Morphological Operators characterized by neighborhood graphs

JUNIOR BARRERA[1]
FRANCISCO DE ASSIS ZAMPIROLLI[2]
ROBERTO DE ALENCAR LOTUFO[2]

[1] IME-Instituto de Matemática e Estatîstica
Universidade de São Paulo
Rua do Matão, 1010 05508-900 - São Paulo, S.P.
jb@ime.usp.br

[2] FEEC-Faculdade de Engenharia Elétrica e de Computação
6101 - 13083-970 - Campinas, SP, Brasil
{fz,lotufo}@dca.fee.unicamp.br

Abstract. Mathematical Morphology is a theory that studies the decomposition of lattice operators in terms of some families of elementary lattice operators. When the lattices considered have a sup-generating family, the elementary operators can be characterized by structuring functions. The representation of structuring functions by neighborhood graphs is a powerful model for the construction of image operators. This model, that is a conceptual improvement of the one proposed by Vincent, permits a natural polymorphic extension of classical softwares for image processing by Mathematical Morphology. These systems constitute a complete framework for implementations of connected filters, that are one of the most modern and powerful approaches for image segmentation, and of operators that extract information from populations of objects in images. In this paper, besides presenting the formulation of the model, we present the polymorphic extension of a system for morphological image processing and some applications of it in image analysis.

1. Introduction

Mathematical Morphology [Ser88] is a theory that studies the decomposition of lattice *operators* (i.e., mappings) in terms of some families of simple lattice operators: erosions, dilations, anti-erosions and anti-dilations. The combination of these operators, that are called *elementary operators of Mathematical Morphology*, via the operations of intersection, union and composition permits the representation of any lattice operator [BB93]. When the lattices considered have a *sup-generating family* (i.e., a set of elements that is enough to create any other element of the lattice via the supremum operation) the elementary operators can be characterized by functions from the sup-generating family into the lattice, that are called *structuring functions*. A natural data structure for representing a structuring function is an oriented 1-graph, where the vertices are the elements of the sup-generating family and the edges are pairs of vertices (x, y) such that y is in the *sup-decomposition* (i.e., subset of the sup-generating family whose elements are less or equal a lattice element) of the image of x by the structuring function.

A particular example of lattice that has a sup-generating family is the set of functions from a finite set $E \subset \mathbb{Z}^2$ to an interval $K \subset \mathbb{Z}^+$, with the partial order inherited from the usual order relation between integer numbers. When the structuring functions considered are flat (i.e., are defined by a translation in K of a subset of E) the graphs necessary to represent structuring functions are reduced to the ones that have vertices in E.

In image processing applications, these graphs are in fact neighborhood graphs that represent the adjacency between objects or regions of the image. For example, E may represent the *flat zones* of the image (i.e., connected regions of the image with the same gray-level) and the edges may represent the adjacency between flat zones. This is the model used to represent *connected filters* (i.e., filters that do not affect the pixels of a flat zone or affect all the pixels of the flat zone of the same value) [SS95], that are one of the most modern and powerful approaches for image segmentation. In other applications, E could represent image objects, as cells or blobs, and the edges could be given by the application of some tessellation algorithm on these objects, reflecting their proximity under some distance measure. This is the model used to extract information from populations of objects in images.

In fact, these models constitute natural representations for some problems and so simplify the conception of their solutions. Additionally, when the objects or regions considered are composed of several pixels, the graph structure is compact and concentrates just the relevant information, which imply in fast algorithms. For example, the implementation of a connected filter as the inf-reconstruction [BBLJ97] may be many times faster in the graph structure than in the conventional image representation, since the iterative manipulation of image pixels is substituted by the iterative manipulation of flat zones of the image.

A similar model of representation was proposed by Vincent [Vin89]. In the original model of Vincent, the graph structure was associated to the function domain and, in our model, it is used to describe the structuring function. This new model is mathematically more consistent, since it is a particular case of the general representation of operators on lattices that have a sup-generating family and generalizes the representation of classical morphological image processing operators. The classical operators are represented by particular homogeneous structuring functions that are characterized just by a function, known in the literature by structuring element.

This new formulation permits a natural polymorphic extension of the softwares for morphological image processing and a complete equality between theory and implementation. The polymorphism is characterized, since both the classical morphological operators and the new ones have as arguments exactly a function and a structuring function, with particular characteristics in each case.

