A BIT PLANE ARCHITECTURE FOR AN IMAGE ANALYSIS PROCESSOR IMPLEMENTED WITH P.L.C.A. GATE ARRAY

J.C. Klein, F. Collange & M. Bilodeau

Ecole des Mines de Paris
Centre de Morphologie Mathématique
35, rue Saint Honoré 77305 Fontainebleau Cedex
Tél. (1) 64 22 48 21, Téléfax (1) 64 22 39 03, Télex MINEFON 694 736.

Abstract

As Image Analysis resorts to increasingly powerful algorithms, the processing time is correspondingly extended. Consequently, system designers are constantly looking for new technologies and new architectures capable of improving processing speed without increasing the complexity and the cost of the machines.

To achieve this objective, the Centre de Morphologie Mathématique (Ecole des Mines de Paris) has designed and developed, a new image processor for Mathematical Morphology based on Programmable Logic Cell Array (PLCA) technology. This processor, incorporated into the Cambridge Instruments Quantimet 570, is capable of performing complex morphological transformations on 512 x 512 images of 8 bits per pixel at the rate of 27 msec per image. This speed, associated with extensive algorithmic software support, makes it an extremely powerful tool in the field of image analysis.

1. INTRODUCTION

Mathematical Morphology [1,2,3] is a technique for image analysis, which was initially based on set transformations. The results obtained within this framework have been extended to the study of functions [4]. The usual approach consists of transforming the image so as to simplify it, segment it and measure some of its components [5,6]. In order to achieve this, the pixels of the image are computed and modified according to their neighbourhood and, in some cases, according to information taken from another image. The transformation is obtained by the action of a structuring element composed of N x N points on each pixel. A value depending on the type of operation to be performed and on the relationships existing between the structuring element and the neighbourhood of the pixel is then assigned to the centre pixel. The architecture of the processor is based on the structuring element concept [7,17,18].

2. GENERAL STRUCTURE

The morphological processing unit [8,9,10] described in this paper has been implemented (Fig.1) into the Cambridge Instruments Quantimet 570 which is briefly presented below.

2.1 The Quantimet 570

The Quantimet 570 image processor consists of :
- 68000-based processor monitoring unit
- Graphic overlay and binary image memory
- Measurement processor
- Acquisition and display system
- Peripheral control interfaces
- 386-based PC compatible host computer serving as the user's interface.

2.2 The morphological processing unit

The morphological processing unit consists of :
- An image memory containing up to 8 banks of 8 512*512 images of 8

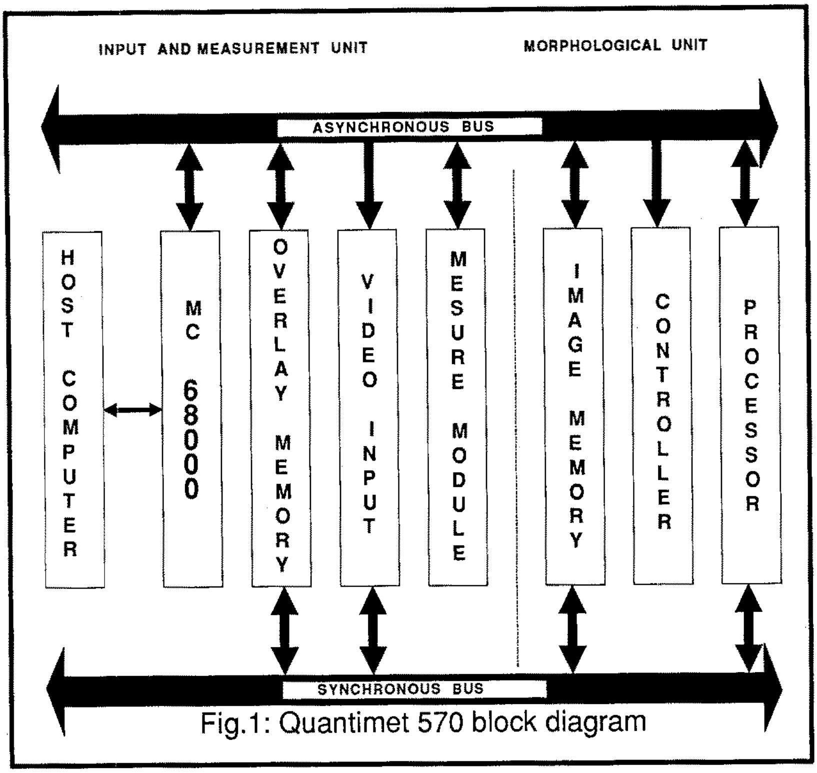


Fig.1: Quantimet 570 block diagram

bits. This represents a maximum capacity of 16 megabytes. Fast access time memory is used allowing 3 read-cycles and 1 write-cycle of 8 pixels each of 8 bits to be performed in 800ns. Two read-accesses and the write-access are reserved for the morphological processor, the third read-access is used to provide the synchronous display of a numerical image.

