Neuroscience, Vol. 5, pp. 747-757, 2004.

[11] A Zarándy, Cs. Rekeczky, P. Földesy, I. Szatmári, "The new framework of applications – The Aladdin system," J. Circuits Syst. Comput. Vol. 12, pp. 769-782, 2003.

[12] Cs. Rekeczky, I. Szatmári, D. Bálya, G. Tímár, and Á. Zarándy, "Cellular Multiadaptive Analogic Architecture: a Computational Framework for UAV Applications," IEEE Transactions on Circuits and Systems I: Regular Papers, Vol. 51, pp.864-884, 2004