

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/252577038>

# Optoelectronic implementation of cellular automata for complex vision algorithms

Article in *Proceedings of SPIE - The International Society for Optical Engineering* · June 2001

DOI: 10.1117/12.432874

---

CITATIONS

0

---

READS

10

3 authors, including:



**Pierre H. Chavel**

Institut d'Optique Graduate School

250 PUBLICATIONS 4,024 CITATIONS

[SEE PROFILE](#)



**Alvaro Cassinelli**

City University of Hong Kong

68 PUBLICATIONS 508 CITATIONS

[SEE PROFILE](#)

Some of the authors of this publication are also working on these related projects:



Furniture learns to move itself with vibration [View project](#)



Novel Human Computer Interfaces / opportunities [View project](#)

# Optoelectronic implementation of cellular automata for complex vision algorithms

Pierre Chavel<sup>a</sup>, Alvaro Cassinelli<sup>a</sup>, I. Glaser<sup>b</sup>

<sup>a</sup>Laboratoire Charles Fabry de l'Institut d'Optique (CNRS), BP147, F 91403 Orsay cedex, France

<sup>b</sup>MMRI, Jerusalem, Israel

## ABSTRACT

In this work, we elaborate on the compromise between the efficiency of a multiprocessor computer architecture for handling large classes of computing tasks and the good performance of optics for implementing shift invariant operations, in particular convolution. We derive a class of processors, optoelectronic cellular automata, that can efficiently implement intensive, low level vision tasks in a time compatible with application constraints up to standard video rates. As one illustration, a parallel simulated annealing task performing motion detection on an image sequence is demonstrated.

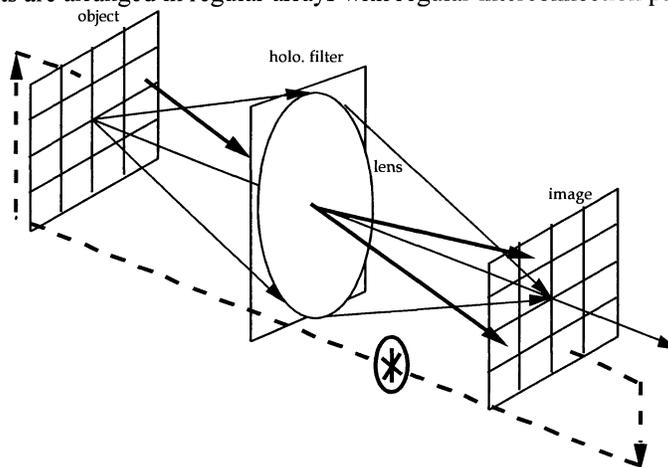
**Keywords:** correlation, random numbers, low level vision, motion detection

## 1. SHIFT INVARIANCE AND COLOURING IN PARALLEL PROCESSORS

It is commonplace that computer architectures are mostly organized in relatively large sections composed of identical units.

This is most evident when looking at the layout of a memory chip or a cache memory section in a processor, but can be found also in other parts of computer integrated circuits where fine grain parallelism at the level of registers and logic units is best implemented using a high degree of symmetry and more specifically, quite often, a high degree of shift invariance.

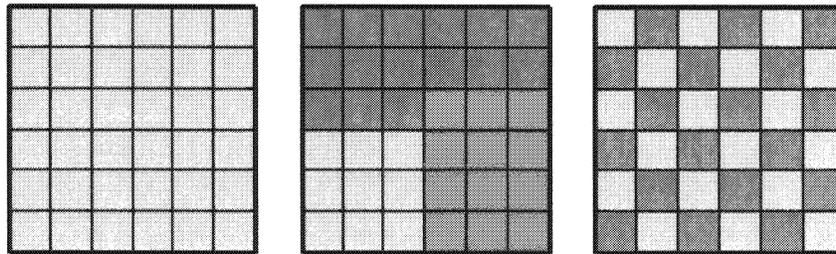
The same property of local shift invariance appears at higher levels in the architecture hierarchy as well, where for example random access memory circuits are arranged in regular arrays with regular interconnection paths.