- The control and synchronization of the memory and the processor. The size of the image is programmable, up to 512 lines x 512 pixels.
- A morphological processor based on a programmable architecture, which gives it optimum adaptive properties for processing the different types of algorithms used in Mathematical Morphology.

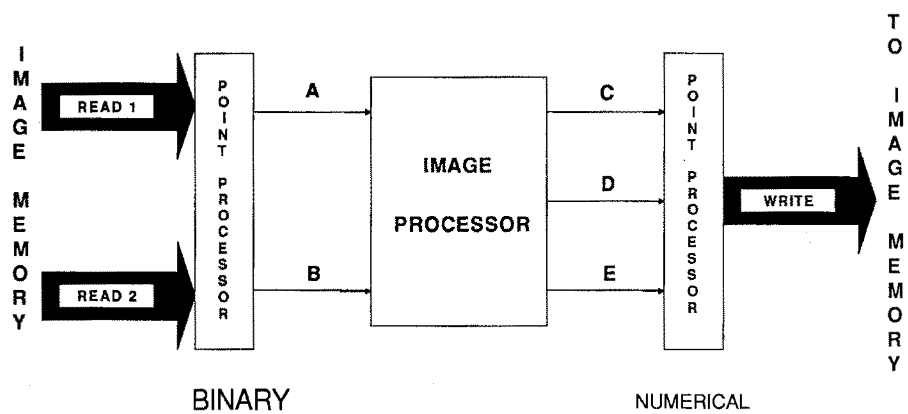
3. THE MORPHOLOGICAL PROCESSOR

The use of Programmable Logic Cell Arrays allows considerable computing capacity to be obtained within a small volume. The main specifications of the processor are the following :

- Pixel frequency : 10 Mhz
- Square sample grid
- Programmable architecture

This programmability permits the optimum adaptation of the processor to the treatment of binary and numerical images. This represents a considerable technological innovation, since binary and numerical processors use the same hardware facilities. They are both designed from a basic cell called a binary cell.

Additionally the processor supports euclidean and geodesic image treatment [11].



A: BINARY IMAGE BEING PROCESSED
B: INPUT GEODESIC PLANE
C: BINARY IMAGE AFTER PROCESSING
D: GEODESIC PLANE WITH DELAY
E: NOT USED

