Image Processing and Imaging

Morphological Image Processing

Dominik Söllinger
Fachbereich AIHI
Universität Salzburg

Wintersemester 2022/23

Outline

Image segmentation often results in a binary mask. Unfortunately, the segmentation often not completely satisfactory. An additional enhancement step is required.

Figure: Left: Segmentation of a fingerprint image. The mask is noisy and valley structures exhibit holes. Right: Enhanced version using morphological operators.

In this lecture we will study **morphological operators** for image enhancement.

But what exactly do we mean when we talk about *morphology*?

- In the field of biology, *morphology* deals with the form and structure of animals and plants.
- We use the same word here in the context of *mathematical morphology* as a tool for extracting image components that are useful in the representation and description of region shape, such as boundaries, skeletons, and the convex hull.

Mathematical Morphology:

- Mathematical morphology **operates on sets**
- In case of binary images, an object can be simply defined by a set of <x,y> coordinates.
- In case of grayscale images, the components of the set are in $\mathbb{Z}^3$

- We distinguish between types of sets: *Objects* and *Structuring elements* (SE's)
- The structuring element is similar to a convolution kernel. This means that ...
  - It can moved around (similar to a convolution)
  - It can be of any size
  - Either contains 1's and 0's. (and sometimes so-called "Don't Care" elements)
  - At each pixel position, a specified **logical operation** is performed between the structuring element and the underlying image. The result is stored in the ouput image at the pixel position.

- For this introduction to the subject, we concentrate on the simplest case, namely, the use of a basic 3×3 structuring element containing all 1's
- With this restriction, it is the logical operation that determines the outcome

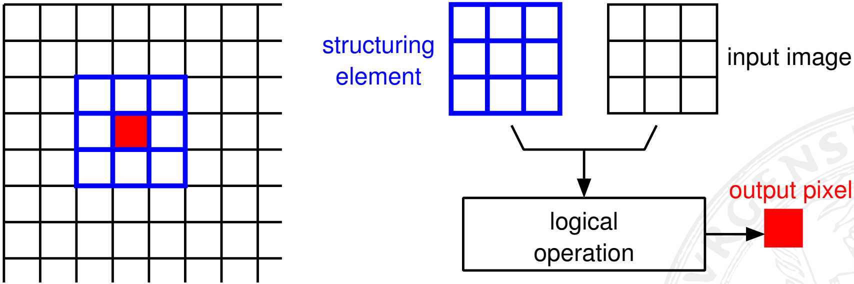


Figure: Morphological image processing

Outline

Outline

Language of Morphological Processing:

- We refer to **S** as a structuring element and **B** as the object on the binary image.
- Both, **B** and **S**, are sets of coordinates obtained from the 2D Cartesian image grid.

- $\mathbf{S}_{xy}$ denotes the structuring element after it has been translated so that its origin is located at the point (x, y). In other words, ...

$$\mathbf{S}_z = \{c | c = s + z \text{ for } s \in S\}$$

where z is the the location of the origin (x, y)

Outline

Basic primitive operators

There are two operators which are fundamental to morphological processing:

- Erosion
- Dilation

Many of the morphological algorithms are based on these two operations.

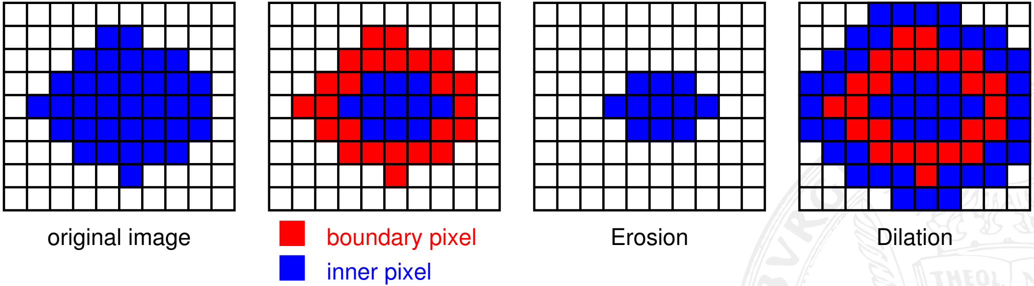


Figure: Comparison of erosion and dilation. Note that, a boundary point is a pixel that is located inside an object, but that has at least one neighbor outside the object

Erosion (1)

Erosion:

Erosion can be used to remove image objects that are too small to be of interest.

