

# A Model for VLSI Implementation of CNN Image Processing Chips Using Current-Mode Techniques

S. Espejo, A. Rodríguez-Vázquez, R. Domínguez-Castro, B. Linares and J.L. Huertas

Dept. of Analog Circuit Design, Centro Nacional de Microelectrónica-Universidad de Sevilla  
Edificio CICA, C/Tarfia sn, 41012-Sevilla, SPAIN  
Phone # 34 5 4623811. FAX # 34 5 4624506. email: angel@cica.us.es

**Abstract-** A new Cellular Neural Network model is proposed which allows simpler and faster VLSI implementation than previous models. Current-mode building blocks are presented for the design of CMOS image preprocessing chips (feature extraction, noise filtering, compound component detection, etc.) using the cellular neural network paradigm. Area evaluation for the new model shows a reduction of about 50% as compared to the use of current-mode techniques with conventional models. Experimental measurements of CMOS prototypes designed in a 1.6 $\mu$ m n-well double-metal single-poly technology are reported.

## I. INTRODUCTION

Cellular Neural Networks (CNN) consist of arrays of elementary processing units (*cells*), each one connected only to a set of adjacent cells (*neighbors*). The reduced connectivity allows high cell density in terms of silicon area, and facilitates CNN physical design. For the class of *translationally invariant* CNNs, in which all inner cells are identical, programmability can be incorporated without a significant routing cost, by just adding a few global control lines, one per weight. CNN properties, and its application to image processing, pattern recognition, motion detection, etc., have been covered in different papers, for instance [1-9]. This communication focuses on CNN VLSI implementation, of which little literature is available [10,11]. First, a new CNN mathematical model is presented which exhibits some advantages for VLSI implementation as compared to the conventional model due to Chua and Yang [1]. Then, we consider the implementation of this model in current domain. Since CNN's are basically aimed to image processing applications, and primary output of image sensor devices (phototransistors [12]) is current, this implementation style is advantageous as compared to previous proposals, where cell input signals are voltages. The results obtained show better area, power and speed figures than previous CNN implementation techniques.

## II. EXTENDED RANGE CNN MODEL

Previously reported CNN IC design approaches focused on the use of gm-C techniques for the implementation of the Chua-Yang's cell circuit model [1], shown in Fig.1, whose

0-7803-1254-6/93\$03.00 © 1993 IEEE

dynamic is given by the following set of nonlinear state-equations,

$$\tau \frac{dx^c}{dt} = -x^c + D^c + \sum_{d \in N_r(c)} \{A_d^c y^d + B_d^c u^d\} \quad (1)$$

$\forall c \in \mathcal{GD}$

where  $N_r(c)$  represents the *cell neighborhood*, including cell  $c$  itself,  $\mathcal{GD}$  is the net *grid domain*, and the *cell outputs* ( $y^d$ ) are obtained from the *state variables*,  $x^d$ , by the following nonlinear function,

$$y^d = f(x^d) = \frac{1}{2} (|x^d + 1| - |x^d - 1|) \quad (2)$$

The coefficients  $A_d^c$  and  $B_d^c$ , with  $d \in N_r(c)$ , can be arranged in two matrices  $A^c$  and  $B^c$ , called the *feedback* and *control templates* respectively, while  $D^c$  is known as the *offset term*. The time constant  $\tau$  is assumed invariant from cell to cell. For translationally invariant CNNs, the offset term and the entries of  $A$  and  $B$  are also cell-invariant.

Computational properties of this model rely on its ability to yield, for  $A^c > 1$ , two stable equilibrium points separated by an instability region, and in the possibility of modifying the stable point attraction regions by changing the neighbor contributions.

$$I = D^c + \sum_{\substack{d \in N_r(c) \\ d \neq c}} \{A_d^c y^d(t) + B_d^c u^d\} + B_c^c u^c \quad (3)$$

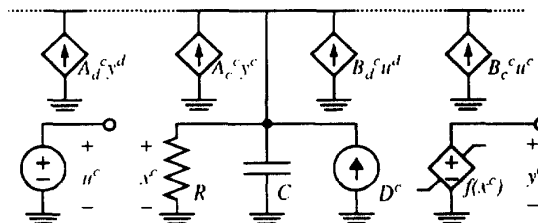


Fig. 1: Chua-Yang CNN Cell Circuit Model [1].