Following this introduction, section two presents the model proposed. Section three describes the architecture of a polymorphic software for morphological image processing. Section four gives some application examples. Finally, section five presents some conclusions and future directions for this research.

2. Characterization of morphological operators by graphs

Let $(\mathcal{L}, \leq)$ or, simply, $\mathcal{L}$ be a complete lattice [Bir67]. The elements of $\mathcal{L}$ will be denoted by the upper case letters X and Y. An *operator* ψ is a mapping from $\mathcal{L}$ to $\mathcal{L}$. The set of operators from $\mathcal{L}$ to $\mathcal{L}$ is denoted $\mathcal{L}^{\mathcal{L}}$.

An *erosion* ϵ is an operator on $\mathcal{L}$ that commutes with the infimum (i.e., $\epsilon(\wedge\mathcal{X}) = \wedge\epsilon(\mathcal{X}), \forall\mathcal{X} \subset \mathcal{L}$). The dual operator of an erosion ϵ is a *dilation* δ, which commutes with supremum (i.e.,$\delta(\vee\mathcal{X}) = \vee\delta(\mathcal{X}), \forall\mathcal{X} \subset \mathcal{L}$).

Let $\alpha, \beta \in \mathcal{L}^{\mathcal{L}}$. The pair (α, β) constitutes an *adjunction* on $\mathcal{L}$ if and only if

$$\beta(Y) \leq X \Leftrightarrow Y \leq \alpha(X), (X, Y) \in \mathcal{L}^2.$$

A fundamental property of the adjunction on $\mathcal{L}$ is that the only pairs of operators that satisfy this relation are the pairs formed by an erosion and a dilation. In other words, if (α, β) constitutes an adjunction, then α is an erosion and β is a dilation [HR89].

A subset ℓ of a complete lattice $\mathcal{L}$ is called a *sup-generating family* if any element of $\mathcal{L}$ can be written as a supremum of elements of ℓ. The elements in ℓ will be denoted by the lower case letters x and y. The set of sup-generating elements that is less or equal an element X in $\mathcal{L}$ is denoted $\ell(X)$. Hence, we have $X = \vee\ell(X)$.

Let E be a finite rectangle in $\mathbb{Z}^2$. The subsets of E represent the binary images and the collection of all binary images constitutes a complete lattice (under the usual inclusion relation on sets) that has as sup-generating family the set of *singletons* (i.e., points) of E. Analogously, a function from E to $K \subset \mathbb{Z}^+$ represents a gray-scale image and the set of all images K^E constitutes a complete lattice (under the partial order inherited from the usual order on integer numbers) that has as sup-generating family the *impulsive functions* (i.e., the set of functions $f_{u,v}$, such that, for all $(u, v) \in E \times K$, $f_{u,v}(u) = v$ and $f_{u,v}(x) = 0, \forall x \neq u$).

The functions from ℓ to $\mathcal{L}$ are called *structuring functions* and denoted by the lower case letter a. A structuring function a will be called a *structuring function with property of dilation* if there exists a dilation $\delta \in \mathcal{L}^{\mathcal{L}}$, such that $a(y) = \delta(y), \forall y \in \ell$.

The following theorem [Bar92] shows how erosions and dilations on lattices with a sup-generating family can be characterized by structuring functions.

Theorem - There exists a bijection between the set of structuring functions with properties of dilation and the set of pairs of operators that constitute an adjunction on $\mathcal{L}$. This bijection is given by, for any $X, Y \in \mathcal{L}$ and $y \in \ell$,

$$\epsilon(X) = \epsilon_a(X) = \vee\{y \in \ell : a(y) \leq X\},$$

$$\delta(Y) = \delta_a(Y) = \vee\{a(y) : y \in l(Y)\}$$

and

$$a(y) = a_{(\epsilon,\delta)}(y) = \delta(y).$$

On lattices that have a *negation* (i.e., that have a decreasing operator that composed with itself gives the identity operator) the anti-dilation and anti-erosion are given just by the composition of, respectively, dilation and erosion with the negation operator. Therefore, on this kind of lattices, the characterization of erosions and dilations by structuring functions and the known results of decomposition of lattice operators in terms of elementary operators imply that any lattice operator can be characterized by a collection of structuring functions. Two examples of lattices that have a negation are the lattices of binary and gray-scale images.