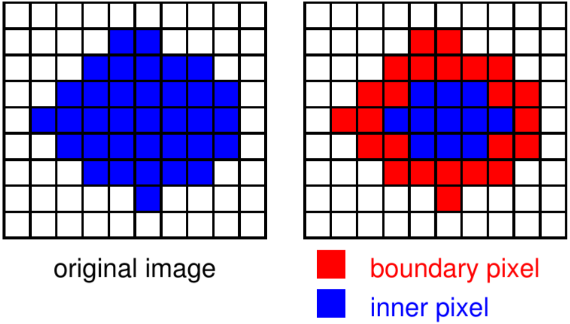
The erosion between the object $\mathbf{B}$ and the structuring element $\mathbf{S}$ can be computed as follows:

$$\mathbf{E} = \mathbf{B} \otimes \mathbf{S} = \{(x, y) | \mathbf{S}_{xy} \subseteq \mathbf{B}\} \quad (1)$$

Note that (x, y) is only contained in $\mathbf{E}$ if $\mathbf{S}_{xy}$ is fully contained in $\mathbf{B}$.

Example:

- With a basic 3 x 3 structuring element, we can eliminate all the boundary points from an object, leaving the object smaller in area by one pixel all around its perimeter.
- If the object is circular, its diameter decreases by two pixels with each erosion.
- If it narrows to less than three pixels thick at any point, it will become disconnected (into two objects) at that point.



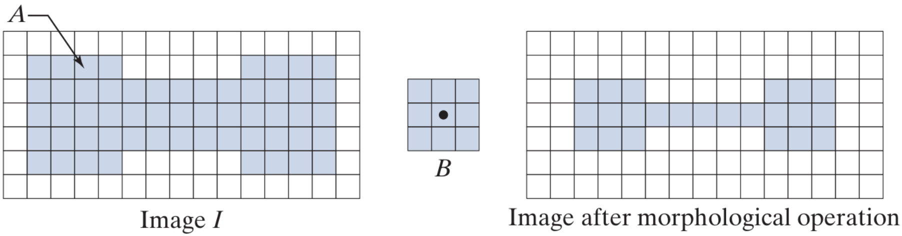


Figure: Example of Erosion. Note that in the figure B denotes the structuring element. If the erosion was applied a second time, the object would get disconnected.

a	b
c	d

FIGURE 9.5 Using erosion to remove image components. (a) A 486×486 binary image of a wire-bond mask in which foreground pixels are shown in white. (b)–(d) Image eroded using square structuring elements of sizes 11×11, 15×15, and 45×45 elements, respectively, all valued 1.

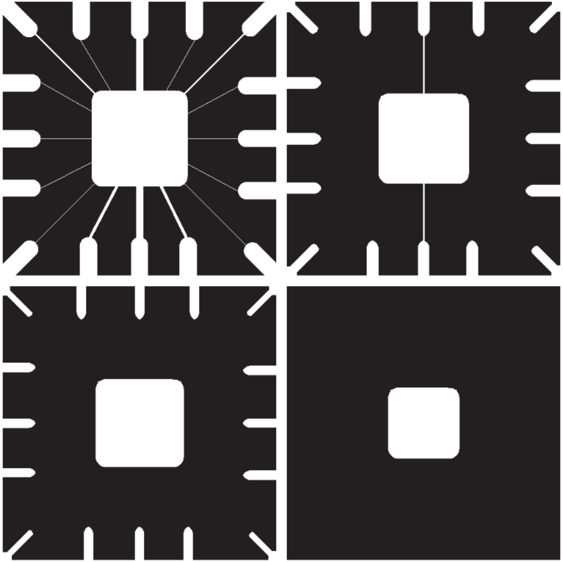


Figure: Erosion used to post-process an image of a wire-bond mask.

Dilation:

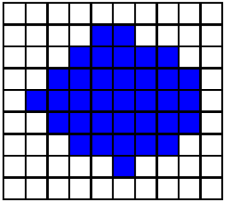
Dilation "grows" or "thickens" objects in an image. It can be used for filling holes in an object.

The dilation between the object **B** and the structuring element **S** can be computed as follows:

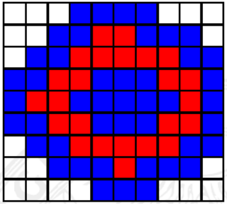
$$\mathbf{D} = \mathbf{B} \oplus \mathbf{S} = \{(x, y) | \mathbf{S}_{xy} \cap \mathbf{B} \neq \emptyset\} \quad (2)$$

Example:

- Once again, a basic 3×3 structuring element is used
- If the object is circular, its diameter increases by two pixels with each dilation
- If the two objects are seperated by less than three pixels at any point, they will become connected (merged into one object) at that point



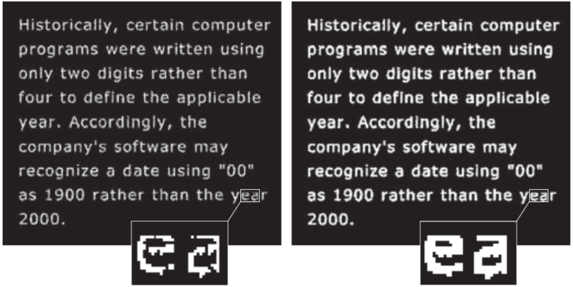
original image



Dilation

a c
b

FIGURE 9.7
(a) Low-resolution text showing broken characters (see magnified view).
(b) Structuring element.
(c) Dilation of (a) by (b). Broken segments were joined.



