



HAL
open science

OPTICAL CELLULAR AUTOMATA : ARCHITECTURAL PRINCIPLES

Pierre Chavel, Jean Taboury, F. Devos, Patrick Garda

► **To cite this version:**

Pierre Chavel, Jean Taboury, F. Devos, Patrick Garda. OPTICAL CELLULAR AUTOMATA : ARCHITECTURAL PRINCIPLES. Journal de Physique Colloques, 1988, 49 (C2), pp.C2-35-C2-38. 10.1051/jphyscol:1988208 . jpa-00227607

HAL Id: jpa-00227607

<https://hal.science/jpa-00227607>

Submitted on 4 Feb 2008

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

OPTICAL CELLULAR AUTOMATA : ARCHITECTURAL PRINCIPLES

P. CHAVEL, J. TABOURY, F. DEVOS* and P. GARDA*

*Institut d'Optique, CNRS-LA, Université Paris-Sud, BP 43,
F-91406 Orsay Cedex, France*

**Institut d'Electronique Fondamentale, CNRS-LA, Bât. 220,
Université Paris-Sud, F-91405 Orsay Cedex, France*

Résumé - On introduit le concept d'automate cellulaire optique et on décrit diverses mises en oeuvre possibles tant sous l'angle des systèmes optiques que sous celui des composants optiques non linéaires nécessaires.

Abstract - The general schematic diagram of an optical cellular automaton is introduced and various implementations are discussed in terms of both optical systems and optical non linear components.

1 - INTRODUCTION :

Clearly, one issue faced by emerging optical computing technology is the effective combination of parallelism with short cycle time in parallel processor arrays. In particular, two-dimensional arrays of bistable or otherwise non linear elements may be interconnected by dense three-dimensional light beams into cellular processors. We suggest that a powerful and rather general approach is that of "optical cellular processors". We define optical cellular processors as arrays of cells where each cell is described by a state variable and a neighborhood. The state variable evolves at given clock times and its new state is a given function of the neighborhood /1/. The most widely investigated optical cellular processors to date are symbolic substitution networks /2/ and neural networks /3/. Here, we discuss optical cellular automata, a class of optical cellular processors that includes symbolic substitution as an special case. We first define the concept and then describe the architectural implementation approaches.

2 - CELLULAR AUTOMATA

The concept of a cellular automaton /4/ can be used to specify the kind of processors considered ; the definition of a cellular automaton includes

- 1 - a regular array of processors C_{ij} , each with a state described by variable c_{ij}
- 2 - parallel evolution of all processors in discrete time according to the state of a given neighborhood \mathbf{N} (not necessarily restricted to nearest neighbors) :

$$c_{ij}(t+1) = f\{c_{i+k, j+l}(t), (i+k, j+l) \in \text{neighborhood of } i,j\}$$

In cellular automata, the neighborhood \mathbf{N} and function f are shift-invariant. The neighborhood is defined by its exact shape ; we shall denote by v its number of pixels.

Even in the simplest case of binary cells, this scheme has been shown to be useful for a number of cases including low-level automated vision tasks, general purpose parallel computation and simulation of continuous two-dimensional physical phenomena /2, 5, 6/. Algebraic operations on binary images, including mathematical morphology, can be cast into the mold of cellular automata/7/. (See also K.S. Huang, B.K. Jenkins, A.A. Sawchuk, "Binary Image Algebra", submitted for publication).. While it definitely is necessary to pursue the quest for even more powerful massively parallel architectures, for example using more sophisticated individual cells (see K.S. Huang, B.K. Jenkins, A.A. Sawchuk, "Digital Optical Cellular Image Processors", submitted for publication) or incorporating the power of stochastic processors /8/, the implementation of cellular automata by purely or partially optical means may already be an important step.