**Fig. 1** An optical cellular automaton, based on optical convolution using an holographic filter and electronic feedback from the image to the object. The star denotes a point non linear operation performed on every pixel, the plain arrows exemplify light rays and the dotted lines arrows denote electronic interconnects. Every image pixel together with its associated electronic feedback and the corresponding object pixel constitute a processor cell, the light rays interconnect the processor cells with each other.

\*Correspondence: E-mail: Pierre.Chavel@iota.u-psud.fr

Optical imaging can be described to a good degree as a linear shift invariant operation, i.e. a convolution. The whole research domain of spatial frequency filtering has emerged from this discovery<sup>1</sup>. With a suitable lens system, that may include holographic filters in the pupil plane, light propagation is therefore a direct physical way to implement linear shift-invariant operations. From this fact, it has been derived<sup>2</sup> that a suitable point non linear operation NL acting on every pixel P of the image plane resulting from an optical convolution C is a good way to implement cellular automata with a degree of parallelism equal to the number N of image pixels, which can be a very large number compared to Fig. 1 usual in parallel computers. The time sequential operation of the automaton should then be implemented by feeding the result of convolution C and non linearity NL from the image back to the object plane and starting the next cycle (see Fig. 1). It has been shown that this process is in principle a general purpose computer<sup>3</sup>, i.e., it has the power of Turing machines, but no very efficient implementation of a Turing machine based on such a globally shift invariant massively parallel array has ever been proposed ; on the contrary, illustrations have been mainly limited to very simple operations of limited practical interest.



**Fig. 2** Shift-invariant (left), semi-shift invariant (center) and interlaced semi-shift invariant (right) processor arrays. All processor cells of the same shade perform the same operation at the same time. In the illustration, there are four patches in the semi-invariant array and two “colors” (distributed patches) in the interlaced semi-shift invariant array

Our driving idea here is therefore to efficiently implement more general classes of parallel machines by arranging the computing sites in a parallel computer into patches, each patch performing a shift-invariant operation that can be implemented with the support of optical convolution in an efficient way<sup>4</sup> (see Fig. 2). It is the main purpose of the present communication to demonstrate such a processor, which can be termed “semi-shift invariant” because of the various patches which are each shift invariant.

There is no reason, however, that the patches form compact areas. On the contrary, it turns out that fairly important image processing tasks can be performed by a semi-shift invariant layout where the invariant parts interlaced into an array of processor cells. While, with suitable spatial sampling, the whole array is globally shift-invariant, at a finer sampling its constituent processor cells are not. Instead, processor cells can be grouped into so-called “ colors ” of the processor and colors are interlaced in a fine, regular fabric. For example, all “ blue ” processor cells of the array perform one given operation at the same time and all “ yellow ” processor cells perform another given operation at the same time. Here we use the terms “blue” and “yellow” in a completely arbitrary manner, without reference to the true colors. We refrain from using “black” and “white” to avoid confusion with binary states of a one-bit processor.

The following is organized in two sections: algorithmic work on massively parallel image processing tasks (section 2) ; an experimental demonstration based on a combination of SLMs and a specialized CMOS circuit incorporating photodetectors (section 3); closely related work aimed at investigating compact implementations of similar processors using graded index optics deserves mentioning at this point<sup>5</sup>.

## 2. MASSIVELY PARALLEL STOCHASTIC APPROACHES TO IMAGE OPTIMISATION

Image optimization is an approach to low level image processing where the quality of the processed image is expressed with respect to a set of desirable criteria in the form of an energy (or cost) function. The energy function should be minimized. It is a challenge to find the global minimum of a problem that depends on a number of variable equal to the number of pixels in a realistic image, therefore sub optimal approaches need to be adopted. We identified stochastic approaches to image optimization as one powerful algorithmic tool to reach good sub optimal solutions in a massively parallel way. Specifically,

every pixel in the energy function affects the energy through its interplay with a limited number of other pixels, called its neighbors; parallel implementation must make sure that two pixels that are neighbors to each other do not evolve in parallel. Therefore, the image array should be “colored,” in the meaning of the word “color” that was just explained, in such a way that two pixels of the same colors are not neighbors to each other. A typical number of colors is a few units, so that massively parallel image processing can be envisioned: the degree of parallelism is linear with the number of pixels in the image.