1	1	1
1	1	1
1	1	1

(a) Reparing broken characters using dilation

Outline

Opening and Closing (1)

Motivation:

Erosion not only removes objects that are smaller than the structuring element, it also reduces the size of the object itself. This often is not desired.

But what if we again perform a dilation after erosion?

Opening

- **Opening** simply means that erosion is applied followed by dilation. In other words, ...

$$\mathbf{B} \circ \mathbf{S} = (\mathbf{B} \otimes \mathbf{S}) \oplus \mathbf{S} \tag{3}$$

- Opening eliminates small thin objects, breaks objects at thin points, and generally smoothing the boundaries of larger objects without significantly changing their area

Figure: Left: Original object. Right: Object after opening using a squared 3x3 structuring element. Note that the size of the two boxes has been retained.

- **Closing** simply means that dilation is applied followed by erosion. In other words, ...

$$\mathsf{B} \bullet \mathsf{S} = (\mathsf{B} \oplus \mathsf{S}) \otimes \mathsf{S} \tag{4}$$

- Closing fills small and thin holes in objects, connects nearby objects, and generally smoothing the boundaries of objects without significantly changing their area

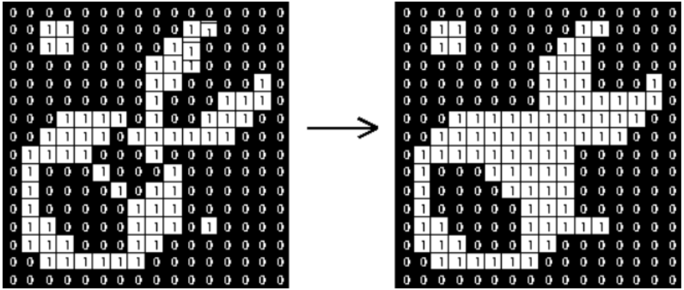


Figure: Left: Original image. Right: Object after closing using a squared 3x3 structuring element

Opening and Closing (2)

When noisy images are segmented by thresholding the...

- resulting boundaries are often quite ragged
- objects have false holes
- background is peppered with small noise objects

Successive openings or closings can improve the situation markedly.

a b
d c
e f

FIGURE 9.11
(a) Noisy image. (b) Structuring element. (c) Eroded image. (d) Dilation of the erosion (opening of A). (e) Dilation of the opening. (f) Closing of the opening. (Original image courtesy of the National Institute of Standards and Technology.)

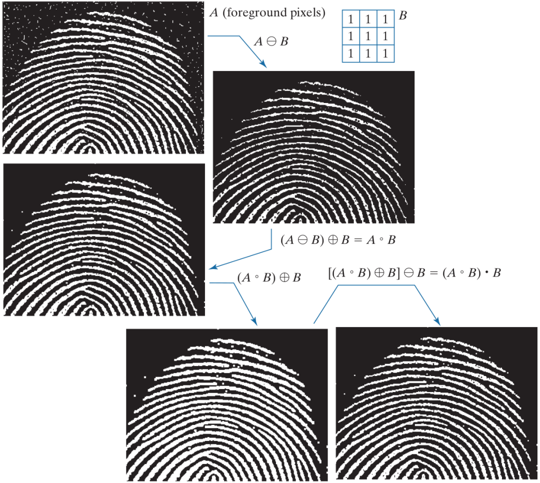


Figure: Example of Opening and Closing used to enhance a fingerprint image

Outline

Hit-or-Miss Transform (1)

- The **Hit-or-Miss Transform** (HMT) is basic tool for shape detection
- Unlike the previous morphological operators, the HMT uses two structuring elements: S_1, for detecting shapes in the foreground and S_2, for detecting shapes in the background.
- However, we can circumvent defining two separate structuring elements by supporting structuring elements with three different states: 1s (Object in foreground), 0s (Object in background), "Don't care" (either foreground or background)

	1	
0	1	1
0	0	

Figure: Example of a structuring element used for HMT. Blank cells denote "Don't care"'s.

- If the foreground (1's) and background (0's) pixels in the structuring element exactly match foreground (1's) and background (0's) pixels in the image, then the pixel at the origin is set to the foreground (1's) color.

