A Focal-Plane Image Processor for Low Power Adaptive Capture and Analysis of the Visual Stimulus

C. M. Domínguez-Matas*, R. Carmona-Galán, F. J. Sánchez-Fernández, A. Rodríguez-Vázquez
Instituto de Microelectrónica de Sevilla-Centro Nacional de Microelectrónica
Consejo Superior de Investigaciones Científicas (CSIC) y Universidad de Sevilla
Av. Reina Mercedes s/n, 41012 Sevilla, Spain
e-mail: rcarmona@imse.cnm.es

Abstract— **Portable applications of artificial vision are limited by the fact that conventional processing schemes fail to meet the specifications under a tight power budget. A bio-inspired approach, based in the goal-directed organization of sensory organs found in nature, has been employed to implement a focal-plane image processor for low power vision applications. The prototype contains a multi-layered CNN structure concurrent with 32×32 photosensors with locally programmable integration time for adaptive image capture with on-chip local and global adaptation mechanisms. A more robust and linear multiplier block has been employed to reduce irregular analog wave propagation ought to asymmetric synapses. The predicted computing power per power consumption, 142MOPS/mW, is orders of magnitude above what rendered by conventional architectures.**

I. Introduction

In higher primates, e. g. humans, vision is the dominant sensory modality. This means that: a) it is the principal pathway for the acquisition of information from the environment, and b) when conflict with other senses occurs, the interpretation of the physical world rendered by the visual system is favored [1]. However, as wireless sensor networks develop [2] and the so-called ambient intelligence is starting to be implemented [3], vision is not yet widely applied, although it would provide the most useful port of entry of the physical data into the virtual world. The reason for this is the high cost, the requirement of a high power budget and the low efficiency of vision devices based upon conventional digital signal processing. Such drawbacks are particularly limitative for applications such as robotic vision [4], or retinal prosthesis for the blind [5], where efficiency is a must. Where conventional digital processors, with a serial processing scheme, fail to meet the specifications, biological systems, which are goal-directed, achieve unpaired performance by means of adaptation to the nature of the stimuli. This adaptation is both architectural and circuital. On one side, multidimensional sensory signals are processed by aggregates of cells of the equivalent dimensionality. On the other, optimized biological circuitry implements the operators involved by means of their physical constitution, mainly by analogy.

In order to overcome the limitations experienced in conventional image processing schemes, we have adopted an bio-inspired architecture that provides for spatio-temporal processing in the focal-plane [6]. We have also devised a robust implementation of the model that outperforms previous VLSI implementations in terms of accuracy, at the expense of some programming flexibility.

II. Focal-Plane Processing Unit

The architecture of the multi-layered array processor that we have implemented is, basically, that of the CNN Universal-Machine [6]. It consists in a central matrix of processing elements, or cells, containing analog and logic operators and memories (Fig. 1), which function in parallel according to a global program. This program encloses the parameters that determine the network dynamics. The outcome of the signal processing is the final state of the evolution of the network. These dynamics can be described in terms of the input ($\mathbf{u}_k$), state ($\mathbf{x}_k$) and output ($\mathbf{y}_k$) variables. Each layer, k, of the array follows the evolution law expressed by:

$$\tau_k \frac{d\mathbf{x}_k(t)}{dt} = -\mathbf{g}[\mathbf{x}_k(t)] + \sum_n [\mathbf{A}_{kn} \otimes \mathbf{y}_n + \mathbf{B}_{kn} \otimes \mathbf{u}_n] + \mathbf{z}_k \qquad (1)$$

where the symbol ⊗ stands for the linear convolution between the feedback and feedforward templates, $\mathbf{A}_{kn}$ and $\mathbf{B}_{kn}$, with the output and input matrices of layer n, where n can be 1, 2 or 3:

$$\begin{aligned}[\mathbf{A}_{kn} \otimes \mathbf{y}_n](i,j) &= \sum_{l=-r}^{r}\sum_{m=-r}^{r} \mathbf{A}_{kn}(l,m)\mathbf{y}_n(i+l, j+m) \\ [\mathbf{B}_{kn} \otimes \mathbf{u}_n](i,j) &= \sum_{l=-r}^{r}\sum_{m=-r}^{r} \mathbf{B}_{kn}(l,m)\mathbf{u}_n(i+l, j+m)\end{aligned} \qquad (2)$$

* Now with Anafocus Ltd. (Spain)

This work is partially supported by projects TIC2003-09817-C02-01 of the Spanish MCyT, and N-00014-02-1-0884 of ONR.

