UNIT-6

Morphological Image Processing

Introduction

The word *morphology* commonly denotes a branch of biology that deals with the form and structure of animals and plants. Morphology in image processing is a tool for extracting image components that are useful in the representation and description of region shape, such as boundaries and skeletons. Furthermore, the morphological operations can be used for filtering, thinning and pruning. The language of the Morphology comes from the set theory, where image objects can be represented by sets.

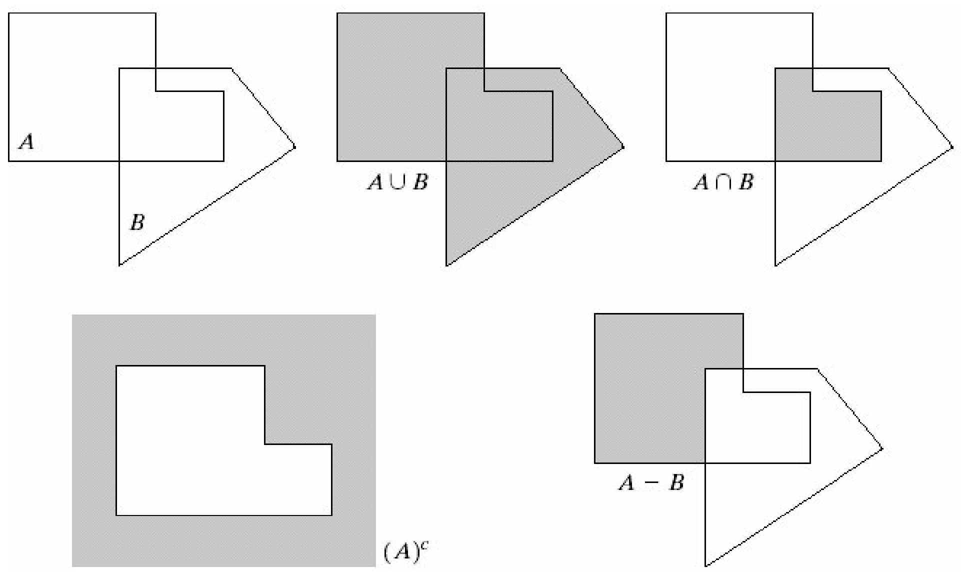
Some Basic Concepts form Set Theory :

- If every element of a set A is also an element of another set B, then A is said to be a subset of B, denoted as $A \subseteq B$
- The union of two sets A and B, denoted by $C = A \cup B$
- The intersection of two sets A and B, denote by $D = A \cap B$
- Disjoint or mutually exclusive $A \cap B = \emptyset$
- The complement of a set A is the set of elements not contained in

$$A^c = \{\omega | \omega \notin A\}$$

- The difference of two sets A and B, denoted A - B, is defined as

$$A - B = \{\omega | \omega \in A, \omega \notin B\} = A \cap B^c$$



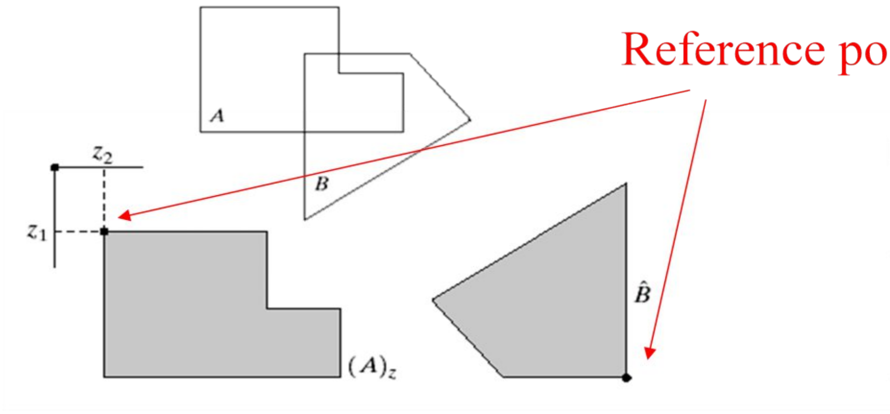
a b c
d e

FIGURE 9.1 (a) Two sets *A* and *B*. (b) The union of *A* and *B*. (c) The intersection of *A* and *B*. (d) The complement of *A*. (e) The difference between *A* and *B*.

Reflection and Translation by examples:

- Need for a reference point.
- Reflection of B: = {x|x=-b, for b∈B}
- Translation of A by $x=(x_1,x_2)$, denoted by $(A)_x$ is defined as:

$$(A)_x = \{c \mid c=a+x, \text{ for } a\in A\}$$



a b

FIGURE 9.2
(a) Translation of A by z.
(b) Reflection of B. The sets A and B are from Fig. 9.1.

Dilation

Dilation is used for expanding an element A by using structuring element B. Dilation of A by B and is defined by the following equation:

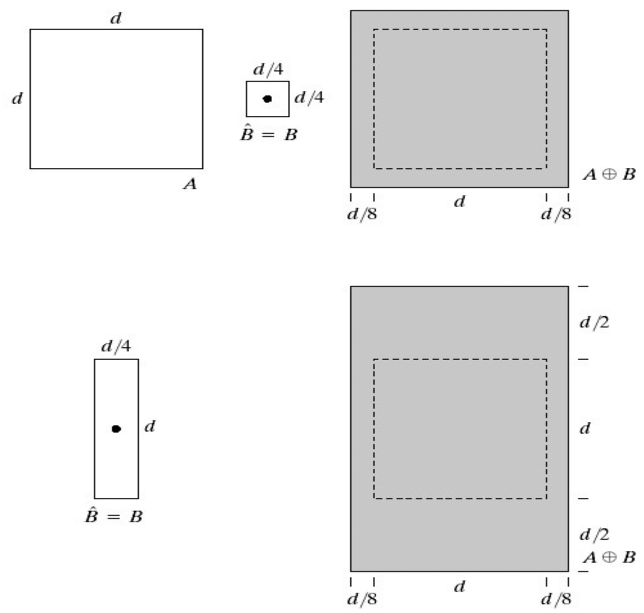
$$A \oplus B = \{z \mid (\hat{B})_z \cap A \neq \emptyset\} \quad (9.2-1)$$

This equation is based on obtaining the reflection of B about its origin and shifting this reflection by z. The dilation of A by B is the set of all displacements z, such that and A overlap by at least one element. Based On this interpretation the equation of (9.2-1) can be rewritten as:

$$A \oplus B = \{z \mid [(\hat{B})_z \cap A] \subset A\} \quad (9.2-2)$$

a b c
d e

FIGURE 9.4
(a) Set A.
(b) Square structuring element (dot is the center).
(c) Dilation of A by B, shown shaded.
(d) Elongated structuring element.
(e) Dilation of A using this element.



Dilation is typically applied to binary image, but there are versions that work on gray scale image. The basic effect of the operator on a binary image is to gradually enlarge the boundaries of regions of foreground pixels (i.e. white pixels, typically). Thus areas of foreground pixels grow in size while holes within those regions become smaller.

