Morphological Grayscale Reconstruction in Image Analysis: Applications and Efficient Algorithms

Luc Vincent

Abstract

Morphological reconstruction is part of a set of image operators often referred to as *geodesic.* In the binary case, reconstruction simply extracts the connected components of a binary image I (the mask) which are "marked" by a (binary) image J contained in I. This transformation can be extended to the grayscale case, where it turns out to be extremely useful for several image analysis tasks. This paper first provides two different formal definitions of grayscale reconstruction. It then illustrates the use of grayscale reconstruction in various image processing applications and aims at demonstrating the usefulness of this transformation for image filtering and segmentation tasks. Lastly, the paper focuses on implementation issues: the standard parallel and sequential approaches to reconstruction are briefly recalled; their common drawback is their inefficiency on conventional computers. To improve this situation, a new algorithm is introduced, which is based on the notion of regional maxima and makes use of breadth-first image scannings implemented via a queue of pixels. Its combination with the sequential technique results in a hybrid grayscale reconstruction algorithm which is an order of magnitude faster than any previously known algorithm.

Published in the *IEEE Transactions on Image Processing,* Vol. 2, No. 2, pp. 176–201, April 1993.

1 Introduction

Reconstruction is a very useful operator provided by mathematical morphology [18, 19]. Although it can easily be defined in itself, it is often presented as part as a set of operators known as *geodesic* ones [7]. The reconstruction transformation is relatively well-known in the binary case, where it simply extracts the connected components of an image which are "marked" by another image (see § 2). However, reconstruction can be defined for grayscale images as well. In this framework, it turns out to be particularly interesting for several filtering, segmentation and feature extraction tasks. Surprisingly, it has attracted little attention in the image analysis community.

The present paper has three major goals: the first one is to provide a formal definition of grayscale reconstruction in the discrete case. In fact, we propose below two equivalent definitions: the first one is based on the threshold superposition principle and the second one relies on grayscale geodesic dilations. The second part of the paper illustrates the use of binary and especially grayscale reconstruction in image analysis applications: examples proving the interest of grayscale reconstruction for such tasks as image filtering, extrema, domes and basins extraction in grayscale images, "top-hat" by reconstruction, binary and grayscale segmentation, etc., are discussed. Lastly, in § 4, we introduce efficient algorithms for computing reconstruction in grayscale images. Up to now, the execution times required by the known grayscale reconstruction algorithms make their practical use rather cumbersome on conventional computers. Two algorithms are introduced to bridge this gap. The first one is based on the notion of regional maxima and uses breadth-first image scannings enabled by a queue of pixels [25]. The second one is a combination of this scanning technique with the classical sequential one [14], and it turns out to be the fastest algorithm in almost all practical cases.

We shall exclusively be concerned here with the discrete case. The algorithms are described in 2D, but their extension to images of arbitrary dimensions is straightforward. In order to be as precise as possible, Vincent90:Thesis algorithm descriptions are done in a pseudo-langage which bares similarities to *C* and *Pascal.*

2 Definitions

2.1 Notations used throughout the paper

In the following, an image I is a mapping from a finite *rectangular* subset D_I of the discrete plane $\mathbb{Z}^2$ into a discrete set $\{0, 1, \ldots N-1\}$ of gray-levels. A binary image I can only take values 0 or 1 and is often regarded as the set of its pixels with value 1. In this paper, we often present notions for discrete sets of $\mathbb{Z}^2$ instead of explicitely referring to binary images. Similary, gray-level images are often regarded as *functions* or *mappings.*

The discrete grid $G \subset \mathbb{Z}^2 \times \mathbb{Z}^2$ provides the neighborhood relationships between pixels: p is a neighbor of q if and only if $(p, q) \in G$. Here, we shall use square grids, either in 4- or in 8-connectivity (see Fig. 1), or the hexagonal grid (see Fig. 2). Note however that the algorithms described below work for any grid, in any dimension. The distance induced by G on $\mathbb{Z}^2$ is denoted d_G: $d_G(p, q)$ is the minimal number of edges of the grid to cross to go from pixel p to pixel q. In 4-connectivity, this distance is often called *city-block distance* whereas in 8-connectivity, it is the *chessboard* distance [2]. The elementary ball in distance d_G is denoted B_G, or simply B. We denote by $N_G(p)$ the set of the neighbors of pixel p for grid G. In the following, we often consider two disjoined subsets of $N_G(p)$, denoted $N_G^+(p)$ and $N_G^-(p)$. $N_G^+(p)$ is the set of the neigbors of p which are reached before p during a raster scanning of the image (left to right and top to bottom) and $N_G^-(p)$ consists of the neighbors of p which are reached after p. These notions are recalled on Fig. 3.

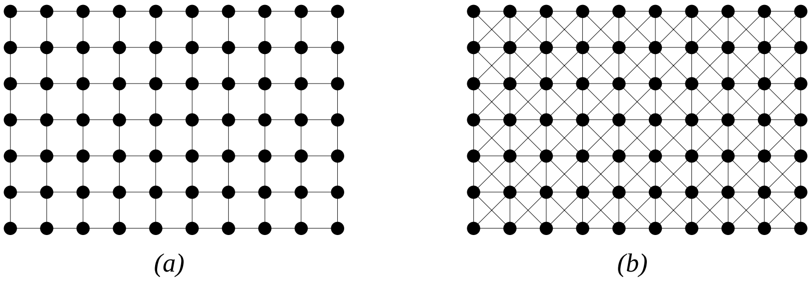


Figure 1: *Portion of square grid in 4- (a) and 8-connectivity (b).*

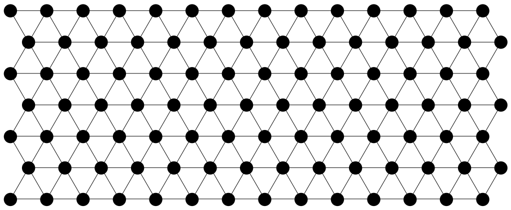


Figure 2: *Portion of hexagonal grid.*

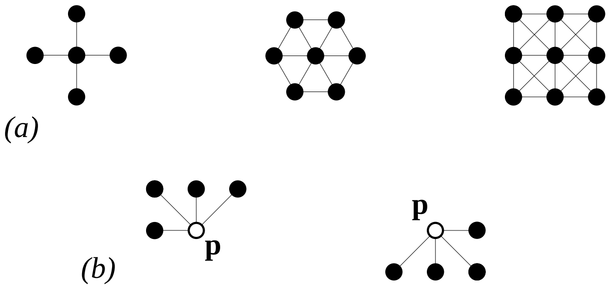


Figure 3: *(a) The elementary ball B in 4-, 6- and 8-connectivity; (b)* $N_G^+(p)$ *and* $N_G^-(p)$ *in 8-connectivity.*

2.2 Reconstruction for binary images

2.2.1 Definition in terms of connected components

Let I and J be two binary images defined on the same discrete domain D and such that $J \subseteq I$. In terms of mappings, this means that: $\forall p \in D, \quad J(p) = 1 \Longrightarrow I(p) = 1$. J is called the *marker* image and I is the *mask*. Let $I_1, I_2, \ldots, I_n$ be the connected components of I.

Definition 2.1 *The reconstruction $\rho_I(J)$ of mask I from marker J is the union of the connected components of I which contain at least a pixel of J:*

$$\rho_I(J) = \bigcup_{J \cap I_k \neq \emptyset} I_k.$$

This definition is illustrated by Fig. 4. It is extremely simple, but gives rise to several interesting applications and extensions, as we shall see in the following sections.

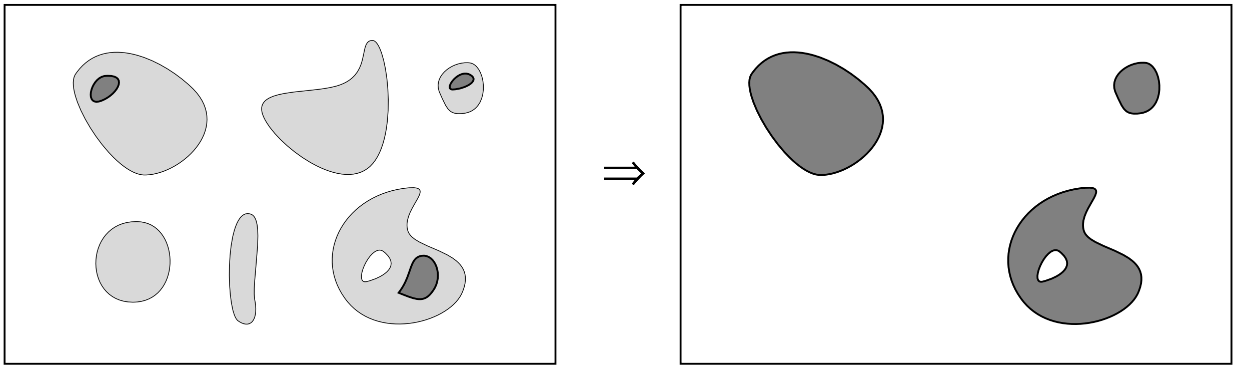


Figure 4: Binary reconstruction from markers.