When E and K are Abelian groups, the most common structuring function used in image processing is the one that characterizes the translation invariant elementary operators and that is given by

$$a(f_{u,v}) = g_u + v,$$

where $(u, v) \in E \times K$ and g_u denotes the spatial translation of g by u, with $g \in K^B$ and $B \subset E$.

This kind of structuring function is called *translation invariant*, since it is built by translation of a function g.

Another interesting choice of structuring function is the one based on a neighborhood graph and given by

$$a(f_{u,v}) = g/D_G(u, r) + v,$$

where E is an arbitrary finite subset of $\mathbb{Z}^2$, $g/D_G(u, r)$ denotes the restriction of a function $g \in K^E$ to a disk $D_G(u, r)$ of center u and radius r, under the distance d_G defined on a graph G of vertices E. If g is a constant function equal to zero, then a will be a flat structuring function. Figure 2.1 shows an example of a dilation by a structuring function represented by a graph.

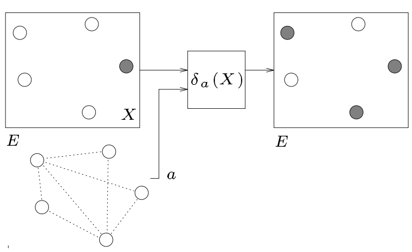

Figure 2.1. Dilation by a flat structuring function, with r=1.

The operators that are translation invariant only in K are characterized by the following structuring function

$$a(f_{u,v}) = a(f_{u,0}) + v.$$

Note that the other two structuring functions presented are particular cases of this one, that is called *K-translation invariant structuring function*.

The translation invariant structuring function is characterized by an homogenuous graph, that can be represented just by the function g, called in the literature *structuring element*. The flat structuring functions can be characterized by a neighborhood graph defined on E. The K-translation invariant structuring functions are characterized by a graph with edges of the type $(x, f_{u,v})$, where x is an element of E.

A general structuring function a from ℓ to $\mathcal{L}$ is characterized by a graph whose vertices are the elements of ℓ and the edges are pairs of vertices (x, y) such that $a(x) = Y$ and $y \in \ell(Y)$.

The classical morphological operators are characterized by structuring elements, while the ones studied by Vincent are characterized by flat structuring functions.

Note that using the characterization of lattice operator by structuring functions the polymorphic extension of a software for morphological image processing is quite natural. It is a question of adding algorithms for the elementary operators characterized by different classes of structuring functions and choosing the right algorithm based on the analysis of the given structuring function.

3. Polymorphic implementation of a morphological image processing software

The morphological operators based on generic structuring functions are useful in image analysis to extract properties of families of objects or to segment images. In both cases the function to be transformed constitutes a model of the image, that represents geometrical or gray-scale properties. In the case of image segmentation, the function has a point for each flat zone, which is associated to the gray-value of this flat-zone, and the structuring function is given by the flat zones adjacency graph. Transformations using this model permit the construction of any connected filter [SS95] and, consequently, solve any image segmentation problem without distortions on the edges. Another useful model for image segmentation is the function which has a point associated to each catchmen basin created by the watershed operator [Vin89]. In this case, the function may represent the mean gray-value or other statistical measurement in each region, and the flat structuring function is given by the adjacency graph of the watershed.

In order to apply this approach to image analysis problems, we need to extend the conventional morphological image processing softwares to systems that support structuring functions represented by neighborhood graphs. Additionally, we need tools to build the functions that will be processed, to create the neighborhood graphs, and to visualize graphs and functions.

We have implemented an extension of the MMach toolbox [BBL94] for dealing with structuring functions based on neighborhood graphs [Zam97]. Figure 3.2 presents a diagram showing the complete process from the creation of the image model function and the structuring function to the processing of the function and visualization of the results.

In the diagram of Figure 3.2, we see an input image that is represented by an array of pixels. From this image, we construct a function, for further processing, and a neighborhood graph, to represent the structuring function.

