A Cellular Neural Network System for Real-Time Image Processing

Rodrigo Montúfar-Chaveznava and Domingo Guinea

***Abstract*— In this paper, the design of a Cellular Neural Network System intended for real-time image processing is presented. Real-time image processing can be possible with this system due the powerful capabilities of the Cellular Neural Network Universal Machine considered in the design. Additionally, the design contemplates the use of an analog RAM to reduce time processing when D/A or A/D conversions are required.**

***Index Terms*—Cellular Neural Networks, Image Processing, Real Time Systems.**

I. Introduction

IMAGE processing is a basic tool in areas where vision and scene interpretation are essential. Usually, images contain a huge quantity of data and information; in consequence, the study of images by means of computer processing can require a big undesirable quantity of time. Such inconvenience represents a handicap in systems that demand real-time processing or immediate image interpretation such as robotics. In this paper, we present the design of a real-time system for image processing. The system is the Cellular Neural Network Computer (CNN-C) made up of a CNN Universal Machine (CNN-UM), an analog memory (ARAM), an imager and a control unit.

The CNN-C was designed considering the similarity between both robotic and human being vision process. Imager and eyes have the same function: to sense the scene and capture the image. Captured image is interpreted before to go to brain; in humans, such interpretation corresponds to nerves that translate the sensed scene to electrical signals. The brain makes a decision according to the perceived signals and perform or command an action. Some times we employ the memory as an auxiliary tool to remember or recover information. Figure 1 shows such similarity.

In general, the vision process in robotics involves the following tasks: (i) image acquisition by an analog media as analog data, (ii) analog data conversion to digital data, (iii) digital data processing by a digital computer and (iv) result interpretation for decision and action.

Rodrigo Montufar-Chaveznava is with the Computer Science Department, Instituto Nacional de Astrofísica, Óptica y Electrónica, Santa María Tonanzintla, 72840 Mexico (e-mail: rodrigo@inaoep.mx).

Domingo Guinea is with the Systems Department, Instituto de Automática Industrial-CSIC, Madrid, 28500 Spain (e-mail: domingo@iai.csic.es).

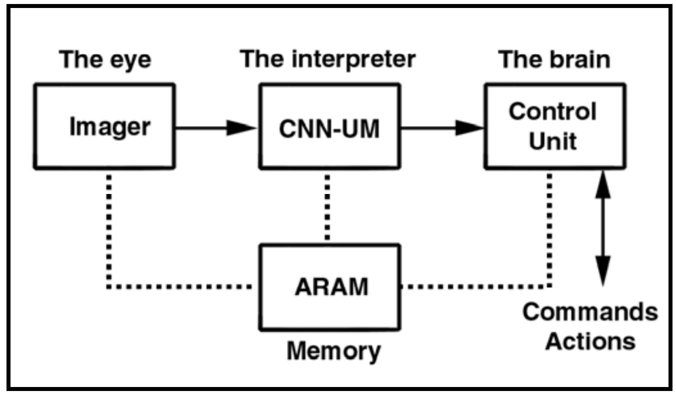


Fig. 1. Representation of human and CNN-C vision processes similarity.

II. CNN computing

The classical framework in image processing is not enough powerful to solve complex image processing tasks. For example, segmentation, filtering, singularity detection, coding or enhancement are carried out by the convolution operation between the image data and a specific kernel; this operation requires a lot of computing. That situation cannot be allowed for some specialized tasks. Then, considering that a CNN-UM can perform real- time convolution, the CNN-C is a good solution.

The first CNN chip [1]-[2] opened up the possibility to put into practice the CNN-C physical implementation with promising results, principally because the 22×20 chip [1] was tested in texture classification successfully [3]. Actually there are more complex chips consisting of a 48×48 or 64×64 array [4]-[5]. The analog nature of the CNN could be a drawback for the computer design. Fortunately, an analog memory has been designed [6] which will reduce the complexity of the design and it will improve the performance of the CNN-C.