1-4244-0921-7/07 $25.00 © 2007 IEEE.

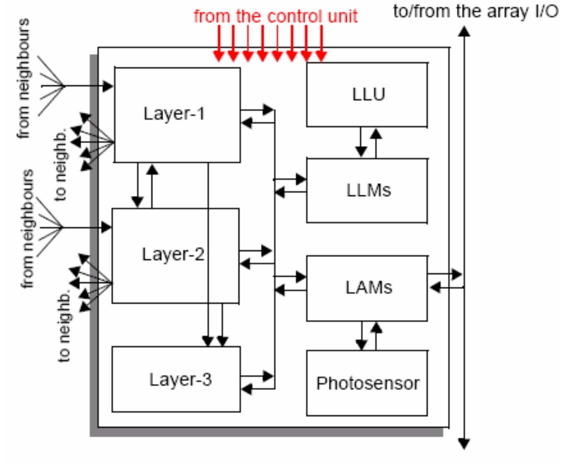

Figure 1. Conceptual diagram of the basic cell.

where r is the neighborhood radius. In this particular implementation, τ_1 and τ_2 are comparable while τ_3 is much smaller than the others. If the full-signal-range CNN model is employed [7], the output and state variables can be identified. In this conditions, each matrix element in Eq. (2) is obtained from the multiplication of the state (or input) variable by a programmable weight. These operators, responsible for multiplying the state (or input) variable by a programmable weight, are termed synapses or synaptic blocks in this context. They are basically four quadrants multipliers in which linearity with the state (or input) variable and a symmetric characteristic are strongly desired. The effect of varying the programmed weights is to modify the network dynamics, and thus, changing the type of processing realized by the array.

Continuing with the evolution law, there is the losses term:

$$g[\mathbf{x}_k(i,j)] = \lim_{m_o\to\infty}\begin{cases} m_o[\mathbf{x}_k(i,j)-1]+m_c & \text{if} \quad \mathbf{x}_k(i,j)>1 \\ m_c\mathbf{x}_k(i,j) & \text{if} \quad |\mathbf{x}_k(i,j)|\le 1 \\ m_o[\mathbf{x}_k(i,j)+1]-m_c & \text{if} \quad \mathbf{x}_k(i,j)<-1 \end{cases} \quad (3)$$

and the activation function, to generate the output:

$$\mathbf{y}_k(i,j) = f[\mathbf{x}_k(i,j)] = \frac{1}{2}\lim_{m\to m_c}\left\{\frac{1}{m}\left[|\mathbf{x}_k(i,j)+m| - |\mathbf{x}_k(i,j)-m|\right]\right\} \quad (4)$$

In both equations, m_c can be 0 or 1 for hard or sigmoidal type nonlinearity, respectively.

The physical realization of the elementary processing unit of the CNN starts with the selection of the appropriate format for the representation of the signals. On one side, voltages can be easily delivered to neighboring areas by connecting wires to high-impedance nodes. Therefore, input, output and state variables are chosen to be represented by the matrices of voltages $\mathbf{V}_u$, $\mathbf{V}_y$ and $\mathbf{V}_x$, respectively. On the other side, signal addition can be easily realized in the form of currents wired together to a virtual ground. Hence, the summands in the second member of Eq. (1) should be represented by currents. And then, this sum of currents will be integrated in the state capacitor to obtain the instantaneous value of the state voltage:

$$C_k\frac{d\mathbf{V}_{x,k}(t)}{dt} = -\mathbf{G}_g[\mathbf{V}_{x,k}(t)] + \sum_n[\mathbf{G}_{A,kn}\otimes\mathbf{V}_{y,n} + \mathbf{G}_{B,kn}\otimes\mathbf{V}_{u,n}] + \mathbf{I}_{z,k} \quad (5)$$

As stated, the elements of the feedback and feedforward templates are now programmable linear transconductances, $\mathbf{G}_{A,kn}(i,j)$ and $\mathbf{G}_{B,kn}(i,j)$. Multiplied by the input and output voltages, they render the neighborhood contributions in the form of currents. Thus, the synaptic block is a transconductor whose output current is proportional, in the ideal case, to the product of the state (or input) variable and the weight. The double transformation implicit in Eq. (5), V-I and then I-V, allows for a compact realization of the processing node, achieving higher cell densities, meaning an array size of practical interest and, besides, a tolerable fill factor.