2.2.2 Definition in terms of geodesic distance

Reconstruction is most of the time presented using the notion of *geodesic distance*. Given a set X (the mask), the geodesic distance betweeen two pixels p and q is the length of the shortest paths joining p and q which are included in X. Note that the geodesic distance between two pixels within a mask is highly dependent on the type of connecticity which is used. This notion is illustrated by Fig. 5. Geodesic distance was introduced in the framework of image analysis in 1981 by Lantuéjoul and Beucher [6] and is at the basis of several morphological operators [7]. In particular, one can define geodesic dilations (and similarly erosions) as follows:

Definition 2.2 *Let $X \subset \mathbb{Z}^2$ be a discrete set of $\mathbb{Z}^2$ and $Y \subseteq X$. The geodesic dilation of size $n \geq 0$ of Y within X is the set of the pixels of X whose geodesic distance to Y is smaller or equal to n:*

$$\delta_X^{(n)}(Y) = \{p \in X \mid d_X(p, Y) \leq n\}.$$

From this definition, it is obvious that geodesic dilations are extensive transformations, i.e. $Y \subseteq \delta_X^{(n)}(Y)$. In addition, geodesic dilation of a given size n can be obtained by iterating n elementary geodesic dilations:

$$\delta_X^{(n)}(Y) = \underbrace{\delta_X^{(1)} \circ \delta_X^{(1)} \circ \ldots \circ \delta_X^{(1)}}_{n \text{ times}}(Y). \tag{1}$$

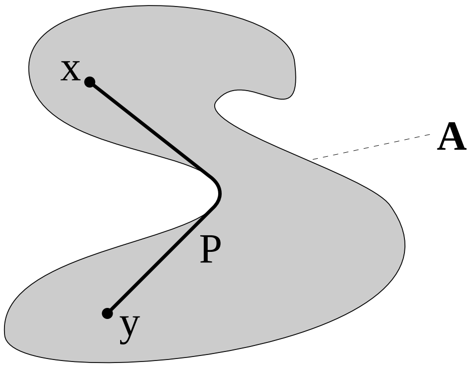
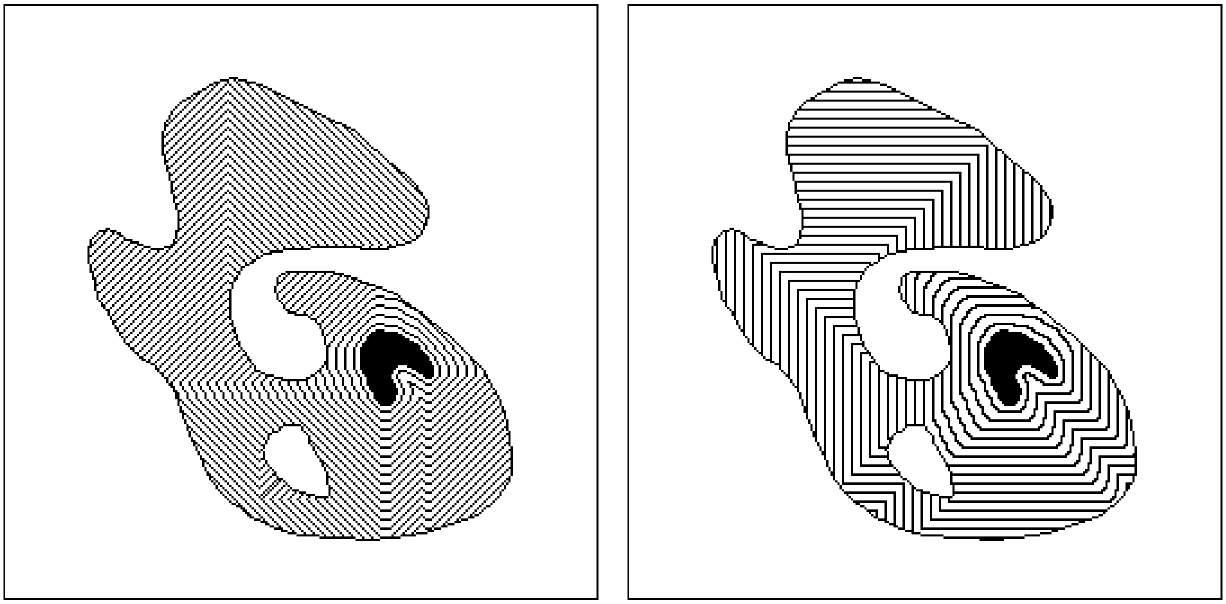


Figure 5: *Geodesic distance $d_G(x,y)$ within a set A.*

Fig. 6 illustrates successive geodesic dilations of a marker inside a mask, using 4- and 8-connectivity. The elementary geodesic dilation can itself be obtained via a standard dilation of size one followed by an intersection:

$$\delta_X^{(1)}(Y) = (Y \oplus B) \cap X. \qquad (2)$$

This last statement is absolutely wrong when non-elementary geodesic dilations are considered. In this latter case, one merely gets the *conditional* dilation of set Y, defined as the intersection of X and the standard dilation of Y. Note that some authors use a different terminology and utilize the word "conditional" for what this paper calls "geodesic" [5].



(a) 4-connectivity

(b) 8-connectivity

Figure 6: *Boundaries of the successive geodesic dilations of a set (in black) within a mask.*

When performing successive elementary geodesic dilations of a set Y inside a mask X, the connected components of X whose intersection with Y is non empty are progressively flooded. The following proposition can thus be stated:

Proposition 2.3 *The reconstruction of X from $Y \subseteq X$ is obtained by iterating elementary geodesic dilations of Y inside X until stability. In other words:*

$$\rho_X(Y) = \bigcup_{n \geq 1} \delta_X^{(n)}(Y).$$

This proposition is at the basis of one of the simplest algorithms for computing geodesic reconstructions in both the binary and the grayscale cases (see § 4.1).

2.3 Grayscale reconstruction

2.3.1 Definition using threshold superposition

It has been known for several years that—at least in the discrete case—any increasing transformation defined for binary images can be extended to grayscale images [18, 19, 31, 20]. By increasing, we mean a transformation ψ such that

$$\forall X, Y \subset \mathbb{Z}^2, \quad Y \subseteq X \Longrightarrow \psi(Y) \subseteq \psi(X). \tag{3}$$

In order to extend such a transformation ψ to grayscale images I taking their values in $\{0, 1, \ldots, N-1\}$, it suffices to consider the successive thresholds $T_k(I)$ of I, for $k = 0$ to $N-1$:

$$T_k(I) = \{p \in D_I \mid I(p) \geq k\}. \tag{4}$$

They are said to constitute the *threshold decomposition* of I [10, 11]. As illustrated by Fig. 7, these sets obviously satisfy the following inclusion relationship:

$$\forall k \in [1, N-1], \quad T_k(I) \subseteq T_{k-1}(I).$$

When applying the increasing operation ψ to each of these sets, their inclusion relationships are preserved. Thus, we can now extend ψ to grayscale images as follows:

$$\forall p \in D_I, \quad \psi(I)(p) = \max\{k \in [0, N-1] \mid p \in \psi(T_k(I))\}. \tag{5}$$

In the present case, binary geodesic reconstruction is an increasing transformation in that it satisfies:

$$Y_1 \subseteq Y_2, X_1 \subseteq X_2, \quad Y_1 \subseteq X_1, Y_2 \subseteq X_2 \quad \Longrightarrow \quad \rho_{X_1}(Y_1) \subseteq \rho_{X_2}(Y_2). \tag{6}$$

Therefore, following the threshold superposition principle of equation (5), we define grayscale reconstruction as follows [27]:

Definition 2.4 (grayscale reconstruction, first definition) *Let J and I be two grayscale images defined on the same domain, taking their values in the discrete set $\{0, 1, \ldots, N-1\}$ and such that $J \leq I$ (i.e., for each pixel $p \in D_I, J(p) \leq I(p)$). The grayscale reconstruction $\rho_I(J)$ of I from J is given by:*

$$\forall p \in D_I, \quad \rho_I(J)(p) = \max\{k \in [0, N-1] \mid p \in \rho_{T_k(I)}(T_k(J))\}.$$

Fig. 8 illustrates this transformation. Just like binary reconstruction extracts those connected components of the mask which are marked, grayscale reconstruction extracts the *peaks* of the mask which are marked by the marker-image. This characteristics is taken into account in the examples of application presented in § 3.

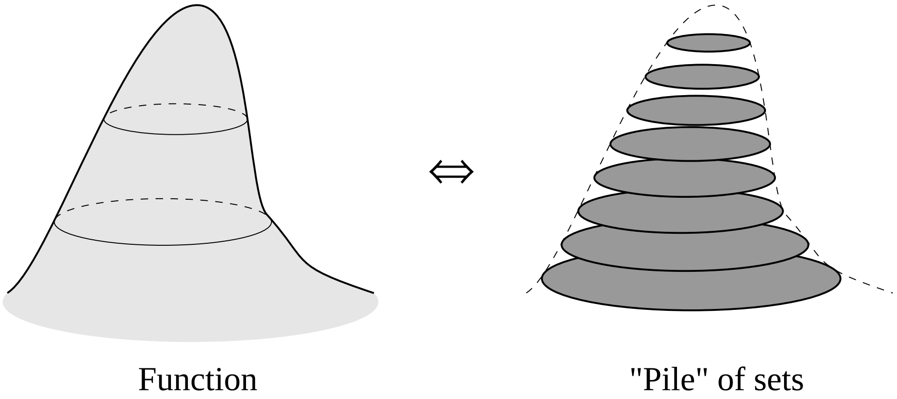


Figure 7: *Threshold decomposition of a graylevel image.*

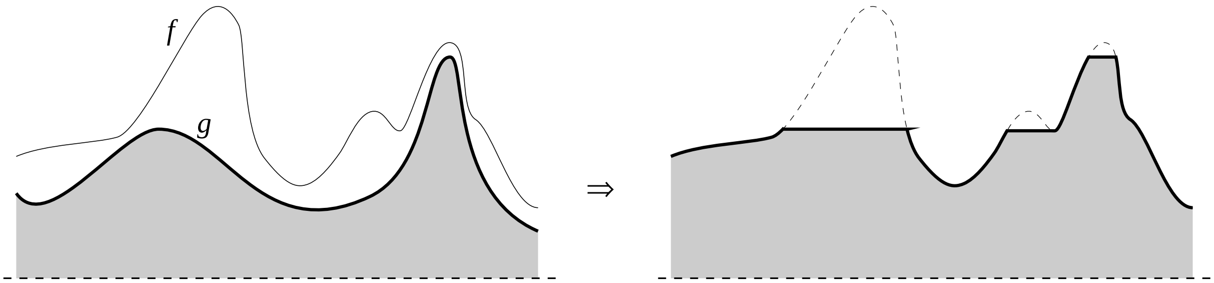


Figure 8: *Grayscale reconstruction of mask f from marker g.*

2.3.2 Alternative definition of grayscale reconstruction

The former definition does not provide any interesting computational method to determine grayscale reconstruction in digital images. Indeed, even if a fully optimized binary reconstruction algorithm is used, one has to apply it 256 times to determine grayscale reconstruction for images on 8 bits! Therefore, it is most useful to introduce this transformation using the geodesic dilations presented in section 2.2.2.