At present, we can find a variety of CNN systems based on many CNN chips: the 6×6 DPCNN for stereo vision [7], ZISC [8], 64×64 CNN-UM [9] or DCNN with an optoelectronic interface [10]. Most of them with a cell array size equal or lower to 64×64. The design presented in this paper pretends to surpass the characteristics of the actual systems, reducing the complexity in circuitry, avoiding as much as possible D/A and A/D conversions and including a novel mixed programmable device as the control unit.

III. THE CNN DEFINITION

The CNN is a large-scale nonlinear analog circuit able to process signals in real time. It is considered an analog parallel-computing paradigm defined in space, and characterized by the interaction of processing elements (cells) locally connected.

The CNN is defined as a n-dimensional array of cells that satisfies two properties: (i) most interactions are local within a finite radius r, and (ii) all state variables are continuous valued signals [11]. Templates specify the interaction between each cell and all its neighbors in terms of their input, state, and output variables.

In a CNN: (i) the space variable is always discretized; (ii) the time variable t may be continuous or discrete; (iii) the interconnection effect represented by the cloning template may be a nonlinear function of the state x, output y, and input u of each cell, within the neighborhood N_r of radius r, as well as that of the time t; (iv) the cell is governed uniquely by an evolution law; and (v) occasionally, the cells and the interconnections may be perturbed by some noise sources of known statistics. Then, there are many choices for the array grid, the cell dynamics, the interaction, and the operation mode.

The general form of the cell dynamical equations is:

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

where x^c represents the state of the generic c cell; y^d represents the state of the cells located in the interaction neighborhood of the cell c; A_d is the feedback template; B_d is the control template; D_A and D_B are the offset or bias terms; and $g(x^c)$ is defined as follow:

$$g(x^c) = \begin{cases} -\infty, x^c < -1 \\ 0 \ , \left|x^c\right| < 1 \\ \infty \ , \ x^c > 1 \end{cases} \qquad (2)$$

Equation (1) corresponds to the Full Signal Range (FSR) CNN model, considered a CNN generalization with non-linearities in the local losses [15]. Considering the electronic implementation of the CNN, this model presents some advantages over the original model presented in [16]: (i) the state and output signals are merged into a single signal; (ii) the realization of the non linear function $y(x)$ is not required; (iii) the normalized dynamic-range of all signals is ± 1, and invariant with the application; (iv) the linear dissipative term is merged with the self feedback coefficient; (v) proportional variations of all coefficients have no effect on the I/O mapping of the network; and (vi) processing time is reduced approximately at half in common applications.

IV. THE CNN DESIGN

An imager, a CNN-UM, an ARAM and a mixed programmable device essentially compose the design. Digital, analog and control buses interconnect these devices. Fig. 2 shows the schematic of the CNN-C design.

The CNN-C consider the use of the next circuits:

1) CNN-UM: The CNNUC3 [5], a prototype designed to work with 3.3 V power supplies whereas the power consumption will remain around 0.8 W. Every cell includes circuitry for CNN processing, binary and gray scaled images storage, login operations among binary images and the necessary configuration circuitry for electrical I/O and control of the different operations. Every cell in the 64×64 array includes the necessary circuitry to implement a FSR CNN model [12]. Neighborhood radius unitary. Feedback and control templates include weights for all the nine adjacent neighbors. The offset or bias term is modeled by two summands.

2) Imager: The IMAGAV [13], a new imager intended for averaging operations. It includes a 64×64 array of active pixels and the circuitry for selection, control and output. The prototype includes random access to all the pixels by selecting the row and column. The read out process can be done pixel by pixel or directly reading a group of pixels, giving the averaged illumination value at the output.

3) ARAM: The AM8192C [6] has an array of 32×256 analog cells. Every cell consists of a capacitor, a pass transistor and some local logic for address decoding. The random access to any memory location is done coding the proper address. I/O analog data transmission is done either by a 16-line wide I/O bus or a serial I/O channel. An I/O MUX/DEMUX with the aid of the row selection decoder scans the 32 array data lines.

