9. Segmentation

The *segmentation* of an image can be defined as its partition into different regions, each having certain properties. In a segmented image, the elementary picture elements are no longer the pixels but connected sets of pixels. Once the image has been segmented, measurements are performed on each region and adjacency relations between regions can be investigated. Image segmentation is therefore a key step towards the quantitative interpretation of image data.

In mathematical terms, a *segmentation* of an image f is a *partition* of its definition domain $\mathcal{D}_f$ into n disjoint nonempty sets $X_1, X_2, \ldots, X_n$ called *segments* such that the union of all segments equals $\mathcal{D}_f$ (see Sec. 2.5.3, page 31). Usually, an image that has been segmented is represented as a label image whereby each segment X_i is given a value different from all other segments. An alternative is to display the segment boundaries only. This enables a binary representation of a segmented image which can be overlaid on the original image.

The design of an algorithm for segmenting an image into meaningful regions requires some prior knowledge about the image objects that are to be recognised. This knowledge concerns features such as shape, size, orientation, grey level distribution, and texture. Ideally, these features should allow one to discriminate two different image objects. Unfortunately, features enabling the recognition of all image objects are seldom available in practical applications. Moreover, the value of a given feature is not always clearly defined since it may depend on the lighting, the resolution of the acquisition device, the view angle, the noise level, etc. As a consequence, there exists no general algorithm producing the optimal segmentation for all images. Existing algorithms are therefore *ad hoc* by nature and based on simplifying assumptions as well as a priori knowledge. A common assumption of many segmentation techniques consists in defining image objects as regions satisfying some uniformity predicate such as homogeneous grey level distributions.

In this chapter, we concentrate on segmentation techniques based on the direct processing of the image definition domain. These techniques are briefly reviewed in Sec. 9.1. There are essentially two approaches depending on whether the core of the segments (i.e., region growing techniques) or their boundaries (i.e., edge detection techniques) are searched. The key morphological transformation for segmenting an image is called the watershed transformation and is presented in Sec. 9.2. It consists of a combination of both region growing and edge detection approaches. The marker-controlled segmentation based on the watershed transformation is described in Sec. 9.3 and exemplified in Sec. 9.4. Bibliographical notes and references are given in Sec. 9.5. Morphological segmentation techniques based on the processing of a feature space defined from feature vectors calculated for all pixels are the scope of Chap. 10 while the segmentation of textured images is developed in Chap. 11.

9.1 Image segmentation techniques

Assuming that image objects are connected regions of little grey level variations, one should be able to extract these regions by using some neighbourhood properties. Indeed, a high grey scale variation between two adjacent pixels may indicate that these two pixels belong to different objects. This assumption does not hold directly for textured objects because grey level variations within a textured object may be higher than those occurring at the object boundaries. However, local texture measurements can be performed so as to obtain similar values for pixels belonging to similar textures and therefore high variations between two neighbour pixels belonging to two different textured regions (see Chap. 11).

In the case of *region growing*, homogeneous regions are first located. The growth of these regions is based on similarity measurements combining spatial and spectral attributes. It proceeds until all pixels of the image are assigned to a region. Region boundaries are created when two growing regions meet.

Edge detection techniques proceed the opposite way. As image objects are assumed to show little grey level variations, their edges are characterised by high grey level variations in their neighbourhood. The task of edge detection is to enhance and detect these variations. Local grey level intensity variations are enhanced by a gradient operator. The gradient image is then used to determine an edge map. A basic approach consists in thresholding the gradient image for all gradient values greater than a given threshold level. Unfortunately, the resulting edges are seldom connected. An additional processing is then required to obtain closed contours corresponding to object boundaries.

The morphological approach to image segmentation combines region growing and edge detection techniques: it groups the image pixels around the regional minima of the image and the boundaries of adjacent groupings are precisely located along the crest lines of the gradient image. This is achieved by a transformation called the *watershed transformation*.

