Toward Visual Microprocessors

TAMÁS ROSKA, FELLOW, IEEE, AND ÁNGEL RODRÍGUEZ-VÁZQUEZ, FELLOW, IEEE

Invited Paper

This paper outlines motivations and models underlying the design of visual microprocessors based on the cellular neural network universal machine. We also overview the state of the art regarding the realization of these microprocessors in the form of very large-scale integration chips. Examples corresponding to measurements realized on these chips are enclosed for illustration purposes.

***Keywords**—Analogic cellular supercomputing, cellular neural networks, CNN technology, visual microprocessors.*

I. INTRODUCTION

For more than 100 years, the living visual system of mammals has been intensively studied by neuroscientists and biophysicists alike. Recently, computer engineers have been active creating machine vision systems. Still, although many ideas have been proposed and implemented in silicon [1]–[3], including resistive grid "silicon retinas," programmable cellular neural/nonlinear network (CNN)[1] models of the visual pathway, as well as many "smart optical sensors," no complete neuromorphic model of the topographic parts of the visual pathway has been made available. The reason is simple: the lack of understanding of the detailed operation of many key components located at the front-end of the visual system, notably, the retina and the lateral geniculate nucleus (LGN). Hence, the representation of the visual scene from the input to the higher layers has been unknown. Of the many exciting partial results related to the visual pathway, some recent findings (see, for instance, [4]) suggest a few sound principles.

- Sensing and processing are interactive processes, and the processing is mainly analog, combined with masks of binary (yes/no) maps.
- The basic structure is composed of several stacks of layers of neurons connected by local receptive field organizations with different spatial distributions and time constants.
- The processing strategy is a kind of "multiscreen theater"; namely, from a given visual scene, several parallel maps are generated and then further processed. This is true even in the mammalian retina [4] where about a dozen parallel channels are organized.

To implement neuromorphic visual models on silicon, we have two ways:

- Pick up a specific task and its model and implement it on silicon. This is the usual way, leading to very useful, task-specific smart sensors.
- Make mixed-signal[2] visual microprocessors. That is, processors which combine optical sensing with analog cellular spatial-temporal dynamics and some form of logic (they are called analogic processors because they combine analog and logic processing structures), which have receptive fields like elementary instructions, and the possibility of storing and executing user-selectable sequences of instructions (programs).

Clearly, the second approach is more demanding in terms of architecture, very large-scale integration (VLSI) chip design, and computational infrastructure, leading to a new type of hardware/software system design.

This paper focuses on the second approach. Namely, we will briefly review the analogic cellular computer architecture, some CMOS prototype chips related to that architecture, and the accompanying computational infrastructure. Some examples measured from the so-called ACE4K chip [10] and the CACE1K chip [11] are included for illustration purposes. The former has a one-layer architecture, while the latter has a three-layer architecture inspired by the CNN model of the

---

[1]Cellular neural/nonlinear network (CNN) models were introduced by Chua and Yang in 1988 [5], and then generalized and used as a model for bionic eyes by Chua, Roska, and Werblin [6]–[8]. Their principles and applications for visual processing are covered in [9].

---

[2]Mixed-signal means that analog and digital signal representations are combined, and hence analog and digital signal processing.

mammalian retina proposed in [12] based on the discoveries about the functionality of the inner part of this retina as reported in [4].

II. CNN-Based Visual Microprocessors

Back in the 1960s, the building blocks for logic design had been the various logic circuits (micromodules) implementing different "smart" logic tasks. These had also been used to make digital computers. The digital computer has a key attribute due to J. Von Neumann, namely stored programmability. It means that the same core architecture, via algorithms coded in software, can be used for a myriad of tasks. Or, to put it in another way, the architecture is open to the human intellect for millions of algorithmic innovations. This is the functional secret behind the success of the digital microprocessor, first made in the early 70s. Visual microprocessors aim to mimic this functional secret. However, they are mixed-signal devices which realize analog-and-logic spatial/temporal processing tasks (wave processing), and hence require quite different building blocks [3].

The front-end "devices" encountered in natural vision systems are capable of acquiring and processing images in a concurrent manner. The retina contains photoreceptors and dynamically coupled processing cells of different types. Among many other tasks, the early processing realized at the retina serves to extract important features from the raw sensory data and, thus, to reduce the amount of information transmitted for subsequent processing. In contrast to that, image acquisition and processing are usually separated in conventional artificial vision systems. One key aspect of visual microprocessors is the integration of sensing and stored programmable processing (SPP) at the analog signal array level—the integrated SPP principle. Among many other things, this allows us to tune the sensors dynamically, pixel by pixel, depending on the content and even on the context of the changing scene. Some of the key architectural aspects have been discussed in [13].

Some features which make the visual microprocessors addressed in this paper different from other topographic smart sensors [1], [2] include the following.

- They use a core analog processing array (a CNN [5]–[7]) with tunable interaction weight patterns and embedded pixel-wise data memories.
- This programmable and reconfigurable array is embedded in a computer architecture resulting in the so-called CNN univesal machine (CNN-UM).
- The CNN-UM is stored programmable and capable of implementing analogic spatial–temporal algorithms through the smart synergy of hardware and software.

All the signal variables are continuous, except for the discreteness in space (pixels or voxels). At the same time, visual microprocessors retain the extraordinary strength of digital computers, their unconstrained variability via programming or software. Obviously, such software and related algorithms are different from conventional ones.

Below we summarize the main architectural and algorithmic ideas underlying CNN-based visual microprocessors. It is worth mentioning that although most of their present-day applications are related to vision, many other
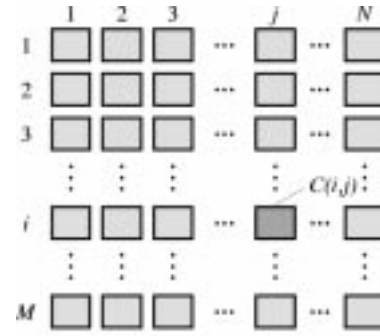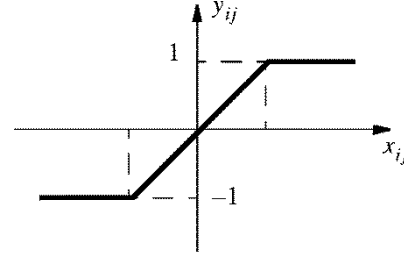

Fig. 1. A typical simple CNN structure.

Fig. 2. The standard output nonlinearity.

topographic problems (tactile and auditory), including topographic optimization, are among the emerging applications.

A. CNN Dynamics

CNNs can be either single-layer or multilayer. Consider first a single layer consisting of a two-dimensional (2-D), regular grid of cells $C(ij)$, where i and j are the row and column coordinates. The topography of such a structure is shown in Fig. 1.

Assume each cell hosts a processor with its real-valued input, state(s), and output signals, $u_{ij}(t)$, $x_{ij}(t)$, and $y_{ij}(t)$, respectively. In such a 2-D layer, each cell processor is connected to its neighbors (in a 3×3 or 5×5, etc., neighborhood or sphere of influence), denoted by $S_r(ij)$. The simplest first-order cell state dynamics is given by[3]

$$\dot{x}_{ij} = -x_{ij} + \sum_{C(kl) \in S_r(ij)} A(ij; kl) \cdot y_{kl}$$
$$+ \sum_{C(kl) \in S_r(ij)} B(ij; kl) \cdot u_{kl} + z_{ij} \quad (1)$$

where $z_{ij} \in R$ is called the threshold of the cell $C(ij)$ · $A(ij; kl)$ and $B(ij; kl)$ are called the feedback and feedforward synaptic operators or templates; in case of a 3×3 neighborhood of radius 1, they are 3×3 matrices.

The state and the output signals of each cell are typically related through the following nonlinear output equation:

$$y_{ij} = f(x_{ij}) = \frac{1}{2}[|x_{ij} + 1| - |x_{ij} - 1|] \quad (2)$$

depicted in Fig. 2. However, the nonlinearity could be of several types and it could also be included in a simpler dynamic equation form. Namely, the standard nonlinearity

[3]The time is scaled in the relative time unit τ_{CNN} which is the time constant of the simple first-order cell dyanmics.
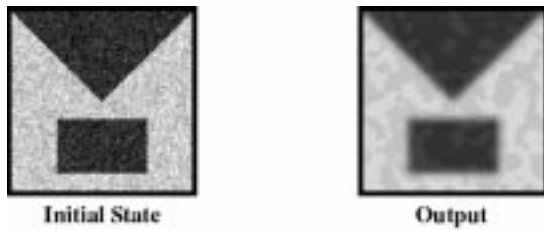
Fig. 3. The initial picture and the diffused picture using a diffusion template defined by gene G_D.

in (2) and the cell-state dynamics represented by (1), the so-called Chua–Yang model, could be replaced by the full-range model which means that $y_{ij}(t) = x_{ij}(t)$, and that the first term in (1) is replaced by a nonlinear function $g(x_{ij})$ whose shape is the inverse of that used for the standard nonlinearity [14].