In this section we select two stochastic image optimization examples as appropriate cases to illustrate our point. Both use simulated annealing, a stochastic procedure that was theoretically and experimentally shown to provide excellent results in various optimization problems. Knowledge of simulated annealing is assumed here; explanations and basic references can be found in references 6 and 7. The first algorithm is binary noise cleaning in a binary image. It is of academic interest only, and is used solely for the purpose of explanation. The second is motion detection in a gray level image sequence, adding to every gray level pixel a binary value that gives its state of motion. It is appropriate here to mention the difference between motion detection and the simple subtraction of successive images in a time sequence: a non zero difference between two such images may indicate motion, but it may as well be due to time variable noise or a change in illumination conditions; the purpose of motion detection is to give a complete estimate of all pixels in the scene that are moving at a given time and follow their motion in terms of compact, but deformable, moving objects.

Specifically, we started from a class of motion detection algorithms based on Markov Random Fields and first proposed by Bouthemy and Lalande<sup>6</sup>. The original algorithm starts by differentiating in time a gray level image to obtain a time gradient  $o(\underline{i},t)$ , where  $\underline{i}$  is a 2-D pixel coordinate and  $t$  is discrete time. A binary field  $e(\underline{i},t)$  is attached to the pixels, denoting the existence of absence of motion. The algorithm proceeds by compromising at every pixel between:

- the magnitude of the gradient
- the previous state of motion
- the present state of motion of its neighbors.

After some simplifications of the Bouthemy and Lalande algorithm that were needed to envisage optoelectronic implementation, we concluded from simulations that it is worthwhile to investigate a model where all fields, except the input image, are binary, i.e. the gradient magnitude is thresholded. In this model, the input to the photodetectors in the image plane of Fig. 1 is called a “force” and combines a convolution with a point operation on the  $e$  and  $o$  values of the pixel considered:

$$F(\underline{i},t) = \beta_s \sum_{j \in neighbor(\underline{i})} [2e(\underline{j},t) - 1] + \beta_t [2e(\underline{i},t-1) - 1] - \beta_a [2e(\underline{i},t-1) - 1] [2 Tresh o(\underline{i},t) - 1]$$

where  $Tresh$  denotes the binary thresholding operation of the time gradient with a given adjustable threshold. The  $\beta_\varepsilon$  are adjustable parameters.

### 3. OPTOELECTRONIC IMPLEMENTATION

Every parallel optoelectronic implementation of a stochastic image optimization processor relies on three ingredients: an optical imaging or convolution setup, an optical random number generator and an optoelectronic device to implement the required nonlinear operations. For the random number generator, based on extensive previous experience<sup>7</sup>, we selected laser speckle. In the following, we concentrate on the optical imaging and optoelectronic parts of the implementation.

#### 3.1. Device selection

Because of the general availability and maturity of silicon technology, in an earlier collaborative project with Institut d'Electronique Fondamentale, CNRS/Université Paris Sud<sup>8</sup>, we had developed a CMOS chip (see 7 and references therein), named SPIE24, consisting of 24\*24 identical processor cells for the optoelectronic simulation of an Ising spin with two phototransistors per cell serving as optical inputs. Of course, because light emission by silicon is far from being well mastered, there is no optical output. The optical inputs, however, are sufficient for the Ising spin problem with the spin interaction coefficients being limited to the four nearest neighbors and provided electronically on the chip. This work has been published earlier, demonstrating video-real time massively parallel implementation of an Ising spin array<sup>9</sup>.