This is illustrated in Fig.2, showing the  $\tau(dx^c/dt)$  vs.  $x^c$  characteristic for the different qualitative situations possible. The displayed dynamic routes show that the attraction region for the equilibrium point on the right (equilibrium points are at the intersection of the characteristics to the horizontal axis) becomes narrower for  $I$  decreasing; for  $I=I_1$ , this point becomes virtual. A similar situation happens for the equilibrium point on the left, in case  $I$  increases.

Analog VLSI implementation of (1) must handle different variation ranges for the state and the output variables, as illustrated in Fig.2: while output variables are restricted to  $[-1,1]$  by (2), state variable excursions are not constrained, although it can be shown [1] that they remain bounded by the following normalized maximum value,

$$x^c_{max} = 1 + |D^c| + \sum_{d \in N_p(c)} \{|A_d^c| + |B_d^c|\} \quad (4)$$

which, for typical templates, ranges between 5 and 10. Thus, in the previously reported  $g_m$ -C circuits [10, 11], where all resistive components in Fig.1 are realized by differential input transconductors, largely different biasing conditions and design equations must be considered for the transconductors, thus complicating the sizing process and yielding non-optimum power consumption and area figures.

A new continuous time (CT) CNN cell model is proposed here to overcome drawbacks of (1) while preserving its computational properties. The cell state equation for this new model is given by,

$$\tau \frac{dx^c}{dt} = g(x) + D^c + \sum_{d \in N_p(c)} \{A_d^c y^d + B_d^c u^d\} \quad (5)$$

$$\forall c \in \mathcal{GD}$$

where  $g(\bullet)$  is a three pieces piecewise linear characteristics given by,

$$g(x^c) = \lim_{m \rightarrow \infty} \begin{cases} -m(x^c + 1) + 1 & x^c < -1 \\ -x^c & \text{otherwise} \\ -m(x^c - 1) - 1 & x^c > 1 \end{cases} \quad (6)$$

This model yields two stable equilibrium points separated by an instability region, as (1) does. Also as in (1), the attrac-

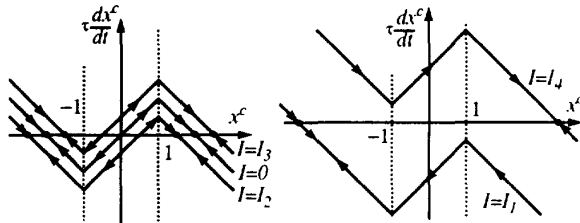


Fig. 2: Dynamic Routes for the Chua-Yang CNN model:  $I_1 < I_2 < 0 < I_3 < I_4$ .

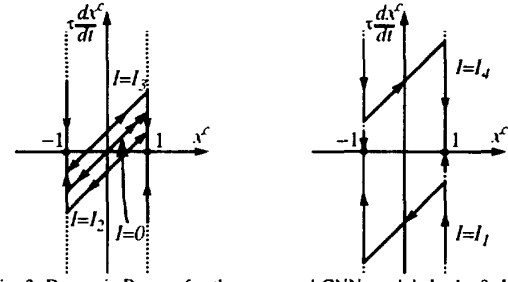


Fig. 3: Dynamic Routes for the proposed CNN model:  $I_1 < I_2 < 0 < I_3 < I_4$ .

tion region for the stable equilibria can be modified by changing the  $I$  parameter. This is illustrated in Fig.3. As a difference to (1), in the new model all variables have the same variation range, what, when it comes to silicon, is very appealing to reduce power and cell area and to simplify the design process. Larger production yields are also found for circuits based on this new model, specially when dealing with low voltage technologies.

### III. CURRENT MODE BASIC BUILDING BLOCKS

The operations required for the implementation of (5) and (6) are: *summation*, *weighted replication*, *nonlinear limitation* and *integration*. In current-mode, summation is simply achieved by routing signals to a common node. Weighted replication is obtained by using *current mirrors*. In MOS technologies, the scaling factor is controlled by transistor sizes. Since in current mirrors currents always flow in the same direction, bias-shifting is required to allow for the symmetric existence-interval of the variables. These concepts are illustrated in Fig.4(a) and (b). Several output currents with the same or different scaling factors can be obtained by using different output transistors.