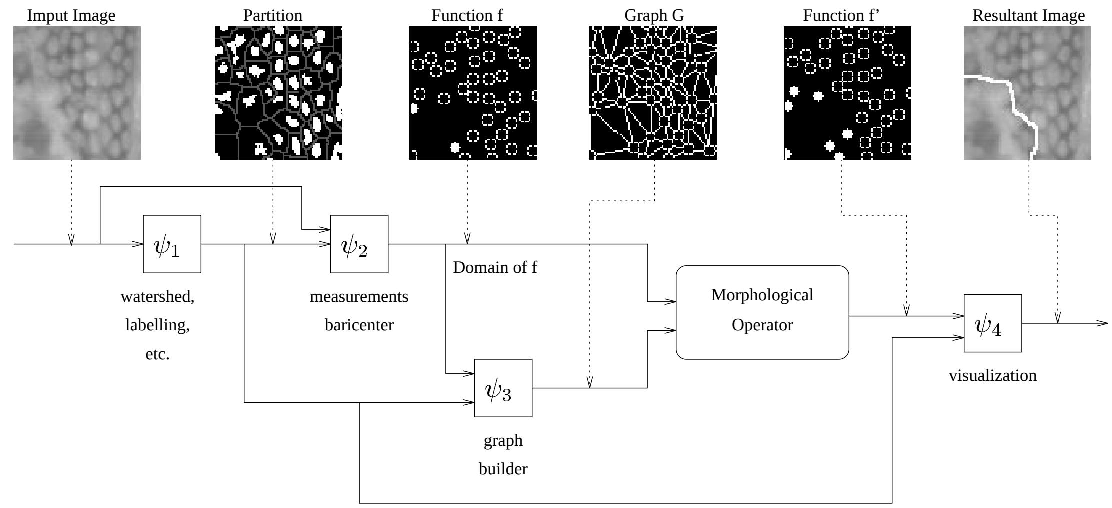

Figure 3.2. Graph based image processing.

The function is represented by an array which contains the coordinates of the centroid (or of another internal point) of each region or object, and its associated values. These values may represent geometrical properties of objects (e.g, area, perimeter, projections, number of adjacent regions, etc.) or gray-scale properties (e.g, mean, variance, median, maximum value, minimum value, value of a particular point, entropy, etc.).

We have two possible forms to construct neighborhood graphs: one based on tecellations of the function domain and the other based on image region adjancency. We implemented two kinds of tecellations: one based in the minimum distance and the other based on k nearest neighbours. In the minimum distance method, we define an edge between two points if they have a distance less than a fixed value, according with some distance measure. In the k nearest neighbour method, we define edges between a point and the k nearest neighbours of it, according with some distance measure. For the construction of graphs based on region adjacency, we use the partition of the image domain given by the watershed or by the flat-zones of the image. An edge is defined between two regions if they are adjacent.

The next step is the morphological processing of the function by the operators characterized by the neighborhood graph. Once Mathematical Morphology on functions leads to a hierarchical construction of operators from dilations and erosions, the extension of the software is simple. It is enough to create specialized algorithms for these two elementary operators and build a polymorphic structure integrating all the algorithms for each of them. Procceding in this way, the code for the other operators that depend on erosion and dilation do not need to be changed.

Additionally, other operators which were implemented in a non hierarchical way were extended by specialed algorithms based on the graph structure. Among these operators are distance function, geodesic distance function, watershed, labeling, area opening, etc.

Finally, we visualize the results of the processing. The function may be presented in two ways: as an image that contains a graphical representation of the graph or as an image such that for each pixel in a region is given the value associated to the point corresponding to that region in the function. This second kind of visualization is important, for example, for the visual verification of the results of the segmentation procedures.

In the next section, we show some applications of this system to image analysis problems.

4. Application examples

In this section, we present the solution of three image analysis problems by the methodology presented: detection of fracture lines in porous materials; detection of non productive parts of Eucalypt pulpwood for the production of celulosis; and segmentation of the faces of a block.

4.1. Fracture Lines

The first problem is the detection of the most probable fracture lines in a porous material (i.e., a material that has holes), when submited to mechanical efforts. The image (Figure 4.3) used in the experiment is a random simulation of a real image of such material. This problem was first studied by Vincent [Vin90].

The idea of the method used in the solution is that the material should break on lines that pass over the holes.

From all these lines, the most probable are the ones that have the minimum number of holes.

The method consists in building a neighborhood graph, defining two extreme points, computing the graph distance of the paths between these two points and choosing the paths of minimum distance. By choosing all possible pairs of points, we can compute the length of all possible fracture lines and choose the shortest (i.e., the one the pass over the smallest possible number of holes).

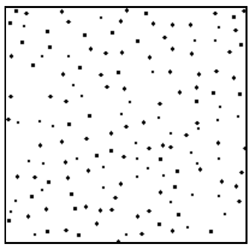

Figure 4.3. Image of the porous material.