It appeared that the same chip could be used for the binary noise in binary image problem by simply:

- setting all interaction coefficients of the Ising chip to unity,
- optically inputting the image to be processed onto the photodetectors on the chip.

This was done as the first full experimental demonstration of stochastic parallel image processing using the concept of semi-shift invariance by coloring<sup>10</sup>. There was, however, no real time image input ; instead, the data to be processed were recorded on a binary photographic mask that was projected onto the chip. In the present work, video real time operation was achieved and optical convolution and optoelectronic feedback were added to implement the motion detection algorithm mentioned in section 2. This is explained in detail below.

## **3.2. System design**

### **3.2.1. Purpose of the system**

With respect to the Ising spin demonstration mentioned in section 3.1 that used the same SPIE24 CMOS chip, two major improvements were required:

- the optical part of the demonstrator and the associated mechanics had to be designed for easy adjustment, stable operation over a long period of time, and easy transportation e.g. for presentation at seminars and conferences ;
- the input image should be changed in video real time. This was implemented through commercial magneto optic spatial light modulators (SLM).

In this subsection, the general design is described.

### **3.2.2. Operation of the system**

It is appropriate at this stage to review the operation of the SPIE24 CMOS chip (Fig. 3). It consists of an array of 24\*24 processor cells, hardwired in two “ colors ” according to the right-hand side scheme of Fig. 2. Each cell has six inputs and one output. The inputs are:

- two (analogue) photodetectors labeled + and -, i.e., the first optical input is considered positive and the second negative,
- and four digital binary connections from the four nearest neighbors that can take on values +1 or -1. Each input from a neighbor is the binary product of the output of the neighbor times a weight that is set electronically before processing starts ; in our case, all weights are identical.

The output is a thresholding “ sign ” operation based on the algebraic sum of the inputs. This output is digital; it can be read from the outside in raster scanned mode and is available as input to the nearest neighbors.