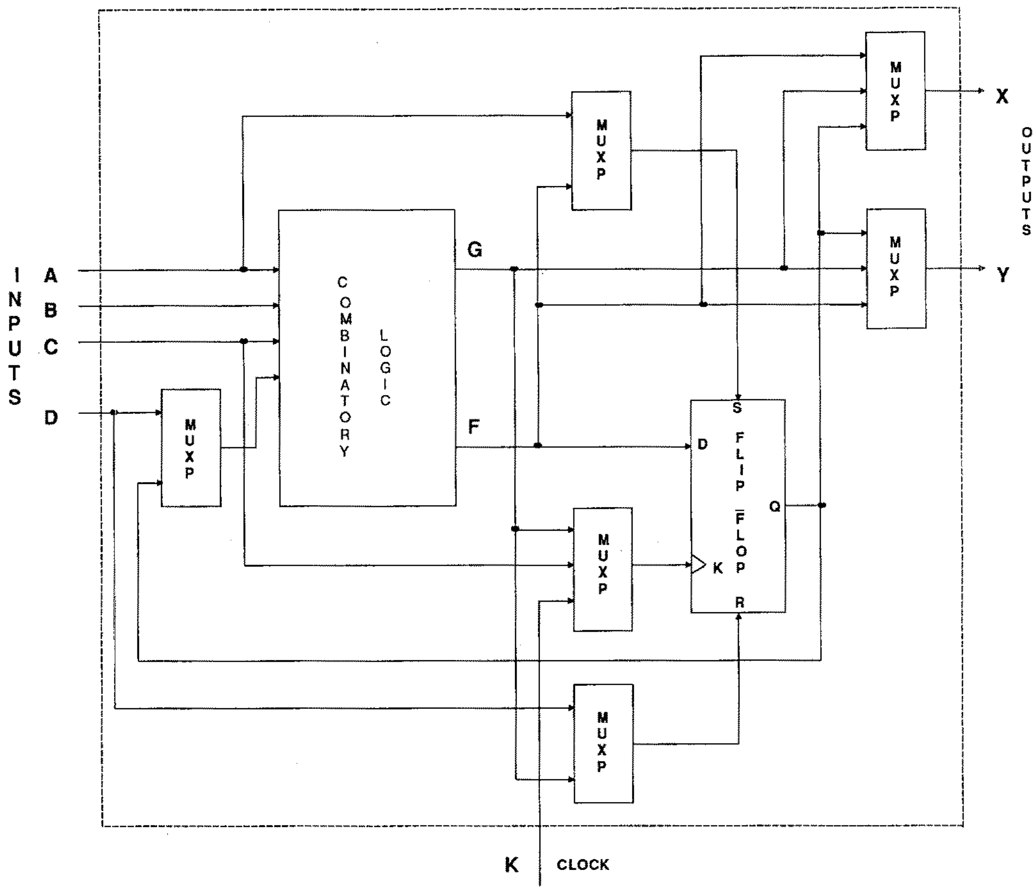
A: NUMERICAL IMAGE BEING PROCESSED
B: GEODESIC INPUT PLANE
C: ERODED NUMERICAL IMAGE
D: DILATED NUMERICAL IMAGE
E: CENTRAL POINT

Fig. 2 Morphological Processor

In numerical or binary mode, the neighbourhood morphological processor is followed and preceeded by a point processor (Fig. 2). These point processors, which are not fully described here, are look-up tables and logic and arithmetic operators.

3.1 Programmable Logic Cell Arrays

Programmable Logic Cell Arrays are a family of integrated circuit digital products, which combine high performance with a high level of integration. These circuits are also versatile, since their programming is not irreversible. This allows the configuration to be modified and loaded in its new form, at any time.



MUXP: PROGRAMMABLE MULTIPLEXOR

Logical unit programmable for one function of 4 variables or two fonctions of 3 variables.

Fig 3: Logic cell configuration

There are 3 main classes of programmable components contained in one such circuit. The number of components and their functions are dependent on the type.

- Logic blocks (numbering from 64 to 320) each contain one or two registers and a combinatory logic section. These blocks constitute the main functional element of the product. A logic block of the XILINX 2064 or 2018 series is presented in Fig. 3.

- Logic blocks take chargeof the I/O of the circuits (numbering from 58 to 144) and contain all the facilities required to interface the product with its environment. These blocks possess one or two registers to insure, if needed, the synchronization of I/O signals with a clock integrated in the gate array.

- Connections provide for the required interconnection network between the different logic blocks.

3.2 Binary cell

Fig. 4 represents the simplified diagram of a binary cell developed from the Logic Cell Arrays logic blocks. This constitutes the main element of the neighbourhood processor. This binary cell is used in both binary mode and numerical mode by the bin/num signal.

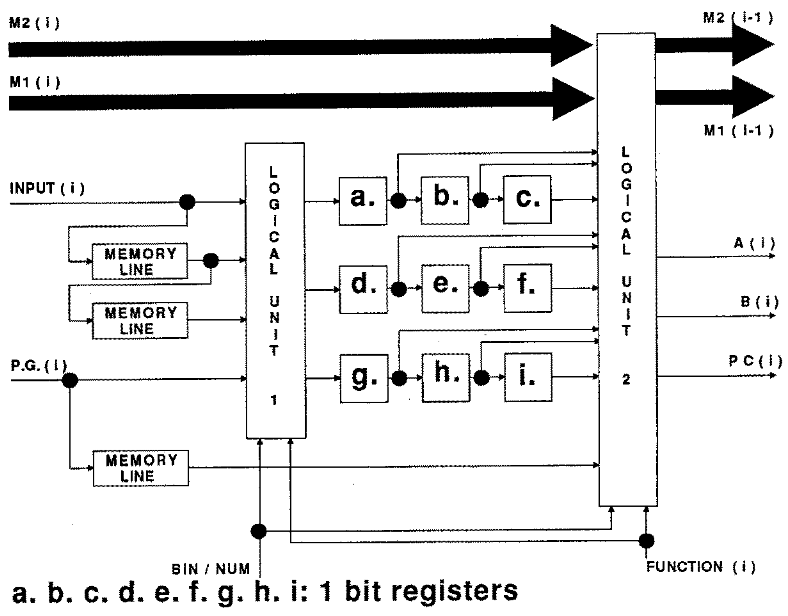


Fig. 4: Binary cell

The binary cell contains an input logical unit and an output logical unit. The part played by these logical units depends on the operating mode.