With reference to binary cellular automata, one further consideration is noteworthy : since the function f can only yield the value 0 or the value 1 for the result $c_{ij}(t+1)$, the automaton is completely defined by the set \mathcal{S} of all neighborhood configurations $\mathbf{N}_1, \mathbf{N}_2, \dots, \mathbf{N}_M$ yielding value 1. $c_{ij}(t+1)$ can then be written as the arithmetic sum - or the boolean OR - of M elementary cellular automata, each taking value 1 -or TRUE - for one and only one neighborhood configuration ; in other words, each elementary cellular automaton recognizes one given binary pattern in its neighborhood.

Figure 1 illustrates the above idea : for a neighborhood \mathbf{N} composed of the pixel ij itself and its eight nearest

neighbors k_l such that $\text{Max}(|k-i|, |l-j|) \leq 1$, we consider the cellular automata aimed at recognizing all isolated pixels. It is the sum of two elementary cellular automata : the elementary cellular automaton recognizing "0" pixels surrounded by eight "1" pixels and the elementary cellular automaton recognizing "1" pixels surrounded by eight "0" pixels.

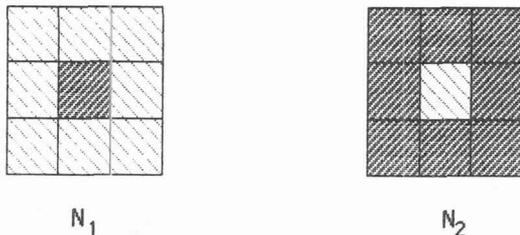


Figure 1. N_1 and N_2 are the neighborhood configurations of the two elementary cellular automata composing the cellular automaton "detection of isolated pixels".

3 - OPTICAL IMPLEMENTATION OF BINARY CELLULAR AUTOMATA

It is always possible to implement a given binary cellular automaton by its decomposition into elementary cellular automata, provided adequate memory planes (e.g. arrays of bistable pixels) are available to store intermediate results. While in principle, with a neighborhood size v , that decomposition may involve up to 2^v steps to be performed in sequence or in parallel, it is reasonable to expect that in most cases suitable algorithmic design can reduce that number of steps to a tolerable value.

In the following, we therefore discuss the optical implementation of cellular automata in terms of elementary cellular automata. Two basic approaches are available and differ by their requirements in real-time devices.

The correlation approach :

The general diagram shown on figure 2, where the arrows indicate spatially modulated optical beams, involves three active elements within the optical processing loop :

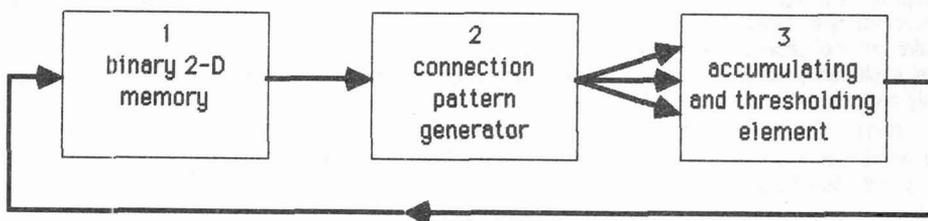


Figure 2. Box diagram of the correlation approach to optical binary cellular automata.

1 - the one-bit memory plane contains the data to be processed and therefore constitutes the object plane ; it is read by an adequate light source unless it is self-luminous ; it can for example be implemented by bistable optical elements ;

2 - the connection pattern generator acts as an imaging device whose impulse response is the required connection pattern. In practice, it may consist of holograms /9/, beam splitters /2/ or rely on shadow-casting according to the principles of geometrical optics /10/. Figure 3 suggests the operation of a programmable holographic pattern generator (3a) and of a shadow-casting pattern generator illuminated by LEDs (3b), both generating an X-shaped pattern. In case (3a)/1/, programmability is achieved using a set of programmable shutters in front of a set of multiplexed holograms ; an additional beam splitter, possibly itself holographic, is then needed to send light from the object onto the connection holograms ;

3 - the thresholding element acts as a multilevel accumulating and inverting element ; each pixel may receive light from any number of its connected neighbors. The incoming interconnect beams allow to recognize the presence of a given match pattern. The thresholding element should be capable of discriminating between the presence and the absence of light. This element also temporarily stores the result before the latter is fed back to element 1, thereby acting as the "master" stage of a master-slave flip-flop to avoid feedback of information through the loop while the

pattern recognition takes place.