9.2 The watershed transformation

Let us consider the topographic representation of a grey tone image (Sec. 2.2.2). Now, let a drop of water fall on such a topographic surface. According to the

law of gravitation, it will flow down along the steepest slope path until it reaches a minimum. The whole set of points of the surface whose steepest slope paths reach a given minimum constitutes the *catchment basin* associated with this minimum. The *watersheds* are the zones dividing adjacent catchment basins. This is illustrated in Fig. 9.1.

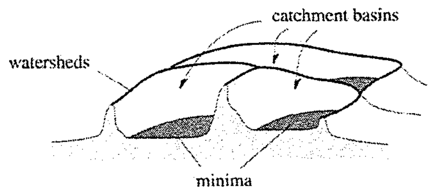


Fig. 9.1. Minima, catchment basins, and watersheds on the topographic representation of a grey scale image.

The watershed transformation appears to be a very powerful segmentation tool. Indeed, provided that the input image has been transformed so as to output an image whose minima mark relevant image objects and whose crest lines correspond to image object boundaries, the watershed transformation will partition the image into meaningful regions. This approach to the segmentation of grey scale images is detailed in Sec. 9.3. In the present section, we introduce two equivalent formal definitions of the watershed transformation (Secs. 9.2.1 and 9.2.2) and detail a fast implementation (Sec. 9.3.3).

9.2.1 Definition in terms of flooding simulations

The concept of watersheds expressed in terms of water flows is not suitable for a simple formal definition because there are many situations where the flow direction at a given point is not determined (e.g., plateau pixels or pixels having more than one neighbour pixel with the lowest grey scale value). However, a definition in terms of flooding simulations alleviates these problems.

Consider again the grey tone image as a topographic surface and assume that holes have been punched in each regional minimum of the surface. The surface is then slowly immerged into water. Starting from the minima at the lowest altitude, the water will progressively flood the catchment basins of the image. In addition, dams are raised at the places where the waters coming from two different minima would merge (see Fig. 9.2). At the end of this flooding procedure, each minimum is surrounded by dams delineating its associated catchment basin. The whole set of dams correspond to the watersheds. They provide us with a partition of the input image into its different catchment basins.

We now formalise this flooding process. The smallest value taken by the grey scale image f on its domain $\mathcal{D}_f$ is denoted by $h_{\min}$ and the largest

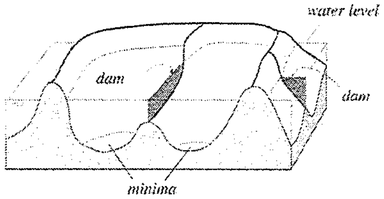


Fig. 9.2. Building dams at the places where the water coming from two different minima would merge.

by $h_{\max}$. The *catchment basin* associated with a minimum M is denoted by $\mathrm{CB}(M)$. The points of this catchment basin which have an altitude less than or equal to h are denoted by $\mathrm{CB}_h(M)$:

$$\mathrm{CB}_h(M) = \{p \in \mathrm{CB}(M) \mid f(p) \leq h\} = \mathrm{CB}(M) \cap T_{t \leq h}(f).$$

We denote by X_h the subset of all catchment basins which have a grey scale value less than or equal to h:

$$X_h = \cup_i \mathrm{CB}_h(M_i).$$

Finally, the set of points belonging to the regional minima of elevation h are denoted by $\mathrm{RMIN}_h(f)$.

The catchment basins are now progressively built by simulating the flooding process. The first image points that are reached by water are the points of lowest grey scale value. These points belong to the regional minima of the image at level $h_{\min}$. They are also equivalent to $X_{h_{\min}}$:

$$X_{h_{\min}} = T_{h_{\min}}(f) = \mathrm{RMIN}_{h_{\min}}(f).$$

The definition of $X_{h_{\min}+1}$ is based on the analysis of the flooding process up to the elevation $h_{\min}+1$. The water either expands the regions of the catchment basins already reached by water or starts to flood the catchment basins whose minima have an altitude equal to $h_{\min}+1$. This is illustrated in Fig. 9.3. More precisely, there are three possible relations of inclusion between a connected connected component Y of $T_{t \leq h_{\min}+1}(f)$ and the intersection between Y and $X_{h_{\min}}$:

(a) $Y \cap X_{h_{\min}} = \emptyset$ (Fig. 9.3a). It follows that Y is a new regional minimum of f at level $h_{\min}+1$ since

$$i)\ \forall\, p \in Y, \quad \begin{cases} p \notin X_{h_{\min}} \Longrightarrow f(p) \geq h_{\min}+1, \\ p \in Y \quad\ \Longrightarrow f(p) \leq h_{\min}+1. \end{cases}$$

$$ii)\ \forall\, p \in \delta^{(1)}(Y) \setminus Y,\ f(p) > h_{\min}+1.$$

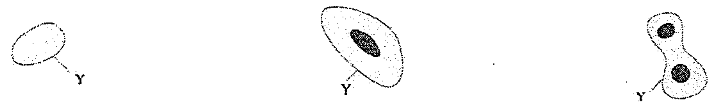
The set of all minima at level $h_{\min}+1$, i.e., $\mathrm{RMIN}_{h_{\min}+1}(f)$, will be considered for defining $X_{h_{\min}+1}$.

(b) $Y \cap X_{h_{\min}} \neq \emptyset$ and is connected (Fig. 9.3b). In this case, Y exactly corresponds to the pixels belonging to the catchment basin associated with the minimum $Y \cap X_{h_{\min}}$ and having a grey level less than or equal to $h_{\min}+1$:

$$Y = \mathrm{CB}_{h_{\min}+1}(Y \cap X_{h_{\min}}) = \mathrm{IZ}_Y(Y \cap X_{h_{\min}}).$$

(c) $Y \cap X_{h_{\min}} \neq \emptyset$ and is *not* connected (Fig. 9.3c). Therefore, Y contains more than one minimum of f at level $h_{\min}$. Denote by $Z_1, Z_2, \ldots, Z_k$ these minima, and let Z_i be one of them. At this point, the best possible approximation for $\mathrm{CB}_{h_{\min}+1}(Z_i)$ corresponds to the geodesic influence zone of Z_i inside Y:

$$\mathrm{CB}_{h_{\min}+1}(Z_i) = \mathrm{IZ}_Y(Z_i).$$



(a) $Y \cap X_{h_{\min}} = \emptyset$. (b) $Y \cap X_{h_{\min}} \neq \emptyset$, and is connected. (c) $Y \cap X_{h_{\min}} \neq \emptyset$, and is not connected.

Fig. 9.3. Inclusion relations appearing when flooding a grey scale image. Dark grey regions indicate pixels that have already been flooded at level $h_{\min}$, i.e., pixels of $X_{h_{\min}}$. Light grey regions are flooded by water when it reaches the level $h_{\min}+1$. (a) A new minimum has been discovered at level $h_{\min}+1$. (b) The flooded region of a catchment basin whose minimum is at level $h_{\min}$ is expanding. (c) The flooded regions of the catchment basins of two distinct minima at level $h_{\min}$ are expanding and merged together.

The two last inclusion relation correspond to the expansion of the regions of the catchment basins already reached by water. These expanded regions can be defined in terms of a unique geodesic influence zone, i.e., the influence zone of $X_{h_{\min}}$ within $T_{t \leq h_{\min}+1}$. It follows that $X_{h_{\min}+1}$ is defined as the union of these geodesic influence zones with the newly discovered regional minima:

$$X_{h_{\min}+1} = \mathrm{RMIN}_{h_{\min}+1}(f) \cup \mathrm{IZ}_{T_{t \leq h_{\min}+1}(f)}(X_{h_{\min}}).$$

This recursion formula holds for all levels h. It is illustrated in Fig. 9.4.

The set of *catchment basins* of a grey scale image f is equal to the set $X_{h_{\max}}$, i.e., once all levels have been flooded:

(i) $X_{h_{\min}} = T_{h_{\min}}(f)$,
(ii) $\forall\, h \in [h_{\min}, h_{\max}-1],\ \ X_{h+1} = \mathrm{RMIN}_{h+1}(f) \cup \mathrm{IZ}_{T_{t \leq h+1}(f)}(X_h)$.