The input logical unit deals with the problems related to the field border in euclidean and geodesic mode [11].

The output logical unit enables the programming of the morphological operator and of the structuring element.

Binary mode:

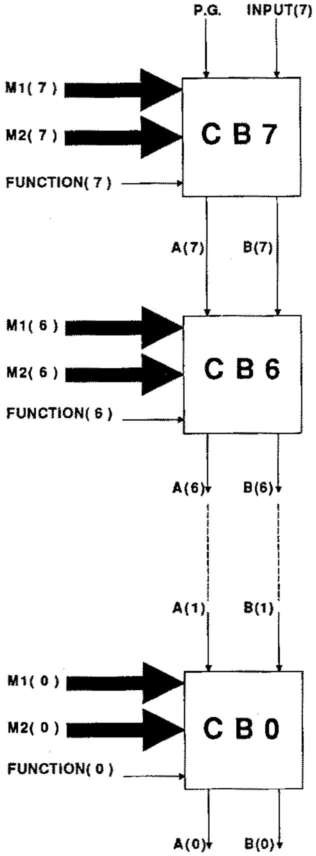


Fig. 5: Binary architecture

-INPUT(i) : Input of the binary processor located at position i in the pipe-line. Depending on the position of the cell, this input either stems from the image memory via the point processor, or from the preceding cell.

-FUNCTION(i): Program control of the type of binary transformation at step i.

-P.G.(i) : Geodesic binary plane used by the processor i.

-M1(i) : 8-bit data indicating the eighbourhood points that must be at 0 in the binary transformation of step i.

-M2(i) : 8-bit data indicating the neighbourhood points that must be at 1 in the binary transformation of step i.

-M1(i-1) : Not used

-M2(i-1) : Not used

-A(i) : Result of the binary transform at step i.

-B(i) : Geodesic bit plane phased by processor i

-PC(i) : Not used.

The input and output logical units serve to manage and compute the masks that indicate the neighbouring pixels that are still considered in the computation of the erosion and of the dilation. (Note-dilation corresponds to the examination of the neighbourhood points to determine the maximum value and erosion to the minimum).

3.3 Binary architecture

The interconnection of the binary cells is shown in Fig. 5. There are 8 elementary binary cells in a pipe-line. Each is a complete processor capable of performing any binary morphological operation on a 3*3 kernel [16,21].

An example of the application of this processor is its ability to perform in one cycle the 8 directions of a skeleton [12].

3.4 Numerical architecture

The interconnection of the binary cells to form the numerical processor is shown in Fig. 6. Each binary cell processes one bit plane of the numerical image. The numerical processor may be considered as a processor in slices of the type MISD (Multiple Instruction Single Data) [19,20].

The 8-bit numerical processor is capable of performing simultaneously 2 morphological operations on two programmable 9-bit kernels. This corresponds to computating the numerical erosion and dilation on two identical or distinct kernels.

An example of the application of this processor is its ability to perform in one cycle only a morphological gradient, a numerical thickening or a numerical thinning [12,1].

4. DATA FLOWS

4.1 Communication with the host system

A first communication channel of the asynchronous type, allows the Control Processor of the Quantimet to control the morphological processor via its 68000 microprocessor.

Numerical mode :

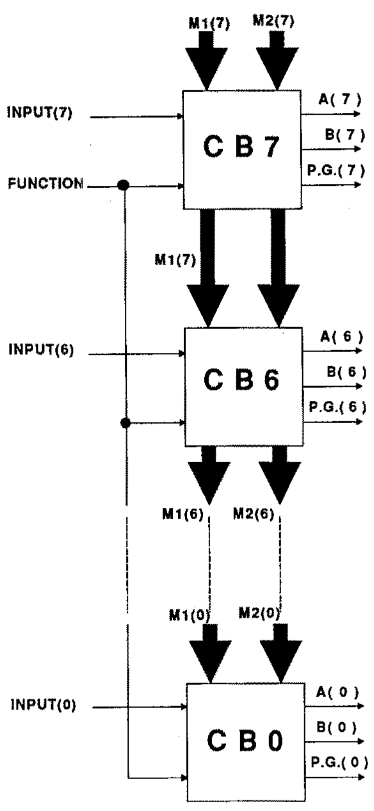


Fig.6: Numerical Architecture

-INPUT(i) : Input of the bit plane number i coming from the image memory via the point processor.