4) Mixed Programmable Device: The FIPSOC system [14] is a new approach to a system prototyping for mixed signal applications. It consists of an on-chip microprocessor, suitable user-friendly CAD tools and a set of library macros and applications to provide an easy path to migrate to ASIC. The chip comprises a multicontext dynamically reconfigurable lookup table section and includes some large granularity logic cells targeted for synthesis programs; it also includes programmable analog blocks with configurable interconnectivity and an optimized interface to the on-chip 8051-microprocessor core.

All circuits are connected via analog, control and digital buses as noted in Fig. 2. Interconnection is important because analog and digital data must be handled carefully. For instance, analog data is captured by the imager and stored into the ARAM; next, the CNN takes the data from the ARAM, process it and turn it back to the ARAM; finally, control unit takes processed data to convert it in digital data in order to interpret or transmit it to an external device and perform an action or command. The described process is controlled via the digital and control bus. Additionally, digital bus is employed when binary images are processed.

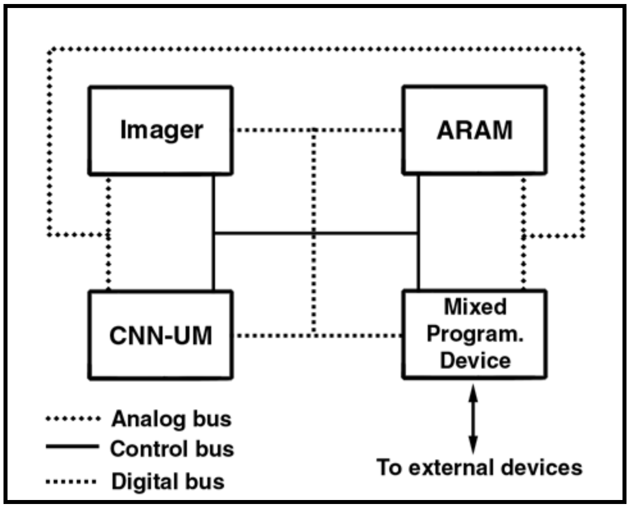


Fig. 2. The CNN-Computer scheme.

2) Image is captured and stored in ARAM, then transmitted to CNN. Optionally, the image can come from an external device through the FIPSOC. If image is binary, it is stored in digital RAM.
3) FIPSOC sends commands to CNN to process the image.
4) Processed image is stored in external memory (analog or digital memory) and transmitted to FIPSOC.
5) FIPSOC takes some decision and actions before to repeat the process. Results can go to external devices from FIPSOC.

A/D and D/A conversions are carried out only when dealing with gray-scale images and they are transmitted between the FIPSOC and the analog memory. It means the number of conversion operations is minimum, and then processing of the CNN-UM is exploited, obtaining real-time results.

V. The CNN characteristics and operation

The characteristics of the circuits employed in the design produces a special CNN system, which has next properties:

1) The retina area for image acquisition is 64×64 pixels. The acquired image can be averaged on the fly.
2) In particular, the CNN equation that models the CNNUC3 chip will be:

$$\tau \frac{dx^c}{dt} = -g(x^c) + \sum_{d=1}^{9} A_d y^d + \sum_{d=1}^{9} B_d u^d + D_A + D_B \qquad (3)$$

3) The element $A_5 = Á_5$-1, where Á is the value that would be used with the original Chua-Yang model.
4) Network boundary conditions for state and input can be set to any analog value and considered part of the template.
5) The CNNUC3 has 22 parameters to define: 9 for each template, 2 for offset and 2 for boundary conditions.
6) The CNNUC3 can accept up to 32 different templates, which are manipulated with up to 64 digital instructions.
7) The CNNUC3 templates size is 3×3.
8) Coefficient values range from –127 to 127 (7 bits plus sign). When using the CNN chip, maximum accuracy and speed will be obtained by scaling the coefficients so that the maximum absolute value was 127.
9) The ARAM stores the acquired image; the results and transitions produce by the CNNUC3, before to be downloaded to the FIPSOC. The ARAM can store a set of four 64×64 images.
10) The FIPSOC has configurable analog blocks and programmable digital logic, which decrease considerably the complexity of the design. It permits to build a compact CNN-C of lower power consumption
11) The FIPSOC can download the results and communicate to an external device by serial port.