Any pixel in the output image touched by the **dot** in the structuring element is set to ON when any point of the structuring element touches a ON pixel in the original image. This tends to close up holes in an image by expanding the ON regions. It also makes objects larger. Note that the result depends upon both the shape of the structuring element and the location of its origin.

Summary effects of dilation:

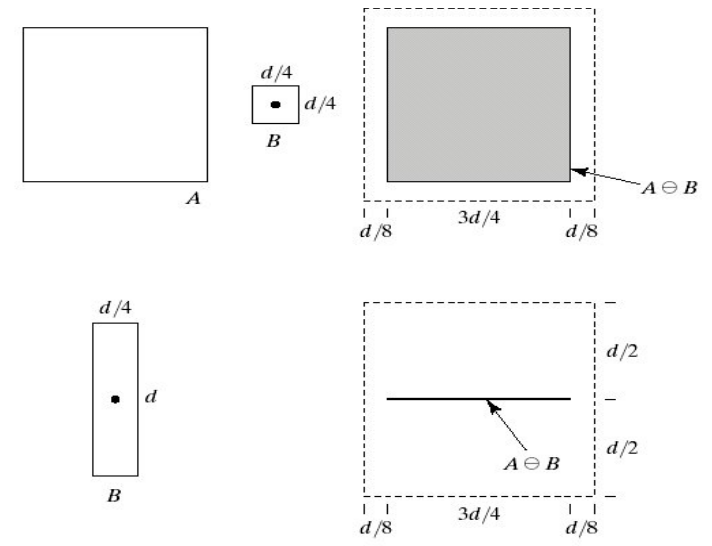
- Expand/enlarge objects in the image
- Fill gaps or bays of insufficient width
- Fill small holes of sufficiently small size
- Connects objects separated by a distance less than the size of the window

Erosion

Erosion is used for shrinking of element A by using element B. Erosion for Sets A and B in Z^2, is defined by the following equation:

$$A \ominus B = \{z|[(B)z \subseteq A\} \quad (9.2-3)$$

This equation indicates that the erosion of A by B is the set of all points z such that B, translated by z, is combined in A.



a b c
d e

FIGURE 9.6 (a) Set A. (b) Square structuring element. (c) Erosion of A by B, shown shaded. (d) Elongated structuring element. (e) Erosion of A using this element.

Any pixel in the output image touched by the · in the structuring element is set to ON when every point of the structuring element touches a ON pixel in the original image. This tends to makes objects smaller by removing pixels.

Duality between dilation and erosion:

Dilation and erosion are duals of each other with respect to set complementation and reflection. That is,

$$(A \ominus B)^c = A^c \oplus \hat{B}$$

a b c

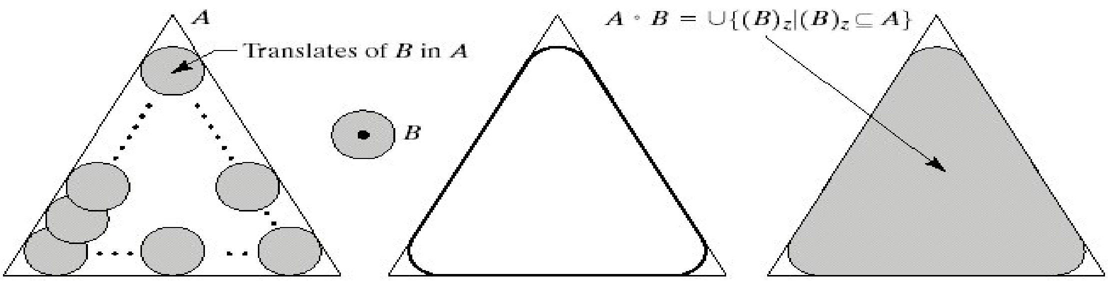
FIGURE 9.7 (a) Image of squares of size 1, 3, 5, 7, 9, and 15 pixels on the side. (b) Erosion of (a) with a square structuring element of 1's, 13 pixels on the side. (c) Dilation of (b) with the same structuring element.

Opening:

An erosion followed by a dilation using the ***same structuring element*** for both operations.

$$A \circ B = (A \ominus B) \oplus B = \bigcup \{(B_z) | (B_z) \subseteq A\}$$

- Smooth contour
- Break narrow isthmuses
- Remove thin protrusion



a b c d

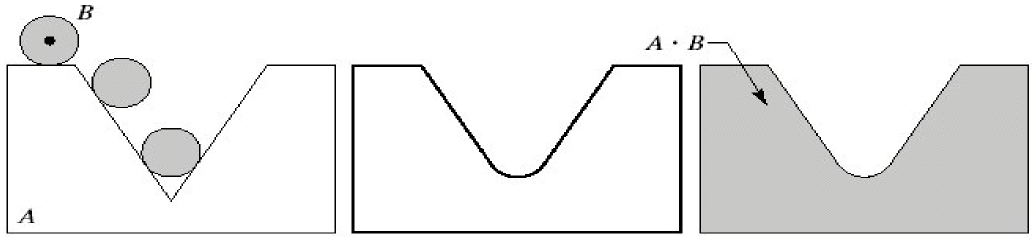
FIGURE 9.8 (a) Structuring element B "rolling" along the inner boundary of A (the dot indicates the origin of B). (c) The heavy line is the outer boundary of the opening. (d) Complete opening (shaded).

Closing:

A Dilation followed by a erosion using the ***same*** structuring ***element*** for both operations.

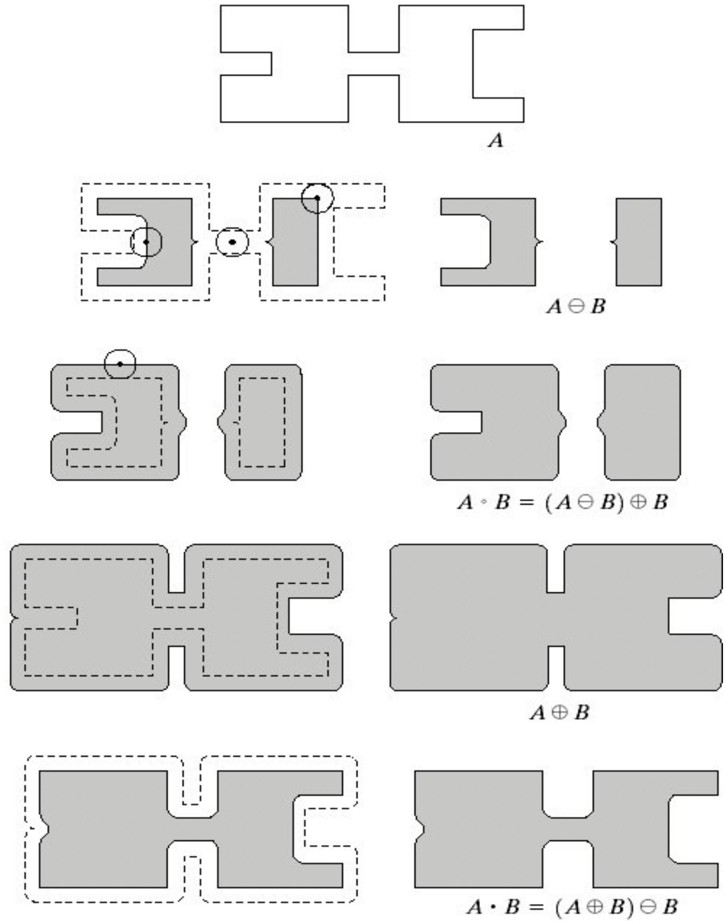
$$A \circ B = (A \oplus B) \ominus B$$

- Smooth contour
- Fuse narrow breaks, and long thin gulfs.
- Remove small holes, and fill gaps.



a b c

FIGURE 9.9 (a) Structuring element B "rolling" on the outer boundary of set A. (b) Heavy line is the outer boundary of the closing. (c) Complete closing (shaded).



a
b c
d e
f g
h i

FIGURE 9.10 Morphological opening and closing. The structuring element is the small circle shown in various positions in (b). The dark dot is the center of the structuring element.