Following the threshold decomposition principle, one can easily define the elementary geodesic dilation $\delta_I^{(1)}(J)$ of grayscale image $J \leq I$ "under" I:

$$\delta_I^{(1)}(J) = (J \oplus B) \wedge I, \tag{7}$$

In this equation, $\wedge$ stands for the pointwise minimum and $J \oplus B$ is the dilation of J by flat structuring element B [18, 19]. These two notions are the direct extension to the grayscale case of respectively intersection and binary dilation by B. The grayscale geodesic dilation of size $n \geq 0$ is then given by:

$$\delta_I^{(n)}(J) = \underbrace{\delta_I^{(1)} \circ \delta_I^{(1)} \circ \ldots \circ \delta_I^{(1)}}_{n \text{ times}}(J). \tag{8}$$

This leads to a second definition of grayscale reconstruction:

Definition 2.5 (grayscale reconstruction, second definition) *The grayscale reconstruction $\rho_I(J)$ of I from J is obtained by iterating grayscale geodesic dilations of J "under" I until stability is reached, i.e.:*

$$\rho_I(J) = \bigvee_{n \geq 1} \delta_I^{(n)}(J).$$

It is straightforward to verify that both this definition and definition 2.4 correspond to the same transformation.

Similarly, the elementary geodesic erosion $\varepsilon_I^{(1)}(J)$ of grayscale image $J \geq I$ "above" I is given by

$$\varepsilon_I^{(1)}(J) = (J \ominus B) \vee I, \tag{9}$$

where $\vee$ stands for the pointwise maximum and $J \ominus B$ is the erosion of J by flat structuring element B [18, 19]. The grayscale geodesic erosion of size $n \geq 0$ is then given by:

$$\varepsilon_I^{(n)}(J) = \underbrace{\varepsilon_I^{(1)} \circ \varepsilon_I^{(1)} \circ \ldots \circ \varepsilon_I^{(1)}}_{n \text{ times}}(J). \tag{10}$$

We are now able to define the *dual* grayscale reconstruction in terms of geodesic erosions:

Definition 2.6 (dual reconstruction) *Let I and J be two grayscale images defined on the same domain D_I and such that $I \leq J$. The dual grayscale reconstruction $\rho_I^*(J)$ of mask I from marker J is obtained by iterating grayscale geodesic erosions of J "above" I until stability is reached:*

$$\rho_I^*(J) = \bigwedge_{n \geq 1} \varepsilon_I^{(n)}(J).$$

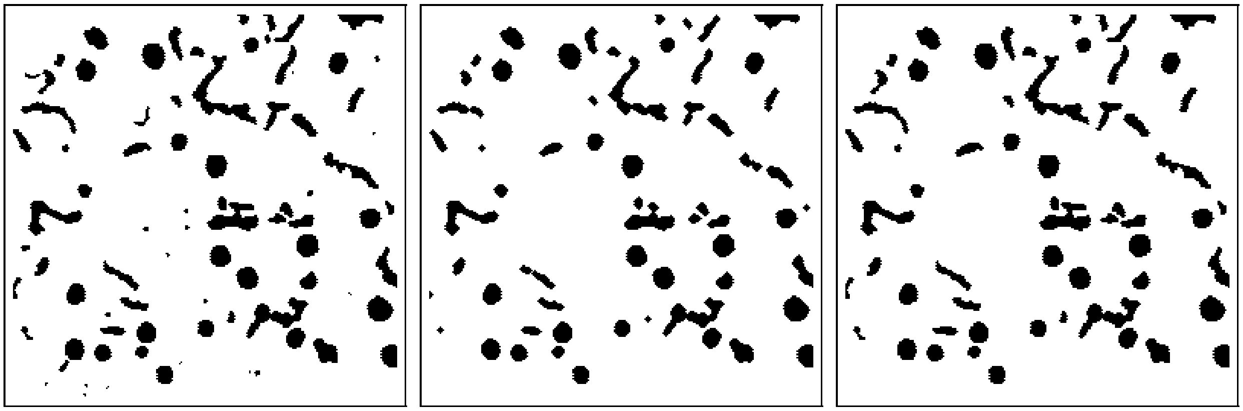
3 Applications

In this section, we underscore the tremendous interest of binary and grayscale reconstruction in image analysis. Surprisingly enough, reconstruction is not a very well-known transformation, particularly in the grayscale case. The examples presented below provide a non-exhaustive catalog of applications of reconstruction in image analysis.

3.1 Filtering by opening-reconstruction

In the binary case, the opening by a disc (or other types of structuring elements) is commonly used to filter out the image parts which cannot hold the disc. Recall that the opening of a binary image I by a disc is the union of all the possible positions of the disc when it is totally included in the image I [18, 28].

In some cases, one wishes to filter out all the connected components which cannot contain the disc *while preserving the others entirely.* The way to do so is to reconstruct the original image I from its opening by the disc. The resulting transformation is often called *opening by reconstruction* and belongs to the category of the *algebraic openings* [20]. An example is shown on Fig. 9: after opening by a disc of radius 2, several connected components of the original image have been removed, but the shape of some of the remaining ones has been dramatically modified. After reconstruction, the original shape of the not totally removed particles is restaured. Opening by reconstruction extends to the grayscale case in a straightforward manner. It turns out to be a particularly interesting operation, especially when large structuring elements are used in the opening step.



(a) original image *(b) opening by a disc of radius 2* *(c) reconstructed image*

Figure 9: Use of opening by reconstruction for filtering.

3.2 Use of top-hat by reconstruction for segmentation

Let us now illustrate on an example one of the possible uses of grayscale reconstruction for picture segmentation: Fig. 10.a represents an angiography of eye blood vessels in which microaneurisms have to be detected. They are small compact light spots which are disconnected from the network of the (light) blood vessels and mainly located in the dark central area of the image. Obviously, it is impossible to detect these micro-aneurisms via simple thresholdings. Similarly, a top-hat transformation [12] consisting in subtracting from the original image its morphological opening with respect to a small disc would extract all the "white" features, i.e. aneurisms *and* blood vessels, which is not desirable.

To correctly segment these micro-aneurisms, one has to account for the fact that they are compact whereas the blood vessels are elongated. A series of morphological openings of Fig. 10.a with respect to segments of different orientations is thus performed. These segments are chosen to be longer than any possible aneurism, so that all the aneurisms are removed by any such opening. On the other hand, there will be at least one orientation for which the vessels are not completely removed by opening. After taking the supremum of these different openings, one gets Fig. 10.b, which is still an algebraic opening of Fig. 10.a [19]. It is used as marker to reconstruct the blood vessels entirely. Fig. 10.c is the result of the grayscale reconstruction of Fig. 10.a from Fig. 10.b. Since the aneurisms are disconnected from the blood vessels, they have not been reconstructed! Thus, by algebraic difference between Fig. 10.a and Fig. 10.c, followed by a relatively simple thresholding, the microaneurisms shown in Fig. 10.d are extracted. The succession of operations used for the present segmentation task is an extension of the family of the top-hat transformations, often referred to as *top-hat by reconstruction.*

3.3 Regional maxima and dome extraction

Reconstruction turns out to provide a very efficient method to extract *regional maxima* and *minima* from grayscale images. Furthermore, the technique extends to the determination of "maximal structures", which we call *h-domes* and *h-basins.* Let us first briefly review the notion of regional maximum:

Definition 3.1 (regional maximum) *A regional maximum M of a grayscale image I is a connected components of pixels with a given value h (plateau at altitude h), such that every pixel in the neighborhood of M has a strictly lower value.*

Regional maxima should not be mistaken with *local* maxima. Recall that a pixel p of I is a local maximum for grid G if and only if its value $I(p)$ is greater or equal to that of any of its neighbors. All the pixels belonging to a regional maximum are local maxima, but the converse is not true: for example, a pixel p belonging to the inside of a plateau is a local maximum, but the plateau may have neighboring pixels of higher altitude and thus not be a regional maximum.

An alternative definition can also be proposed for the notion of regional maximum [23]:

Definition 3.2 *A regional maximum at altitude h of grayscale image I is a connected component C of $T_h(I)$ such that $C \cap T_{h+1}(I) = \emptyset$.*

(Recall from eq. (4) that $T_h(I)$ is threshold of I at level h.)