The operation of the CNN-C is summarized as follows:

1) CNN is programmed storing templates and instructions in internal memories.

VI. Simulations

Simulations have been performed considering the CNN-C characteristics described in this paper. In particular, simulations have been directed to obtain wavelet representations and signal characterization.

Haar wavelet templates were constructed to carry out the wavelet representation. Fig. 3 presents the wavelet representation at different resolution levels along with reconstructions obtained from quantized representations.

We can note high visual quality reconstructions. Considering analogue nature of CNN, quantization was performed in order to distinguish effects when a representation is transmitted to a digital device for storage or digital analysis and it is reloaded from it.

Simulations can be extended to other kind of wavelets considering limitations imposed by the neighborhood radio. Image size is not a limitation because, although CNN-UM operates over a 64×64 cell array, image can be divided in 64×64 pixels regions.

As mentioned above, image characterization has been also simulated. In particular, we simulated an edge detector based in gaussians. Fig. 4 presents gradient modulus and angle maps required to detect edges; a thresholded map is also presented. The principal advantage of this detector lies in the exploitation of the CNN properties. VLSI implementation of CNNs allows obtaining real time results and performing massive processing. At present, edge detection research has developed a wide variety of techniques, which can be also implemented in the CNN-C. For instance, Sobel, Prewitt and Roberts edge detectors can be implemented, because they employ well defined 2×2 and 3×3 kernel windows. The CNN templates design is very important in order to obtain good results.

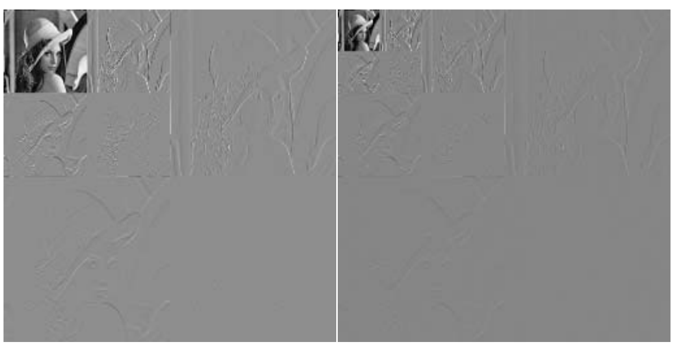
VII. Conclusions and future work

The CNN-C presented in this paper has the ability to perform real-time processing. It is compact, very flexible and can operate independently. These characteristics allow employing the system in special environments or hard situations such as in autonomous robots, on-line industrial inspection or microscopy electronic with an adequate optical system.

In relation with software simulations, we expect to decrease time processing substantially because time employed for convolutions and conversions, the most time consuming tasks, will decrease also substantially with the proposed architecture.

We expect the CNN-UM technology break the 100×100 array barrier to obtain more resolution and expand the action field of the CNN processing. We also expect that cell interconnectivity increases, it will open the possibility to employ bigger kernels, and in consequence, the number of image processing tasks will also increase.

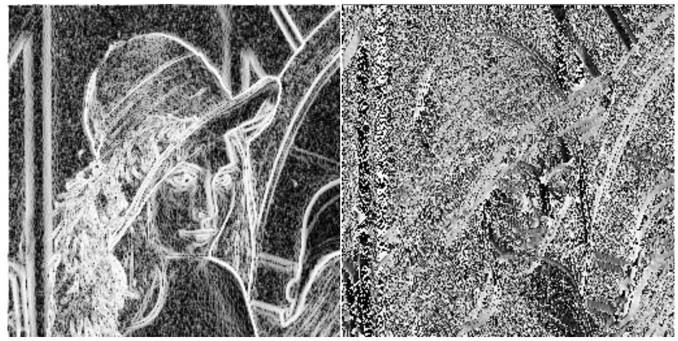
(a)



(b) (c)

(d) (e)

Fig. 3. (a) Original image; (b) wavelet representation at 2 resolution levels; (c) wavelet representation at 3 resolution levels; (d) and (e) reconstructed images from (b) and (c) respectively.



(a) (b)

(c) (d)

Fig. 4 (a) Gradient modulus map; (b) angle map; (c) local maxima; and (d) thresholded map obtained with a threshold value of 60.