Hit-or-Miss Transform:

The hit-and-miss transform is a basic tool for shape detection. The hit-or-miss transform is a general binary morphological operation that can be used to look for particular patterns of foreground and background pixels in an image.

Concept: To detect a shape:

- **Hit object**
- **Miss background**

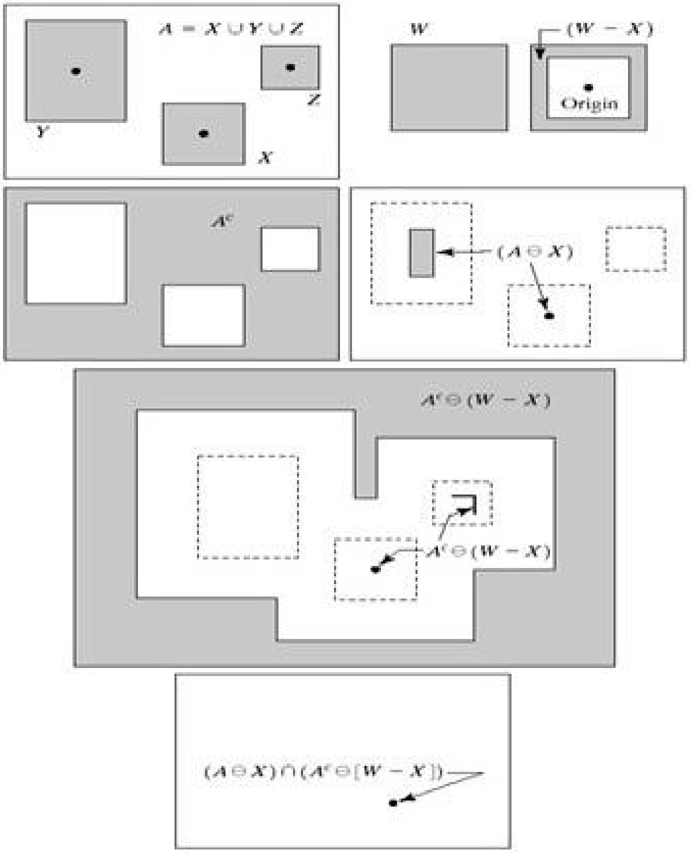
Let the origin of each shape be located at its center of gravity.

- If we want to find the location of a shape– **X** , at (larger) image, **A**
- Let X be enclosed by a small window, say – **W**.
- The ***local background*** of X with respect to W is defined as the *set difference* (**W - X**).
- Apply *erosion* operator of A by X, will get us the set of locations of the origin of X, such that X is completely contained in A.
- It may be also view geometrically as the set of all locations of the origin of X at which X found a match (**hit**) in A.
- Apply *erosion* operator on the *complement of A* by the *local background* set (W – X).
- Notice, that the set of locations for which X **exactly** fits inside A is the ***intersection*** of these two last operators above.
- If B denotes the set composed of X and it's background B = (B1,B2) ; B1 = X , B2 = (W-X).
- The match (or set of matches) of B in A, denoted as

$$A \circledast B = (A \ominus B_1) \cap (A^c \ominus B_2)$$

B1: Object related, B2: Background related

- The reason for using these kind of structuring element B = (B1,B2) is based on definition **two or more objects are distinct only if they are disjoint (disconnected) sets.**
- In some applications, we may interested in detecting certain **patterns (combinations)** of 1's and 0's, not for detecting individual objects.
- In this case a background is not required and the *hit-or-miss transform* reduces to simple *erosion*.
- This simplified pattern detection scheme is used in some of the algorithms for – **identifying characters within a text**.



a b
c d
e
f

FIGURE 9.12 (a) Set A. (b) A window, W, and the local background of X with respect to W, $(W - X)$. (c) Complement of A. (d) Erosion of A by X. (e) Erosion of A^c by $(W - X)$. (f) Intersection of (d) and (e), showing the location of the origin of X, as desired.

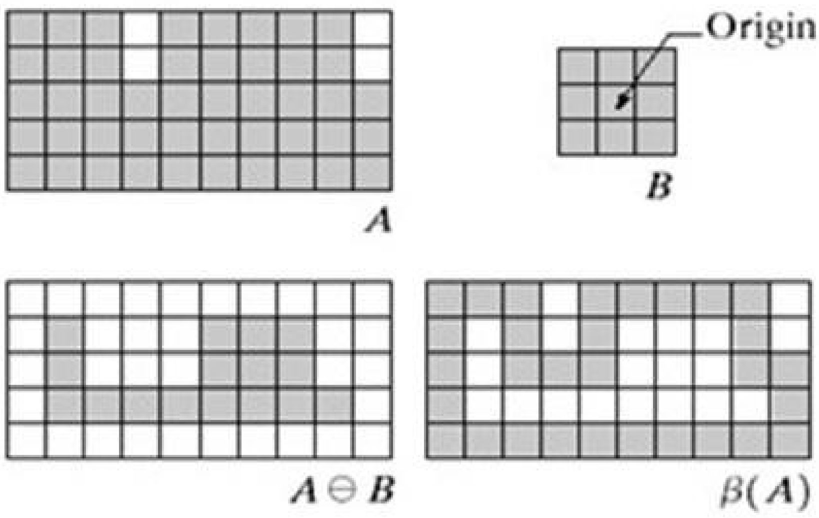
The structural elements used for Hit-or-miss transforms are an extension to the ones used with dilation, erosion etc. The structural elements can contain both foreground and background pixels, rather than just foreground pixels, i.e. both ones and zeros. The structuring element is superimposed over each pixel in the input image, and if an exact match is found between the foreground and background pixels in the structuring element and the image, the input pixel lying below the origin of the structuring element is set to the foreground pixel value. If it does not match, the input pixel is replaced by the boundary pixel value.