Determining the regional maxima of a grayscale image is relatively easy and several algorithms have been proposed in literature, some of which are reviewed in [23]. One of the most efficient methods makes use of grayscale reconstruction and is based on the following proposition:

Proposition 3.3 *The (binary) image $M(I)$ of the regional maxima of I is given by:*

$$M(I) = I - \rho_I(I-1).$$

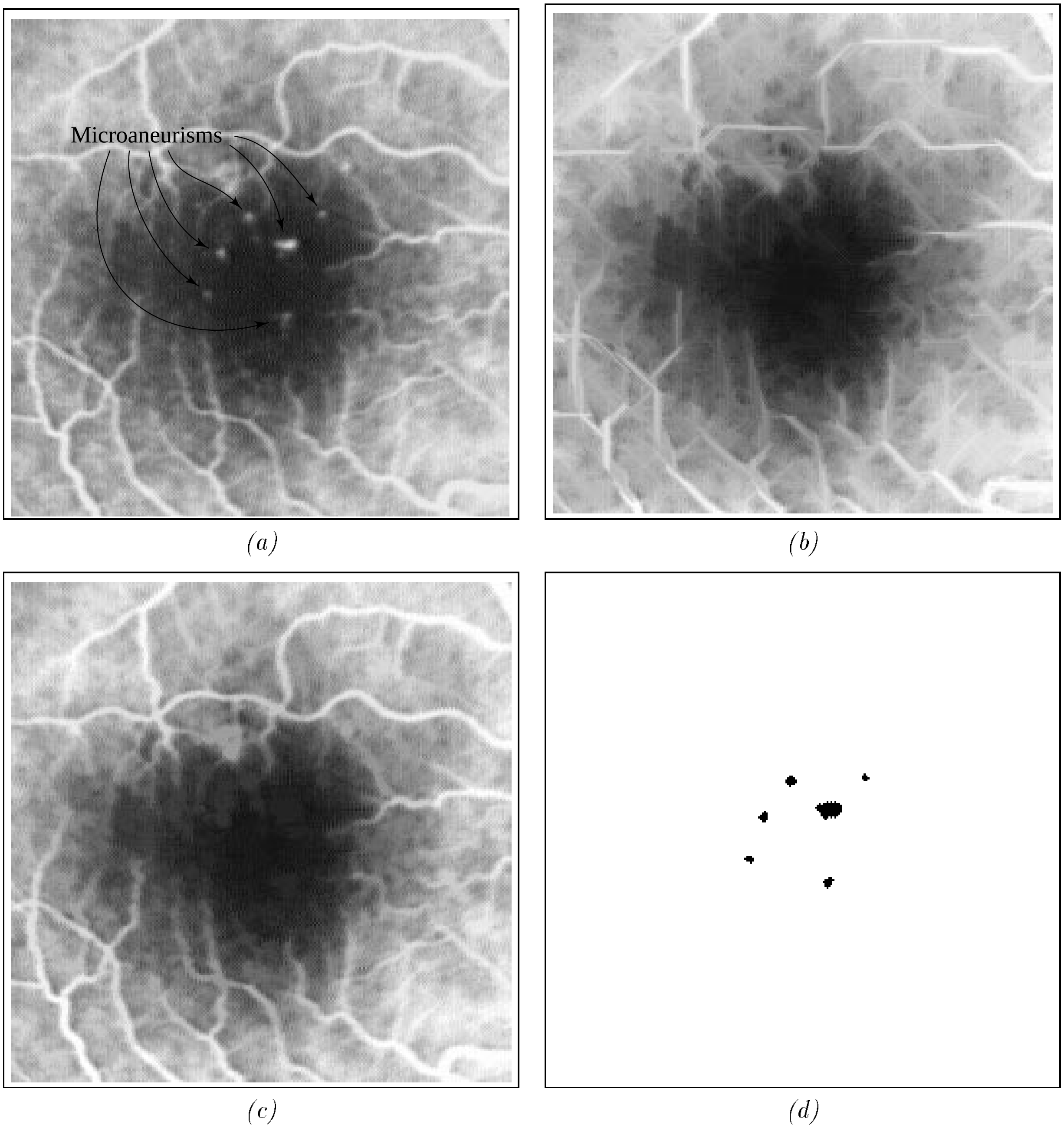
PROOF : According to definition 3.2, a connected component C of $T_h(I)$ is a maximum at level h if and only if $C \cap T_{h+1}(I) = C \cap T_h(I-1) = \emptyset$. In other words, the set M_h of the pixels belonging to a maximum of I at altitude h is given by:

$$M_h = T_h(I) \setminus \rho_{T_h(I)}(T_h(I-1)). \tag{11}$$

Now, for any h, h', $h \neq h'$, $M_h \cap M_{h'} = \emptyset$. This means that by replacing the set difference ($\setminus$) by an algebraic difference and using the threshold superposition principle, formula (11) can be extended to the grayscale case. □

This proposition is illustrated by Fig. 11.

Now, instead of subtracting value 1 in prop. 3.3, an arbitrary gray-level constant h can be subtracted from I. This provides a useful technique for extracting "domes" of a given height, that we call h-domes. The following definition can be proposed:



(a)

(b)

(c)

(d)

Figure 10: *Use of grayscale reconstruction for image segmentation: (a) original image of blood vessels, (b) supremum of openings by segments, (c) reconstructed image, (d) microaneurisms obtained after substraction of (c) from (a) and thresholding step.*

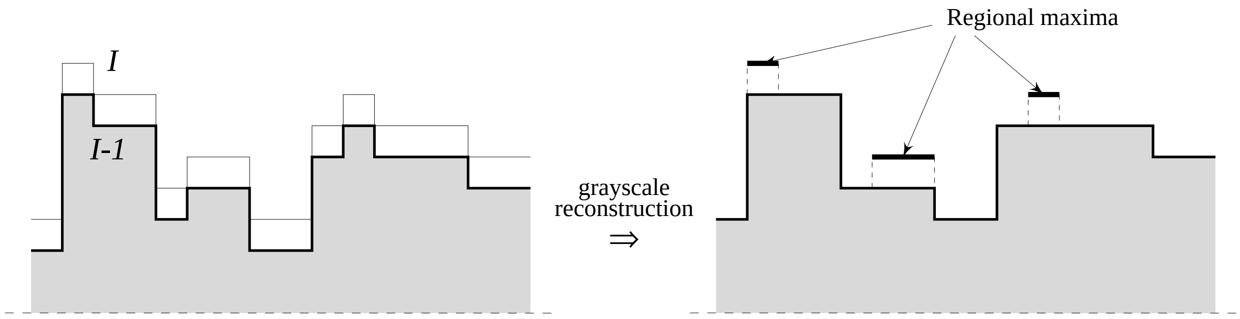


Figure 11: *Extracting the regional maxima of* I *by reconstruction of* I *from* $I-1$.

Definition 3.4 *The h-dome image* $D_h(I)$ *of the h-domes of a grayscale image* I *is given by:*

$$D_h(I) = I - \rho_I(I-h).$$

Geometrically speaking, an h-dome can be interpreted the same way maxima are: an h-dome D of image I is a connected component of pixels such that:

– every pixel p neighbor of D satisfies: $I(p) < \min\{I(q) \mid q \in D\}$,

– $\max\{I(q) \mid q \in D\} - \min\{I(q) \mid q \in D\} < h$.

In addition, the value of pixel p of h-dome D in image $D_h(I)$ is equal to I(p)-min$\{I(q) \mid q \in D\}$.

The h-dome transformation is illustrated on Fig. 12. Unlike classical top-hats, the h-dome transformation extracts light structures without involving any size or shape criterion. The only parameter (h) is related to the height of these structures. This characteristic is of interest for complex segmentation problems.

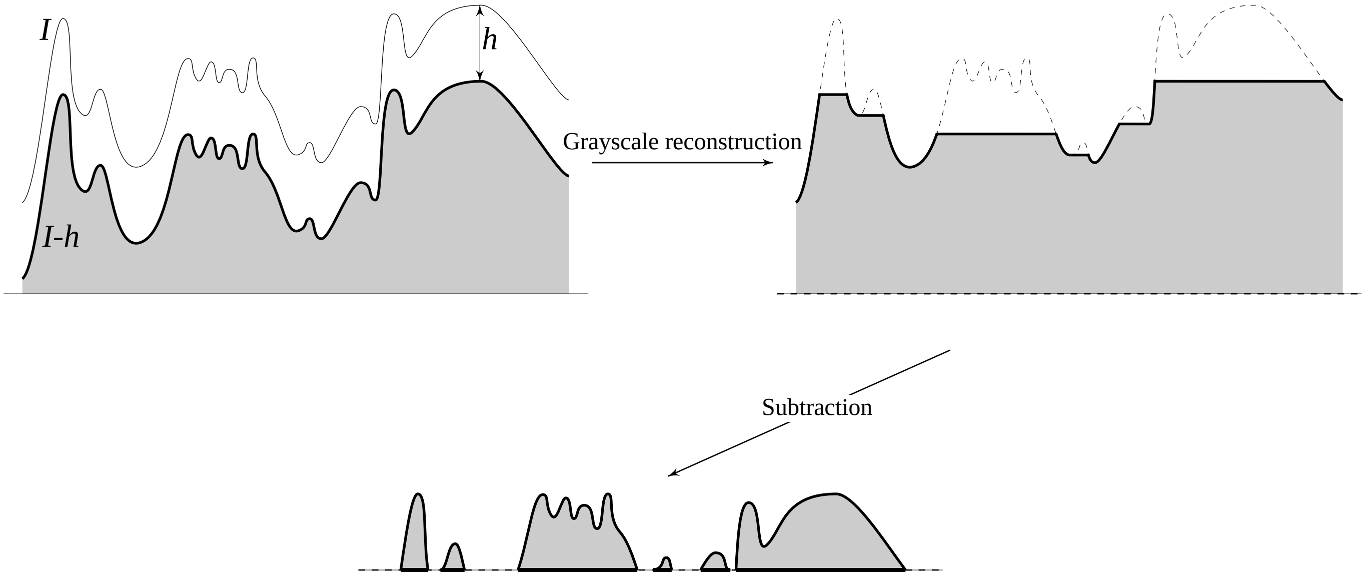


Figure 12: *Determination of the h-domes of grayscale image* I.