Not shown in figure 2 are the light source and the external control unit.

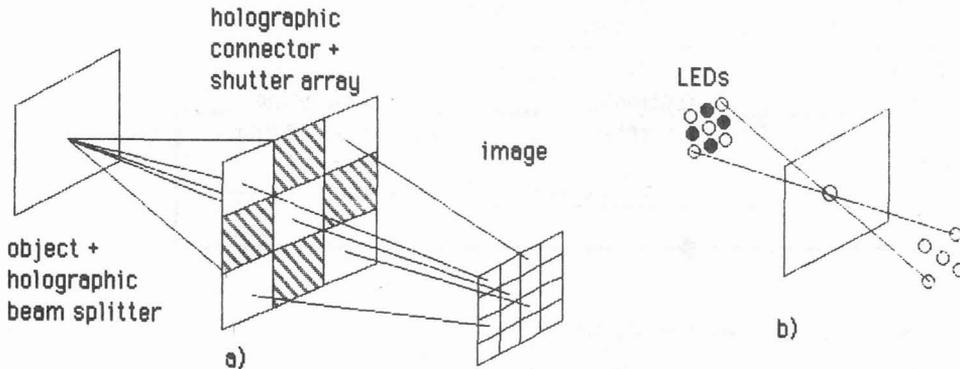


Figure 3. Programmable connection pattern generators.
 a) Holographic approach, b) LED + shadow casting approach.

Let us now describe the operation of such a setup implementing an optical binary cellular automaton by decomposition into elementary cellular automata. At time t , the object plane (element 1) encodes the state of the binary array to be processed, c (boldface letters indicate 2-D arrays of elements, in this case the c_{ij}) as well as its complement \bar{c} . The complement may be introduced by space or by time, or preferably by polarization multiplexing. Element 2 generates the neighborhood configuration N_m required for the m th elementary cellular automaton, $m = 1 \dots M$, as well as its complement \bar{N}_m in such a way that element 3 receives the illuminance :

$c * \bar{N}_m + \bar{c} * N_m$ (* denotes convolution and + denotes usual addition). No light is received if and only if N_m is recognized. The thresholding behaviour of element 3 then causes state 1 to be stored in element 3 at every pixel where N_m has been recognized and 0 at all other pixels. In other words, the operation of the elementary cellular automaton is written as

$$H[c * \bar{N}_m + \bar{c} * N_m] \tag{1}$$

where H is the inverting and thresholding function $H(x) = 1$ if $x \leq 0$, 1 otherwise. This process is iterated for $m = 1$ to M , and the results are accumulated in element 3. After that, the content of element 3 is fed back to element 1 and the cellular automaton is ready for its next cycle.

From the standpoint of optical aberrations and diffraction, we have found that realistic figures are 100×100 pixels of typical size $100 \mu\text{m}$ or less for a neighborhood of up to 25 pixels (i.e. a maximum fan-in and fan-out of 25). This work will be published elsewhere.

From the standpoint of nonlinearity, elements 1 and 3 should provide for the contrast between the on and the off state and element 3 should accommodate the dynamic range imposed by the neighborhood size v .

The shift-and-multiply approach :

Instead of expression (1), the operation of an elementary cellular automaton can also be expressed as a product :

$$\prod_{(k,l) \text{ with } N_{m;k,l} = 1} c * \delta_{k,l} \quad \prod_{(k,l) \text{ with } N_{m;k,l} = 0} \bar{c} * \delta_{k,l} \tag{2}$$

where $N_{m;k,l}$ denotes pixel k,l of neighborhood configuration N_m and $\delta_{k,l}$ denotes a Dirac delta peak at point k,l .