Once the cell dynamics is fixed, the interaction patterns $\mathbf{A}$ and $\mathbf{B}$ and the offset value z define the functionality of the CNN layer. Given an input signal array u_{ij} for $1 \leq i \leq M, 1 \leq j \leq N$, defined as a picture with pixel values u_{ij}, the set of values $(\mathbf{A}, \mathbf{B}, z)$ determines the outcome of the CNN dynamic process. This set is called a cloning template or a gene. In the space-invariant case, the templates are 3×3 (or 5×5 or 7×7) matrices. This means that a CNN array can be defined by the cell dynamics and the 19 (or 51 or 99) numbers of the $\mathbf{A}, \mathbf{B}$ templates and the offset z. The input image could be either static or dynamic; hence, a CNN layer plays the role of an image processor.

The peculiar property of controlling the functionality of a whole array of interconnected cells by means of just a few interconnection weights (e.g., 19 numbers) is very familiar to neurobiologists. Indeed, the cloning template is no more than a receptive field organization in the retinotopic part of the visual pathway [8]. On the other hand, the CNN paradigm is well suited for representing many topographic sensory modalities via their receptive field organizations. The first attempts [15] have been followed by many other useful results.

In a nontrivial case, the CNN dynamics is a wave acting for a finite time T. For example, for a diffusion template or gene G_D we have

$$\mathbf{A} = \begin{bmatrix} 0.1 & 0.15 & 0.1 \\ 0.15 & 0 & 0.15 \\ 0.1 & 0.15 & 0.1 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad z = 0. \tag{3}$$

Fig. 3 shows the initial state and the output image (at $T = 2$ elapsed time). There exists a very wide catalog of templates covering a myriad of applications. Also, because these templates are programmable by definition, learning can be incorporated to adapt the templates either globally, for example, using a genetic algorithm [16], or locally. Thus, not only associative memories can be constructed, e.g., [17], but the plasticity of the brain might be directly modeled [13].
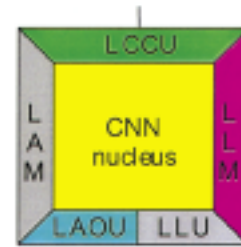
LCCU	Local Communication and Control Unit
LAOU	Local Analog Output Unit
LLU	Local Logic Unit
LAM	Local Analog Memory
LLM	Local Logic Memory

Fig. 4. The extended cell of the CNN-UM.

B. The CNN-Universal Machine (CNN-UM) [7]

If we furnish each CNN cell processor with local memories [local analog memory (LAM) and local logic memory (LLM)] and a local communication and control unit (LCCU) to send/receive information to/from the global analogic programming unit (GAPU), we get the extended CNN cell of the CNN-UM architecture. For practical reasons, in each cell we add a local logic unit (LLU) and a local analog output unit (LAOU) which take inputs and send outputs from/to their local memories, LLM and LAM, respectively. Fig. 4 shows the extended cell schematically.

The GAPU is the conductor of the extended cell array, communicating with each cell via the LCCUs of each cell. The GAPU contains three registers and a global analogic control unit (GACU), the latter of which is the host of the stored program and controls the whole array computer. The three registers store the cloning templates [analog programming-instruction register (APR)], the local logic instructions [logic program-instruction register (LPR)], and the switch configuration codes [switch configuration register (SCR)], respectively.

The CNN-UM can be viewed as an array computer defined on flows [18]. Algorithms can be constructed where the elementary instruction is the solution of a partial differential equation (PDE). This correspondence was highlighted already in the seminal paper [5] for the heat equation; also, in [19], a mechanical system was modeled by a CNN. Later, systematic methods have been devised to convert PDEs defined in continuous space into CNN dynamics [20]. Recent advances in complex image processing show that PDE-based techniques seem to be superior in many respects (e.g., [21]). The drawback is their high computational complexity when implemented in digital processors. Here, using a CNN, solution of a nonlinear PDE is the basic task.

The next example shows a complex analogic spatial/temporal algorithm used for the calculation of the inner boundaries of the left ventricle in an echo-cardiogram [22]. Active waves [23] are used as algorithmic steps. For reference, we
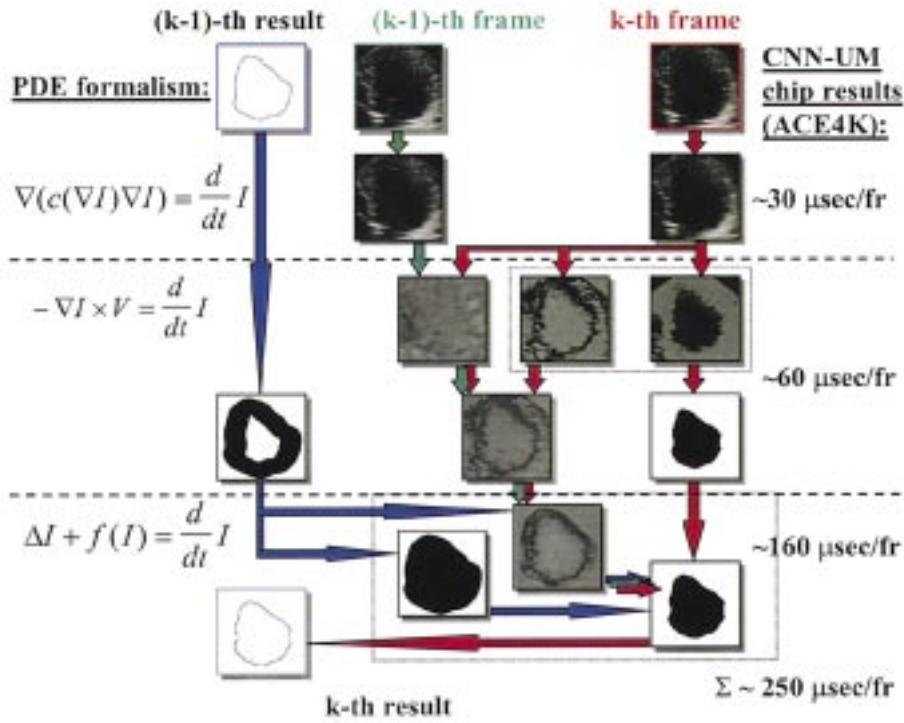
Fig. 5. The bold arrows represent different cloning templates. Some of them are performing the solution of complex nonlinear PDEs as elementary instructions; these are written on the left-hand side of the figure with their execution times on the right-hand side. In addition, several simpler instructions and templates are used, for instance, local logic operations.

also show the execution times of the algorithmic steps on the so-called ACE4k chip [10].

C. Example 1

A flow diagram is depicted in Fig. 5 of the analogic CNN algorithm with some typical intermediate results. Observe that it can be interpreted as a combination of three image flows merging and branching during the processing stage of a single frame. Here the third flow stands for the information calculated from the current frame, the second one for the intermediate results obtained from the previous frame, while the first one represents the binary masks generated from the previous result. The core of the three main processing stages of the algorithm can also be described by PDEs (left): 1) image filtering and reconstruction derived from nonlinear diffusion PDEs; 2) motion estimation derived from optical flow PDEs; and 3) trigger wave-type active contour-based boundary tracking derived from reaction-diffusion nonlinear PDEs. These PDE approximations, executed on the ACE4K chip, can be completed within a millisecond, allowing the processing system to reach its peak performance around four thousand frame/sec (right).

D. Multilayer and Complex Cell CNN-UM

The multilayer CNN structure was already introduced in [5]. It is used when several 2-D CNN layers are necessary
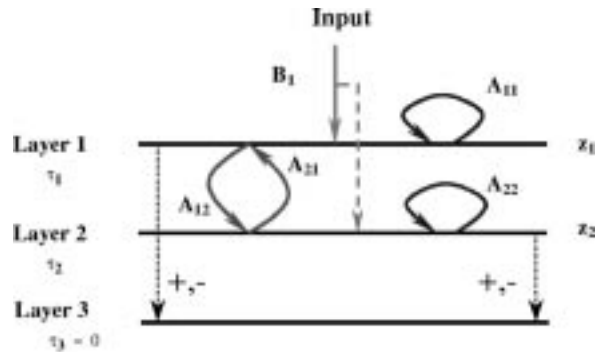

Fig. 6. Fig. 3 shows the initial state and the output image (at $T = 2$ elapsed time). There exists a very wide catalog of templates covering a myriad of applications. Also, because these templates are programmable by definition, learning can be incorporated to adapt the templates. Either globally, for example, using a genetic algorithm [16], or locally. Thus, not only associative memories can be constructed, e.g., [17], but the plasticity of the brain might be directly modeled [13].

to describe the spatial-temporal dynamics. In many cases, the layers are just cascaded, and the consecutive instructions of the CNN-UM are adequate to model the same process. However, in those cases where interlayer feedback does exist, we need the multilayer CNN structure. Such a multilayer CNN is useful for modeling the vertebrate retina [12].