As an example, let us consider Fig. 13.a, which is an image of the corneal endothelial tissue of the eye, obtained using a wide-field specular microscope. The analysis of images of this kind is detailed in [29]. The first step of their segmentation consists in extracting a *marker* for each cell. As explained in [29], the large variations in contrast and in cell sizes across the image make it difficult to use top-hat transformations. On the other hand, the cells can be viewed as light *domes* separated by thin dark *valleys*. The previously described h-dome operator can thus be automatically applied: after subtraction of a constant h from Fig. 13.a and reconstruction, one gets Fig. 13.b. The choice of h turns out not to be a critical operation, since a range of values yield correct results. Then, subtracting Fig. 13.b from Fig. 13.a results in Fig. 13.c, the h-domes of Fig. 13.a. An easy thresholding of this image yields Fig. 13.d, which is an accurate set of cell markers.

Additional examples of application of the h-dome transformation can be found in [23, 26, 3], and more details can be found in [4]. Note that the results of this section can easily be "inverted" to extract minima and h-basins in grayscale images.

3.4 Grayscale reconstruction and binary segmentation

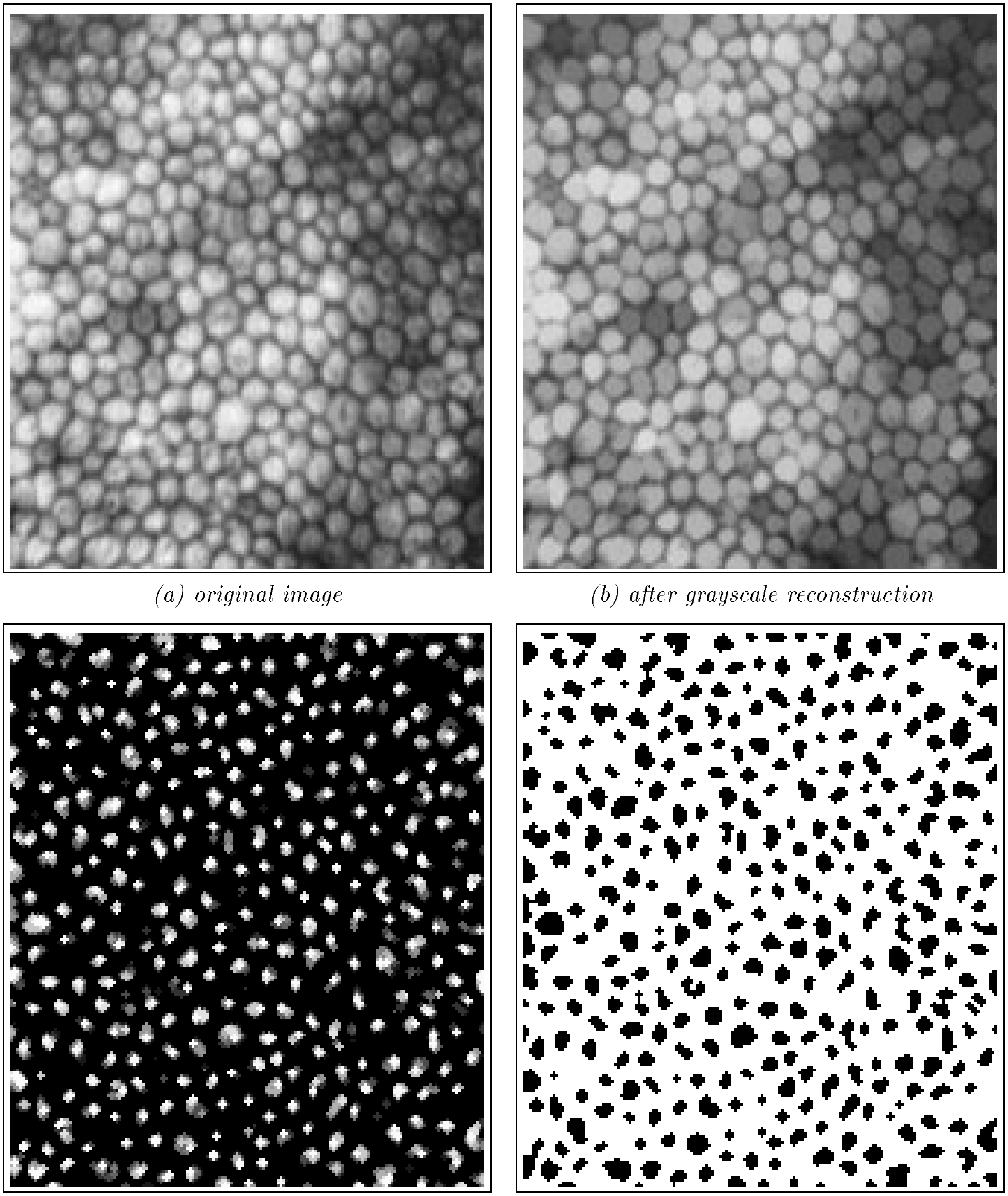
The examples reviewed above illustrate the use of grayscale reconstruction in several image analysis tasks. However, it is probably for binary and grayscale segmentation that this operation is most useful. This section briefly deals with binary segmentation, which morphologists often refer to as the task consisting in separating overlapping particles in a binary image [28, 23]. For example, Fig. 14.a is a binary image of coffee beans, and one may want to disconnect them properly in order to perform individual measurements on them. To achieve this goal, following the approach described in [28], the idea is to first mark the coffee beans. By marker of an object, we mean a connected component of pixels located inside the object to be extracted. Once correct markers have been obtained, the *watershed transformation* [1] allows us to achieve the desired segmentation automatically.

It is therefore crucial to design robust marking procedures. In the case where the objects to be separated are roughly convex, the *ultimate erosion* usually provides a satisfactory marking [28]. This transformation is obtained as the regional maxima of the distance function of the original binary image [2, 15]—Recall that the distance function $dist(I)$ of binary image I assigns with every pixel p its distance to the background, i.e.. to the closest pixel with value 0. However, in many cases, due to the fact that we work in a discrete space, the resulting markers are poor (see Fig. 14.c): there is not a unique marker per object to be extrated! To get rid of this drawback, one cannot simply perform a small dilation of the ultimate erosion image. Indeed, the components of the ultimate erosion which have to be connected into a single marker can be arbitrarily far away from each other. The correct method consists in reconstructing the distance function $dist(I)$ from $dist(I) - 1$. The maxima of the resulting function provides a correct marking of our objects, thereby yielding a correct segmentation, as illustrated by Figs. 14.d–e. For more details on the use of watersheds and grayscale reconstruction for binary segmentation, see [28, 30].

3.5 Watershed segmentation of grayscale images

Similarly, the markers/watersheds methodology applies to grayscale segmentation. This task consists in extracting objects from a gray-level image as precisely as possible. Let us consider for example the classic image shown in Fig. 15.a. It represents a 2-D electrophoresis gel, whose spots need to be extracted as precisely as possible. Its watershed segmentation was originally proposed by Beucher in [1].

The watershed lines are the highest crest-lines separating the regional minima [30]. It seems therefore natural to compute the watersheds of the morphological gradient of Fig. 15.a, which is shown in Fig. 15.b. This gradient, obtained by algebraic difference between a unit dilation and a



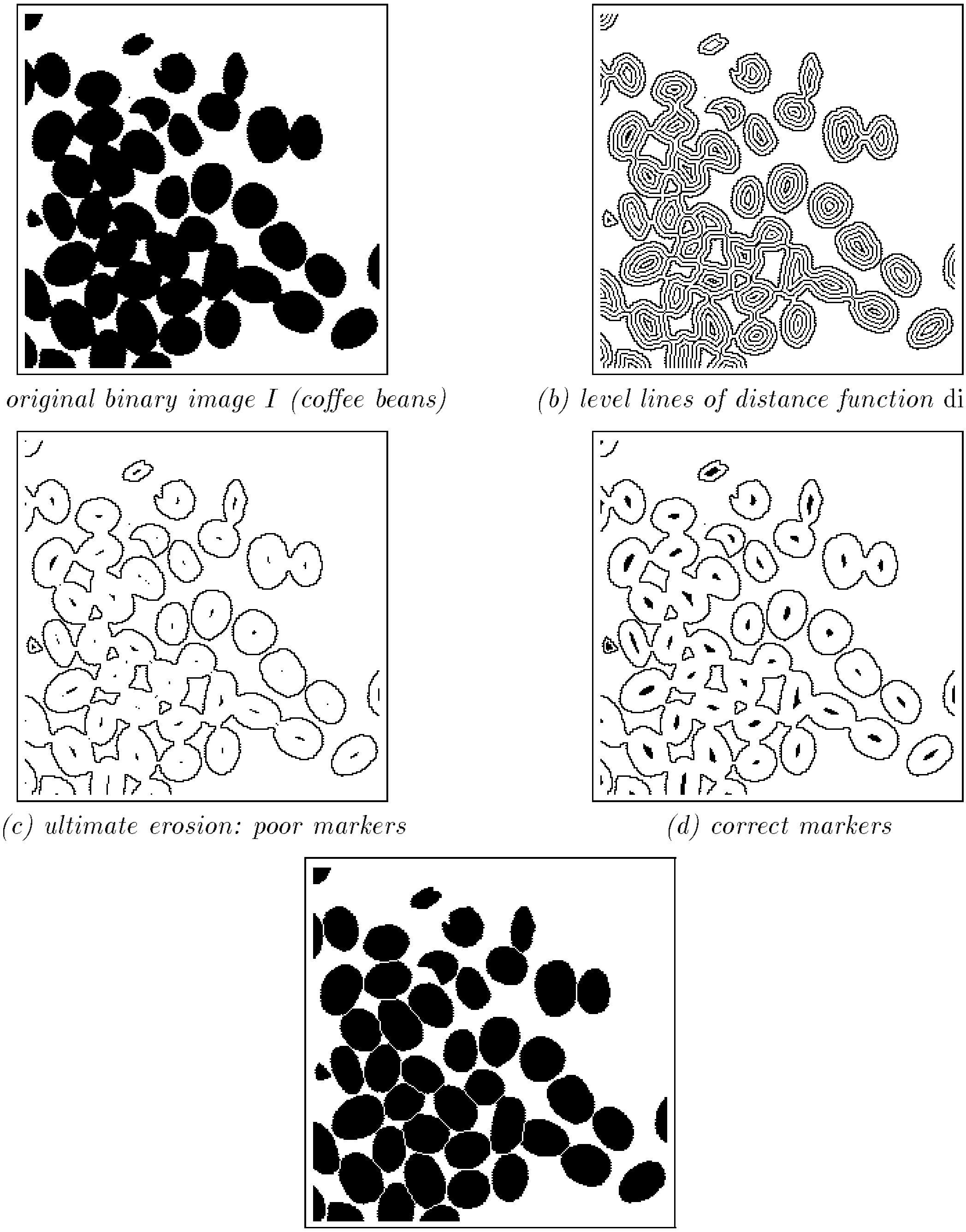
(a) original image

(b) after grayscale reconstruction

(c) h-domes

(d) cell markers

Figure 13: *Extraction of cell markers in images of corneal endothelial tissue.*



(a) original binary image I (coffee beans)