Basic Morphological Algorithms

Boundary Extraction:

The boundary of a set A is obtained by first eroding A by structuring element B and then taking the set difference of A and it's erosion. The resultant image after subtracting the eroded image from the original image has the boundary of the objects extracted. The thickness of the boundary depends on the size of the structuring element. The boundary of a set A is obtained by first eroding A by structuring element B and then taking the set difference of A and it's erosion. The resultant image after subtracting the eroded image from the original image has the boundary of the objects extracted. The thickness of the boundary depends on the size of the structuring element. The boundary β (A) of a set A is

$$\beta(A) = A - (A \ominus B)$$



a b
c d

FIGURE 9.13 (a) Set *A*. (b) Structuring element *B*. (c) *A* eroded by *B*. (d) Boundary, given by the set difference between *A* and its erosion.

a b

FIGURE 9.14 (a) A simple binary image, with 1's represented in white. (b) Result of using Eq. (9.5-1) with the structuring element in Fig. 9.13(b).

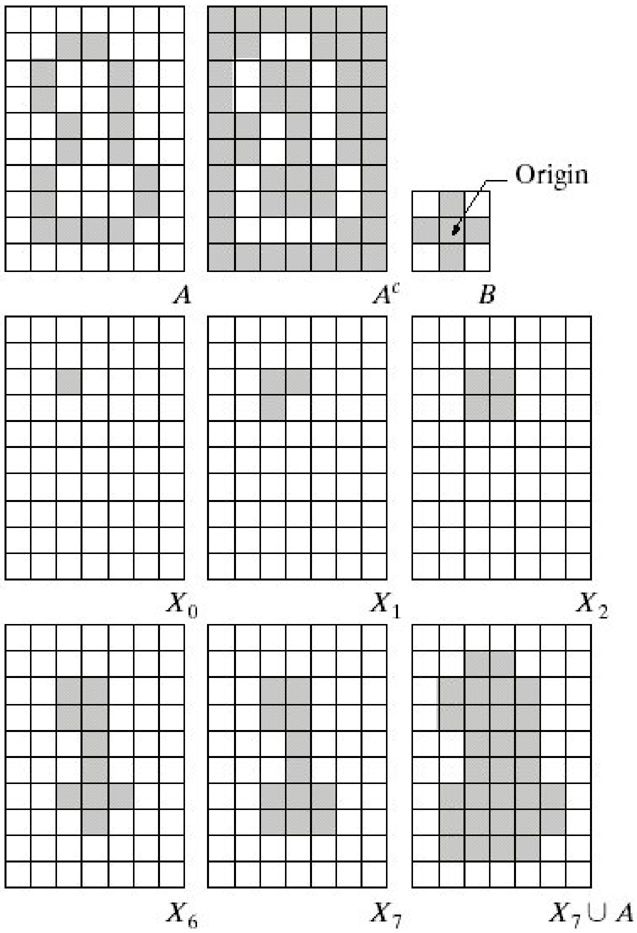
Region Filling or Hole Filling:

A Hole may be defined as a background region surrounded by connected border of foreground pixels. This algorithm is based on a set of dilations, complementation and intersections. Let P is the point inside the boundary, and that is filled with the value of 1.

$$X_k = (X_{k-1} \oplus B) \cap A^c$$

- The process stops when $X_k = X_{k-1}$

- The set X_k contains all the filled holes
- The result that given by union of A and X_k , is a set contains the filled set and the boundary.



a b c
d e f
g h i

FIGURE 9.15 Region filling. (a) Set *A*. (b) Complement of *A*. (c) Structuring element *B*. (d) Initial point inside the boundary. (e)–(h) Various steps of Eq. (9.5-2). (i) Final result [union of (a) and (h)].

Extraction of Connected Components:

Extraction of connected components from a binary image is central to many automated image analysis applications.

- Let A be a set containing one or more connected components and form an array X_0 whose elements are 0s except at each location known to correspond to a point in each connected component A which is set to 1.
- The objective is to start with X_0 and find all the connected components.

$$X_k = (X_{k-1} \oplus B) \cap A$$

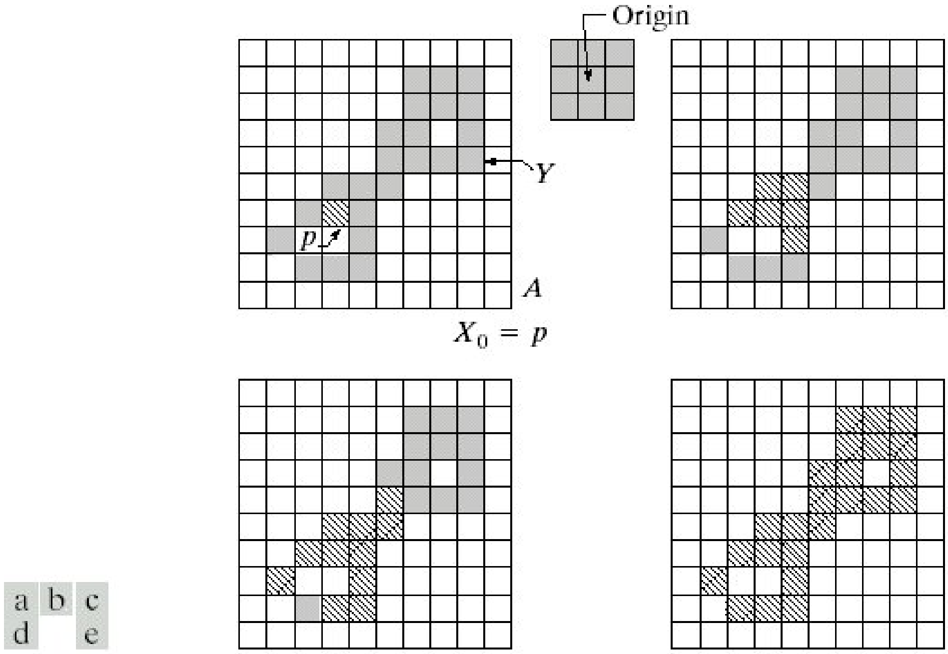


FIGURE 9.17 (a) Set A showing initial point p (all shaded points are valued 1, but are shown different from p to indicate that they have not yet been found by the algorithm). (b) Structuring element. (c) Result of first iterative step. (d) Result of second step. (e) Final result.

Convex Hull:

A is said to be convex if a straight line segment joining any two points in A lies entirely within A.

- The convex hull H of set S is the smallest convex set containing S
- The set difference H-S is called the convex deficiency of S

The convex hull and convex deficiency useful for object description. This algorithm iteratively applying the hit-or-miss transforms to A with the first of B element, unions it with A, and repeated with second element of B.