The catchment basin image CB of a grey tone image is represented as a label image whereby each labelled region corresponds to the catchment basin of a

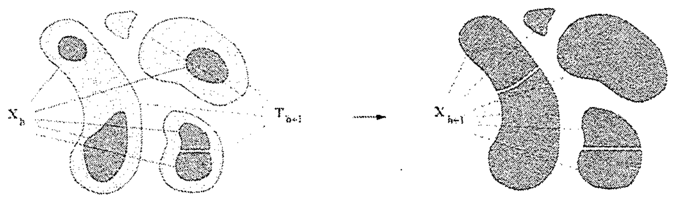


Fig. 9.4. Recursion relation between X_h and X_{h+1}.

regional minimum of the input image. The *watersheds* WS of f correspond to the boundaries of the catchment basins of f. Figure 9.5 illustrates the iterative construction of the catchment basins on a synthetic image. A fast algorithm for implementing the flooding simulation is presented in Sec. 9.2.3.

9.2.2 Definition in terms of generalised geodesy

The catchment basins of a grey scale image can also be considered as the influence zones of the regional minima of the image. In this sense, the watershed lines are nothing but a skeleton by influence zones of the image minima. This idea can be formalised using the notions of generalised geodesy described in Sec. 7.3.

The regional minima RMIN of the input image f are first set to the minimum image value $h_{\min}$. The resulting image is denoted by f':

$$f'(\mathrm{x}) = \begin{cases} h_{\min}, \text{ if } p \in \mathrm{RMIN}(f), \\ f(\mathrm{x}), \text{ otherwise.} \end{cases} \tag{9.1}$$

Figures 9.6a and b illustrates this transformation on a 1-D signal. In Sec. 7.3.1, we have seen that the geodesic time function corresponds to an integration of the intensity values of the geodesic mask, starting from a given reference set. By performing this process on the internal gradient of f' and using the regional minima as reference set, the image f' can be reconstructed:

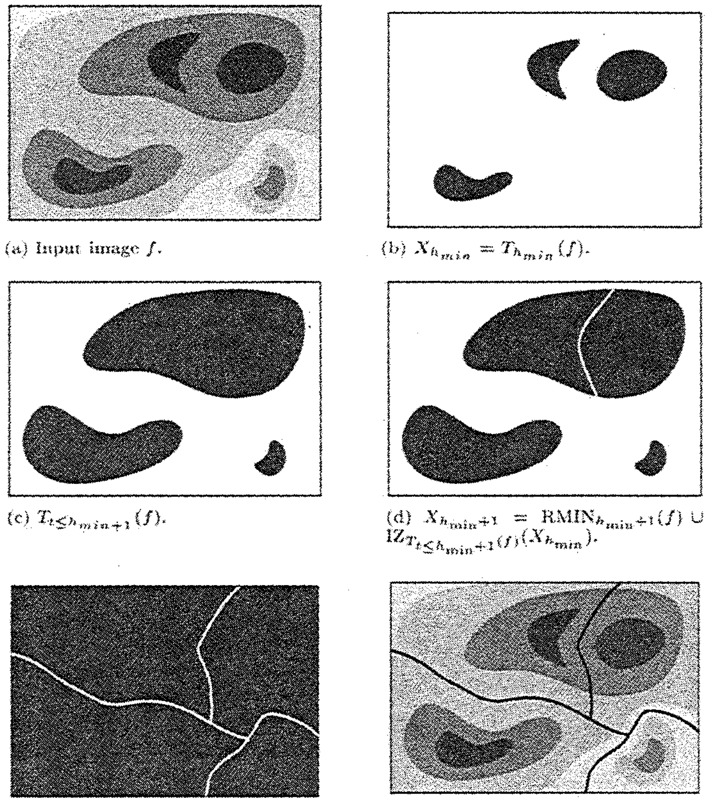
$$f' = \mathcal{T}_{\rho^-(f')}[\mathrm{RMIN}(f)].$$