The functions that model this image are binary functions, whose domain are the centroids of the holes. We work with two binary functions: one that is zero only in one of the end points considered and the other that is zero just in the other end point. Figure 4.4 shows the two end points chosen.

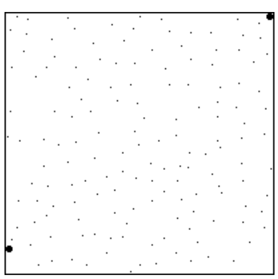

Figure 4.4. End points.

The neighborhood graph was constructed from the adjacency between regions of the watershed of the input image. Figure 4.5 presents the result of the watershed operator and Figure 4.6 presents the graph built from the watershed.

The next step is to compute the geodesic distance functions for the two images constructed. Note that the value associated to each point after the geodesic transform is the distance between that point and the zero end point. Figures 4.7 and 4.8 show the two geodesic distance functions produced.

Adding the two geodesic distance functions we get a new function, such that the value associated to each point is the length of the shortest path that goes from one end point to the other and pass over this point. Extracting the minimum value of this function, we get the path with minimum density of holes. Figure 4.9 presents the fracture line found in our example.

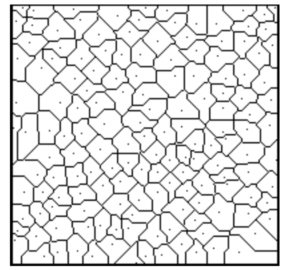

Figure 4.5. Watershed.

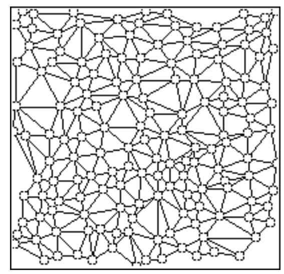

Figure 4.6. Neighborhood graph.

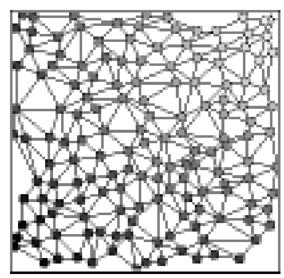

Figure 4.7. Geodesic distance.

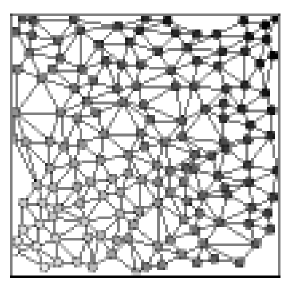

Figure 4.8. Geodesic distance.

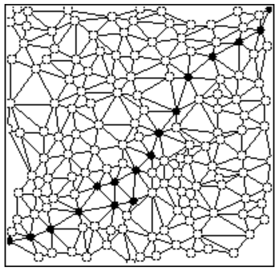

Figure 4.9. Fracture lines detected.

4.2. Eucalypt pulpwood image segmentation

The second problem is related to the evaluation of the capacity of production of celulosis from the trees of a given forest by the analysis of microscopic pulpwood images taken from these trees. Figure 4.10 shows a microscopic image of an Eucalypt pulpwood. Note that it has several small blobs immersed in a background. The non productive regions are the ones where the density of blobs is small. The goal is to segment the image to find these regions. This problem was proposed and first studied by [Jon96].

The idea of the method is to creat a graph, where the edges are the blobs, and to separate the regions such that the distance between blobs is larger than a given value.

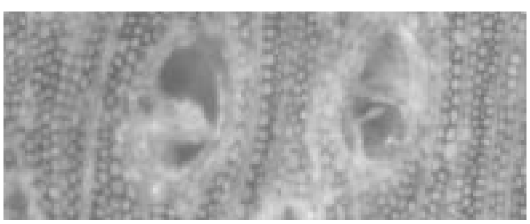

Figure 4.10. Microscopic image of the Eucalypt pulpwood.

The graph is created by the watershed of a first segmentation of the blobs, that is performed by a filtering (top-hat) followed by a threshold. This result is presented in Figure 4.11. The result of the watershed is presented in Figure 4.12.

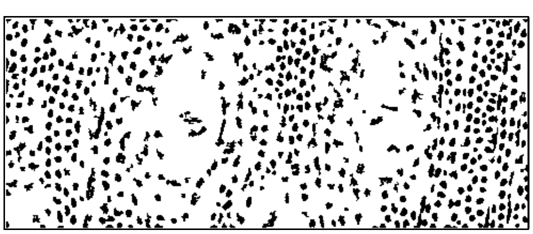

Figure 4.11. Segmentation of the blobs.