This exactly corresponds to the common demonstration of binary pattern recognition in a conference using superimposed, shifted, identical transparencies on an overhead projector and hence suggests the following alternative to the schematic diagram of figure 4 :

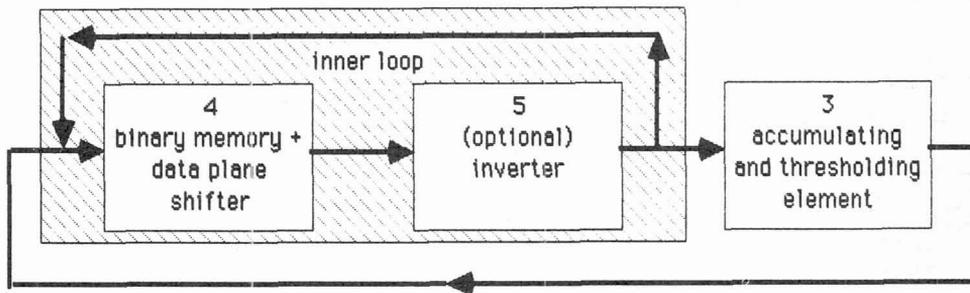


Figure 4. Box diagram of the shift-and-multiply approach to optical binary cellular automata.

Element 4 is a binary 2-D memory like element 1, but this time some mechanism is required to shift the data pixels by the required amount $k.l$. The shift may in fact be implemented either in the element, or in the feedback loop. Element 5 is an inverter, required for the second part of equation 2.

The operation of this setup consists in shifting the data with respect to the illuminating beam by $k.l$, then simply sending again the light around the loop for the next $k.l$. In fact, the loop may be replaced by a stack of elements. While in figure 2 the feedback loop is used only at the end of each elementary cellular automaton operation, the inner loop is used here for each pixel of each elementary cellular automaton. The benefit is that there is only one light beam. While level restoration may still be required at places, the fan-in and the fan-out is unity. From one loop to the next or one element in the stack to the next, only ordinary imaging is needed and geometrical shadow casting may even suffice. The loop is performed v times, and a pixel in the resulting image has level "1" if and only if neighborhood configuration N_m has been recognized. The remainder of figure 4 differs from figure 2 only by the fact that no inversion is required in element 3. We plan to investigate the potential of this approach in the near future.

Acknowledgement : this work was supported in part by the French Ministry for Research and Higher Education (M.R.E.S.).

REFERENCES

- 1/1 P. Chavel, J. Taboury, F. Devos, J.M. Wang, P. Garda, to be published in Applied Optics. 1st may 1988.
- 1/2 K.H. Brenner, A. Huang and N. Streibl, Appl. Opt. 25 (1986) 3054.
- 1/3 D. Psaltis and N. Farhat, Opt. Let. 10 (1985) 98.
- 1/4 S. Wolfram, Theory and Applications of Cellular Automata, World Scientific, Singapore, 1986.
- 1/5 F. Devos and P. Garda, proceedings Intern. Joint Conf. Pattern Recog., Paris, 1986.
- 1/6 D. d'Humieres, P. Lallemand and Y. Pomeau, bull. Soc. Franç. Phys. 60 (1986) 14.
- 1/7 J. Serra, Image Analysis and Mathematical Morphology, Academic Press, London, 1982.
- 1/8 F. Devos, P. Garda, P. Chavel, Opt. Let. 12 (1987) 152.
- 1/9 B.K. Jenkins, A.A. Sawchuk, T.C. Strand, R. Forchheimer and B.H. Soffer, Appl. Opt. 23 (1984) 2455.
- 1/10 Y. Ichioka and J. Tanida, Proc. I.E.E.E. 72 (1984) 787.