Figures 9.6c and d shows an example on a 1-dimensional signal. The points where the wavefronts coming from two distinct minima meet define the watersheds of the original grey scale image:

$$\mathrm{WS}(f) = \mathrm{SKIZ}_{\rho^-(f')}[\mathrm{RMIN}(f)]. \tag{9.2}$$

The catchment basins are defined using the corresponding geodesic influence zones:

$$\mathrm{CB}(f) = \mathrm{IZ}_{\rho^-(f')}[\mathrm{RMIN}(f)].$$

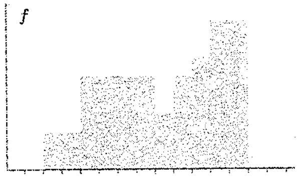


(a) Input image f.
(b) $X_{h_{min}} = T_{h_{min}}(f)$.
(c) $T_{t\leq h_{min+1}}(f)$.
(d) $X_{h_{\min}+1} = \mathrm{RMIN}_{h_{\min}+1}(f) \cup \mathrm{IZ}_{T_{t\leq h_{\min}+1}(f)}(X_{h_{\min}})$.
(e) $X_{h_{max}} = \cup_i \mathrm{CB}(M_i)$.
(f) WS(f) superimposed on f.

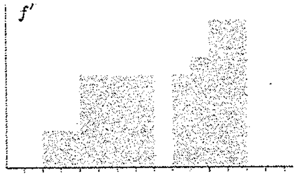
Fig. 9.5. Watersheds on a synthetic image having only four grey scale levels: illustration of the definition based on successive thresholds and geodesic influence zones.

The first step which consists in setting all image minima to the minimum image value ensures that the integration process will be delayed proportionally to the height of these minima. This mimics the flooding process which starts from the lowest minima and progressively reaches all image pixels. An example on a grey scale image is illustrated in Fig. 9.7. The algorithm for computing geodesic time functions (Sec. 7.3.1) can be easily extended for the computation of geodesic time influence zones by propagating the labels of the pre-labelled regional minima.

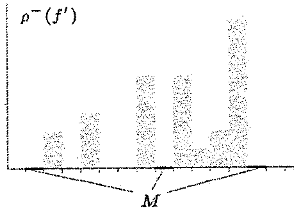
There exists also a link between the watersheds of an image and its grey tone skeleton (obtained by homotopic thinnings). Indeed, the watersheds are a subset of the pixels of the grey tone skeleton that do not belong to a regional



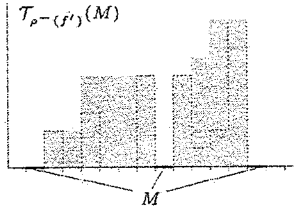
(a) 1-D signal f



(b) Minima of f set to zero



(c) Internal gradient $\rho^-(f')$



(d) $\mathcal{T}_{\rho^-(f')}(M)$

Fig. 9.6. Catchment basins as generalised influence zones. The internal gradient of the input signal is first computed and used as a grey scale geodesic mask for computing the geodesic influence zones of the minima M of the input image.

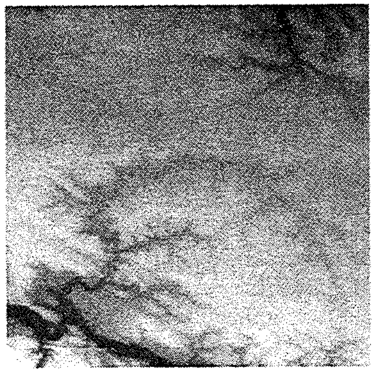
minimum:

$$\mathrm{WS}(f) \subseteq [\mathrm{RMIN}[\mathrm{SK}(f)]]^c.$$

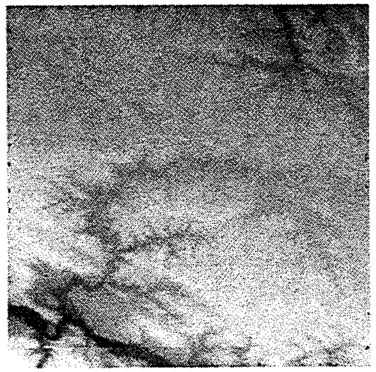
This has already been illustrated in Fig. 5.17 on page 167.