Nonlinear limitation is also obtained using current mirrors. In Fig.4(b), a saturation nonlinearity appears as a result of the cut-off of the input transistor. By cascading two of these structures, the nonlinearity in (2) is obtained, as shown in Fig.5.

Finally, a current-mode lossy integrator can be easily implemented by exploiting reactive parasitic behavior of a current mirror, as illustrated in Fig.6, whose small signal behavior is described by the following equation,

$$\tau \frac{dI_o}{dt} = -I_o + I_{in} \quad (7)$$

where  $\tau$  is given by  $\tau = C/g_m$ ,  $g_m$  being the small signal transconductance of the transistors in the current mirror. Under

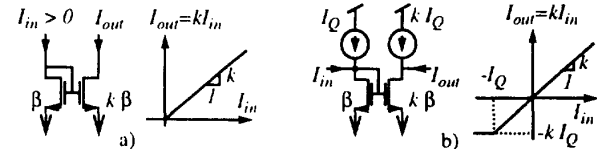


Fig. 4: a) MOS current mirror; b) MOS current mirror with bias shifting.

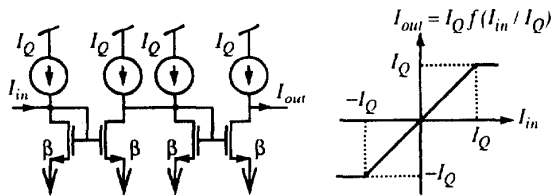


Fig. 5: Implementation of the nonlinear operator  $f(\bullet)$ .

large signal operation, the lossy integrator time constant becomes a function of the output current (through  $g_m$ ). However, it can be shown that this nonlinearity does not alter the computational properties of current-mode CNNs, but just produces slight changes in the transient response.

Current saturation at the lossy integrator output can be exploited for the implementation of the nonlinearity required in the new model, given by (6). Note that bias current in the integrator of Fig.6 determines the lower extreme of the state variable range ( $I_o > -I_Q$ ). The upper extreme is determined by the loading device. Thus, by using the same bias current  $I_Q$  for this loading device, we force the integrator output current to remain confined inside the interval  $[-I_Q, I_Q]$ , thus having a very simple way to implement (6). Note also that, since output and state variables are confined to the same interval, the physical implementation of (2) is not required.

The above mentioned building blocks can also be realized by enhanced CMOS mirrors (self-biased cascode, cascode, servo-mirrors, etc.), as can be required to reduce errors due to finite Early voltages. The resulting area and speed penalizations are usually not severe.

#### IV. CELL ARCHITECTURE

Fig.7 shows a conceptual diagram for a CNN cell, according to either (1) or (5). Weighted replication is achieved with the circuit in Fig.4(b). When sign inversion is required, an additional replication stage is cascaded. For the realization of (5), the nonlinear operator is implemented by properly loading the integrator block, and hence, the nonlinear operator block is not required.

For simplicity, I/O circuitry has not been included in the diagram in Fig.7. Initial state values  $x^c(0)$  and external inputs  $u^c$  can be electrically set, or optically transmitted using photoactive devices [12]. Output of each cell can be evaluated with the help of an additional replication branch. Further I/O considerations are needed to reduce pinage of large-net chips.

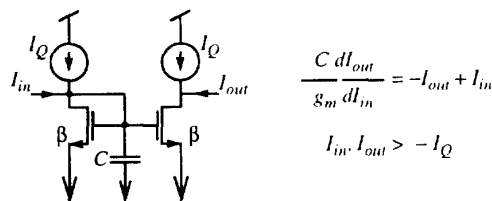


Fig. 6: Lossy integrator.

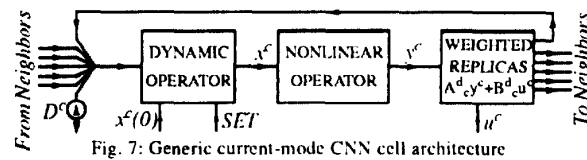


Fig. 7: Generic current-mode CNN cell architecture

Note that weighted replication is performed at the output of each cell. In other words, each cell produces a different output, with the required weight, for each neighbor. At each cell, neighbor contributions are added at the input node as they are received. In order to avoid confusion when designing current-mode CNNs, it is convenient to work with new template matrices, obtained interchanging  $A_d^c$  by  $A_c^d$  and  $B_c^d$  by  $B_d^c$ .