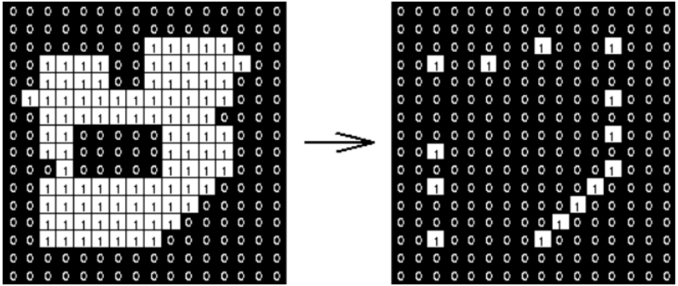


Figure: Effect of the hit-and-miss based right angle convex corner detector on a simple binary image. The used structuring elements are shown on the next slide.

	1	
0	1	1
0	0	

	1	
1	1	0
	0	0

	0	0
1	1	0
	1	

0	0	
0	1	1
	1	

Figure: The four structuring elements used for corner finding in previous images using the hit-and-miss transform. Note that they are all the same element, but rotated by different amounts.

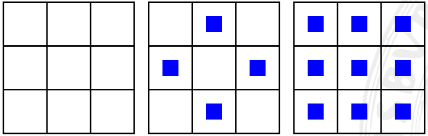
Most morphological operators operating on NxN neighbourhood can be efficiently implemented using **lookup tables**.

Example:

Let's we consider an 3x3 neighbourhood in a **binary** image. As a result, there are $2^9 = 512$ possible "configurations" we might encounter in the binary image.

We can now simply precalculate the result for a given structuring element and morphological operator for each possible configuration. The neighbourhood operation can then be implemented with 512-entry lookup table.

16	8	4
32	1	2
64	128	256



0   128+32+8+2=170   511

Outline

Boundary extraction (1)

The boundary of a set **B** can be obtained by first eroding **B** by a suitable structuring element **S**, and then performing the set difference between **B** and its erosion.

That is,

$$Bound(\mathbf{B}) = \mathbf{B} - (\mathbf{B} \otimes \mathbf{S}) \tag{5}$$

where "$-$" denotes the difference between the two sets.

a b
c d

FIGURE 9.15
(a) Set, A, of foreground pixels.
(b) Structuring element.
(c) A eroded by B.
(d) Boundary of A.

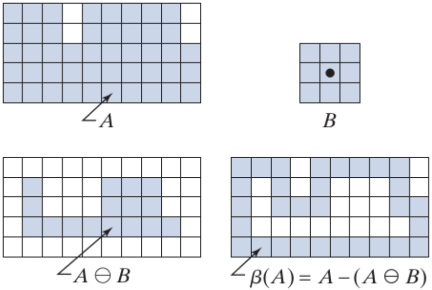


Figure: Example of boundary extraction using erosion

a b

FIGURE 9.16
(a) A binary image.
(b) Result of using Eq. (9-18) with the structuring element in Fig. 9.15(b).

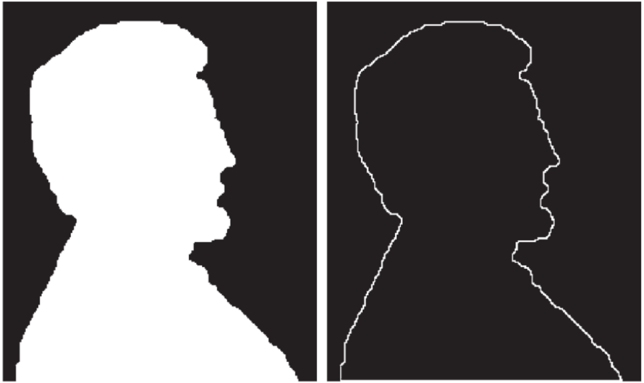


Figure: Boundary extraction used to extract the contours of the person

Outline

Thinning (1)

We can think of thinning as an **erosion which does not break objects**. Typically thinning is applied iteratively until a skeleton is obtained.

We can think of thinning as a two-step process:

1st step: Apply an erosion, but instead of eliminating the pixels only mark them as candidates for removal.

2nd step: Check which of the marked pixels can be removed with breaking the connectivity of the object.

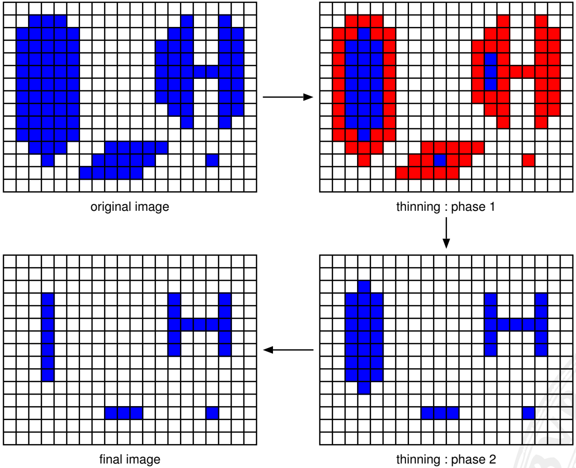


Figure: Two-step thinning process

Thinning (3)

In practice, a thinning "step" is typically implemented by means of a Hit-or-Miss transform.

$$Thin(\mathbf{B}) = \mathbf{B} \cap HMT(\mathbf{B}, \mathbf{S}) \tag{6}$$

Each thinning step is then repeated using different structuring elements. An example of such structuring element is shown in the figure.