Fig. 6 shows the conceptual architecture of a second-order dynamics, three-layer cell which has been prototyped in the
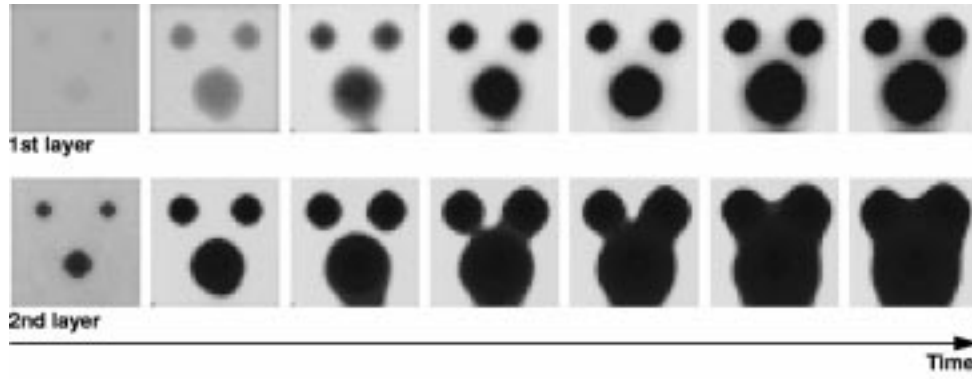
Fig. 7. Using the CACE1K chip, programming the layer time constants and the A-templates on the two dynamic layers, a double wave propagation can be programmed. The resulting sequence of snaphots shows the different speed and the different types of waves on the two layers.

chip called CACE1K [11]. The dynamic operation is given according to the following expressions:

$$\dot{x}_{1,ij} = -x_{1,ij} + \sum_{C(kl) \in S_r(ij)} A_{11}(ij; kl) \cdot y_{1,kl} + b_0 \cdot u_{ij}$$
$$+ a_{21} \cdot y_{2,ij} + z_1$$

$$\dot{x}_{2,ij} = -x_{2,ij} + \sum_{C(kl) \in S_r(ij)} A_{22}(ij; kl) \cdot y_{2,kl} + b_0 \cdot u_{ij}$$
$$+ a_{12} \cdot y_{1,ij} + z_2$$

$$x_{3,ij} = f_a(x_{1,ij}, x_{2,ij}) \qquad (4)$$

where $f_a(\cdot)$ represents the built-in difference arithmetic. The operation of this prototype is hence controlled by the 23 parameters involved in (4), given as

$$\mathbf{A}_{11} = \begin{bmatrix} a_{1,1} & a_{1,2} & a_{1,3} \\ a_{1,4} & a_{1,5} & a_{1,6} \\ a_{1,7} & a_{1,8} & a_{1,9} \end{bmatrix} \quad \mathbf{A}_{22} = \begin{bmatrix} a_{2,1} & a_{2,2} & a_{2,3} \\ a_{2,4} & a_{2,5} & a_{2,6} \\ a_{2,7} & a_{2,8} & a_{2,9} \end{bmatrix}$$
$$\mathbf{A}_{12} = a_{12} \quad \mathbf{A}_{21} = a_{21} \quad \mathbf{B}_1 = b_0 \quad z_1 \quad z_2 \qquad (5)$$

plus the relative values of the time constants of Layers 1 and 2, totaling 25 different parameters. Many types of nonlinear waves (trigger-, traveling-, auto-, and spiral-waves) can be obtained by properly controlling these parameters [23].

E. Example 2

This example illustrates the generation of double-wave propagation using the CACE1K chip [11]. The template element values for this operation are

$$\mathbf{A}_{11} = \mathbf{A}_{22} = \begin{bmatrix} 0,25 & 0,25 & 0,25 \\ 0,25 & 3,00 & 0,25 \\ 0,25 & 0,25 & 0,25 \end{bmatrix}$$
$$a_{12} = -5 \quad a_{21} = 3 \quad b_0 = 0 \quad z_1 = -1.25 \quad z_2 = 2.25$$
$$(6)$$

and the ratio between the time constants of the two layers is $\tau_1/\tau_2 = 1/5$. Using the same chip, very recently we have been able to implement some of the key inner retinal effects, impossible to realize on first-order layers. More detailed results are reported elsewhere [24].

Our quest to make a programmable prototype spatial-temporal computer which could also serve as a visual micropro-

cessor could be justified in two ways. On the one hand, we have proven earlier that the CNN-UM is universal. In a sense, it is equivalent to the Turing machine. The proof was realized by implementing the game of life. On the other hand, in each cell, with not more than four layers, we can implement any nonlinear multi-input single-output operator with fading memory. This is only one side of the story. On the other side, which is similar to the digital computers or Turing machines in which the μ-recursive functions are the formal descriptions of the algorithms with proven capabilities, we have also determined the equivalent formal notion of algorithms as the α-recursive functions with similar properties [18]. Hence, we have all the theoretical background to establish our new type of computer for topographic operations, in particular for vision. Moreover, it has turned out that the neuromorphic constructs for most of the topographic senses with accompanying processing are quite similar to those of CNN models [9].

III. Analogic Visual Microprocessor in Silicon

CNN-based analogic visual microprocessors have similarities with the so-called single instruction multiple data (SIMD) systems [25], although they work directly on analog signal representations obtained through embedded optical sensors and hence do need neither a front-end sensory plane nor analog-to-digital converters. The architecture of these visual microprocessors is illustrated in Fig. 8 through two prototype chips, namely, ACE4K [10] and ACE16K [26]. In both cases, as in other related chips [11], [27]–[29], the architecture includes a core array of interconnected elementary processing units, surrounded by a global circuitry. This latter circuitry is intended for:

- control and timing;
- adressing and buffering of the core cells;
- input/output;
- storage of user-selectable instructions (programs) to control the sequence of operations of the processing core;
- storage of user-selectable analogic programming parameter configurations (templates).
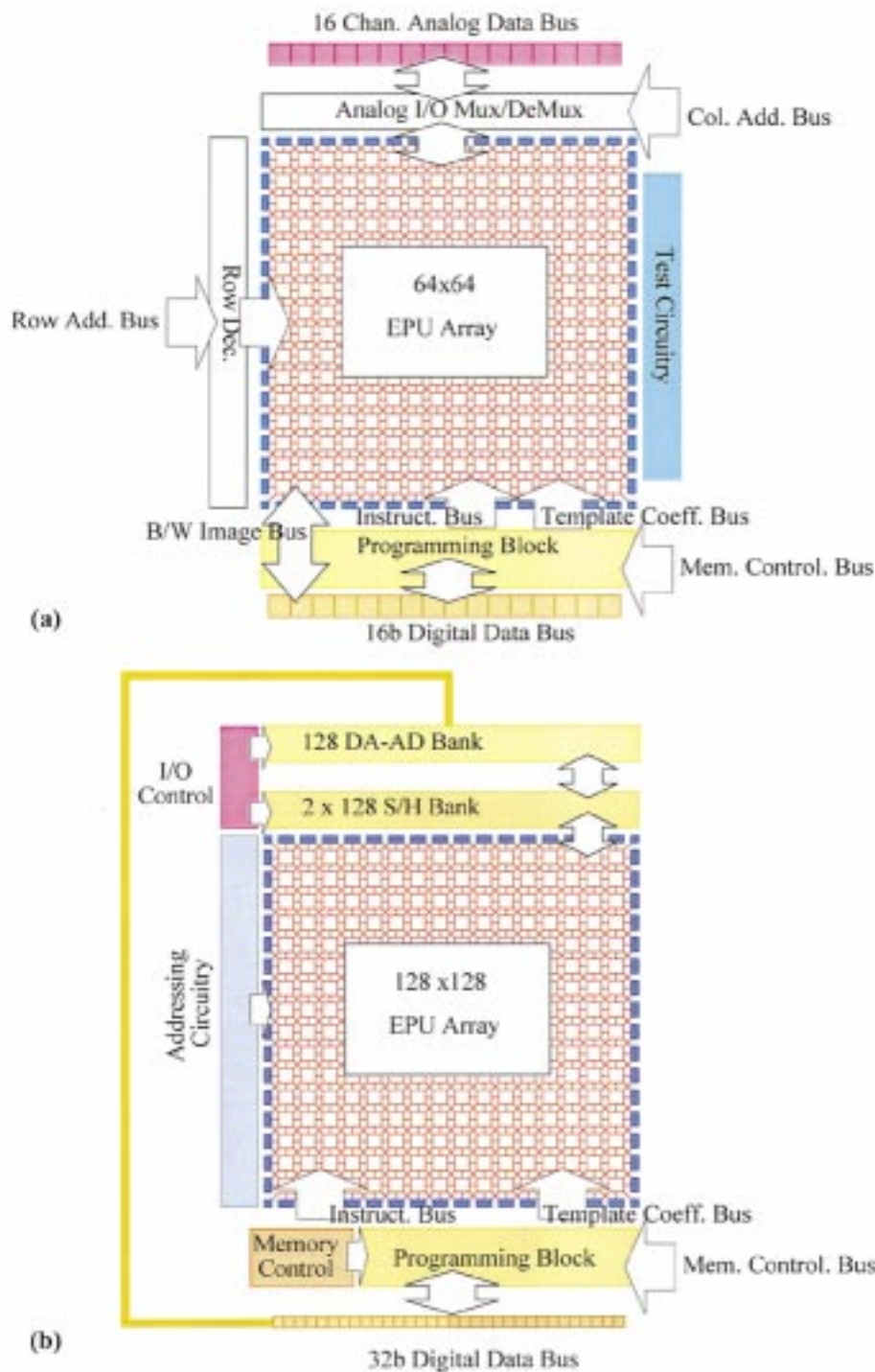
Fig. 8. Architectures of analogic visual microprocessor chips: (a) ACE4K [10] and (b) ACE16K [26].