Fig.8(a) shows the complete cell schematic of a connected component detection (CCD) CNN [6], using single transistor mirrors. For the sake of illustration, Table I gives the total cell area for different templates in a  $1.6\mu\text{m}$  CMOS technology, corresponding to the use of current-mode techniques for the implementation of both the Chua-Yang and the extended range model. Cascode transistors with  $W=4\mu\text{m}$  and  $L=3.2\mu\text{m}$  are used for the current mirror building blocks, as illustrated in Fig.8(b). Although data in Table I does not include the area occupied by the initialization circuitry, approximate pixel-densities range from 60 to 160 cells/ $\text{mm}^2$  (depending on the particular template) when the proposed model is used.

#### V. EXPERIMENTAL RESULTS

Experimental results of a CCD CNN prototype, designed in a  $1.6\mu\text{m}$  double-metal single-poly CMOS technology, are summarized here. The design actually includes two prototypes, each with sixteen cells in a row. First prototype (P1) corresponds to the schematic in Fig.8(a) while the second one (P2) corresponds to the conventional CNN model by Chua and Yang. Both prototypes were designed using the current-mode technique described above.

Design process begins by choosing a unitary bias current ( $I_Q$ ) which in our case was  $2\mu\text{A}$ . A current mirror capable of driving  $2I_Q$  and a  $I_Q$  current source must then be designed. Cascode structures were used in our design to implement both the current mirror and the current source. All transistors had

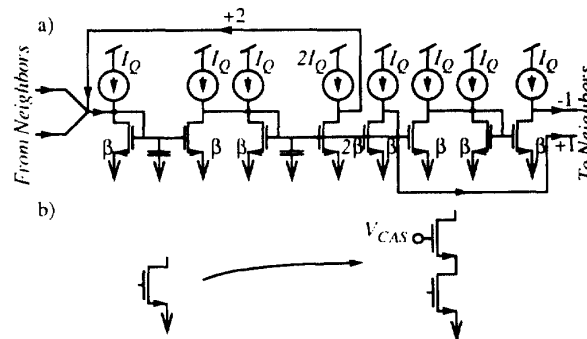


Fig. 8: a) Schematic of a CNN cell for CCD; b) Cascode structure.

TABLE I  
CELL AREA FOR DIFFERENT TEMPLATES AND CNN MODELS.

	Proposed Model Area ( $\mu\text{m}^2$ )	Chua-Yang Model Area ( $\mu\text{m}^2$ )
C.C. Detector.	5916	12691
Shadow Detector.	7736	16533
Borders Extrac.	15471	26291
Corners Extrac.	16381	28212
Hole Filling	10921	24774
Noise Filtering	5460	14258

$W=4.0\mu\text{m}$  and  $L=3.2\mu\text{m}$ . After this steps, implementing any CNN is just a matter of combining the same building blocks.

Power calculation is extremely simple from cell schematic. With a 5V power supply, P1 consumes  $90\mu\text{W}/\text{cell}$ , while P2 takes  $290\mu\text{W}/\text{cell}$ . Figs.9(a) and (b) show the layouts of P1 and P2 single cells respectively, including the initialization circuitry, an extra output for evaluation purpose, and the necessary spacing and lines to be connected by abutment with neighboring cells. Area figures are  $8303\mu\text{m}^2/\text{cell}$  (equivalently  $120\text{ cells}/\text{mm}^2$ ) for P1, and  $16.368\mu\text{m}^2/\text{cell}$  (equivalently  $61\text{ cells}/\text{mm}^2$ ) for P2. The slight difference with Table I is due to the incorporation of the initialization and test circuitry.

In order to test the prototypes with standard equipment, interfaces were used in the design so that chip I/O is performed in voltage form.

Fig.10 shows an experimental measurement of prototype P1. The sixteen outputs are shown in order from top ( $O_1$ ) to bottom ( $O_{16}$ ). Changes at the right side of the graphs correspond to a new cycle of the set/process control signal. High voltage (5V) corresponds to  $+2\mu\text{A}$ , while low voltage (0V) corresponds to  $-2\mu\text{A}$ . Boundary cells are set to a low state value. Note that convergency time is only  $1.6\mu\text{s}$ , with 16 cells in a row. Identical results were observed from P2, although small speed-differences have been observed for some initial states. None of the prototypes can be said to be generally faster.