9.2.3 Computation of watersheds

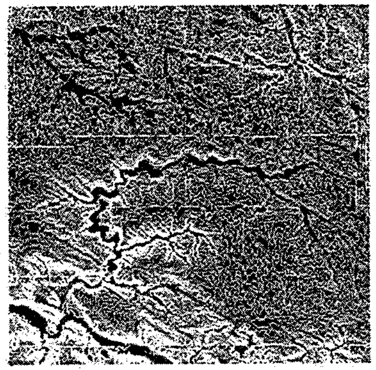
The definition of watersheds in terms of a flooding process is well suited to their efficient computation. We have to consider the successive thresholds of the image and compute the geodesic influence zones of each threshold inside the next one. As the image is processed grey level by grey level, only a small number of pixels are effectively treated at each step. Thus, rather than scanning the entire image to modify the value of a few pixels only, one must have a direct access to these pixels. This is achieved by means of a preliminary sorting of the pixels in the increasing order of their grey values. Among the numerous sorting techniques, the distributive sort (Isaac and Singleton, 1956) is particularly suited to the present problem. The procedure first determines the frequency distribution of each image grey level. Then, the cumulative frequency distribution is computed. This induces the direct assignment of each pixel to a unique cell in the sorted array. As memory and time requirements to compute frequency distributions are generally negligible compared to those required for images, this sorting technique constitutes one of the best choices for sorting discrete image data.



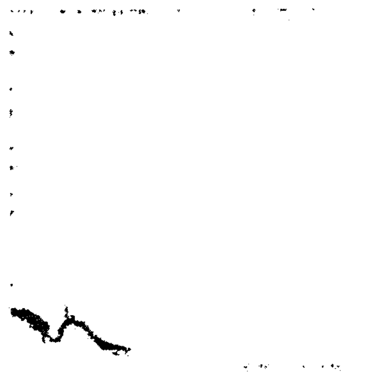
(a) Original digital elevation model f.



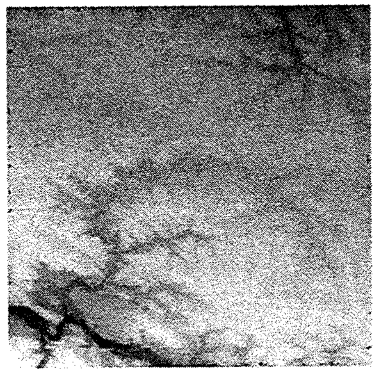
(b) f' following Eq. 9.1.



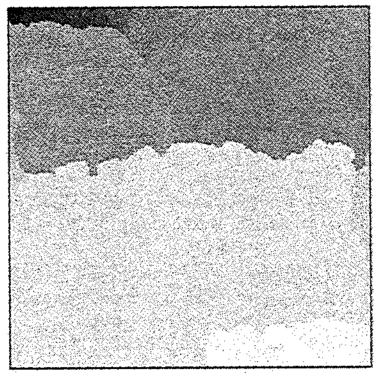
(c) Internal gradient ρ^- of f'.



(d) Minima of f: $\mathrm{RMIN}(f)$.



(e) Geodesic time function, $\mathcal{T}_{\rho^-(f')}[\mathrm{RMIN}(f)]$.



(f) Geodesic influence zones, $\mathrm{IZ}_{\rho^-(f')}[\mathrm{RMIN}(f)] = \mathrm{CB}(f)$.

Fig. 9.7. The catchment basins CB of a grey scale image seen as geodesic influence zones.

Once the pixels have been sorted, a fast computation of geodesic influence zones is enabled by a breadth-first scanning of each threshold level T_h. This scanning can be implemented thanks to a queue of pixels. Suppose the flooding has reached a given level h. Therefore, a label has already been assigned to each catchment basin whose regional minimum has an altitude less than or equal to h. Thanks to the prior sorting, the pixels of altitude $h+1$ are accessed directly. These pixels are given a special value denoted by `mask`. Those among them which have an already labelled pixel as one of their neighbours are put into the queue. Starting from these pixels, the queue structure enables to extend the labelled catchment basins inside the mask of pixels having value `mask`, by computing geodesic influence zones. After this step, only the minima at level $h+1$ have not been reached. Indeed, they are not connected to any of the already labelled catchment basins. Therefore, a second scanning of the pixels at level $h+1$ is necessary to detect the pixels which still have value `mask`, and to give a new label to the thus discovered catchment basins. Moreover, a particular value is assigned to the pixels where two different catchment basins would merge. This exactly corresponds to the construction of dams, as explained in Sec. 9.2.1.