-FUNCTION(i): Program the type of numerical transformation (identical at each step).

-P.G.(i) : Not used.

-M1(i) : 8-bit data indicating the neighbourhood points that are taken into account at step (i) in computing the numerical dilation.

-M2(i) : 8-bit data indicating the neighbourhood points that are taken into account at step (i) in computing the numerical erosion.

-M1(i-1) : 8-bit data indicating the neighbourhood points that are taken into account at step (i-1) in computing the numerical dilation.

-M2(i-1) : 8-bit data indicating the neighbourhood points that are taken into account at step (i-1) in computing the numerical erosion.

-A(i) : Bit plane number i of the erosion of the neighbourhood.

-B(i) : Bit plane number i of the dilation of the neighbourhood.

-PC(i) : bit plane number i of the central pixel.

A second communication channel, synchronous with the video control signals, allows the transfer a 512x512 image in 40-milliseconds. This channel is used :

- to transmit numerical images, acquired by the Quantimet 570, to the memory of the morphological processor.
- to transmit the images from the morphological processor memory to the display of the Quantimet 570
- to exchange binary images between the morphological processor and the Quantimet 570.

The communication channels are shown in Fig. 1.

4.2 Communication between the image memory and the morphological processor

The communication between the image memory and the processor are managed by a specific controller. Pixel data is propagated at a rate

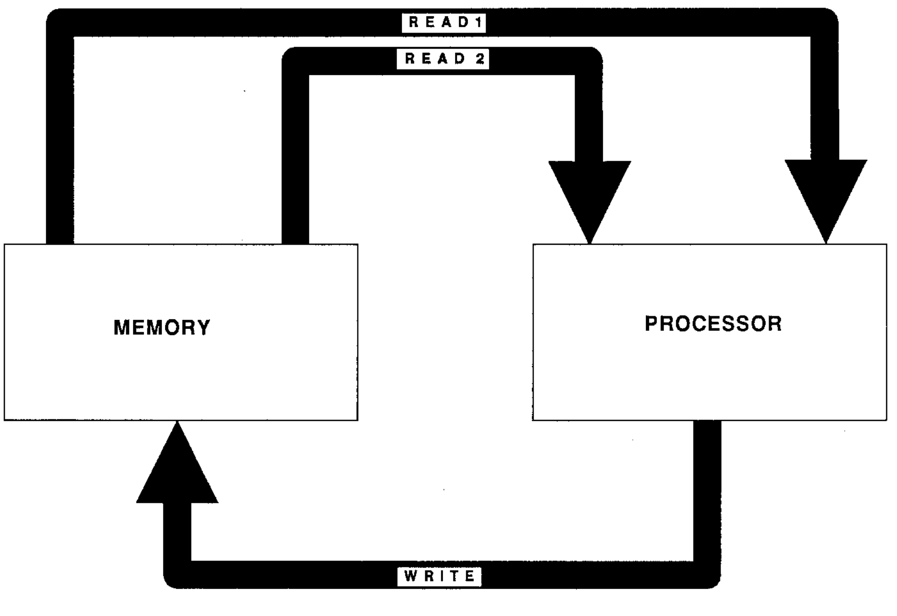


Fig. 7: Communication between memory and morphological processor

of 100ns per pixel on three buses (Fig. 7) which are asynchronous with respect to the video control signals. This architecture was chosen to permit the simultaneous reading (READ1 and READ2) of two 8-bit numerical images, as morphological transformations give a resulting image (WRITE) by using one or two input images.

5. THE SOFTWARE

The morphological processing unit is controlled from a library of procedures installed in the 68000 microprocessor of the Q570. This unit communicates with the PC-type Control Processor via a parallel port and executes the instructions sent by the host computer. Messages made of instructions may be sent by any user program, for example, the QUIC menu program used by the Q570 or the QBasic interpreter. During the development, we interfaced to the library with a Pascal compiler. It includes approximately 85 procedures classified into the following categories.