Other results include Montecarlo simulations, which showed 100% and 90% yield (out of 30 trials) for prototypes P1 and P2 respectively, and extensive simulations using other templates.

It can be concluded that the circuits obtained are advantageous in terms of area, speed and power over previous CNN models and implementation techniques.

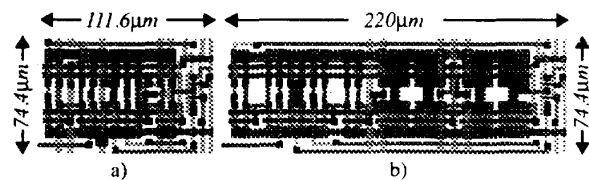


Fig. 9: Cell layouts: a) prototype P1; b) prototype P2

## REFERENCES

- [1] L.O. Cua and L. Yang: "Cellular Neural Networks: Theory". *IEEE Trans. Circuits and Systems*, Vol. CAS-35, pp 1257-1272, 1988.
- [2] L.O. Cua and L. Yang: "Cellular Neural Networks: Applications". *IEEE Trans. Circuits and Systems*, Vol. CAS-35, pp 1273-1290, 1988.
- [3] L.O. Chua and T. Roska: "Stability of a Class of Nonreciprocal Cellular Neural Networks". *IEEE Trans. Circuits and Systems*, Vol. CAS-37, pp 1520-1527, 1990.
- [4] L.O. Chua and P. Thiran: "An Analytical Method for Designing Simple Cellular Neural Networks". *IEEE Trans. Circuits and Systems*, Vol. CAS-38, pp 1332-1341, 1991.
- [5] J.A. Nossek et al.: "Cellular Neural Networks: Theory and Circuit Design". *Int. J. Circuit Theory Applications*, 1992 (to appear)
- [6] T. Matsumoto et al.: "CNN Cloning Template: Connected Component Detector". *IEEE Trans. Circuits and Systems*, Vol. CAS-37, pp 633-635, 1990.
- [7] T. Matsumoto et al.: "CNN Cloning Template: Hole Filler". *IEEE Trans. Circuits and Systems*, Vol. CAS-37, pp 635-638, 1990.
- [8] T. Matsumoto et al.: "CNN Cloning Template: Shadow Detector". *IEEE Trans. Circuits and Systems*, Vol. CAS-37, pp 1070-1073, 1990.
- [9] S. Matsui and T. Okumoto: "A Two-Dimensional Segmentation-Free Learning Recognition System by a Cellular Automaton Array using Eigenvectors of the Second Moment Matrix". *IEICE Transactions*, Vol. E-74, pp 2432-2440, 1991.
- [10] J.M. Cruz and L.O. Chua: "A CNN Chip for Connected Component Detection". *IEEE Trans. Circuits and Systems*, Vol. CAS-38, pp 812-817, 1991.
- [11] K. Halonen et al.: "VLSI Implementation of a Reconfigurable Cellular Neural Network Containing Local Logic". *Int. J. Circuit Theory Applications*, 1992 (to appear)
- [12] A.H. Sayles and J.P. Uyemura: "An Optoelectronic CMOS Memory Circuit for Parallel Detection and Storage of Optical Data". *IEEE J. Solid-State Circuits*, Vol. SC-26, pp 1110-1115, 1991.
- [13] S. Espejo: *Ph.D. Dissertation*. University of Seville, Spain. (In Progress.)

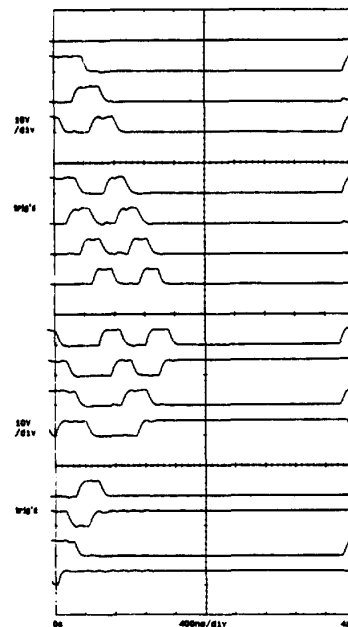


Fig. 10: Experimental measurements from prototype P1.