Let B^i , i=1,2,3,4 represents the four structuring elements. Then we need to implement the

$$X_k^i = (X_{k-1} \circledast B^i) \cup A$$

Let us consider

$$X_0^i = A.$$

and the procedure terminates when

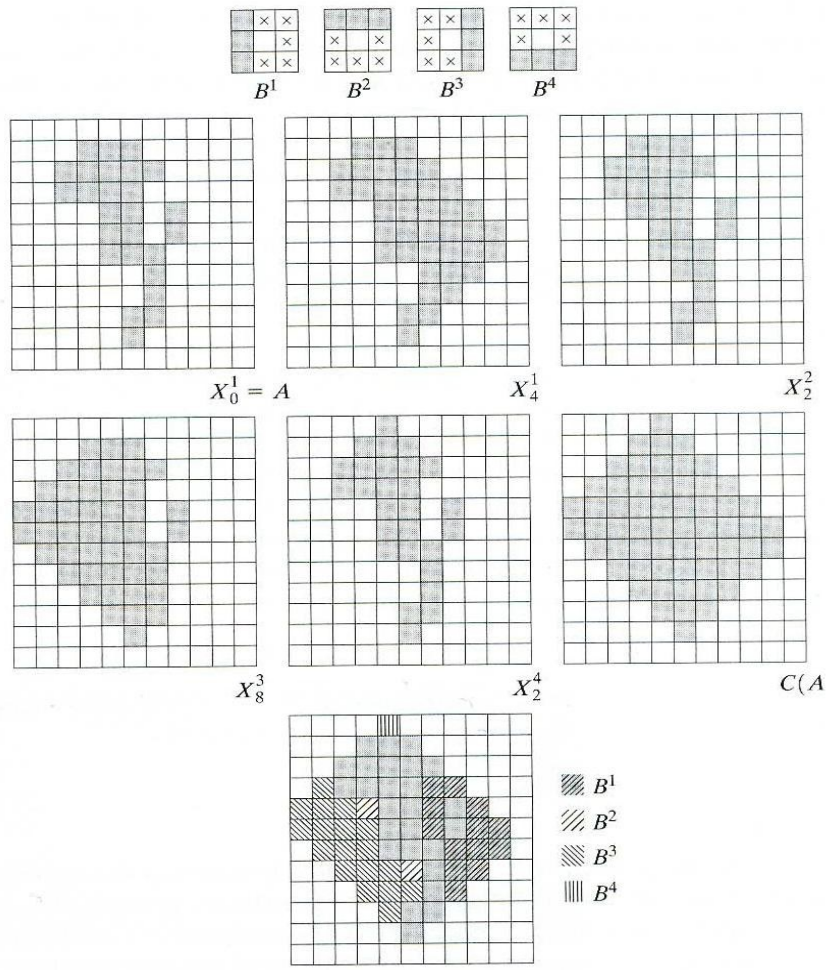
$$X_k^i = X_{k-1}^i$$

If

$$D^i = X_k^i$$

then the convex hull of A is defined as

$$C(A) = \bigcup_{i=1}^{4} D^i$$



a
b c d
e f g
h

FIGURE 9.19 (a) Structuring elements. (b) Set A. (c)–(f) Results of convergence with the structuring elements shown in (a). (g) Convex hull. (h) Convex hull showing the contribution of each structuring element.

Thinning:

The thinning of a set A by a structuring element B, can be defined by terms of the hit-and-miss transform:

$$A \otimes B = A - (A \circledast B) = A \cap (A \circledast B)^c$$

A more useful expression for thinning A symmetrically is based on a sequence of structuring elements:

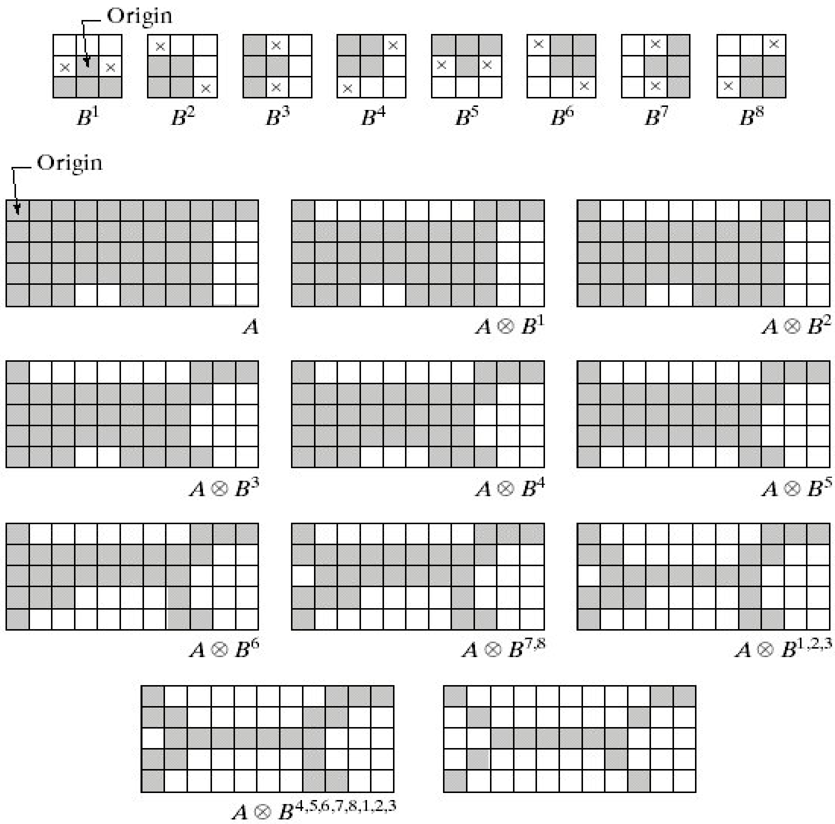
$$\{B\}=\{B^1, B^2, B^3, \ldots, B^n\}$$

Where B^i is a rotated version of B^{i-1}. Using this concept we define thinning by a sequence of structuring elements:

$$A \otimes \{B\} = ((\ldots((A \otimes B^1) \otimes B^2) \ldots) \otimes B^n)$$

The process is to thin by one pass with B^1, then thin the result with one pass with B^2, and so on until A is thinned with one pass with B^n. The entire process is repeated until no further changes occur. Each pass is preformed using the equation:

$$A \otimes B = A - (A \circledast B) = A \cap (A \circledast B)^c$$



a
b c d
e f g
h i j
k l

FIGURE 9.21 (a) Sequence of rotated structuring elements used for thinning. (b) Set A. (c) Result of thinning with the first element. (d)–(i) Results of thinning with the next seven elements (there was no change between the seventh and eighth elements). (j) Result of using the first element again (there were no changes for the next two elements). (k) Result after convergence. (l) Conversion to m-connectivity.

Thickening:

Thickening is a morphological dual of thinning and is defined as

$$A \odot B = A \cup (A \circledast B)$$

As in thinning, thickening can be defined as a sequential operation:

$$A \odot \{B\} = ((\ldots((A \odot B^1) \odot B^2) \ldots) \odot B^n)$$

the structuring elements used for thickening have the same form as in thinning, but with all 1's and 0's interchanged.