5.1 Image memory access

The images can be accessed by transferring pixels, or blocks of variable size, or complete images. For example, the procedure LOADIMAGE img. "fred" transfers the image "fred", stored on disk, into the image memory number "img". The acquisition of an image from a camera can be performed with or without summation. Display instructions allow the display binary and numerical images, with an option of superimposing the binary image.

Instructions for the transfer of images between the Q570 and the morphological processor, allow the user to directly use the facilities provided by each system.

5.2 Neighbourhood transformations

The library contains all the basic transformations in binary and numerical modes. Table 1 summarises the main procedures. The use of elementary transformations allows the creation of new algorithms. The watershed is a technique of image segmentation which is similar to the grey level skeleton. The Top Hat transformation is used to produce spatially adaptive thresholding. The example given in Fig. 8 is a sequential alternating algorithm, which alternates openings and closings of increasing size. The five parameters are : source and target images (img.1, img. 2), the order according to which transformations must be performed (opening-closing or closing-opening) determined by "ind", "elst", which defines the shape of the structuring element, and an integer (n) specifying the number of iterations (3).

```
procedure Asf(img1,img2,elst,ind,n:integer;
var i:integer;
begin
   Greymove(img1,img2);
   if ind=0 then for i:=1 to n do
        begin
             Greyopen(img2,img2,elst,i);
             Greyclose(img2,img2,elst,i);
        end;
   else for i:=1 to n do
        begin
             Greyopen(img2,img2,elst,i);
             Greyclose(img2,img2,elst,i);
        end;
end.
```

Fig. 8: Example of morphological filtering program

Laplacian, gradient and user-defined image convolution functions are integrated in the library.

Watershed	Watershed divide line
Maxima	Detection of local maxima
Tophat	Top hat transformation
Greybuild	Reconstruction of numerical image
Gradient	Morphological gradient
Greythresh	Numerical image thresholding
Greyskel	Numerical image skeleton
Greyprune	Numerical pruning
Binskel	Binary skeleton
Binprune	Binary pruning
Skiz	Skeleton by zone of influence

Table 1 - Examples of algorithms available in the library

5.3 Arithmetic transformations

All the functions : addition, subtraction, multiplication and division between two images or between an image and a constant are available for 8 and 16 bit images. Utility programs allow an image to be converted from the signed value to the absolute value.

6. PERFORMANCE

We have used the first prototype as a development tool for our research in image analysis. Mathematical morphology applies to numerous fields, let us mention, among others, biology [13], medicine [14] and robotics [15].

Table 2 gives some examples of the time needed to compute 512x512 images. The execution time for each transformation is proportional to the size of the image. Some algorithms cannot be characterized by a fixed execution time as they depend on the image content. For example, skeletonization algorithms are of this type.

Operation	Size	msec
Grey dilate/erode	3 x 3	27
Grey open/close	3 x 3	54
Tophat transform	3 x 3	81
Gradient transform	3 x 3	27
Binary dilate/erode	17 x 17	27
Binary open/close	9 x 9	27
Geodesic dilate/erode	17 x 17	27
Point transform	-	27
Image add/subtract	-	27

Table 2 - Timings of typical operations

In order to estimate the processing potential of the machine, we have used the system for a particular application: the segmentation of proteins in an electrophoresis gel. The algorithm was studied at the Centre de Morphologie Mathematique, by S. Beucher [6]. Fig. 9 illustrates the results obtained after each step and thus allows the performance of the processor to be evaluated.

- Initial step: On the filtered image, regional minima are detected. Time: 1.5 s (Fig. 9b).