0	0	0
	1	
1	1	1

	0	0
1	1	0
	1	

Figure: Structuring elements for skeletonization by morphological thinning. At each iteration, the image is first thinned by the left hand structuring element, and then by the right hand one, and then with the remaining six 90 deg rotations of the two elements. The process is repeated in cyclic fashion until none of the thinnings produces any further change.

If the described process is applied iteratively until convergence we obtain a **skeleton**.

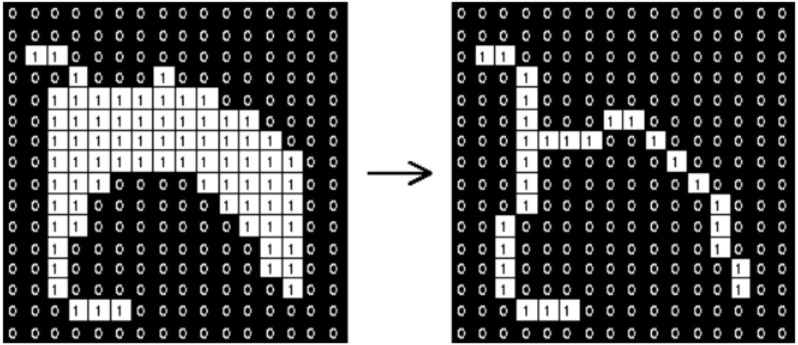


Figure: Thinning iteratively applied to the left image until convergence.

Outline

Skeletonisation (Medial Axis Transform) (1)

An operation closely related to iterative thinning is **Skeletonization**, also known as **Medial Axis Transform (MAT)**. However, the MAT often produces different results than standard thinning.

Medial Axis: The medial axis of an object is the set of all points that have *more than one* closest point on the object's boundary.

Hint: Think of the largest circle with its origin at position (x,y) that still fits inside the object. Does the circle touch the boundary at at least one point? If yes, (x,y) lies on the medial axis.

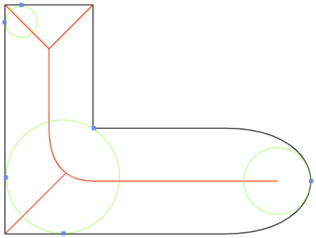


Figure: Example of MAT: Points on the medial axis are marked red

The **MAT** seeks finding all point which lie on the medial axis to construct the skeleton. As a result, the resulting skeleton is different from standard thinning.

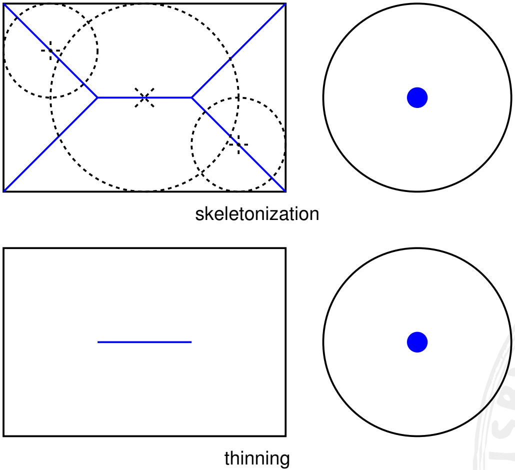


Figure: Skeletonization (MAT) vs. iterative thinning. The resulting skeleton is marked blue.

Skeletonization can be either implemented by means of a **distance transform** or, as with thinning, with a two-pass conditional erosion.

However, the rule for deleting pixels is slightly different.

Outline

Pruning (1)

Thinning or skeletonization often leaves unwanted parasitic components (a.k.a. spurs). Spurs are short branches having an endpoint located within three or so pixels of an intersection.