The accuracy of these terms is very important to accomplish a correct operation of the network, since the synapse offsets, as well as every mismatch on ideally symmetric weights, are integrated in the state capacitor. Precisely, in the implementation of four-quadrant multipliers, one of the common difficulties is to maintain the symmetry with respect to the origin of the weights. A mismatch in weights having the same absolute value but opposite signs can modify the dynamic routes of the cells in the network, ending in displaced equilibrium points, and thus, distorting the prescribed processing. The main linearity concerns are found in the V-I conversion, as linear current integration, and thus I-V transformation, can be provided by available highly linear double-poly capacitors. In this design, we have employed a linearized OTA [8] in order to generate the unitary current contribution. Though the elementary transconductor achieving V-I conversion has a larger number of transistors than the single-transistor synapse in [9], advantages in the linearity with the state (or input) variable and symmetry of the V-I characteristic justify its use. In addition, the supporting circuitry can be simplified resulting in a more robust implementation finally without any area penalty.

Fig. 2 depicts the schematics of the elementary dynamic processor. The transconductor responsible of transforming the state capacitor voltage V_x into a differential current is a source-degenerated differential pair [8] with diode-connected loads. The operation of this circuit alone is inherently symmetric if working in fully-differential mode, representing an enhancement from what have been achieved by previous implementations. This symmetry though is broken by using a single-ended input voltage, but still the resulting V-I characteristic maintains symmetry levels beyond those of other implementations.

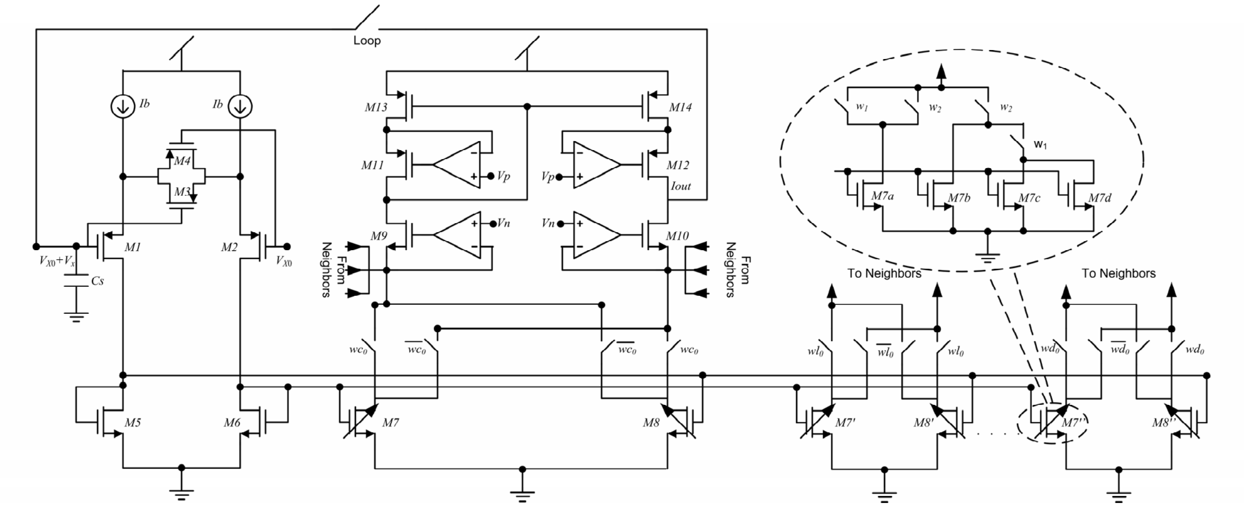

Figure 2. Schematic of the linearized OTA and synaptic blocks

The implementation of the weights is based on geometrical relations between transistors. This has the advantage of being less influenced by process parameter variations both inter- and intra-die. It has also the drawback of only permitting the use of a discrete set of weight values, namely –4, –2, –1, 0, 1, 2 and 4. Opposite-sign contributions are obtained by crossing output wires, achieving a symmetric operation by architecture. Finally, the sum of all the currents from the neighborhood is injected into the target state capacitor. But before that, differential to single-ended current conversion is realized with the help of gain-boosted Cascode transistors to reduce the effect of the finite output resistance of the mirror.