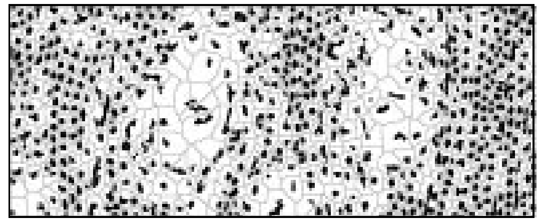

Figure 4.12. Watershed.

The function that models the image is created by associating to the centroid of each blob, the maximum Euclidean distance to its adjacent blobs under the graph constructed. The first operator applied to the function is the threshold, which separates the more isolated blobs from the others. Figure 4.13 presents the result of the thresholding of the image model function.

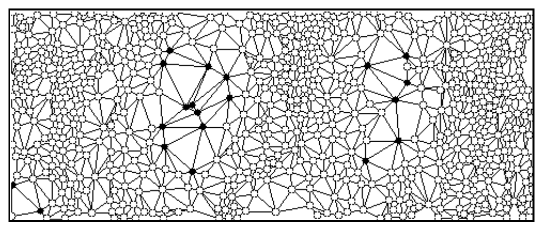

Figure 4.13. Threshold.

The next step is a closing (i.e., dilation followed by erosion) of size three, followed by a dilation of size one. Figure 4.14 presents the result of this operator.

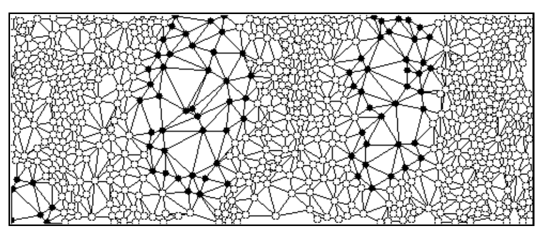

Figure 4.14. Dilation.

Finally, Figure 4.15 shows the result of the superposition of the original image of the regions detected from the graph (Figure 4.14).

4.3. Segmentation of a block

The third problem is the segmentation of the faces of a block. This segmentation may be useful, for example, in robotic vision for positioning of the robot arms relatively to the object. The image considered for the segmentation is the image presented in Figure 4.16.

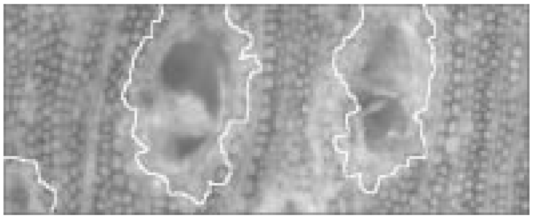

Figure 4.15. Segmentation.

The approach that we use for this problem is based on connected filtering. We build a graph of the flat zones and from a marker for each face we reconstruct the complete faces.

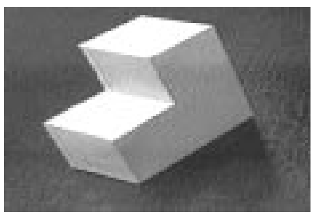

Figure 4.16. Image of a block.

The first step is the filtering by an alternated sequential filter. The result of this filter is the elimination of very small flat zones. Figure 4.17 presents the image obtained by this filter. From this filtered image we create the flat zone model for the image.

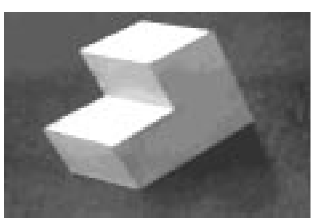

Figure 4.17. Filtering.

In the following, we label the image model function and choose some flat zones as markers for the faces, one for each face. Figure 4.18 presents the labeling of the flats zones and Figure 4.19 presents the flat zones chosen as markers.

Finally, from the image model and the marker we applied a connected expansion that preserves the number of markers, as proposed by Crespo [Cre93]. The result of this procedure is the segmentation of the block faces as shown in Figure 4.20.

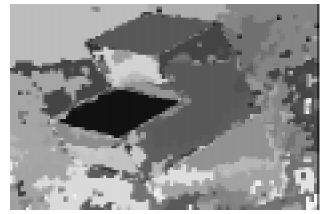

Figure 4.18. Labeling.