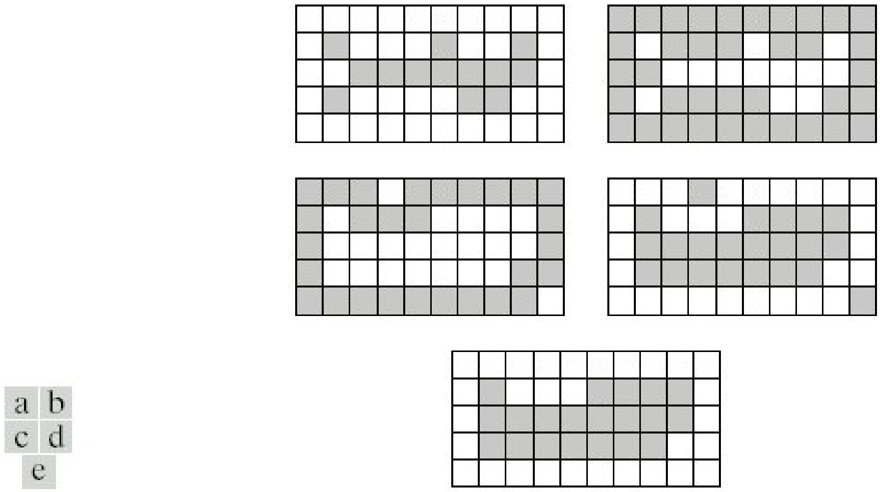


FIGURE 9.22 (a) Set A. (b) Complement of A. (c) Result of thinning the complement of A. (d) Thickened set obtained by complementing (c). (e) Final result, with no disconnected points.

Skeletons:

The skeleton of A is defined by terms of erosions and openings:

$$S(A) = \bigcup_{k=0}^{K} S_k(A)$$

with

$$S_k(A) = (A \ominus kB) - (A \ominus kB) \circ B$$

Where B is the structuring element and $(A \ominus kB)$ indicates k successive erosions of A:

$$(A \ominus kB) = (\ldots((A \ominus B) \ominus B) \ominus \ldots) \ominus B$$

k times, and K is the last iterative step before A erodes to an empty set in other words:

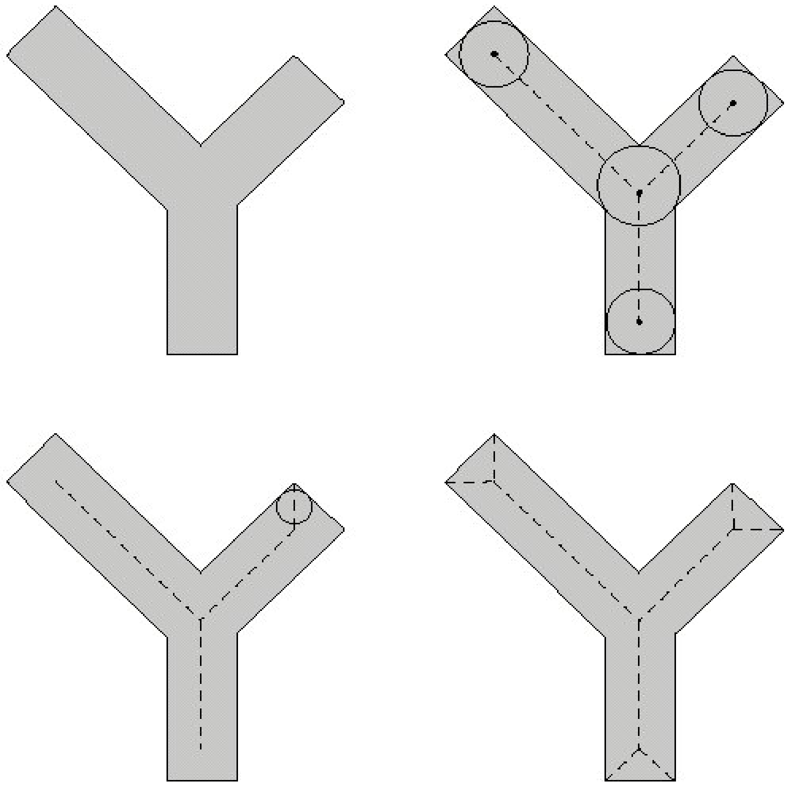
$$K = \max\{k \mid (A \ominus kB) \neq \emptyset\}$$

The S(A) can be obtained as the union of skeleton subsets $S_k(A)$. A can be also reconstructed from subsets $S_k(A)$ by using the equation

$$A = \bigcup_{k=0}^{K} (S_k(A) \oplus kB)$$

Where $(S_k(A) \oplus kB)$ denotes k successive dilations of $S_k(A)$ .that is:

$$(S_k(A) \oplus kB) = ((\ldots((S_k(A) \oplus B) \oplus B) \oplus \ldots) \oplus B$$



a b
c d

FIGURE 9.23 (a) Set A. (b) Various positions of maximum disks with centers on the skeleton of A. (c) Another maximum disk on a different segment of the skeleton of A. (d) Complete skeleton.

GRAY SCALE MORPHOLOGY

Gray Scale Images:

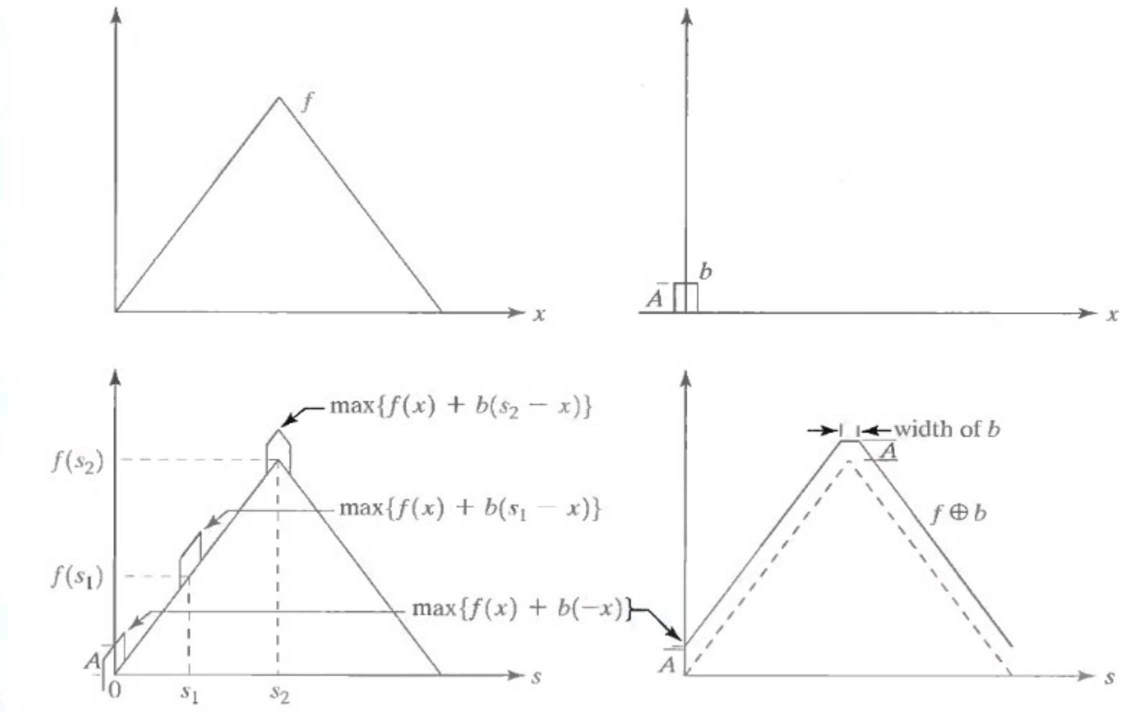
In gray scale images on the contrary to binary images we deal with digital image functions of the form f(x,y) as an input image and b(x,y) as a structuring element. (x,y) are integers from Z*Z that represent a coordinates in the image. f(x,y) and b(x,y) are functions that assign gray level value to each distinct pair of coordinates. For example the domain of gray values can be 0-255, whereas 0 – is black, 255- is white.