(b) level lines of distance function dist(I)

(c) ultimate erosion: poor markers

(d) correct markers

(e) segmented image

Figure 14: *Use of grayscale reconstruction in binary segmentation. The correct markers are obtained as the regional maxima of the reconstruction of the distance function* dist(I) *from* dist$(I) - 1$.

unit erosion of Fig. 15.a, is shown in Fig. 15.b. The spot contours are located on crest-lines of this gradient, but far too many of these crest lines are due to noise in the original data. Therefore, the watersheds of Fig. 15.b yield the over-segmented result of Fig. 15.c.

As explained in numerous recent publications [28, 20, 30, 13], the correct way to use watersheds for grayscale image segmentation consists in first detecting *markers* of the objects to be extracted. The design of robust marker detection techniques involves the use of knowledge specific to the series of images under study. Not only object markers, but also background markers need to be extracted. In the present case, describing the marking technique used for Fig. 15.a would go beyond the scope of this paper. The extracted marker image is shown in Fig. 15.d.

After marker extraction, the rest of the segmentation can proceed automatically as follows: grayscale reconstruction is used to modify the gradient image G into an image G' such that:

– its only minima are located on the extracted markers,

– its highest crest-lines separating markers are preserved.

More specifically, denote by G the gradient image and by M the binary marker image. Let m be the maximal value of the pixels of G. The image G' is defined as the dual reconstruction of $\min(G+1, (m+1)M)$ from $(m+1)M$ (see definition. 2.6):

$$G' = \rho^*_{\min(G+1,(m+1)M)}((m+1)M). \qquad (12)$$

In this process, pixels located on markers are given value 0 in G' and non-marked catchment basins get filled up. More details on this process are given in [28, 23, 20]. The resulting modified gradient is shown in Fig. 15.e. Its watersheds now provide the desired segmentation, as illustrated by Fig. 15.f. This segmentation methodology is commonly used in morphology and has been successfully applied to various types of images: NMR images [23], digital elevation models [21], corneal endothelial images [29], succession of images used for motion estimation [3], and many others.

4 Computing reconstruction in digital images

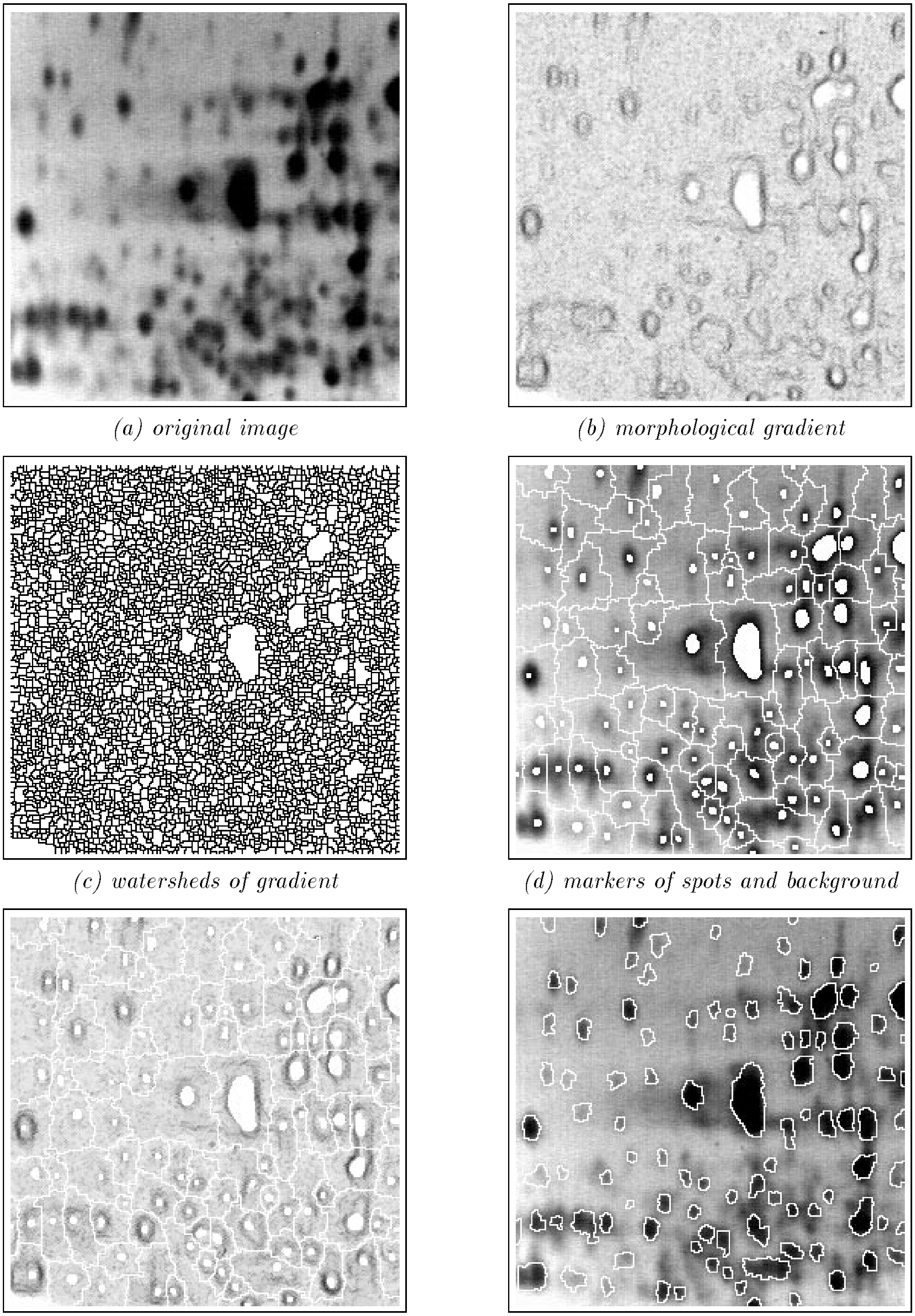
In this section, we are concerned with both the binary and the grayscale case, but the emphasis is put on grayscale reconstruction. Indeed, in the binary case, a straightforward efficient implementation of morphological reconstruction can be proposed as follows:

1. Label the connected components of the mask image, i.e., each of these components is assigned a unique number. Note that this step can itself be implemented very efficiently by using algorithms based on chain and loops [16] or queues of pixels [23, 26].

2. Determine the labels of the connected components which contain at least a pixel of the marker image

3. Remove all the connected components whose label is not one of the previous ones.

As mentioned earlier, such an algorithm could be extended to the grayscale case by working on the different thresholds of the images. However, it would be extremely inefficient, making grayscale reconstruction a too cumbersome transformation to be used in practice. This is the reason why we are now interested in implementing this transformation as efficiently as possible. Note however that the algorithms described below also work in the binary case if one considers binary images as grayscale images taking only values 0 and 1. In the algorithm descriptions of this section, '←' refers to the assignment symbol.



(a) original image

(b) morphological gradient

(c) watersheds of gradient

(d) markers of spots and background

(e) modified gradient

(f) final segmentation

Figure 15: Watershed segmentation of 2-D electrophoresis gels.

4.1 Standard technique

Proposition 2.3 and definition 2.5 directly yield a computational technique for determining grayscale reconstruction in digital images. The corresponding algorithm is part of a set of classical methods referred to as *parallel* ones [23, 25, 26]. It basically works by iterating elementary dilation followed by pointwise minimum until stability as follows:

Algorithm: parallel reconstruction

```
• $I$: mask image (binary or grayscale)
• $J$: marker image, defined on domain $D_I$, $J \leq I$.
  Reconstruction is determined directly in $J$
• Allocate work image $K$ defined on $D_I$
• Repeat until stability (i.e. no more pixel value modifications):
    – Dilation step: for every pixel $p \in D_I$
        – $K(p) \leftarrow \max\{J(q), q \in N_G(p) \cup \{p\}\}$
    – Pointwise minimum: For every pixel $p \in D_I$
        – $J(p) \leftarrow \min(K(p), I(p))$
```

In each of the above steps, the image pixels can be scanned in an arbitrary order, so that the implementation of this algorithm on a parallel machine is extremely easy and efficient. However, it requires the iteration of numerous complete image scannings, sometimes several hundreds! It is therefore not suited to conventional computers, where its execution time is often of several minutes.