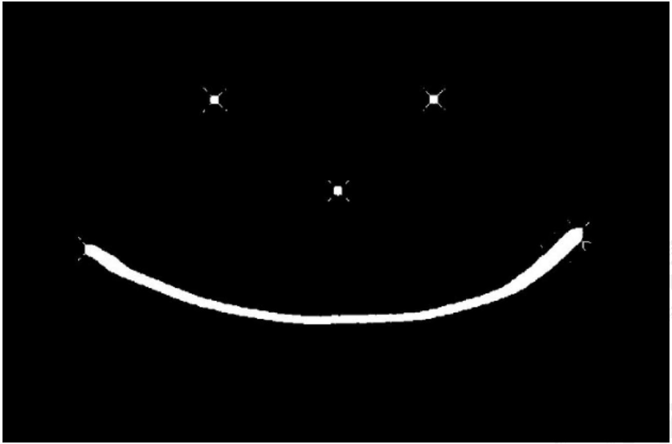


Figure: Thinned image with spurs

These spures can be removed by a series of 3×3 operations that remove endpoints (thereby shortening all the branches), followed by reconstruction of the remaining branches. Since there is no endpoint to grow back from, the spur is not reconstructed.

This process is called **pruning**.

Outline

Thickening (1)

Thickening is the morphological dual of thinning. We can think of it as a **dilation that does not merge nearby objects**.

Similarily to thinning, thickening can be conveniently implemented using a hit-or-miss transform.

$$Thick(\mathbf{B}) = \mathbf{B} \cup HMT(\mathbf{B}, \mathbf{S}) \tag{7}$$

Thickening (2)

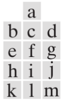


FIGURE 9.23
(a) Structuring elements.
(b) Set A.
(c) Result of thinning A with B^1 (shaded).
(d) Result of thinning A_1 with B_2.
(e)–(i) Results of thinning with the next six SEs. (There was no change between A_7 and A_8.)
(j)–(k) Result of using the first four elements again.
(l) Result after convergence.
(m) Result converted to m-connectivity.

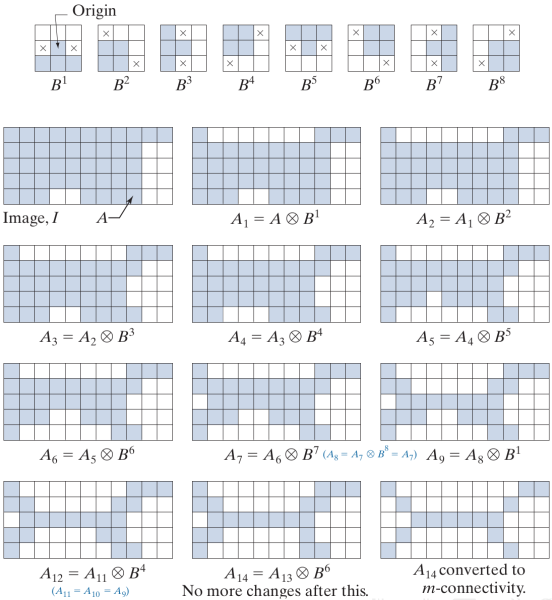


Figure: Example of thinning

Thickening (3)

An alternative is to complement the image and use the apply the thinning operation on the background.

a b
c d
e

FIGURE 9.24
(a) Set A.
(b) Complement of A.
(c) Result of thinning the complement.
(d) Thickened set obtained by complementing (c).
(e) Final result, with no disconnected points.

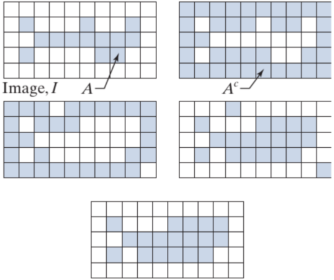


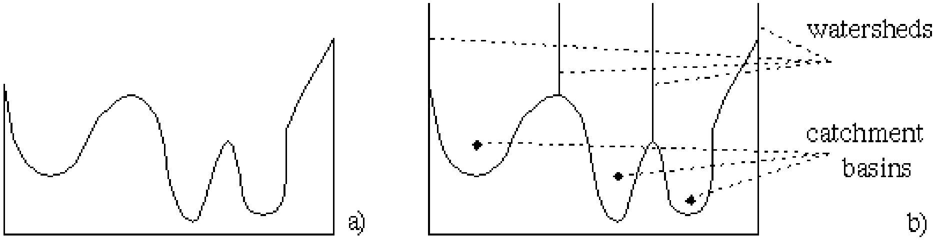
Figure: Thickening by thinning of the complement

Note that depending on the structure of the object, thickening can result in disconnected points (see Subfigure (d)). Hence, the method is usually followed by post-processing.