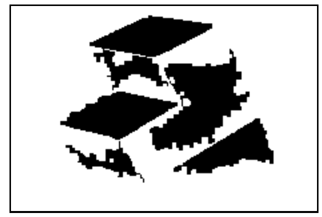

Figure 4.19. Markers.

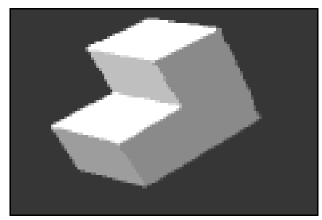

Figure 4.20. Segmentation.

5. Conclusion

In this paper we have presented a conceptual improvement of the model proposed by Vincent for the construction of morphological operators based on neighborhood graphs. This new model permits a complete equality between theory and implementation, and leads to a natural polymorphic extension of morphological image processing softwares. Furthermore, the complete class of connected filters can be implemented by this approach.

Using the conceptual model proposed, we have implemented an extension of the MMach toolbox and used it for the solution of some image analysis problems: detection of fracture lines on porous materials, identification of non productive regions of Eucalypt pulpwood and segmentation of the faces of a block.

The next step of this research will be to use this approach to study techniques for the design of connected filters.

6. Acknowledgments

We acknowledge Dr. Ronald Jones for kindly providing the Eucalypt pulpwood images. This project was partialy supported by ProTeM-CC/CNPq through the AnIMoMat project, contract 680067/94-9, and by Olivetti do Brasil S.A.

7. References

[Bar92] J. Barrera. *Uma abordagem unificada para problemas de visão computacional: A morfologia matemática*. Tese de Doutorado em Engenharia Elétrica apresentada ao Departamento de Engenharia Eletrônica da EPUSP, São Paulo, 1992.

[BB93] G.J.F. Banon and J. Barrera. Decomposition of mappings between complete lattices by mathematical morphology; part 1: General lattices. *Signal Processing*, 30:299–327, 1993.

[BBL94] J. Barrera, G.J.F. Banon, and R. A Lotufo. Mathematical morphology toolbox for the khoros system. *Conf. on Image Algebra and Morphological Image Processing V, International Symposium on Optics, Imaging and Instrumentation - SPIE's Annual Meeting*, 24-29 July, 1994.

[BBLJ97] J. Barrera, G. F. Banon, R. A. Lotufo, and R. Hirata Jr. MMach: a Mathematical Morphology Toolbox for the KHOROS System. *Submited to Jornal of Eletronic Image*, 1997.

[Bir67] G. Birkhoff. *Lattice Theory*. American Mathematical Society, Providence, Rhode Island, 1967.

[Cre93] José Crespo. *Morphological Connected Filters and Intra-Region Smooting for Image Segmentation*. A Thesis Presented to The Academic Faculty for Doctor of Philosophy in Eletrical Engineering, Georgia Institute of Technology, 1993.

[HR89] H.J.A.M. Heijmans and C. Ronse. The algebraic basis of mathematical morphology. i. dilations and erosions. *Comput. Vision, Graphics and Image Processing*, 50:245–295, 1989.

[Jon96] Ronald Jones. A graph-based segmentation of eucalypt pulpwood images. *CSIRO Division of Mathematics and Statistics - North Ryde - Australia*, 1996.

[Ser88] J. Serra. *Image analysis and mathematical morphology - Theoretical advances*, volume 2. Academic Press, London, 1988.

[SS95] P. Salembier and J. Serra. Flat zones filtering, connected operators, and filters by reconstruction. *IEEE-Transactions on image processing*, 4 - No 8:1153–1160, August 1995.

[Vin89] L. Vincent. Graphs and mathematical morphology. *Signal Processing*, 16:365–388, 1989.

[Vin90] L. Vincent. *Algorithmes Morphologiques A Base de Files D'Attente et de Lacets. Extension Aux Graphes*. These présentée a l'Ecole Nationale Supérieure des Mines de Paris par obtenir le titre de Docteur en Morphologie Mathématique, 1990.

[Zam97] F. de A. Zampirolli. *Operadores Morfológicos baseados em grafos de vizinhança - Uma extensão da MMach Toolbox*. Dissertação de Mestrado em Matemática Aplicada apresentada ao Departamento de Matemática Aplicada a Computação do IME/USP, São Paulo, 1997.