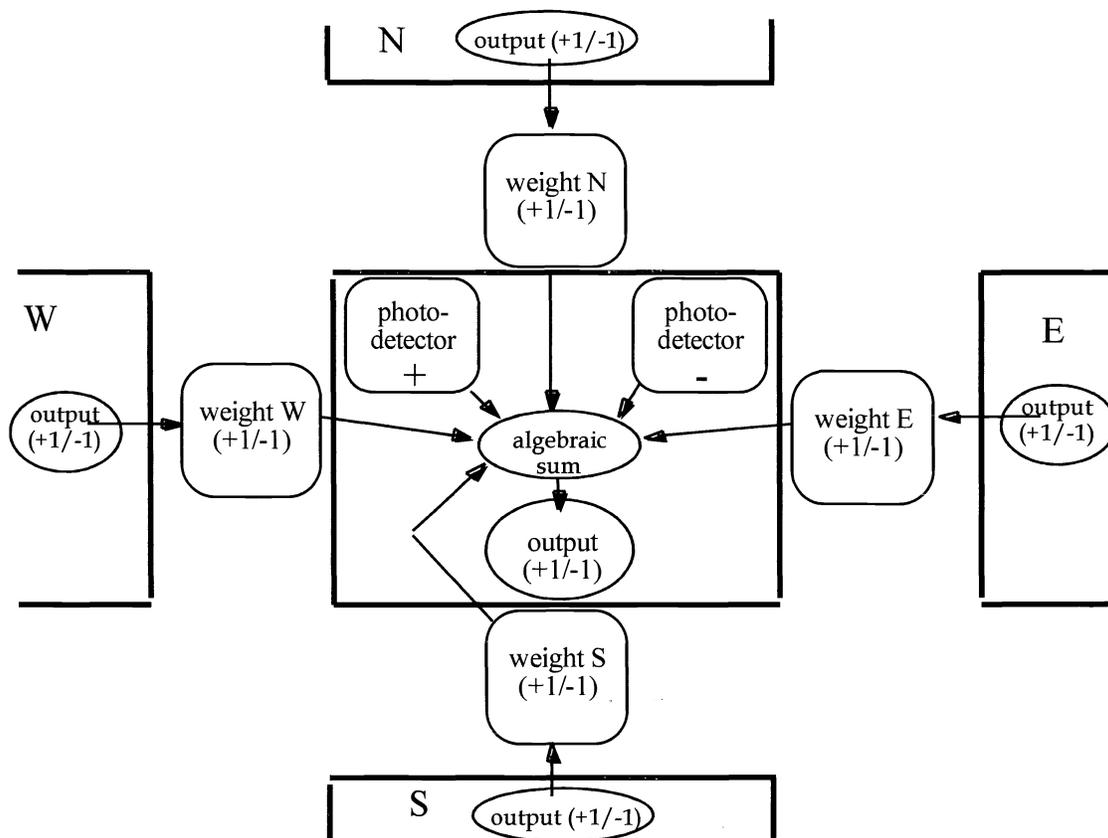


Fig. 3 SPIE24 chip, functionality of one processing cell. The output destinations, not shown, are to the W, E, S, N neighbors and to a common line for reading out the state of the whole array

Let us now review the desired operation of the system:

- one binary image to be processed is presented at the input every 40 ms and projected onto the SPIE24 chip photodetectors ;
- the SPIE24 chip performs one full simulated annealing operation in 40 ms. The output can then be observed or assessed and the processor is ready for the next input image.

One simulated annealing operation consists in typically 2000 cycles of 10  $\mu$ s each over the two “ colors ” interlaced in the image. It should be reminded here that two “ neighboring ”, i.e. interconnected, processor cells never belong to the same “ color ”. Each cycle is performed in parallel, first over all pixels belonging to the “ blue ” color, then over all pixels belonging to the “ yellow ” color, and it comprises:

- one convolution operation, which can be performed either electronically using the four nearest neighbors connection of the SPIE24 chip, or optically using a suitable filter ;
- one random number generation, which is performed optically using laser speckle projected onto the SPIE24 chip photodetectors ;
- and one thresholding “ sign ” operation at each pixel as just explained.

### 3.2.3. Imaging and convolution modes

Fig. 4 is a general overview of the system with two SLM inputs. Under this mode of operation, optics is used both for random number generation and for convolution, with the advantage that larger neighborhoods can in principle be reached (provided the "coloring" on the chip can be altered accordingly); one convolution filter and an additional SLM are then required. The electronic nearest neighbors interconnects available on the chip are then disabled.