Outline

A good (animated) visualisation and several examples may be found at:
`http://cmm.ensmp.fr/~beucher/wtshed.html`

- Watersheds separate different (water)basins
- Transfer this notion into image processing context →
- Image is interpreted as three dimensional structure:
  - Luminance values are interpreted as height measure (elevation at the corresponding image coordinates)
  - In many cases a gradient image is used for further processing
  - Region borders (i.e. edge chains) correspond to "high valued" edge chains and inner areas with low gradient correspond to basins



Watershed Segmentation - Two Different Approaches

- Basins are homogeneous in the sense that all pixels belonging to a catchment basin are connected to the minimum value of the basin by a path, the pixel of which are monotonically decreasing in direction to the minumum
- Basins represent the regions of the segemented image, the watersheds represent region borders

There are two differnt approaches to watershed transform:

1. Determines a "downstream" path to a minimum for each pixel; A catchment basin is defined as the set of pixels the path of which leads to the same minimum; problem is the unique determination of the path (which can be facilitated by local gradients in the contineous case).
2. Dual to the first one and uses "flooding": catchment basins are flooded from below (assume that holes are at the location of minima in the 3D surface, when submerging the surface, water enters through the holes); as soon as two catchment basins would be fused due to rising water, a dam is constructed to prevent this; value of the dam pixels is set to the maximum value of the image.

Watershed Segmentation - Flooding Approach (1)

We focus onto the second technique:

- Preprocessing: pixels are sorted according to their gray-scale, gray-scale histogram is computed, and a list of pointers is built pointing at pixels with gray-scale h
- In this manner, all pixels with a specific gray-scale can be addressed efficiently

- Assume that flooding has been propagated until gray-scale k
- Each pixel with gray-scale $\leq k$ has been uniquely assigned to a basin and carries its label
- In the next step all pixels with gray-scale $k+1$ are processed
- A pixel with this gray scale can be assigned to basin l in case at least one direct neighbour carries label l

- In order to determine membership to a basin, zones of influence are defined:
  - Zone of influence of a basin l are the positions of the not-assigned but connected pixels with gray-scale $k+1$, the distance to l of which is smaller than to any other basin.

Watershed Segmentation - Flooding Approach (2)

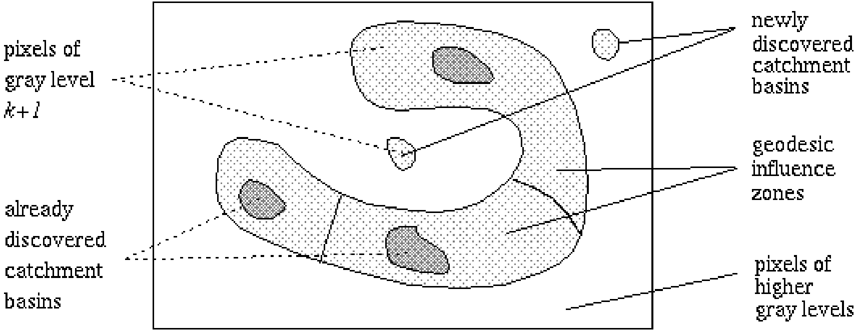


Figure: Zones of influence of basins

- All pixels with gray-scale $k + 1$ belonging to the influence zone of the basin l are assigned the label l, which means that catchment basins grow
- Not-yet assigned pixels are processed successively, pixels without label correspond to new basins and get a new label
- Borders separating the catchment basins are the watersheds

Watershed Segmentation - Flooding Approach (3)

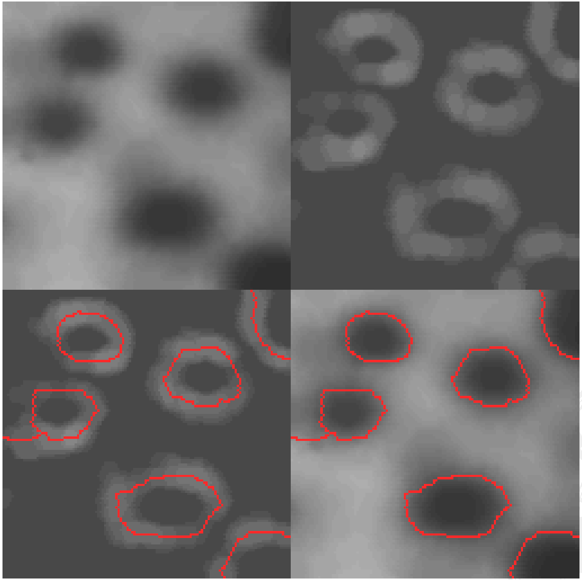
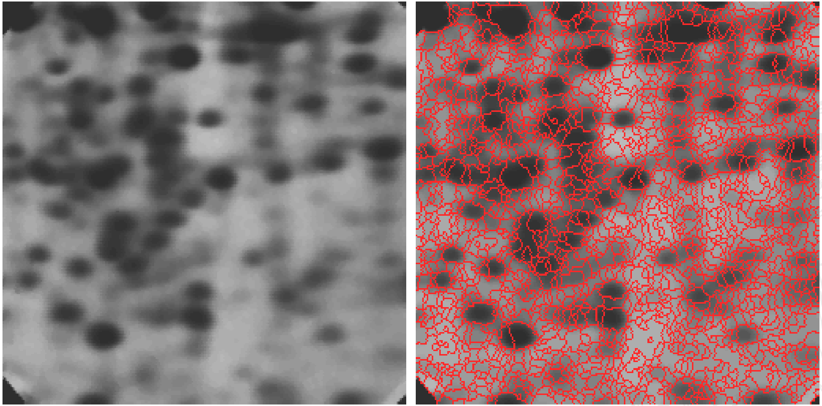