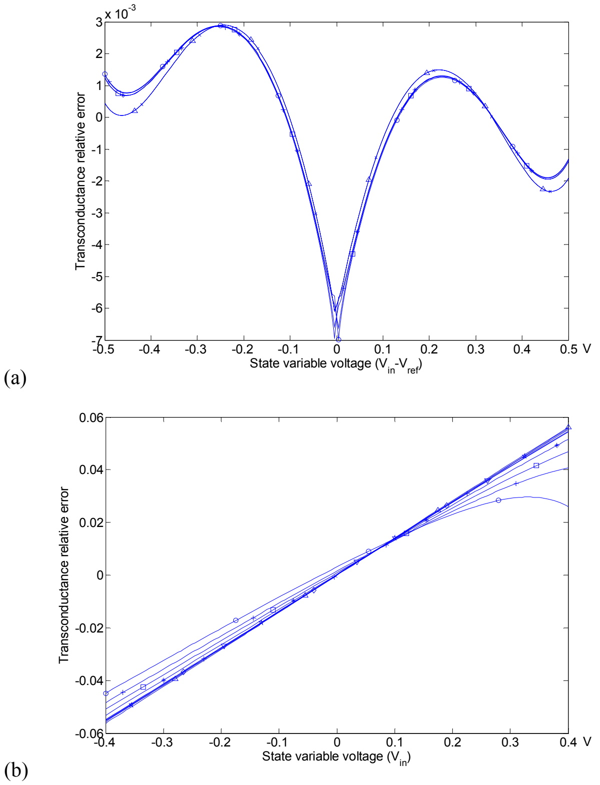

Figure 3. Transconductance relative error of the (a) OTA-based synapse, (b) Single-transistor synapse.

As a result, the output current has a high linearity. Fig. 3 depicts the transconductance relative error —relative to the slope of the best fit line— for the OTA-based synapse (above) and for the single-transistor synapse (below). It can be seen that this error resembles the inherent symmetry of the OTA-based synapse and the dissymmetry of the characteristic of the MOSFET. Quantitatively, the transconductance relative error in large signal is kept below 0.7%, which is an order of magnitude better than that shown by the single-transistor synapse. Concerning the symmetry of the characteristic, the difference of the output currents corresponding to weights with the same absolute value but opposite sign is zero on average because offset cancellation, the standard deviation, obtained by Monte Carlo simulation, being 2% of the absolute value of the individual currents.

III. Adaptive Image Capture

Adaptive image capture in this chip is based in the local and global control of the photosensor gain, through selectively controlling the integration time for each photosensor. The elementary light sensing device is a regular n-diffusion/p-substrate inverse biased diode. Operating in integration mode, the voltage representing the value of the pixel depends linearly on the generated photocurrent, and thus, on the average power being irradiated over the sensor area. The slope of this relation is determined by the integration time. We have provided each pixel with the ability to automatically adjust its own integration time according to a locally-stored voltage that can be the result of fast analog computations realized at the focal-plane. The local support for this consists in an analog memory for the control voltage and a comparator [10]. A global signal, an inverted ramp, is delivered across the array to allow the pixels to start integrating at different time instants, according to the magnitude of the local control voltage. The ramp can be shaped, e. g. with an exponential decay as in this chip, to accomplish signal compression.

The major computational effort to accomplish adaptation to illumination conditions is realized at the parallel array, by computing the average value of the pixels, and at the periphery, by an all-digital circuit that introduces the appropriate correction into the average integration time, and generates the global ramp with the help of a DAC.

The average pixel voltage, otherwise a hard computing task, is realized in the array by means of linear diffusion, either programming it into the CNN, or with a built-in resistive grid. Once the average is available, it is compared with an upper and a lower threshold, V_{up} and V_{down}. If the resulting average falls between the thresholds, the average integration time will not change. If the voltage falls above/below the upper/lower limit a digital circuit, triggered by these comparators, corrects the frequency division realized onto the systems master clock, T_{clk}, employed to generate the inverse ramp, in the proper sense, resulting in:

$$T_{ramp} = 2^{p}(24+q)T_{clk} \tag{7}$$

where $p \in \{0,1,2,\ldots,9\}$ and $q \in \{0,1,2,\ldots,23\}$. Adaptation loops start at T_{ramp} (p=0 and q=23) and begin with growing p until the average pixel voltage falls below V_{down}. Then q is decreased until the [V_{down}, V_{up}] region is reached again. These combined exponential and linear scales allow for a fast response to sudden changes in the global illumination conditions while maintaining a damped settling in order to avoid any flickering. Fig. 4 depicts the evolution of the average exposure time, in an adaptive capture loop, starting at different integration times and illumination conditions. In all cases, there is an initial exponential trend towards the appropriate order of magnitude and then a slow (damped) decrease towards the final value. Bearing in mind that the captured scene is fixed, the absence of flickering can be seen as a necessary condition for the convergence of the adaptive capture loop.