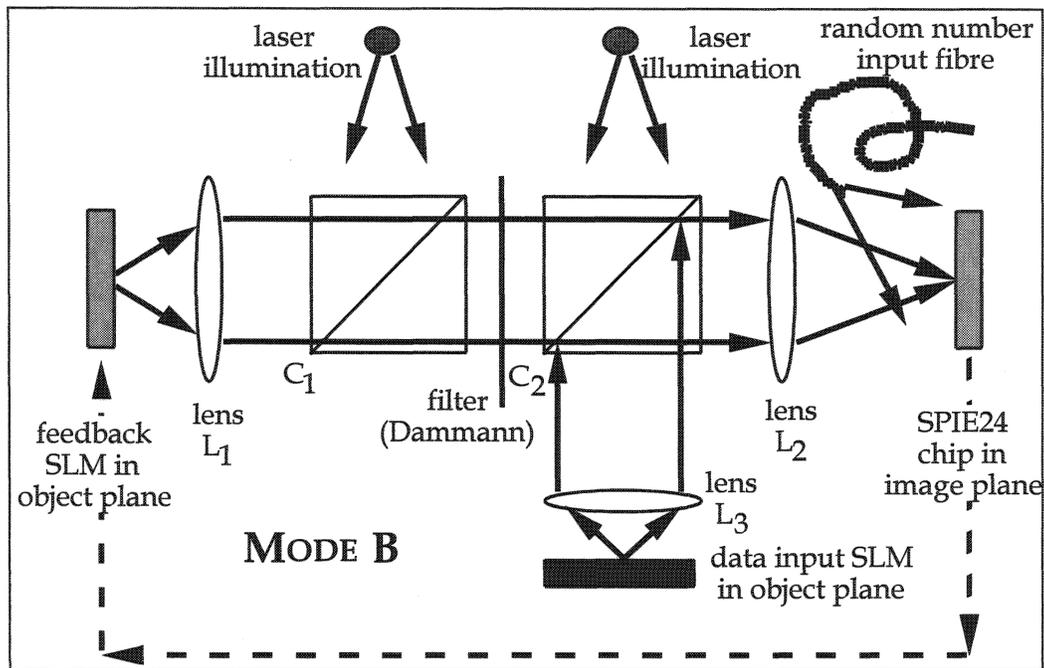


Fig. 4 General overview of the system. This mode of operation includes optical input, optical random number generation and optical convolution

The system is designed as follows. The input data are sent to the data input SLM at video rate. Lenses  $L_2$  and  $L_3$  together image the object on the image, i.e. the SLM on the SPIE24 chip. Because reflective mode SLMs are used, illumination has to be provided from the front, whence the need of beam splitter cubes. Cube  $C_2$  is used for alignment and input control using a CCD camera. Polarizing beam splitters are used to increase the overall system throughput. The optical random number input needed for simulated annealing is provided from the side directly onto the SPIE24 chip. CodeV™ simulations of the optical system showed satisfactory imaging quality over the whole field (the square photodetectors on the SPIE24 chip are  $35\ \mu\text{m}$  on a side). A second SLM is been placed on the left hand side for image data feedback from the SPIE24 chip. These data are convolved onto the SPIE24 chip using lenses  $L_1$ ,  $L_2$  and the convolution filter, which is of the general class of Dammann gratings in all applications that we considered so far.

### 3.2.4. Sampling effects and magnification

One difficulty derives from the fact that both the SLM and the SPIE24 chip are spatially sampled into pixels. Undesirable moiré effects may result if the magnification is not selected properly, and this condition determines the desirable system magnification, which is also the focal length ratio of lenses  $L_2$  and  $L_1$  (and  $L_3$  and  $L_1$  as well).

The relevant parameters are:

For the SPIE24 chip :

- $24 \times 24$  processor rectangular cells, each  $214\ \mu\text{m} \times 216\ \mu\text{m}$  ;
- total size  $5.136 \times 5.184\ \text{mm}$  ;
- detectors are square,  $35\ \mu\text{m}$  on a side ;
- center to center spacing of the detectors is  $128\ \mu\text{m}$ .

For the SLM :

- $256 \times 256$  square pixels, each  $14 \times 14\ \mu\text{m}$ , spaced  $15 \times 15\ \mu\text{m}$  ;
- total size  $3.84\ \text{mm}$ .

We found that with a magnification of round 1.4, the whole SPIE24 chip area is illuminated, and each detector receives the image of between  $2 \times 2$  and  $3 \times 3$  SLM pixels. SLM pixels were in fact be grouped by software into patches of  $3 \times 3$  pixels without any risk of overlapping at detector side. These groups will be called “ macropixels ” in the following. A slight cell to cell illumination nonuniformity results from the uncontrolled position of the dead zones between SLM pixels with respect to the SPIE24 photodetectors, but because the dead zone is so small this can easily be shown to be less than 5% ; because we have been working on binary images, it has not resulted in any degradation of the performance. This mode of operation was adopted for the demonstrator.

Needless to say, the rotation of the SLM image with respect to the chip should be fine tuned to better than one SLM pixel accuracy.

### 3.3. The components

Lenses, beam splitter, optical benches, holders were purchased from optical component vendors, with some mechanical parts made locally. Here we focus on the two more specific components, SLM and optical convolution filters (Dammann gratings).