A pseudo-code of the flooding simulation is presented hereafter. The queue which is used is a First-In-First-Out (FIFO) data structure as described on page 195. The pseudo-code of the algorithm is given hereafter:

Fast watersheds using flooding simulations

```
# define mask      -2   ; initial value of a threshold level
# define wshed      0   ; value of pixels belonging to watersheds
# define init     -1   ; initial value of f_o
# define inqueue  -3   ; value assigned to pixels put into the queue
•– input:   f_i, grey tone image (nonnegative integers)
 – output:  f_o, image of labelled catchment basins; labels are 1,2,...
• Initialisations:
  – Value init is assigned to each pixel of f_o: ∀ p ∈ D_im_o, f_o(p) ← init
  – current_label ← 0
  – flag: Boolean variable
  – N_G(p) is the set of neighbours of p

• Sort the pixels of f_i in the increasing order of their grey values.
• For h ← h_min to h_max { ; geodesic SKIZ of level h−1 inside level h
  ∀ pixel p such that f_i(p) = h { ; direct access thanks to sorting
    f_o(p) ← mask;
    if there exists p' ∈ N_G(p) such that f_o(p') > 0 or f_o(p') = wshed {
      f_o(p) ← inqueue; fifo_add(p);
    }
  }
  while fifo_empty() = false {
    p ← fifo_retrieve();
    ∀ pixel p' ∈ N_G(p) {
      if f_o(p') > 0 { ; i.e., p' belongs to an already labelled basin
        if (f_o(p) = inqueue or (f_o(p) = wshed and flag = true))
          f_o(p) ← f_o(p');
        else if (f_o(p) > 0 and f_o(p) ≠ f_o(p')) {
          f_o(p) ← wshed; flag ← false;
        }
      }
      else if f_o(p') = wshed
        if f_o(p) = inqueue {  f_o(p) ← wshed;  flag ← true;  }
      else if f_o(p') = mask {  f_o(p') ← inqueue;  fifo_add(p');  }
    }
  }
  ∀ pixel p such that f_i(p) = h { ; check for new minima
    if f_o(p) = mask {
      current_label ← current_label + 1;
      fifo_add(p); f_o(p) ← current_label;
      while fifo_empty() = false {
        p' ← fifo_retrieve();
        ∀ pixel p'' ∈ N_G(p') {
          if f_o(p'') = mask {  fifo_add(p'');  f_o(p'') ← current_label;  }
        }
      }
    }
  }
}
```


In the above algorithm, the Boolean variable `flag` is used to detect whether the value `wshed`, which is currently assigned to a pixel p, comes from another `wshed`-pixel in the neighbourhood of p, rather than from two neighbouring pixels with distinct labels. The original description of this algorithm is presented in (Soille and Vincent, 1990). An extension requiring distance calculations on plateaus for better accuracy is detailed in (Vincent and Soille, 1991) together with a review of many other watershed algorithms.

9.3 Marker-controlled segmentation

We first present the principle of the marker-controlled approach (Sec. 9.3.1). Some hints about the way to produce appropriate marker and segmentation functions are then given (Sec. 9.3.2). Finally, the implementation of a watershed algorithm incorporating the use of markers is briefly introduced (Sec. 9.3.3).

9.3.1 Principle

The basic idea behind the marker-controlled segmentation is to transform the input image in such a way that the watersheds of the transformed image correspond to meaningful object boundaries. The transformed image is called the *segmentation function*. In practice, a direct computation of the watersheds of the segmentation function produces an over-segmentation which is due to the presence of spurious minima. Consequently, the segmentation