4.2 Sequential reconstruction algorithm

In an attempt to reduce the number of scannings required for the computation of an image transform, sequential or recursive algorithms have been proposed [14]. They rely on the following two principles:

> - *the image pixels are scanned in a predefined order, generally raster or anti-raster,*
> - *the new value of the current pixel, determined from the values of the pixels in its neighborhood, is written directly* in the same image, *so that it is taken into account when determining the new values of the as yet unconsidered pixels.*

Here, unlike for parallel algorithms, the scanning order is essential! This type of algorithm was first introduced for the computation of distance functions [15] and then extended to a number of morphological transformations [8, 23]. Among others, binary and grayscale reconstruction can be obtained sequentially by using the following algorithm, where information is first propagated downwards in a raster scanning and then upwards in an anti-raster scanning.

Algorithm: sequential reconstruction

```
• $I$: mask image (binary or grayscale)
• $J$: marker image, defined on domain $D_I$, $J \leq I$.
  Reconstruction is determined directly in $J$
```

```
• Repeat until stability:
    – Scan $D_I$ in raster order:
        – Let $p$ be the current pixel;
        – $J(p) \leftarrow \Big(\max\{J(q), q \in N_G^+(p) \cup \{p\}\}\Big) \wedge I(p)$
    – Scan $D_I$ in anti-raster order:
        – Let $p$ be the current pixel;
        – $J(p) \leftarrow \Big(\max\{J(q), q \in N_G^-(p) \cup \{p\}\}\Big) \wedge I(p)$
```

This algorithms usually only requires a few image scannings (a dozen typically) until stability is reached, and is therefore much more efficient than the parallel algorithm presented in the previous section. However, like several other sequential algorithms [26], it does not deal well with "rolled-up structures" (connected components in the binary case and crest-lines in the grayscale case): as illustrated by Fig. 16 in the binary case, the sequential reconstruction of a rolled-up component may require several complete image scannings in which the value of only very few pixels is actually modified.

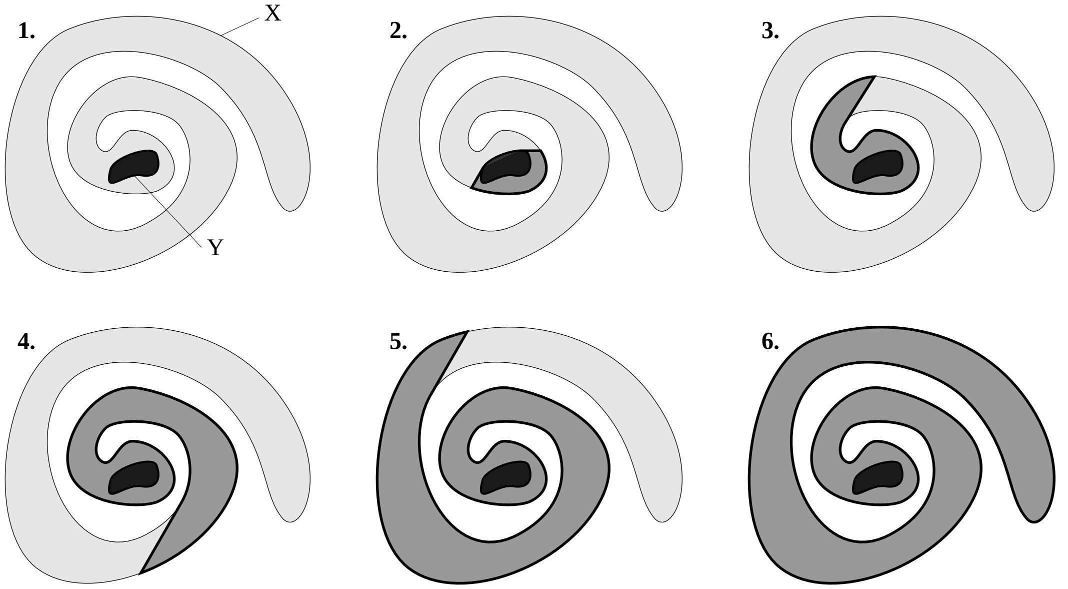


Figure 16: *The sequential computation of a binary reconstruction in a rolled up mask may involve several complete image scannings: here, the hatched zones represent the pixels which have been modified after each step.*

4.3 Regional maxima and reconstruction

As detailed in [25, 26], a further step in the design of efficient morphological processing consists in trying to *consider only the pixels whose value may be modified.* A first scanning is used to detect the pixels which are the process initiators and are typically located on the boundaries of the objects or regions of interest. Then, starting from these pixels, information is propagated *only* in the relevant image parts. Two categories of algorithms relying on this principle have been proposed in literature: the first ones are based on the encoding of the objects boundaries as loops and the propagation of these structures in the image or in some given mask [16], whereas the algorithms of the second category regard the images under study as graphs and realize breadth-first scannings of these graphs starting from strategically located pixels [23, 26, 22]. These two class of methods can be used to efficiently implement such complex morphological operations as propagation functions [9, 16], watersheds [30], skeletons [24] and many others [17].

Here, we shall be concerned with the second class of algorithms. The breadth-first scannings involved are implemented by using a *queue of pixels,* i.e., a *First-In-First-Out (FIFO)* data structure: the pixels which are first put into the queue are those which can first be extracted. In other words, each new pixel included in the queue is put on one side whereas a pixel being removed is taken from the other side [25, 26]. In practice a queue is simply a large enough array of pointers to pixels, on which three operations may be performed:

> - *fifo_add(p): puts the (pointer to) pixel p into the queue.*
> - *fifo_first(): returns the (pointer to) pixel which is at the beginning of the queue, and removes it.*
> - *fifo_empty(): returns* true *if the queue is empty and* false *otherwise.*

In the binary case, it is extremely easy to implement reconstruction using a FIFO-algorithm: it suffices to initialize the queue by loading it with the boundary pixels of the marker-image. Then, the value of these pixels is propagated in the relevant connected components of the mask image. The corresponding algorithm works as follows:

Algorithm: binary reconstruction using a queue of pixels

```
• $I$:  binary mask image
• $J$:  binary marker image, defined on domain $D_I$, $J \subseteq I$.
  Reconstruction is determined directly in $J$
• Initialization of the queue with contour pixels of marker image:
  For every pixel $p \in D_I$:
    – If $J(p) = 1$ and $\exists q \in N_G(p), J(q) = 0$ and $I(p) = 1$:
        – $fifo\_add(p)$
• Propagation:  While $fifo\_empty() = false$
    – $p \leftarrow fifo\_first()$
    – For every $q \in N_G(p)$ (neighbor of $p$):
        – If $J(q) = 0$ and $I(q) = 1$
            – $J(q) \leftarrow 1$
            – $fifo\_add(q)$
```

This algorithm is extremely efficient, since after the initialization of the queue, only the relevant pixels are considered. Besides, the same technique can be implemented using loop-based algorithms. The typical execution time of this algorithm is of 1/4 second on a *Sun IPC Workstation,* for images of size 256×256 pixels.

The extension of this algorithm to grayscale is not immediate: we need to consider the regional maxima of the marker image (see definition 3.2). Denoting by $R(I)$ the following image:

$$\forall p \in D_I, \quad R(I)(p) = \begin{cases} I(p) & \text{if } p \text{ belongs to a maximum,} \\ 0 & \text{otherwise.} \end{cases} \tag{13}$$

we can state:

Proposition 4.1 *Let I and J be two grayscale images such that $J \leq I$. Then:*

$$\rho_I(J) = \rho_I(R(J)).$$

PROOF : According to definition 2.5 of grayscale reconstruction, it suffices to prove that for every threshold level $h \in \{0, 1, \ldots, N-1\}$:

$$\rho_{T_h(I)}(T_h(J)) = \rho_{T_h(I)}(T_h(R(J))).$$

$R(J) \leq J$ implies that $T_h(R(J)) \subseteq T_h(J)$. Thus, binary reconstruction being an increasing transformation, we have $\rho_{T_h(I)}(T_h(J)) \supseteq \rho_{T_h(I)}(T_h(R(J)))$.

Similarly, let C be a connected component of $T_h(J)$. Let $h_{\max} = \max\{J(q), q \in C\}$ and let $C_{\max}$ be the set of the pixels of C with value $h_{\max}$. Let C' be a connected component of $C_{\max}$. C' is obviously a regional maximum of J at altitude $h_{\max}$. Thus, by definition, $\forall p \in C', R(J)(p) = h_{\max}$. Since $h \leq h_{\max}$, this implies: $\forall p \in C', T_h(R(J))(p) = 1$. Therefore, $C \cap T_h(R(J)) \neq \emptyset$. This being true for every connected component C of $T_h(J)$, definition 2.1 implies:

$$\rho_{T_h(I)}(T_h(J)) \subseteq \rho_{T_h(I)}(T_h(R(J))),$$

which completes the proof. □

In practice, the above proposition means that only the regional maxima of the marker image J need to be taken into account for the computation of $\rho_I(J)$. The algorithm introduced below takes advantage of this fact by propagating the values of the regional maxima of J using a FIFO structure:

Algorithm: grayscale reconstruction using a queue of pixels

```
• $I$:  grayscale mask image
• $J$:  grayscale marker image, defined on domain $D_I$, $J \leq I$.
  Reconstruction is determined directly in $J$
• Compute regional maxima of $J$:  $J \leftarrow R(J)$;
• Initialization of the queue with boundaries of maxima:
  For every pixel $p \in D_I$:
    – If $J(p) \neq 0$ and $\exists q \in N_G(p), J(q) = 0$
        – $fifo\_add(p)$
• Propagation:  While $fifo\_empty() = false$
    – $p \leftarrow fifo\_first()$
    – For every pixel $q \in N_G(p)$
        – Look if $q$ is lower than $p$ and if it is necessary to propagate it:
          If $J(q) < J(p)$ and $I(q) \neq J(q)$
            – $J(q) \leftarrow \min\{J(p), I(q)\}$
            – $fifo\_add(q)$;
```

The above algorithm constitutes a very clear improvement with respect to the sequential algorithm presented in the previous section. Its typical execution time on a *Sun IPC Workstation* is of 2.5 seconds for a 256 × 256 image whereas the sequential one may require as much as 10 seconds in some cases.