On the other hand, the core of interconnected processing units embeds different functions on a common silicon substrate (see Fig. 9 for illustration purposes), namely:

- 2-D sensing;
- 2-D analog/digital array processing concurrent with the signal sensing;
- 2-D spatio-temporal processing determined by local, receptive-field-like programmable interconnections;

- 2-D memory banks for concurrent online uploading and downloading of short-term analog and digital data.

Several analogic visual microprocessor chips in different CMOS technologies have been reported during the last few years. Particularly, [10], [11], and [26]–[29] report those implementations with at least 20×20 pixels. Table 1 presents a summary of some of their most relevant data. Some columns
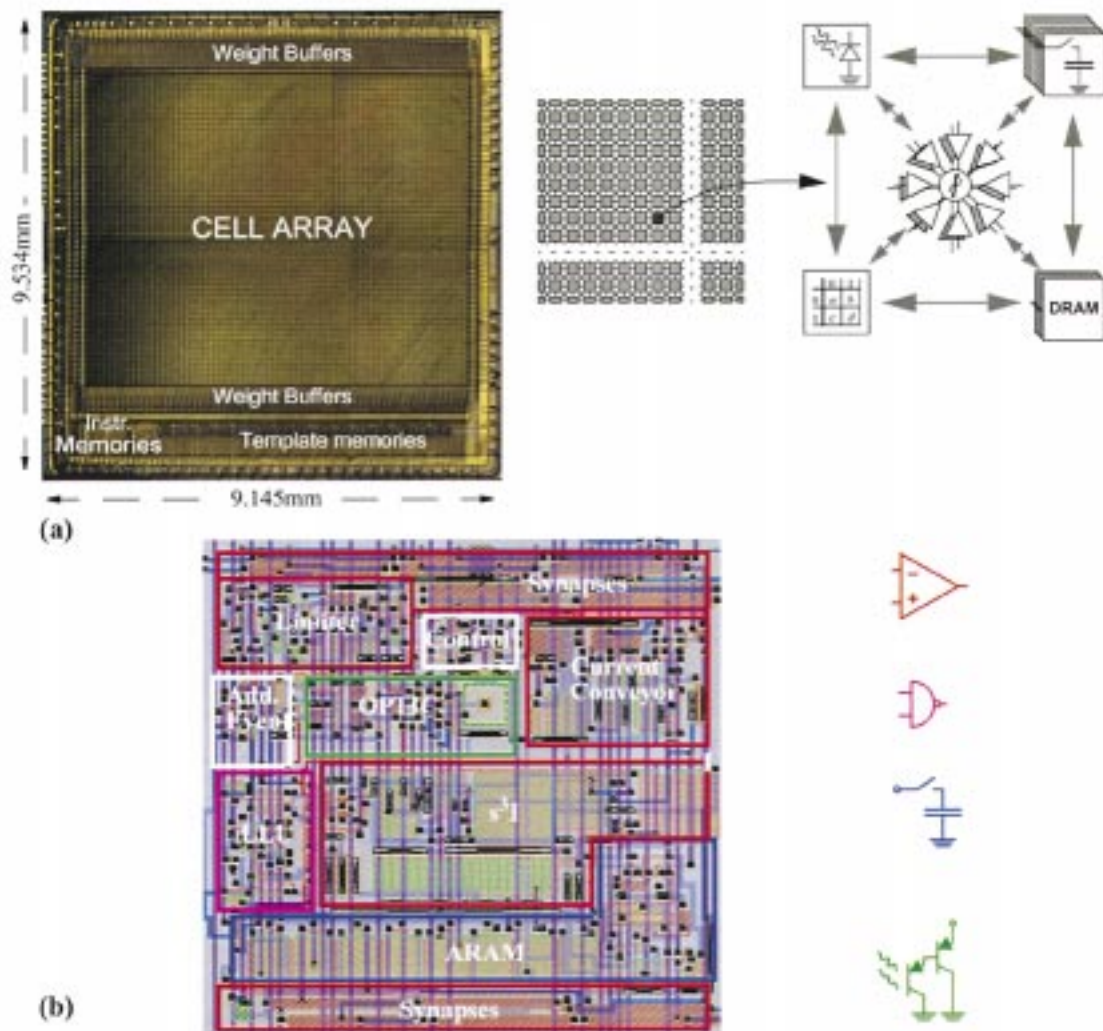
Fig. 9. llustrating the embedding of different functional features at the core processing array of visual microprocessors. (a) Microphotograph of the ACE4K chip (left) and conceptual representation of the distributed functions embedded in the core array (right). (b) Layout of a processing unit of the ACE16K showing the areas occupied by the different functions realized concurrently by the core array.

correspond to chips intended for black and white input images, while others are for chips which accept gray-scale input images. As with any other analog processing circuit, figures of merit about performance must contemplate accuracy and area occupation in addition to speed and power consumption. The speed measure here is proportional to the number of cells, the inverse of the time constant, and a weighted number of multipliers per cell. Any comparison must refer to the number of operations per second and to the accuracy. The data in the table highlights the following.

- There is a tradeoff between area occupation (cell density) and accuracy, on the one hand, and speed of operation and power consumption, on the other. This tradeoff is typical of analog integrated circuits [33].
- The evolution toward scaled-down technologies reports advantages in terms of speed and cell density. Actually, the ACE16K chip has 128×128 resolution and

is capable of realizing sequences of 64 instructions; using up to 32 different templates (each template consisting of 24 8-bit-coded analog programming values) during a sequence; loading and downloading full-size gray-scale images to and from the cache memory, and having always eight full-size images available for usage during the flow; with an internal processing time of 160 ns, and providing digitally coded output images (obtained with a battery of internal converters) with a downloading time of 0.128 ms.

The capability to design cells with maximum density, speed and accuracy, and minimum area and power consumption relies basically on the exploitation of all functional features offered by the MOS transistor. This is very different from digital design, in which only the switching capability of the MOS transistor is exploited. The design of the entities which interconnect the cells (synapses) defines one of the

Table 1
Summary and Comparison of Chip Implementations

Ref.	[28]	[27]	[29]	[10]	[26]
Tech, μm	0.8	0.7	0.5	0.5	0.35
Array Size	20 x 22	20 x 20	48 x 48	64 x 64	128x128
Pix. Format[a]	B	A[b]	B	A[c]	A[d]
Weight. Prog.[e]	8-b	Continuous [-4,4][c]	6-b	8-b	8-b
Memory per Cell	4 LLMs 1 State Capacitor	1-State 1-Input	2 In/2 Out Registers	4 LLMs 4 LAMs 1-State Cap 1-Inp. Cap	8 LAMs 2 LLU Op. 2 Flags 3 Pix. Cap.
Multipliers per cell[f]	9, 1[g]	5, 5, 1[h]	9, 9, 1	9, 9, 2	9, 1, 1, 1
Photo Sensors.	Yes	No	No	Yes	Yes Multimode Sensor
Program. Memory	8 Templ.	No	1 Templ.	32 Templ. 64 Digital Instructions	32 Templ. 4096 Dig. Instructions
τ [i]	250ns	5μs	50ns	1.2μs CNN 280ns Conv	0.8μs CNN 160ns Conv
Cells/mm^2	27.5	16.7	295	82	180
Power (W)	~1 W	375 μW/cell	300mW (max.)	250μW/cell 1.2 W Chip	180μW/cell 4W Chip
Speed	15.8 GOPS	0.53 GOPS	0.5 TOPS	40 GOPS CNN	0.19 TOPS CNN
XPS/area[j]	0.98 GOPS/mm^2	22 10^6 OPS/mm^2	64 GOPS/mm^2	1 GOPS/mm^2	3.6 GOPS/mm^2
XPS/Pow[k] (OP/J)	1.58 10^{10}	3.5 10^9	1.6 10^{12}	3.95 10^{10}	8.25 10^{10}
Electr. I/O	22 Lines Binary Bus	20 Lines Analog Bus	48-b Binary Bus	16b B. Bus 16 Lines Analog Bus	32b Digital Data Bus

a. A=Analog, B=Binary (B/W), D=Digital
b. Only B/W results are available.
c. 7.7b Equivalent Accuracy.
d. 8b Equivalent Accuracy.
e. It refers to the number of bits used to define weight parameters.
f. A, B, and z multipliers.
g. A and B multipliers are the same. The chip uses a time-multiplexing scheme.
h. Cross-shape neighourhood.
i. The time constant figure, by itself, does not say too much about the global speed of the system -time to load templates, images, etc- but, unfortunately, is the only data normally reported.
j. Speed/Area ratio.
k. Speed/Power ratio.

major issues. In order to do this, different possibilities may be chosen *a priori*, as illustrated in Fig. 10. In all cases, electrical controllability is provided by default. However, the different strategies exhibit quite a different performance in the presence of systematic and random error sources, as well as a different incidence of the global signal transmission errors. Hence, careful analysis and optimization is needed to select the best approach. Such analysis and optimization are needed to achieve the cell density and accuracy levels featured by last generation chips. The background for such procedures can be found in [3], [10], [11], [26], and [28].