REFERENCES

[1] R. Dominguez-Castro, S. Espejo, A. Rodriguez-Vazquez and R. Carmona, A CNN Universal chip in CMOS technology, Proc. of the third IEEE Int. workshop on Cellular Neural Networks and their applicaction (CNN-94), pp. 91-96, Rome, Dec. 1994.

[2] J. M. Cruz, L. O. Chua and T. Roska, A fast, complex and efficient test implementation of the CNN Universal Machine, Proc. of the third IEEE Int. workshop on Cellular Neural Networks and their applicaction (CNN-94), pp. 61-66, Rome, Dec. 1994.

[3] T. Sziranyi and M. Csapodi, Texture clssification by CNN and Genetic Learning, Proc. of IEEE Int. Conf. Pattern Recognition, Jerusalem, ICPR'94, Vol. III, pp. 381-383, 1994.

[4] A. Passio, A. Dawidziuk, K. Halonen and V. Porraiñan, Minimum size 0.5 µm CMOS Programmable 48 × 48 CNN test chip, Proceeding of the 1997 European Conference on Circuit Theory and Design, pp. 154-156, 1997.

[5] G. Liñan, R. Dominguez, S: Espejo and A. Rodríguez, CNNUC3 User Guide, Instituto de Microelectrónica de Sevilla, 1999.

[6] R. Carmona and T. Kozek, The AM8192C Random Access Analog Memory Chip, Technical Report No. 1, Nonlinear Electronics Laboratory, University of California, Berkeley, October 1997.

[7] M. Salerno, F. Sargenti, V. Bonaiuto, S. taraglio and A. Zanela, Design of a board for CNN-based stereo vision, Proceedings of the Sixth IEEE International Workshop on Cellular Neural Networks and their Applications, CNNA-2000, pp. 273-276, Catania, Italy, May 2000.

[8] L. Stevens, High-tec startup has a brainy new chip, North Bay Business Journal, September 18, 2000.

[9] I. Szatmari, A. Zarandy, P. Foldesy and L. Kek, An analogic CNN engine board with the 64×64 analog I/O CNN-UM chip, Proc. IEEE Int. Symp. On Circuits and Systems (ISCAS2000), Vol. 2, pp. 124-127, Geneva, 2000.

[10] S. Jankowski, A. Wielgus, W. A. Pleskacs, R. Buczynsli and M. Wisniewski, IC design of 8×8 digital CNN with optoelectronic interface, roceedings of the Sixth IEEE International Workshop on Cellular Neural Networks and their Applications, CNNA-2000, pp. 431-436, Catania, Italy, May 2000.

[11] L. O. Chua and T. Roska, The CNN paradigm, IEEE Trans. On circuits and systems-I: Fundamental theory and applications, Vol. 40, No. 3, pp. 147-156, March 1993.

[12] S. Espejo, R. Carmona, R. Domínguez-Castro and A. Rodríguez-Vázquez, A VLSI-Oriented Continuous-Time CNN Model, International Journal of Circuit Theory and Applications. Vol. 24, pp 341-356, May-June 1996.

[13] E. Roca, S. Espejo, R. Dominguez-Castro and A. Rodriguez-Vazquez, IMAGAV-1 User´s Guide, Instituto de Microelectrónica de Sevilla, 1998.

[14] J. Faura, P. Naish, P. Paddan, B. Krah, P. van Duong, J. M. Moreno, A. Torralba, J. Launa and J. M. Insenser, "FIPSOC: A New Concept to Mixed Signal Integration", Silicon Design Show'96.

[15] A. Rodriguez-Vazquez, S. Espejo, R. Dominguez-Castro, J. L. Huertas and E: Sanchez-Sinencio, “Current-Mode Techniques for the implementation of continuous and discrete time Cellular Neural Networks”, IEEE Trans. Circuits and Systems II, Vol. 40, pp. 132-146, March 1993.

[16] L. O. Chua and L. Yang, “Cellular Neural Networks: Theory”, IEEE Trans. Circuits and Systems, Vol. 35, pp.1257-1272, October 1988.