4.4 A fast hybrid grayscale reconstruction algorithm

Although much faster than the techniques previously proposed in literature, the above algorithm is slown down by the initial determination of the regional maxima of the marker image. Furthermore, contrary to its binary counterpart, some image regions may be scanned more than once during the breadth-first scanning step. This is true in particular when two regional maxima of J with different elevations are next to each other. On the other hand, the sequential grayscale reconstruction algorithm does not have this drawback, but as mentioned earlier, after the first two image scannings, it requires several additional scannings in which only a few pixels are modified.

These two algorithms have therefore complementary drawbacks and advantages, and this is the motivation for the hybrid algorithm introduced now: the idea is to start with the two first scannings of the sequential algorithm. During the second one (anti-raster), every pixel p such that its current value could still be propagated during the next raster scanning, i.e. such that

$$\exists q \in N_G^-(p), J(q) < J(p) \text{ and } J(q) < I(q),$$

is put into the queue. The last step of the algorithm is then exactly the same as the breadth-first propagation step of the FIFO algorithm proposed in the previous section. However, the number of pixels to be considered during this step is considerably smaller than previously. This algorithm is described below in pseudo-code:

Algorithm: fast hybrid grayscale reconstruction

```
• $I$: mask image (binary or grayscale)
• $J$: marker image, defined on domain $D_I$, $J \leq I$.
  Reconstruction is determined directly in $J$
• Scan $D_I$ in raster order:
    – Let $p$ be the current pixel;
    – $J(p) \leftarrow \Big(\max\{J(q), q \in N_G^+(p) \cup \{p\}\}\Big) \wedge I(p)$
• Scan $D_I$ in anti-raster order:
    – Let $p$ be the current pixel;
    – $J(p) \leftarrow \Big(\max\{J(q), q \in N_G^-(p) \cup \{p\}\}\Big) \wedge I(p)$
    – If there exist $q \in N_G^-(p)$ such that $J(q) < J(p)$ and $J(q) < I(q)$
        – $\mathit{fifo\_add}(p)$
• Propagation step: While $\mathit{fifo\_empty}() = \mathit{false}$
    – $p \leftarrow \mathit{fifo\_first}()$
    – For every pixel $q \in N_G(p)$:
        – If $J(q) < J(p)$ and $I(q) \neq J(q)$
            – $J(q) \leftarrow \min\{J(p), I(q)\}$
            – $\mathit{fifo\_add}(q)$
```

This algorithm seems to offer the best compromise for computing grayscale reconstructions. It takes advantage of the strong points of both the algorithms described in the last two sections without retaining their drawbacks. A mean case complexity analysis would be extremely difficult to perform on this kind of algorithm, since it would involve the design of a model for the different kind of input images that may be used. It would go beyond the scope of the present paper. However, from an experimental point of view, the execution time of this algorithm is of less than a second on a *Sun IPC Workstation,* for almost any input image of size 256 × 256. Note that the algorithm works equally well for binary images and that its extensions to any kind of grid and to multi-dimensional images are straightforward. These characteristics make it the fastest known algorithm on conventional computers.

5 Summary

In this paper, grayscale reconstruction has been formally defined for discrete images. Its relations to binary reconstruction and to morphological geodesic transformations have been underscored. Some of the applications of binary and grayscale reconstruction in image analysis have then be reviewed. They illustrate the flexibility and usefulness of this transformation for such tasks as filtering and segmentation.

The known algorithms for computing binary and grayscale reconstruction are respectively of parallel and sequential type. They have been described, and the study of their drawbacks led us to propose a new method, based on the regional maxima of the marker image and making use of a queue of pixels (FIFO structure). Although more efficient than both the parallel and the sequential method, this new technique is not fully satisfactory. A last "hybrid"algorithm was therefore introduced, which takes advantage of the strong points of both the sequential and the FIFO algorithm. Its execution time is usually of less than a second on a *Sun Sparc Station,* for 256 × 256 images. This is an order of magnitude faster than any previously known technique. All the algorithms described extend to the three-dimensional case in a straightforward manner.

Acknowledgements

This work was supported in part by the *National Science Foundation* under Grant MIPS–86–58150, with matching funds from *DEC* and *Xerox.*

References

[1] S. Beucher and C. Lantuéjoul. Use of watersheds in contour detection. In *International Workshop on Image Processing, Real-Time Edge and Motion Detection/Estimation*, Rennes, France, 1979.

[2] G. Borgefors. Distance transformations in digital images. *Comp. Vis., Graphics and Image Processing*, 34:334–371, 1986.

[3] C.-S. Fuh, P. Maragos, and L. Vincent. Region-based approaches to visual motion correspondence. Technical report, HRL, Harvard University, Cambridge, 1991. Submitted to PAMI.

[4] M. Grimaud. A new measure of contrast: Dynamics. In *SPIE Vol. 1769, Image Algebra and Morphological Image Processing III*, pages 292–305, San Diego CA, July 1992.

[5] R. M. Haralick and L. G. Shapiro. *Computer and Robot Vision.* Addison-Wesley, 1991.

[6] C. Lantuéjoul and S. Beucher. On the use of the geodesic metric in image analysis. *Journal of Microscopy*, 121:39–49, Jan. 1981.

[7] C. Lantuéjoul and F. Maisonneuve. Geodesic methods in quantitative image analysis. *Pattern Recognition*, 17(2):177–187, 1984.

[8] B. Laÿ. Recursive algorithms in mathematical morphology. In *Acta Stereologica Vol. 6/III*, pages 691–696, Caen, France, Sept. 1987. 7th International Congress For Stereology.

[9] F. Maisonneuve and M. Schmitt. An efficient algorithm to compute the hexagonal and dodecagonal propagation function. In *5th European Congress For Stereology*, pages 515–520, Freiburg im Breisgau FRG, Sept. 1989. Acta Stereologica. Vol. 8/2.

[10] P. Maragos and R. Schafer. Morphological filters—part ii: their relations to median, order-statistics, and stack filters. *IEEE Trans. on Acoustics, Speech and Signal Processing*, 35(8):1170–1184, Aug. 1987.

[11] P. Maragos and R. Ziff. Threshold superposition in morphological image analysis. *IEEE Trans. Pattern Anal. Machine Intell.*, 12(5), May 1990.

[12] F. Meyer. Iterative image transformations for the automatic screening of cervical smears. *J. Histochem. and Cytochem.*, 27:128–135, 1979.

[13] F. Meyer and S. Beucher. Morphological segmentation. *Journal of Visual Communication and Image Representation*, 1:21–46, Sept. 1990.

[14] A. Rosenfeld and J. Pfaltz. Sequential operations in digital picture processing. *J. Assoc. Comp. Mach.*, 13(4):471–494, 1966.

[15] A. Rosenfeld and J. Pfaltz. Distance functions on digital pictures. *Pattern Recognition*, 1:33–61, 1968.

[16] M. Schmitt. *Des Algorithmes Morphologiques à l'Intelligence Artificielle*. PhD thesis, Ecole des Mines, Paris, Feb. 1989.

[17] M. Schmitt and L. Vincent. *Morphological Image Analysis: a Practical and Algorithmic Handbook*. Cambridge University Press, (to appear in 1997).

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

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

[20] J. Serra and L. Vincent. An overview of morphological filtering. *Circuits, Systems and Signal Processing*, 11(1):47–108, Jan. 1992.

[21] P. Soille and M. Ansoult. Automated Basin Delineation from DEMs Using Mathematical Morphology. *Signal Processing*, 20:171–182, 1990.

[22] L. van Vliet and B. J. Verwer. A contour processing method for fast binary neighbourhood operations. *Pattern Recognition Letters*, 7:27–36, Jan. 1988.

[23] L. Vincent. *Algorithmes Morphologiques à Base de Files d'Attente et de Lacets: Extension aux Graphes*. PhD thesis, Ecole des Mines, Paris, May 1990.

[24] L. Vincent. Efficient computation of various types of skeletons. In *SPIE Vol. 1445, Medical Imaging V*, pages 297–311, San Jose, CA, 1991.

[25] L. Vincent. New trends in morphological algorithms. In *SPIE/SPSE Vol. 1451, Nonlinear Image Processing II*, pages 158–169, San Jose, CA, Feb. 1991.

[26] L. Vincent. Morphological algorithms. In E. R. Dougherty, editor, *Mathematical Morphology in Image Processing*, pages 255–288. Marcel-Dekker, Inc., New York, Sept. 1992.

[27] L. Vincent. Morphological grayscale reconstruction: Definition, efficient algorithms and applications in image analysis. In *IEEE Int. Computer Vision and Pattern Recog. Conference*, pages 633–635, Champaign IL, June 1992.

[28] L. Vincent and S. Beucher. The morphological approach to segmentation: an introduction. Technical report, Ecole des Mines, CMM, Paris, 1989.

[29] L. Vincent and B. Masters. Morphological image processing and network analysis of corneal endothelial cell images. In *SPIE Vol. 1769, Image Algebra and Morphological Image Processing III*, pages 212–226, San Diego, CA, July 1992.

[30] L. Vincent and P. Soille. Watersheds in digital spaces: an efficient algorithm based on immersion simulations. *IEEE Trans. Pattern Anal. Machine Intell.*, 13(6):583–598, June 1991.

[31] P. Wendt, E. Coyle, and N. Gallagher. Stack filters. *IEEE Transactions on Acoustics Speech and Signal Processing*, 34(4):898–911, Aug. 1986.