IV. ABOUT SCALING DOWN

It is expected that the performance figures featured for these chips can be further enhanced as technology scales
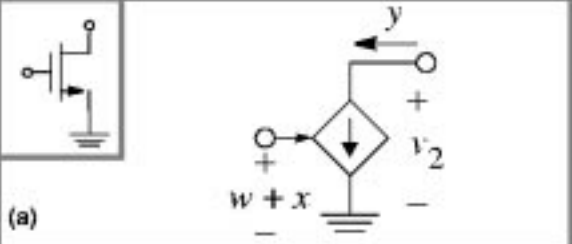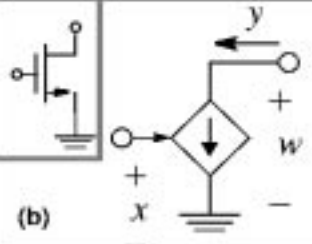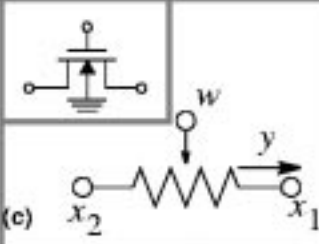
Fig. 10. Using a single NMOST for voltage-to-current transformation. Only first-order terms are included in the displayed behavioral equations.

down. However, one problem arises due to the necessity of maintaining analog accuracy, and hence the quality of the analog design, as transistor sizes decrease. Below we first identify mismatch as the main limit for the analog accuracy and then explore different tradeoffs associated with the analog design in the presence of mismatch.

A. Mismatch Versus Noise as a Limiting Factor

Mismatch makes two nominally identical devices behave differently when they are used in a real integrated circuit. Based on the formulation of mismatch as a function of device geometries in [30], the variance of the large-signal transconductance parameter β, the threshold voltage V_{T0}, and the slope factor[4] n_p as function of the device area and aspect ratio can be represented as

$$\sigma^2(V_{T0}) = \frac{A_{V_{T0}}^2}{A} + \frac{B_{V_{T0}}^2}{\sqrt{A^3 \cdot S}} + \frac{C_{V_{T0}}^2}{\sqrt{A^3 \cdot S^{-1}}}$$

$$\frac{\sigma^2(\beta)}{\beta^2} = \frac{A_\beta^2}{A} + \frac{B_\beta^2}{\sqrt{A^3 \cdot S}} + \frac{C_\beta^2}{\sqrt{A^3 \cdot S^{-1}}}$$

$$\sigma^2(n_p) = \frac{A_{n_p}^2}{A} + \frac{B_{n_p}^2}{\sqrt{A^3 \cdot S}} + \frac{C_{n_p}^2}{\sqrt{A^3 \cdot S^{-1}}} \qquad (7)$$

where A is the transistor channel area and S is the transistor aspect ratio.

Another accuracy limiting factor is noise. The equivalent noise current for an MOS transistor can be expressed as [31]

$$\frac{\overline{i_n(t)^2}}{\Delta f} = 4 \cdot k \cdot T \cdot G_{\mathrm{ch}} + \frac{K_F \cdot g_m^2}{A \cdot C_{\mathrm{ox}}^h} \cdot \frac{1}{f^{A_f}} \qquad (8)$$

where h and A_f vary between 1 and 2, $G_{\mathrm{ch}} = \beta \cdot (V_G - V_{T0} - n_p V_S)$ within the ohmic region and 2/3 of this quantity in saturation, and $g_m = \partial I_{\mathrm{DS}}/\partial V_{\mathrm{GS}}$ is the small-signal transconductance parameter.

---

[4]In the original model, the variance was formulated for the body effect factor γ. $\sigma^2(n_p)$ can be obtained as a function of $\sigma^2(V_{T0})$ and $\sigma^2(\beta)$.

Let us consider that the only significant mismatch error is that of the large-signal transconductance parameter β—as it actually happens in many practical circuits used for establishing interconnections in analog array processors [32], [33]. In terms of the transistor area A and aspect S, this error is expressed as

$$\frac{\sigma_I^2}{I_{\max}^2} = \frac{A_\beta^2}{A} + \frac{B_\beta^2}{\sqrt{A^3 \cdot S}} + \frac{C_\beta^2}{\sqrt{A^3 \cdot S^{-1}}}. \qquad (9)$$

Under similar assumptions, the noise contribution can be approximated by

$$\frac{\sigma_I^2}{I_{\max}^2} = \frac{4 \cdot k \cdot T \cdot [X_c + x_{\max} - V_{T0} - n_p \cdot W_c - n_p \cdot w_{\max}]}{\mu_0 \cdot C_{\mathrm{ox}} \cdot S \cdot x_{\max}^2 \cdot w_{\max}^2}$$

$$\cdot \Delta f + \frac{K_F}{C_{\mathrm{ox}} \cdot x_{\max}^2} \cdot \frac{\ln\left(\frac{f_{\max}}{f_{\min}}\right)}{A}. \qquad (10)$$

Using typical parameters for CMOS 0.5-μm technologies ($X_c = 2.75$ V, $x_{\max} = w_{\max} = 0.4$ V, $V_{T0} = 0.65$ V, $\mu_0 = 588$ cm^{2}V^{-1}s^{-1}, $C_{\mathrm{ox}} = 3.4$fF μm^{-2}, $K_F = 3.6 \times 10^{-25}$ V^{2}F) and considering a bandwidth of 1–5 MHz, we conclude that, for devices with channel areas of about 50 μm^2, the matching level sets an accuracy slightly above 8 b while for this same area and a channel aspect ratio of 0.1 the noise poses a limit in the resolution of 10.48 bit, far beyond from that posed by mismatching phenomena.

B. The Effect of the Scaling Process

Let us assume that lateral dimensions scale as

$$l_{\min}^{\mathrm{new}} = \frac{1}{\lambda} \cdot l_{\min}^{\mathrm{old}}. \qquad (11)$$

Thus, the gate oxide thickness, which approximately evolves in current technologies as $t_{\mathrm{ox}} \propto \sqrt{l_{\min}}$, scales as

$$t_{\mathrm{ox}}^{\mathrm{new}} \cong \frac{1}{\sqrt{\lambda}} \cdot t_{\mathrm{ox}}^{\mathrm{old}}. \qquad (12)$$
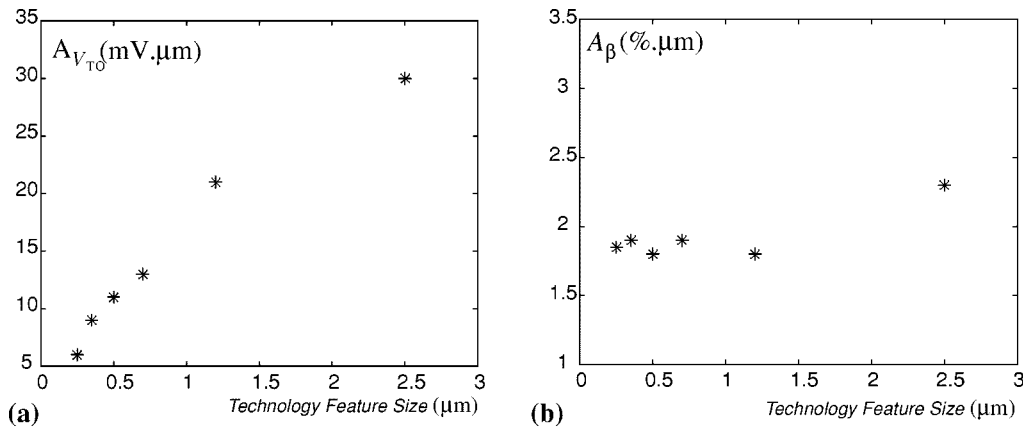
Fig. 11. (a) Historical trend of parameter A_β. (b) Historical trend of parameter $A_{V_{T0}}$ [34].

Assume that the synapse size defines the achievable cell density

$$\text{Density} \propto \frac{1}{L_X \cdot L_Y} \tag{13}$$

where L_X and L_Y are the synapse width and length. As technologies scale down, *Density* might hence evolve as

$$\text{Density}_{\text{new}} = \lambda^2 \cdot \text{Density}_{\text{old}}. \tag{14}$$

Another important parameter is the time constant which can be expressed as

$$\tau = C/g_m. \tag{15}$$

In the case that one transistor is employed to realize the synapse [32], [33], the transconductance parameter is approximately given by

$$g_m = \beta \cdot (V_D - V_S) = \mu_o \cdot C_{\text{ox}} \cdot S \cdot w \tag{16}$$

where w is the weight control signal.