#### SLM, SLM control

We decided to use our system with binary images and, for reasons that will be come clear in this paragraph, we needed a fast SLM response time. Therefore, we selected ferroelectric liquid crystal SLMs. A model by Displaytech was found most appropriate for our needs.

In the commercial version, buffer memories must first be loaded from a computer; with a Cyrix 166 processor, the loading operation takes about 20 seconds. It appeared therefore appropriate to bypass the commercial driving electronics and develop a direct access card to drive the SLM at a rate of thousand of frames per second. In this mode, at each iteration of the simulated annealing algorithms, data must be fed back from the SPIE24 chip to the “ feedback SLM ” of Fig. 4b. The “ data input SLM ” can still be operated at video rate and receive gray level images.

#### The optical filter. Dammann gratings

Whenever a point spread function consisting of a set of peaks with an even distribution of intensity is required, Dammann gratings<sup>11</sup> can advantageously be used as spatial frequency filters, because they avoid the frequent drawback of working off axis with a single side lobe hologram. It is essential for our demonstration of the advantages of optics for convolution operations that the neighborhood size, i.e. the number of orders in set S or the number of peaks in the point spread function, can be larger than four, because microelectronics circuit design can handle four nearest neighbor interconnects easily but gets much more complex when larger connectivity is needed. In the algorithmic cases we considered, the required point spread function did comply with the constraints of Dammann gratings. In the present study, we implemented a mere simulation of the SPIE24 neighborhood, with a four points point spread function that sends equal amounts of light to the N, S, E and W neighbors.

The Dammann grating were fabricated on our photoreduction bench using glass plates spin coated with photoresist. One issue that arises with our spatial filtering setup is coherence. The point spread function is specified in intensity, and we did not want to face the difficulties of coherent noise and hard to control phase differences that are known to plague coherent spatial filtering. Static speckle was found to be sufficient to remove unwanted interference effects.