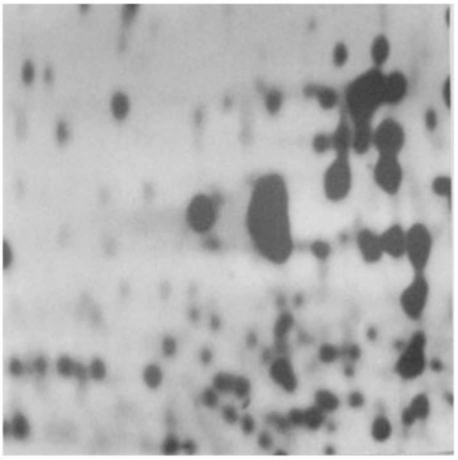


Fig. 9a: Initial image

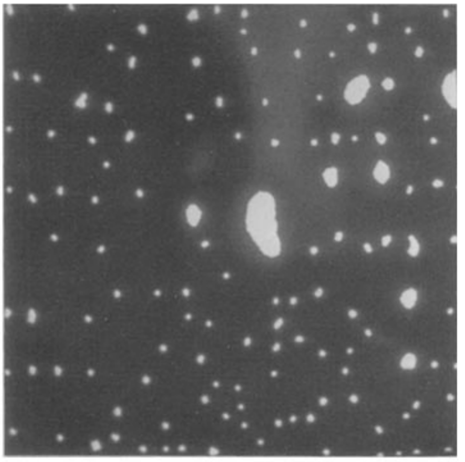


Fig. 9b: Minima of filtered image

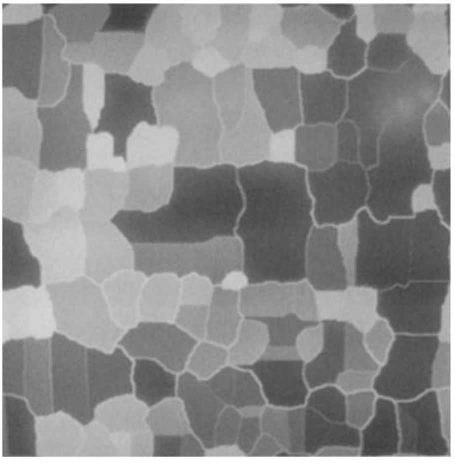


Fig. 9c: Catchment basins

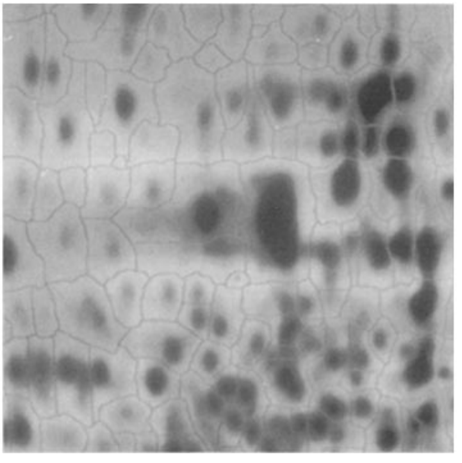


Fig.9d: Watersheds of filtered image

- Second step: The catchment basins of the filtered image are computed. This yields image 9c. Time: 35 s.
- Third step: The watersheds of the filtered image are obtained by an adaptive thresholding of image 9c (Fig. 9d). Time: 81 ms.
- Fourth step: The marker image is obtained by taking the union of the image of the minima and of the watersheds image (Fig.9e). Time: 27 ms.
- Final step: The marker image is used to modify the homotopy of the

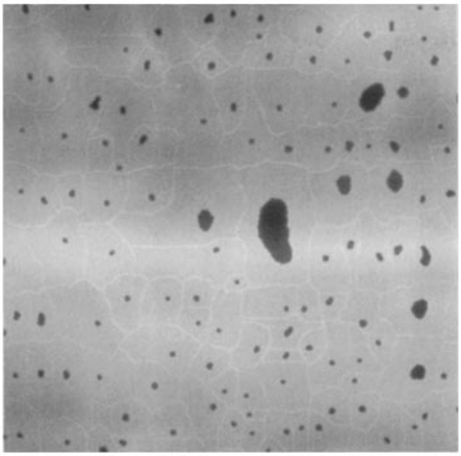


Fig. 9e: marker image

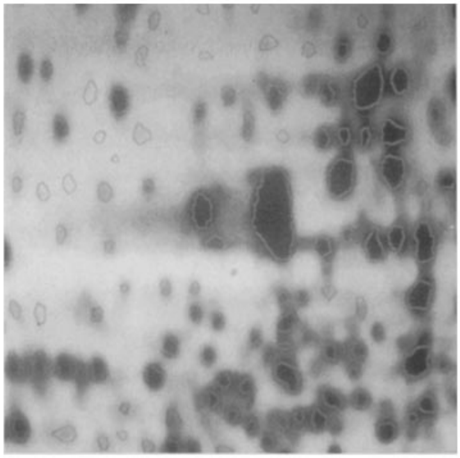


Fig. 9f: Contour of spots

gradient of the original image. Then, the watersheds of this modified gradient provide an accurate detection of the spot contours. This final result is displayed on Fig. 9f.