On the other hand, assuming that the capacitor is implemented by using the gate capacitance of an MOS transistor, the capacitance value neglecting border effects is approximately given by

$$C \cong C_{\text{ox}} \cdot A. \tag{17}$$

From (16) and (17), the time constant becomes

$$\tau = \frac{1}{\mu_o \cdot w} \cdot \frac{A}{S}. \tag{18}$$

Hence, it might ideally scale as

$$\tau_{\text{new}} = \tau_{\text{old}} \cdot \lambda^{-2}. \tag{19}$$

Unfortunately, the density and speed enhancements reported by (14) and (19) cannot be realized in practice due to the necessity of keeping the analog accuracy. The question is, what happens with the technological parameters related to the accuracy when the technology scales down? Do they also scale down? The answer is that not all of them scale as feature size does. The historical trend shows [34] that scaling down produces a reduction of the main parameter related to V_{T0} mismatching, namely the parameter, $A_{V_{T0}}$ [see Fig. 11(a)]. However, as already mentioned, accuracy in the behavior of the one transistor synapse is mainly affected by random fluctuations on the parameter β [32], [33]. Errors of the synapse current are approximately given by

$$\frac{\sigma_I^2}{I_{\text{max}}^2} \approx \frac{A_\beta^2}{A}. \tag{20}$$

Fig. 11(b) shows that the A_β parameter has remained practically unchanged as feature size was scaled down. Hence, synapse errors evolve as

$$\frac{\sigma_I^2\big|_{\text{new}}}{I_{\text{max}}^2\big|_{\text{new}}} \approx \frac{A_{\text{old}}}{A_{\text{new}}} \frac{\sigma_I^2\big|_{\text{old}}}{I_{\text{max}}^2\big|_{\text{old}}}. \tag{21}$$

Consequently, if transistors are designed such that their channel areas are scaled down by λ^2, then, the relative error $\varepsilon = \sigma_I/I_{\text{max}}$ will grow according to

$$\varepsilon_{\text{new}} = \varepsilon_{\text{old}} \cdot \lambda \tag{22}$$

Accuracy can only be kept by maintaining approximately the same absolute channel area. Of course this statement is valid provided that the empirical trend depicted in Fig. 11(b) remains.

C. Design Tradeoffs

Among many other things, analog design art consists mainly in the combination of many design equations involving area occupation, power consumption, speed, and accuracy. Typically, the objective is to meet the design requirements by minimizing (or maximizing) a certain figure of merit (FOM), using the channel areas and aspect ratios of the transistors as design variables.

Unfortunately, as already highlighted in previous section, it is not possible to optimize all figures simultaneously; instead, tradeoffs among the different figures must be considered.

1) Accuracy Versus Density: The dependence of mismatch on the channel aspect ratio is low for moderately large values of the channel areas. Due to this, the channel

area A is constrained by the required accuracy and it may therefore be said that the precision P satisfies

$$P^{-2} = \frac{A_\beta^2}{A} \qquad (23)$$

where P is defined as

$$P = \frac{I_{\max}}{\sigma_I}. \qquad (24)$$

On the other hand, the density of synapses, that is, the number of synapses per area unit, can be basically expressed as

$$\text{Density} = \frac{K_{\text{area}}}{A} \qquad (25)$$

where K_{area} is a constant which includes the influence of the routing lines and diffusion regions on the achievable density.

Hence, a first tradeoff can be formulated as

$$\frac{P^{-2}}{\text{Density}} = A_\beta^2 \cdot K_{\text{area}}. \qquad (26)$$

Accordingly, maximum achievable accuracy and cell density cannot be optimized separately since the greater the accuracy, the smaller the density and vice versa.

2) Speed Versus Power: The maximum power consumption of a synapse is expressed as

$$\begin{aligned}
\text{Pow} &= V \cdot I = w_{\max} \cdot \mu_o C_{\text{ox}} S \cdot x_{\max} \cdot w_{\max} \\
&= \mu_o \cdot C_{\text{ox}} \cdot S \cdot w_{\max}^2 \cdot x_{\max}
\end{aligned} \qquad (27)$$

while the minimum time constant, corresponding to the maximum weight value, is given by

$$\tau = \frac{1}{\mu_o \cdot w_{\max}} \cdot \frac{A}{S} = \text{Speed}^{-1}. \qquad (28)$$

Therefore,

$$\frac{\text{Pow}}{\text{Speed}} = A \cdot C_{\text{ox}} \cdot x_{\max} \cdot w_{\max}. \qquad (29)$$

Consequently, it seems that the only way to minimize this figure, i.e., reduce the power consumption and increase the speed, is by reducing the synapse area. Nevertheless, this automatically leads to a reduction of the achievable accuracy. On the other hand, reducing the signal ranges, $x_{\max}$ or $w_{\max}$, will directly degrade the signal-to-noise ratio (SNR) and thus the accuracy.

A global FOM involving speed accuracy and tradeoff can be formulated in the following way:

$$\frac{\text{Pow} \cdot \text{Speed}^{-1}}{P^2} = A_\beta^2 \cdot C_{\text{ox}} \cdot x_{\max} \cdot w_{\max}. \qquad (30)$$

Since A_β does not show any evolution as technology is scaled down, this FOM only depends on the technology scaling process as C_{ox} does. Therefore, since $C_{\text{ox}} \propto \sqrt{\lambda}$, it is expected that the FOM will worsen in the future.

V. COMPUTATIONAL INFRASTRUCTURE FOR TERAOPS OPERATION

A. Computational Infrastructure

Practical stored programmability requires a standard computational infrastructure and a high-level language, operating system, and software library for the analogic software. Moreover, the computational infrastructure should rely on the existing PC culture and should be transparent to digital systems. The details of the computational infrastructure and the chip set architecture have been published elsewhere [35]. Presently, analogic CNN visual microprocessors are supporting TeraOPS equivalent digital computing speed, and rates of more than 10 000 frames/s have been tested.

B. Programmable Neuromorphic Vision Models

Many parts of the visual pathway, in different animals and in humans, have been recently studied in detail. As to the retina, see, e.g., [36] and the recent breakthrough in [4]. As with the retinotopic neuromorphic vision models, the three basic structures of the spatial–temporal models are as follows:

- layers with given receptive fields combined in a cascade structure;
- allowing interlayer feedback (e.g., in the prototype complex cell structure);
- the combination of an ON and OFF pathway (or an excitatory and an inhibitory flow).

Recognize that in these models there is no discretization in time.

These structures are implementable on CNN (see, e.g., the first results in [15]). On the other hand, it is impractical to build special chips for each visual effect (e.g., for edge detection, histogram equalization, motion detection, length tuning, directional sensitivity, and detecting a typical morphology). Moreover, if we want to make a visual prosthesis, programmability might be mandatory.

In the next example, we show a typical channel of a multilayer CNN retina model reflecting the basic new concepts of mammalian retinal operation [4]. Observe that in the cascade structure there are many interlayer feedback parts. In addition, the two paths of signals represent the ON and OFF visual pathway.

C. Example 3

The flow diagram of a typical vertebrate retina model is shown in Fig. 12. Snapshots of a moving head are also presented. Based on [4], it is known that in a mammalian retina there are about a dozen parallel channels embedded in the inner part of the retina. Here we show one typical and simple channel. The interested reader can consult [24] and its reference publications.