Figure: Original, gradient image, watersheds, original with watersheds

Note: We have not described how to explicitly construct dams – this will be explained subsequently using morphological operators

Oversegmentation: When applying watershed segmentation as described so far, we result in significant oversegmentation quite often (i.e. too may regions are formed), since the number of minima is simply too high (e.g. caused by noise):

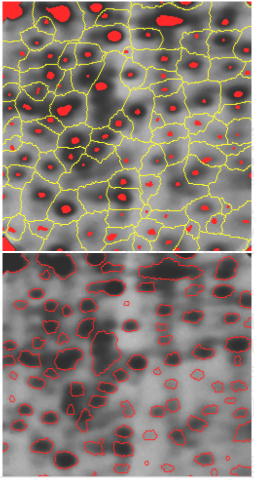


In order to limit this effect, the following strategy can be applied:

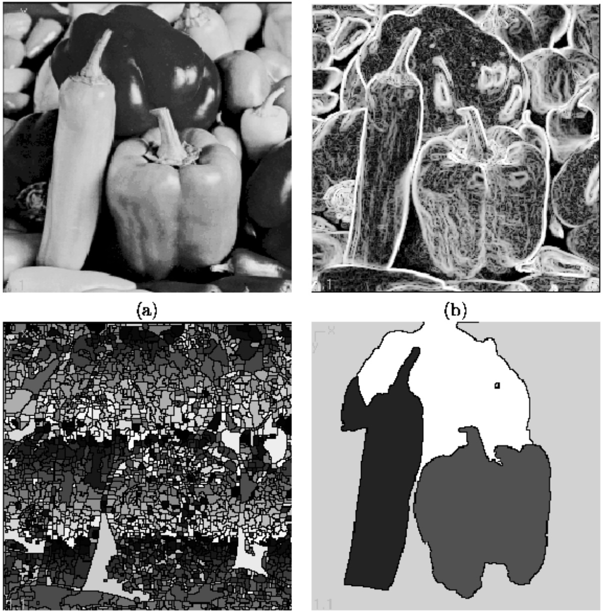
- Image smoothing (e.g. Gauss filtering with large σ)
- Marker: Only "internal markers" are accepted as local minima, i.e. contigous regions with identical gray-scale surrounded by pixels with larger value.

Markers:

- Setting internal markers and subsequently applying the watershed algorithm to the image
- Resulting watershed lines are denoted as "external markers", which partition the image into regions
- Each region contains a single object with its associated background
- Now we can apply a watershed procedure or thresholding to each region to arrive at the desired segmentation
- Additional measure: number of internal markers can be limited or a minimum size can be required



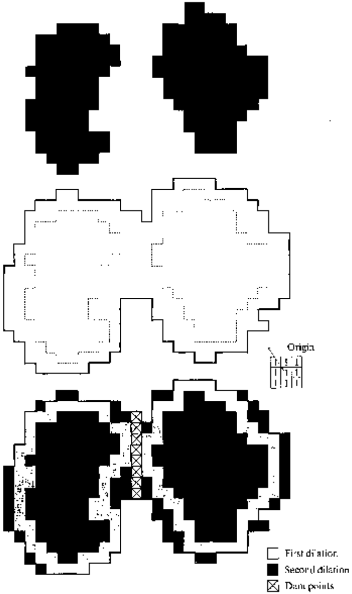
Watershed Segmentation - All Variants Example



(a) (b) (c) (d)

Dam construction

- If requirement of an explicit dam construction process during flooding:
- Flooding step $n-1$ right before a fusion of two catchment basins is taken as the initial stage for dam construction (black regions in figure)
- Fused region after step n is denoted as q (white region in figure)
- Subsequently, dilation is applied to the two black regions using a constant 1 3 x 3 structuring element, satisfying two conditions:
  1. The center of the structuring element resides in q.
  2. Dilation is only applied without a fusion of the two regions.



Example (right):

- First dilation step: can be conducted without any problems
- The two black regions are enlarged consistently
- Second dilation run: several points do not satisfy the first condition
- That is why the perimeter curve is disconnected
- Points satisfying the first condition but not the second one are defined to be dam points
- These points are set to the maximal luminance value in order not to get over-flooded again
- Then, flooding is continued