CONCLUSION

The binary processing cell, wich may be configured in numerical or binary mode, constitutes the core of the system. To develop it, we have used widely available. Programmable Logic Cell Arrays. This allowed an optimum ratio to be acheived between the degree of integration and the cost of the cell implementation.

The organization, programmed according to the processing mode being used, of eight cells in pipeline forms the morphological processor. This structure supports all morphological transformations. The time required for the execution of the basic transformations is much shorter than can be obtained from a classical computer.

The programming software is easy to use and provides the user with the whole range of the morphological tools for a moderate cost compared to its performance.

REFERENCES

[1] G. Matheron, *Eléments pour une Théorie des Milieux Poreux*, Masson, Paris, 1967.967.

[2] J. Serra, *Image Analysis and Mathematical Morphology,* Academic Press, London, 1982.

[3] J. Serra, *Image Analysis and Mathematical Morphology,* Vol 2: *Theoretical Advances*, Academic Press, London, 1988.

[4] S. Beucher, J. Serra, *Shapes and Patterns of Microstructures Considered as Grey-tone Functions,* Proc. 3rd European Symposium on Stereology, Ljubljana (Yougoslavie), june 22-26, 1981.

[5] M. Coster, J-L. Chermant, *Précis d'Analyse d'Images*, Ed. du CNRS, 1985.

[6] S. Beucher, L. Vincent, *Introduction aux Outils Morphologiques de Segmentation,* Journal de Microscopie et de Spectroscopie Electroniques, December 1989, to be published.

[7] M. Golay, *Hexagonal Pattern Transforms*, IEEE trans. on Comp., C18, 8, August 1969.

[8] J-C. Klein, *Conception et Réalisation d'une Unité Logique pour l'Analyse d'Images*, PhD Thesis, Université de Nancy, 1976.

[9] F.Collange, *Système de Vision "temps réel vidéo" Conception et Réalisation d'un Module d'Acquisition et de Mémorisation d'Images,* Phd Thesis, Université de Clermont, 1986.

[10] R. M. Lockeed, D. L. McCubbrey, S. R. Sternberg, *Cytocomputers: Architecture for Parallel Image Processing*, IEEE Worshop on Picture Data Description and Management, 1980.980.

[11] C. Lantuéjoul, S. Beucher, *On the Use of Geodesic Metric in Image Analysis*, J. of Microscopy, 121, 39-49, 1981.

[12] F. Meyer, *Skeletons and Perceptual Graphs*, Signal Processing, Special Issue in Advances on Mathematical Morphology, Vol.16, N°4, 335-364, April 1989.

[13] F. Meyer, *Quantitative Analysis of the Chromatin of Lymphocytes.* An Essay on Comparative Structuralism, Blood Cells 6, 159-172, 1980.

[14] B. Laÿ, *Analyse Automatique des Images Angiofluorographiques au Cours de la Rétinopathie Diabétique*, PhD Thesis, Ecole des Mines, Paris, 1983.

[15] S. Beucher, J. M. Blosseville and F. Lenoir, *Traffic Spacial Measurements Using Video Image Processing*, SPIE Cambridge Symp. on Advances in Intelligent Robotics Systems, November 1987.

[16] International Patent, Automatic scanning device for analyzing textures. Inventor: J. Serra. Assignee: Institut de Recherches de la Sidérurgie Française (I.R.S.I.D.). N°1.449.059, July 2, 1965.

[17] G. S. Gray, *Local Properties of Binary Images in Two Dimensions*, IEEE Trans. Comput. C-20, 551-561, 1971.

[18] International Patent, *Device for the Logical Analysis of textures.* Inventor: J. Serra. Assignees: Association pour la Recherche et le Developpement de Processus Industriels (A.R.M.I.N.E.S.) and J. SERRA. N°70 21 322, June 10, 1970.

[19] M.J.B. Duff, *CLIP4 A Large Scale Integrated Circuit Array Parrallel Processor*, 3 rd International Joint Conference on Pattern Recognition, 728-732, 1976.

[20] B. Kruse, *A Parallel Picture Processing Machine*, IEEE Trans. on Computers, Vol. C-22, N°. 12, 1075-1087.

[21] F.A. Gerritzen, L.G. Aardema, *Design and Use of a Fast Flexible and Dynamically Microprogrammable Pipe-lined Image Processor*, First Scandinavian Conference on Image Analysis, Linköping, Sweden, 1980.