VI. COMPUTATIONAL COMPLEXITY

Classical computational complexity studies are based on the digital computer, in particular the Turing Machine. Recently, a first step in the direction of breaking this powerful
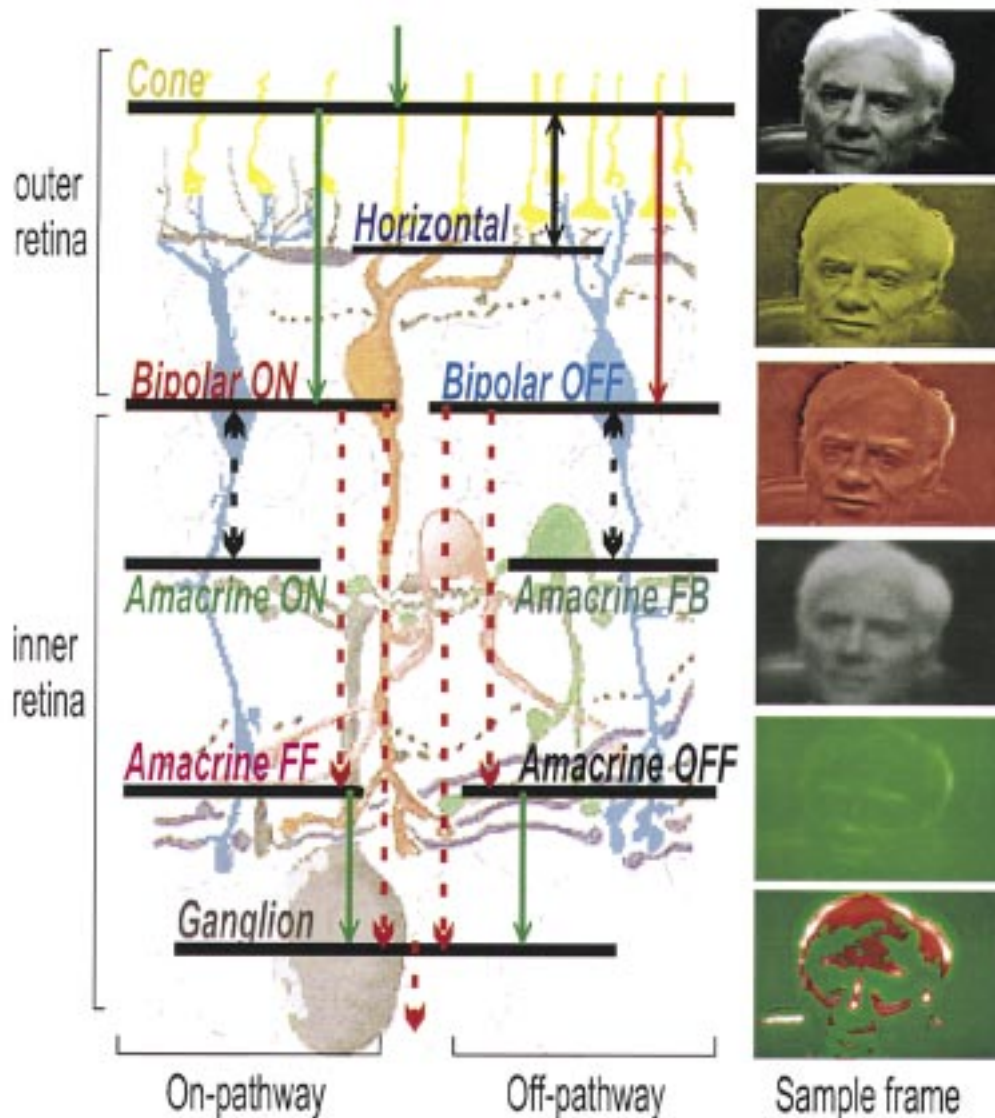
Fig. 12. Retina modeling. The left side, showing also a drawing of the interacting general neuron types in the retina, presents the multilayer CNN structural elements of the ON–OFF retina model. The neurons in the retina are organized into 2-D layers modeled with CNN layer(s). A neuron in a given layer interacts with another neuron in another layer through synapses, which have their own dynamics and temporal characteristics. The layers are depicted by horizontal lines and the interlayer synapses by vertical arrows. The circle represents the intralayer coupling, which is a space-constant-dependent diffusion. The dashed lines stand for nonlinear transfer functions. The right side is a sequence of the sample frames from a processed natural scene video in one particular (local edge detector) model. The topmost picture is the input and the others are the responses in some computed layers. The green color indicates the inhibition, the red regions correspond to the excitation and the white spots stand for the spiking, to the output of the retina.

but rigid framework has been made by introducing a still iterative computational complexity theory based on real values [37]. The CNN-UM defines a computing platform one step further: it is a machine based on flows, or real-valued image flows [18].

Computing is a physical process. While the classical complexity theory was basically good for logic operations and for dealing with the combinatorial complexity, as well as a part of the number-crunching tasks (but still missing the semantic aspects), it cannot even capture the problem of chaotic signals or nonlinear waves. The latter, as we have seen, is completely common in visual models. The principal question is practical: how long does it take to solve a problem on a given piece of silicon within a given power dissipation? The answer is not only dependent on the size of the problem, but more importantly on the parameters of the operator. Recent results show some possible answers in this direction [18]. As a part of this endeavor, the notion of an analogic cellular algo-

rithm has been developed via the α-recursive functions. As the α-recursive function is the basis for digital algorithms (they are basic components of the C language as well), the μ-recursive function is the basis for analogic cellular software and the Alpha language used for it [35]. It has been proven that the CNN-UM is a minimal implementation for the α-recursive functions.

VII. CONCLUSION

We have shown some basic notions, architectures, CMOS implementations, computational infrastructures as well as the biological plausibility for a visual microprocessor. Operating focal plane visual microprocessors and its accompanying computational infrastructure with analogic visual software are available. It has been shown that the integrated sensing and stored programmable processing principle is crucial in any complex vision-related tasks, including the whole process from sensors to visual understanding.

REFERENCES

[1] C. Koch and H. Li, Eds., *Vision Chips, Implementing Vision Algorithms with Analog VLSI Circuits*. New York: IEEE Press, 1995.
[2] A. Moini, *Vision Chips*. Norwell, : Kluwer, 2000.
[3] T. Roska and A. Rodríguez-Vázquez, Eds., *Toward the Visual Microprocessor*. New York: Wiley, 2000.
[4] B. Roska and F. S. Werblin, "Vertical interactions across ten parallel, stacked representations in the mammalian retina," *Nature*, vol. 410, pp. 583–587, Mar. 2001.
[5] L. O. Chua and L. Yang, "Cellular neural networks: Theory and applications," *IEEE Trans. Circuits Syst.*, vol. 35, pp. 1257–1290, 1988.
[6] L. O. Chua and T. Roska, "The CNN paradigm," *IEEE Trans. Circuits Syst. I*, vol. 40, pp. 147–156, Mar. 1993.
[7] T. Roska and L. O. Chua, "The CNN universal machine: An analogic array computer," *IEEE Trans. Circuits Syst. II*, vol. 40, pp. 163–173, Mar. 1993.
[8] F. Werblin, T. Roska, and L. O. Chua, "The analogic cellular neural network as a bionic eye," *Int. J. Circuit Theory Applicat.*, vol. 23, pp. 541–549, 1995.
[9] L. O. Chua and T. Roska, *Cellular Neural Networks and Visual Computing*. Cambridge, U.K.: Cambridge Univ. Press, 2002.
[10] G. Liñán, P. Földesy, S. Espejo, R. Domínguez-Castro, and A. Rodríguez-Vázquez, "A 0.5 μm CMOS 10^6 transistor analog programmable array processor for real-time image processing," in *Proc. 1999 Eur. Solid-State Circuits Conf.*, Sept., pp. 358–361.
[11] R. Carmona, P. Garrido, R. Domínguez-Castro, S. Espejo, and A. Rodríguez-Vázquez, "Bioinspired analog vlsi design realizes programmable complex spatio-temporal dynamics on a single chip," in *Proc. 2002 Conf. Design and Test in Europe*, to be published.
[12] D. Bálya, B. Roska, E. Nemeth, T. Roska, and F. S. Werblin, "A qualitative model framework for spatio-temporal effects in vertebrate retina," *Proc. 2000 IEEE Conf. Cellular Neural Networks and Their Applications*, pp. 165–170, 2000.
[13] T. Roska, "Computer-sensors: Spatial-temporal computers for analog array signals, dynamically integrated with sensors," *J. VLSI Signal Process. Syst.*, vol. 23, pp. 221–238, 1999.
[14] S. Espejo, R. Carmona, R. Domínguez-Castro, and A. Rodríguez-Vázquez, "A VLSI-oriented continuous-time CNN model," *Int. J. Circuit Theory Applicat.*, vol. 24, pp. 341–356, May–June 1996.
[15] T. Roska, J. Hámori, E. Lábos, K. Lotz, L. Orzó, J. Takács, P. Venetianer, Z. Vidnyánszky, and Á. Zarándy, "The use of CNN models in the subcortical visual pathway," *IEEE Trans. Circuits Syst. I*, vol. 40, pp. 182–195, Mar. 1993.
[16] T. Kozek, T. Roska, and L. O. Chua, "Genetic algorithm for CNN template learning," *IEEE Trans. Circuits Syst. I*, vol. 40, pp. 392–402, June 1993.
[17] P. Szolgay, I. Szatmári, and K. László, "A fast fixed point learning method to implement associative memory on CNN's," *IEEE Trans. Circuits Syst. I*, vol. 44, pp. 362–366, 1997.
[18] T. Roska, "Analogic wave computers—Wave-type algorithms: Canonical description, computer classes, and computational complexity," *Proc. 2001 IEEE Int. Symp. Circuits and Systems*, pp. 41–44, 2001.
[19] P. Szolgay, G. Vörös, and Gy. Eröss, "Applications of the cellular neural network paradigm in mechanical vibrating systems," *IEEE Trans. Circuits Syst. I*, vol. 40, pp. 222–227, Mar. 1993.
[20] T. Roska, L. O. Chua, D. Wolf, T. Kozek, R. Tetzlaff, and F. Puffer, "Simulating nonlinear waves and partial differential equations via cnn—Part I: Basic techniques," *IEEE Trans. Circuits Syst. I*, vol. 42, pp. 807–815, Oct. 1995.
[21] L. Alvarez and J. M. Morel, "Morphological approach to multiscale analysis," in *Geometry-Driven Diffusion in Computer Vision*, B. M. H. Romeny, Ed. Norwell, MA: Kluwer, 1994, pp. 229–249.
[22] C. Rekeczky, Á. Tahy, Z. Végh, and T. Roska, "CNN-based spatio-temporal nonlinear filtering and endocardial boundary detection in echocardiography," *Int. J. Circuit Theory Applicat.*, vol. 27, pp. 171–207, 1999.
[23] C. Rekeczky and L. O. Chua, "Computing with front propagation: Active contour and skeleton models in continuous-time CNN," *J. VLSI Signal Process. Syst.*, vol. 23, pp. 373–402, 1999.
[24] D. Bálya, C. Rekeczky, and T. Roska, "Basic mammalian retinal effects on the prototype complex cell CNN universal machine," in *Proc. IEEE 7th Int. Workshop Cellular Neural Networks and Their Applications*, 2002, pp. 251–258.
[25] J. C. Gealow and C. G. Sodini, "A pixel-parallel image processor using logic pitch matched to dynamic memory," *IEEE J. Solid-State Circuits*, vol. 34, pp. 831–839, June 1999.
[26] G. Liñán, R. Domínguez-Castro, S. Espejo, and A. Rodríguez-Vázquez, "ACE16K: An advanced focal-plane analog programmable array processor," in *Proc. 2001 Eur. Solid-State Circuits Conf.*, Villach, Austria, Sept. 2001, pp. 216–219.
[27] P. Kinget and M. Steyaert, *Analog VLSI Integration of Massive Parallel Processing Systems*. Norwell, MA: Kluwer, 1997.
[28] R. Domínguez-Castro *et al.*, "A 0.8 μm CMOS 2-D programmable mixed-signal focal-plane array processor with on-chip binary imaging and instruction storage," *IEEE J. Solid-State Circuits*, vol. 32, pp. 1013–1026, 1997.
[29] A. Paasio, A. Dawidziuk, K. Halonen, and V. Porra, "Minimum size 0.5 μm CMOS programmable CNN test chip," in *Proc. 1997 Eur. Conf. Circuit Theory and Design*, Budapest, Hungary, Sept. 1997, pp. 154–156.
[30] M. J. M. Pelgrom *et al.*, "Matching properties of MOS transistors," *IEEE J. Solid-State Circuits*, vol. 24, pp. 1433–1440, Oct. 1989.
[31] E. A. Vittoz, "Future of analog VLSI in the VLSI environment," *Proc. 1990 IEEE ISCAS*, pp. 1372–1390.
[32] R. Domínguez-Castro, S. Espejo, A. Rodríguez-Vázquez, and R. Carmona, "A one-transistor-synapse strategy for electrically-programmable massively-parallel analog array processors," in *IEEE-CAS 1997 Region 8 Workshop on Analog and Mixed IC Design*, ISBN 0-7803-4240-2, Sept., pp. 117–122.
[33] A. Rodríguez-Vázquez, E. Roca, M. Delgado-Restituto, S. Espejo, and R. Domínguez-Castro, "MOST-based design and scaling of synaptic interconnections in VLSI analog array processing chips," *J. VLSI Signal Process. Syst. Signal, Image Video Technol.*, vol. 23, pp. 239–266, Nov./Dec. 1999.
[34] M. Steyaert *et al.*, "Speed-power-accuracy trade off in high-speed analog-to-digital converters: Now and in the future," in *Proc. 9th Workshop in Analog Circuit Design*, Apr. 2000.
[35] T. Roska, A. Zarándy, S. Zöld, P. Földesy, and P. Szolgay, "The computational infrastructure of analogic CNN computing—Part I: The CNN-UM chip prototyping system," *IEEE Trans. Circuits Syst. I*, vol. 46, pp. 261–268, 1999.
[36] F. Werblin, A. Jacobs, and J. Teeters, "The computational eye," *IEEE Spectrum*, vol. 33, pp. 30–37, May 1996.
[37] L. Blum, F. Cucker, M. Shub, and S. Smale, *Complexity and Real Computation*. New York: Springer, 1998.