Dilation:

Equation for gray-scale dilation is

$$(f \oplus b)(s,t) = \max\{f(s-x,t-y)+b(x,y) \mid (s-x),(t-y) \in D_f, (x,y) \in D_b\}$$

Df and Db are domains of f and b. The condition that (s-x),(t-y) need to be in the domain of f and x,y in the domain of b, is analogous to the condition in the binary definition of dilation, where the two sets need to overlap by at least one element.



a b
c d

FIGURE 9.27 (a) A simple function. (b) Structuring element of height A. (c) Result of dilation for various positions of sliding b past f. (d) Complete result of dilation (shown solid).

We will illustrate the previous equation in terms of 1-D. and we will receive an equation for 1 variable:

$$(f \oplus b)(s) = \max\{f(s-x) + b(x) | (s-x) \in D_f \text{ and } x \in D_b\}$$

The requirements the (s-x) is in the domain of f and x is in the domain of b implies that f and b overlap by at least one element. Unlike the binary case, f, rather than the structuring element b is shifted. Conceptually f sliding by b is really not different than b sliding by f. The general effect of performing dilation on a gray scale image is twofold:

If all the values of the structuring elements are positive than the output image tends to be brighter than the input. Dark details either are reduced or eliminated, depending on how their values and shape relate to the structuring element used for dilation

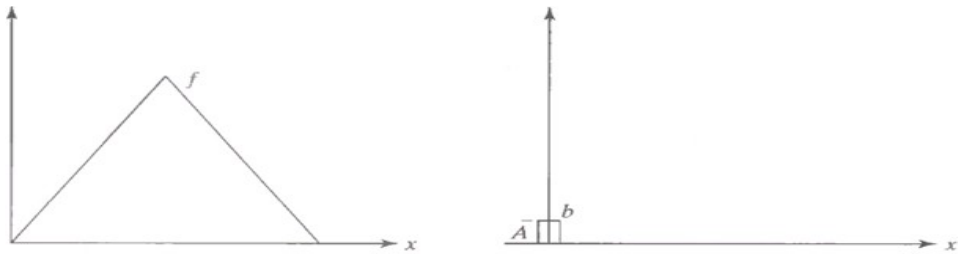
Erosion:

Gray-scale erosion is defined as:

$$(f \ominus b)(s,t) = \min\{f(s+x, t+y) - b(x,y) | (s+x), (t+y) \in D_f, (x,y) \in D_b\}$$

The condition that (s+x),(t+y) have to be in the domain of f, and x,y have to be in the domain of b, is completely analogous to the condition in the binary definition of erosion, where the structuring element has to be completely combined by the set being eroded. The same as in erosion we illustrate with 1-D function

$$(f \ominus b)(s) = \min\{f(s+x) - b(x) | (s+x) \in D_f \text{ and } x \in D_b\}$$



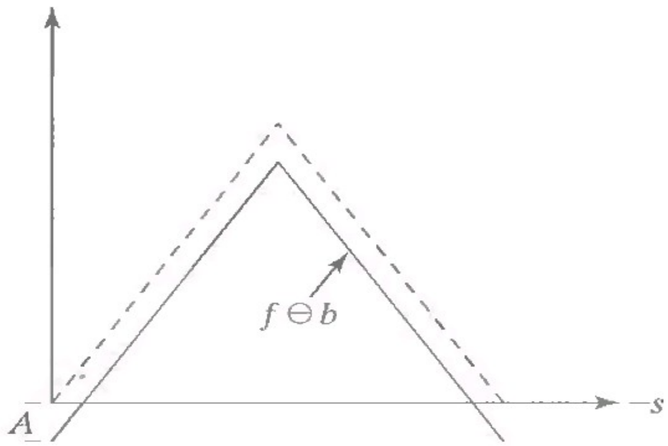


FIGURE 9.28 Erosion of the function shown in Fig. 9.27(a) by the structuring element shown in Fig. 9.27(b).

- General effect of performing an erosion in grayscale images:
  - If all elements of the structuring element are positive, the output image tends to be darker than the input image.
  - The effect of bright details in the input image that are smaller in area than the structuring element is reduced, with the degree of reduction being determined by the grayscale values surrounding by the bright detail and by shape and amplitude values of the structuring element itself.
- Similar to binary image grayscale erosion and dilation are duals with respect to function complementation and reflection.

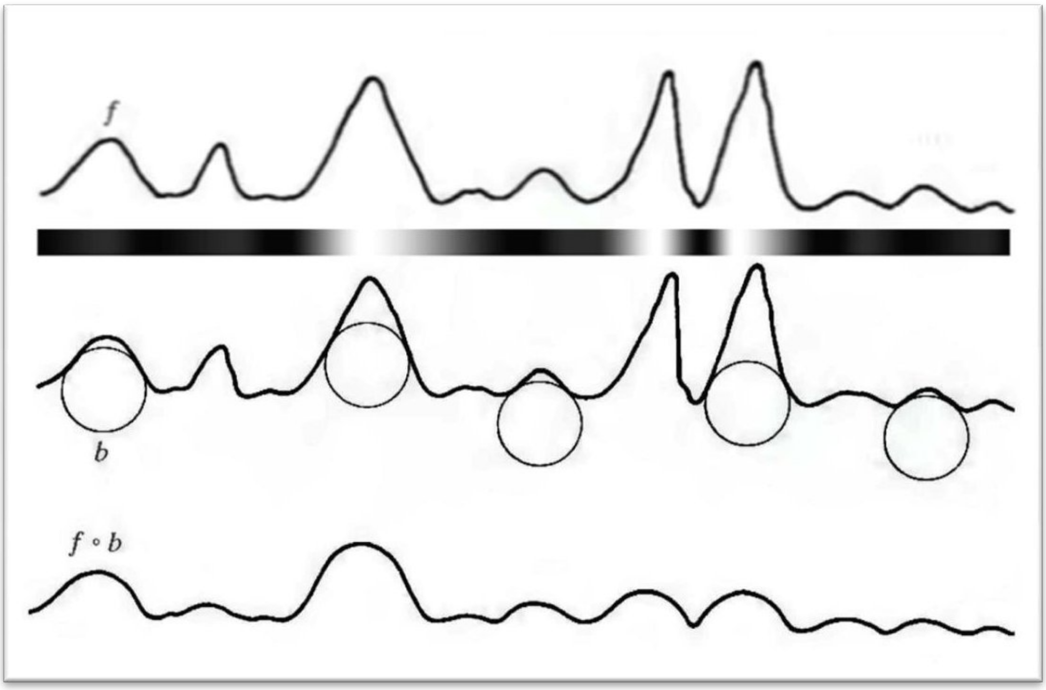
Opening:

In the opening of a gray-scale image, we remove small light details, while relatively undisturbed overall gray levels and larger bright features.

$$f \circ b = (f \ominus b) \oplus b.$$

The structuring element is rolled underside the surface of f. All the peaks that are narrow with respect to the diameter of the structuring element will be reduced in amplitude and sharpness. The initial erosion removes the details, but it also darkens the image. The subsequent dilation again increases the overall intensity of the image without reintroducing the details totally removed by erosion.