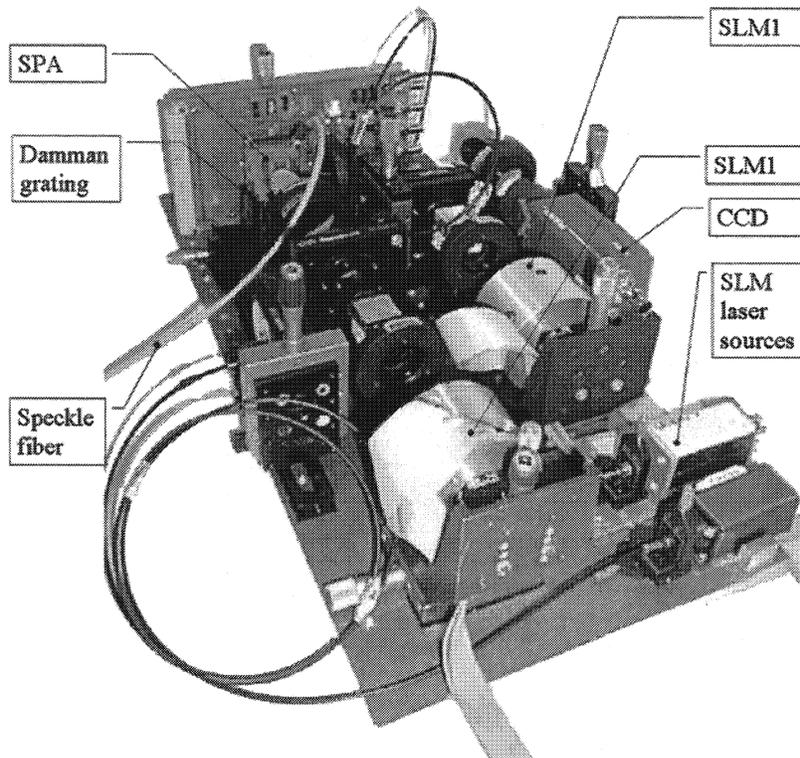


Fig. 5 A view of the setup. SPA is the SPIE24 chip

### 3.4. Demonstration results

The demonstration of the modified Bouthemy-Lalande algorithm mentioned in section 2 was completed. While video rate was compatible with the components, non optimal operation of the Windows computer interface issues with the driving PC slowed the process to about 10 seconds per image instead of the expected video rate. Aside from that problem, good operation of the simulated annealing algorithm for motion detection was observed. Because our images consist of  $24 \times 24$  binary points and the algorithm uses two "colors", the level of parallelism is  $24 \times 24 / 2 = 238$ , but this is restricted only by the cost of dedicated chip design and fabrication. With presently available technology, it would be possible to design a new version of the chip with many more pixels, perhaps several hundreds on a side, and the processor operation would be exactly as fast.

### ACKNOWLEDGEMENTS

Part of this work was done under contract ERBCI1\*CT93-0004 of the European Commission. The essential role of Eric Belhaire, Francis Devos, Antoine Dupret, Patrick Garda and Jean-Claude Rodier in designing and testing the SPIE24 chip is gratefully acknowledged. Philippe Lalanne and Donald Prévost participated actively in the first stages of this work.

### REFERENCES

1. P.M. Duffieux, *L'intégrale de Fourier et son application à l'optique*, Besançon, 1946, A. Maréchal et P. Croce, *C.R. Acad. Sc.* **237** pp. 607-609, 1953 ;
2. A. Huang, *IEEE 10th Optical Computing Conf.* pp. 13-17, 1980 ; J. Taboury, J.M. Wang, P. Chavel, F. Devos and P. Garda, *Appl. Opt.* **27**, pp. 1643-1650, 1988.
3. K.H. Brenner, A. Huang and N. Streibl, *Appl. Opt.* **25**, pp. 3054-3060, 1986 ; M.J. Murdocca *Appl. Opt.* **26**, pp. 682-688, 1987;
4. B.K. Jenkins, P. Chavel, R. Forchheimer, A.A. Sawchuk and T.C. Strand, *Appl. Opt.* **23**, pp. 3465-3474, 1984;

5. A. Kirk, A. Goulet, H. Thienpont, N. McArdle, K.H. Brenner, M. Kuijk, P. Heremans and I. Veretennicoff, *Appl. Opt.* **36**, pp. 3070-3078, 1997;
6. Ph. Bouthemy and P. Lalande, *Opt. Engin.* **32**, pp. 1205-1212, 1993;
7. Ph. Lalanne, D. Prévost, P. Chavel, L. Garnero, "Stochastic Artificial Retinae", *Ann. Phys.*, 1999;
8. 67 A. Dupret, E. Belhaire, J.C. Rodier, P. Lalanne, D. Prevost, P. Garda, P. Chavel, " An optoelectronic CMOS circuit implementing a simulated annealing algorithm, ".*IEEE Journal of solid-state circuits* **31**, pp. 1046-1050, 1996;
9. P. Chavel, Ph. Lalanne, "On parallel algorithms for optical image processors," *Optical Computing 1994*, Edinburgh, August 1994, B.S. Wherrett, ed., pp. 11-16, IoP Publishing, London, 1995;
10. A. Cassinelli, P. Lalanne, P. Chavel, I. Glaser, " Demonstration of video-rate optoelectronic parallel processors for noise cleaning in binary images by simulated annealing, " *Optical Computing 1998*, Brugge, Belgium, 17-20 June 1998, SPIE Proc. Volume 3490, H. Thienpont, ed., pp. 163-166, SPIE, Bellingham, 1998;
11. H. Dammann and K. Görtler, *Opt. Commun.* **3**, pp. 321-323, 1971 ; J.L. Tribillon, *Pure Appl. Opt.* **3**, pp. 389-404, 1994 ; H. Dammann, E. Klotz, *Opt. Acta* **24**, pp. 505-515, 1977 ; U. Krackhardt, N. Streibl, *Opt. Commun.* **74**, pp. 31-36, 1989.