Tamás Roska (Fellow, IEEE) received the Diploma in electrical engineering from the Technical University of Budapest, Budapest, Hungary, in 1964 and the Ph.D. and D.Sc. degrees from the National Qualification Committee, Hungarian Academy of Sciences, Budapest, Hungary, in 1973 and 1982, respectively.

Since 1964, he has held various research positions. During 1964–1970, he was with the Measuring Instrument Research Institute, Budapest, between 1970 and 1982 with the Research Institute for Telecommunication, Budapest (serving also as the head of Department for Circuits, Systems and Computers) and since 1982, he has been with the Computer and Automation Institute of the Hungarian Academy of Sciences where, for 15 years, he has been the head of the Analogic and Neural Computing Research Laboratory. He has taught several courses at various universities, presently, at the Technical University of Budapest, at the University of California, Berkeley, and very recently at the Pázmány P Catholic University in Budapest. He is teaching courses on "Emergent Computations" and "Cellular Neural Networks." In 1974 and each year since 1989, he has been a Visiting Scholar at the Department of Electrical Engineering and Computer Sciences and the Electronics Research Laboratory, and recently a Visiting Research Professor at the Vision Research Laboratory of the University of California, Berkeley. He also presently serves as a Dean of the Faculty of Information Technology at the Pázmány P. Catholic University, Budapest. His main research areas are cellular neural networks, nonlinear circuit and systems, neural circuits, visual computing, and analogic spatial-temporal supercomputing. He has published more than 200 research papers and four books (some as a coauthor), and held several guest seminars at various universities and research institutions in Europe, USA, and Japan. He is a co-inventor of the CNN Universal Machine (with L. O. Chua), a U.S. patent of the University of California with worldwide protection, and the analogic CNN Bionic Eye (with F. Werblin and L. O. Chua), another U.S. patent of the University of California. He has contributed also to the development of various physical implementations of these inventions making this Cellular Analogic Supercomputer a reality.

Dr. Roska received the IEEE Fellow award for contributions to the qualitative theory of nonlinear circuits and the theory and design of programmable cellular neural networks. In 1993, he was elected to be a member of the Academia Europaea (European Academy of Sciences, London, U.K.) and the Hungarian Academy of Sciences. For technical innovations he received the D. Gabor Award for establishing a new curriculum in information technology, and for his scientific achievement he was awarded the A. Szentgyörgyi Award and the Széchenyi Award, respectively. In 1994, he became the elected active member of the Academia Scientiarium et Artium Europaea (Salzburg, Austria). In 2002, he received the Bolyai Award in Hungary. Since 1975, he has been a member of the Technical Committee on Nonlinear Circuits and Systems of the IEEE Circuits and Systems Society. Between 1987–1989, he was the founding Secretary and later he served as Chairman of the Hungary Section of the IEEE. Recently, he has served twice as Associate Editor of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—I, FUNDAMENTAL THEORY AND APPLICATIONS and for 2002–2003 he has been appointed as the Editor-in-Chief of this journal. He has served as Guest Co-Editor of special issues on cellular neural networks of the *International Journal of Circuit Theory and Applications* (1992, 1996, 1998, 2000), the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS (1993 and 1999), and the *Journal of VLSI Signal Processing Systems* (1999). He is a member of the Editorial Board of the *International Journal of Circuit Theory and Applications*. He is a member of the Technical Committee on Multimedia and the Technical Committee on Neural Networks of the IEEE. In 1998, he established and became the first Chair of the Technical Committee on Cellular Neural Networks and Array Computing of the IEEE Circuits and Systems Society. In 2000 he received the IEEE Millenium Medal and the Golden Jubilee Award of the IEEE Circuits and Systems Society.

Ángel Rodríguez-Vázquez (Fellow, IEEE) is a Professor of Electronics at the Department of Electronics and Electromagnetism, University of Seville, Seville, Spain. He is also a member of the research staff of the Institute of Microelectronics of Seville—Centro Nacional de Microelectrónica (IMSE-CNM)—where he is heading a research group on Analog and Mixed-Signal Integrated Circuits. His research interests are in the design of analog interfaces for mixed-signal circuits, CMOS imagers and vision chips, telecom circuits, neuro-fuzzy controllers, symbolic analysis of analog integrated circuits, and optimization of analog integrated circuits. In these fields, he has published 5 books, 23 book chapters in other books, around 100 journal papers, and more than 250 conference papers.

Dr. Rodríguez-Vázquez served as an Associate Editor of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS-I, FUNDAMENTAL THEORY AND APPLICATIONS (IEEE TCAS-I) from 1993 to 1995, as Guest Editor of the IEEE TCAS–I special issues on "Low-Voltage and Low-Power Analog and Mixed-Signal Circuits and Systems" (1995) and "Bio-Inspired Processors and Cellular Neural Networks for Vision" (1999), as Guest Editor of the IEEE TCAS-II special issue on "Advances in Nonlinear Electronic Circuits" (1999), and as chair of the IEEE Circuits and Ssytems Analog Signal Processing Committee (1996). Currently, he is an Associate Editor for IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS-II. He was corecipient of the 1995 Guillemin-Cauer award of the IEEE Circuits and Systems Society, the best paper award of the 1995 European Conference on Circuit Theory and Design, and the 1999 best paper award of the *International Journal on Circuit Theory and Applications*. In 1992 he received also the young scientist award of the Seville Academy of Science.