IV. Chip Data and Expected Performance

A prototype of an analog/mixed-signal parallel array processor of 32×32 cells has been designed and fabricated in a CMOS 0.35µm process. Table I shows a survey of chip features. These data are predicted from the simulation results as the chip is still being tested. We can make use of use of the estimated time constant (100ns) to obtain the expected peak computing power. The number of additions and products being realized simultaneously at the processing element is 81, according to Eq. (1). The necessary time to arrive to the solution with N-bits equivalent resolution is $\tau(N+1)\ln 2$. We arrive to a peak computing power of 131GOPS. Which, using the data in Table I, translates into 3.41GOPS/mm^2 and 142MOPS/mW. Those figures are related, as the lesser the circuitry employed in the implementation, the smaller the current consumption will be. Using analog circuits at the elementary processing units avoids A/D conversion at the pixel level. For moderate accuracy requirements, they occupy less area and consume less power than their digital counterparts, rendering a more efficient implementation.

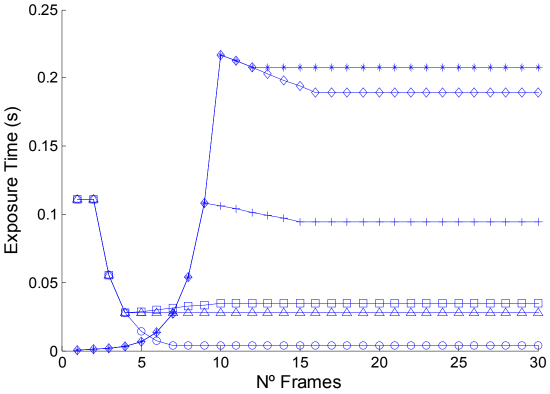

Figure 4: Exposure time vs. No. of frames

TABLE I. Prototype Chip Data

CMOS process	0.35µm
No. of CNN cells	3x32x32
Die area	7.6mm x 7.6mm
Array area	6.2mm x 6.2mm
Processing element size	176µm x 176µm
Current per PE	300µA@3.3V
Weight resolution	7-8b
Image resolution	8b
I/O rates	10MHz
CNN time constant	below 100ns

References

[1] I. Rock, J. Victor, “Vision and Touch: An Experimentally Created Conflict between the Two Senses”. Science, New Series, Vo. 143, No. 3606, pp. 594-596, February 1964.

[2] D. Puccinelli, M. Haenggi, “Wireless Sensor Networks: Applications and Challenges of Ubiquitous Computing”. *IEEE Circuits and Systems Magazine*, Vol. 5, pp. 19-29, Aug. 2005.

[3] E. Aarts, R. Roovers, “IC Design Challenges for Ambient Intelligence”, Design, Automation and Test in Europe (DATE), pp. 2-7, March 2003.

[4] T. Makimoto, T. T. Doi, “Chip Technologies for Entertainment Robots - Present and Future”. Int. Electr. Dev. Meeting, pp. 9-16, Dec. 2002.

[5] Eyal Margalit et al. “Retinal Prosthesis for the Blind”, Survey of Ophthalmology, Vol. 47, No. 4, pp. 335- 356, July-August 2002.

[6] T. Roska and L. O. Chua: “The CNN Universal Machine: An Analogic Array Computer”. IEEE TCAS, Vol. 40 (3), pp. 163-173, March 1993.

[7] S. Espejo et al., "A VLSI Oriented Continuous-Time CNN Model". Int. J. Circ. Th. And Apps. Vol. 24, No. 3, pp. 341-356, May-June 1996.

[8] F. Krummenacher and N. Joehl, “A 4-MHz CMOS Continuous-Time Filter with On-Chip Automatic Tuning”. IEEE J. of Solid-State Circuits, Vol. 23, pp. 750-758, June 1988.

[9] R. Domínguez, S. Espejo, A. Rodríguez and R. Carmona, "Four-Quadrant One-Transistor Synapse for High-Density CNN Implementations". CNNA 98, pp. 243-248, London, April 1998.

[10] R. Carmona, C. M. Domínguez, J. Cuadri, F. Jiménez and A. Rodríguez, "A CNN-Driven Locally Adaptive CMOS Image Sensor". ISCAS'04, Vol. V, pp. 457-460, Vancouver, Canada, May 2004.