Opening a G-S picture is describable as pushing object B under the scan-line graph, while traversing the graph according the curvature of B



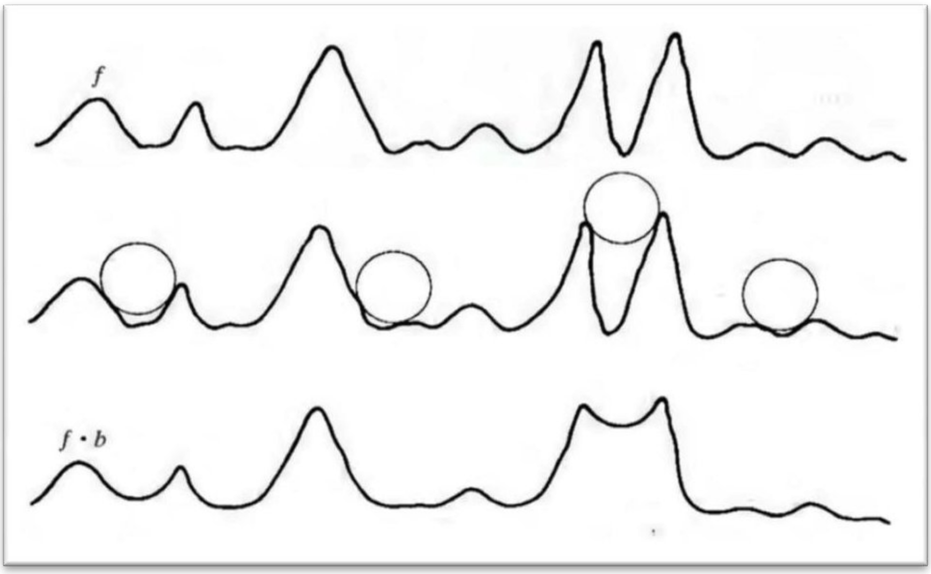
Closing:

In the closing of a gray-scale image, we remove small dark details, while relatively undisturbed overall gray levels and larger dark features

$$f \bullet b = (f \oplus b) \ominus b.$$

The structuring element is rolled on top of the surface of f. Peaks essentially are left in their original form (assume that their separation at the narrowest points exceeds the diameter of the structuring element). The initial dilation removes the dark details and brightens the image. The subsequent erosion darkens the image without reintroducing the details totally removed by dilation

Closing a G-S picture is describable as pushing object B on top of the scan-line graph, while traversing the graph according the curvature of B. The peaks are usually remains in their original form.



Applications of Gray-Scale Morphology:

<u>Morphological smoothing</u>

- Perform *opening* followed by a *closing*
- The net result of these two operations is to **remove or attenuate both bright and dark artifacts and noise**.

Morphological gradient

- *Dilation* and *erosion* are use to compute the *morphological gradient* of an image, denoted g:

$$g = (f \oplus b) - (f \ominus b)$$

- It uses to **highlights sharp gray-level transitions** in the input image.
- Obtained using symmetrical structuring elements tend to depend less on edge directionality.

Top-hat and Bottom-hat transformation:

- Combining image subtraction with opening and closing results are referred as top-hat and bottom-hat transformations.
- The top-hat transformation of gray scale image f is defined as

$$T_{\text{hat}}(f) = f - (f \circ b)$$

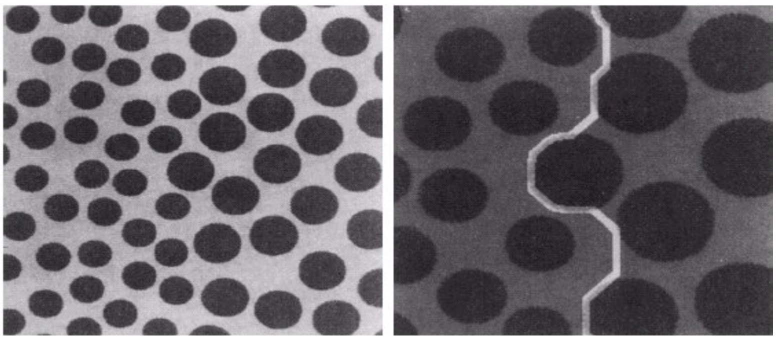
- The bottom-hat transformation of gray scale image f is defined as

$$B_{\text{hat}}(f) = (f \bullet b) - f$$

- The top-hat transform is used for light objects on a dark background and the bottom-hat transform is used for the converse.

Textural segmentation:

- The objective is **to find the boundary between different image regions based on their textural content**.
- *Close* the input image by using successively larger *structuring elements*.
- Then, single *opening* is preformed ,and finally a simple *threshold* that yields the boundary between the textural regions.

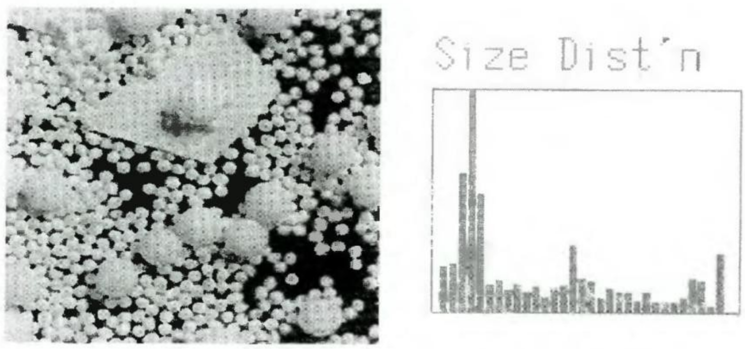


a b

FIGURE 9.35 (a) Original image. (b) Image showing boundary between regions of different texture. (Courtesy of Mr. A. Morris, Leica Cambridge, Ltd.)

Granulometry:

- *Granulometry* is a field that deals principally with **Determining the size distribution of particles in an image.**
- Because the particles are lighter than the background, we can use a morphological approach to determine size distribution. To construct at the end a ***histogram*** of it.
- Based on the idea that *opening* operations of particular size have the most effect on regions of the input image that contain particles of similar size.
- This type of processing is **useful for describing regions with a predominant particle-like character.**



a b
FIGURE 9.36 (a) Original image consisting of overlapping particles; (b) size distribution. (Courtesy of Mr. A. Morris, Leica Cambridge, Ltd.)

Image Segmentation

Image segmentation divides an image into regions that are connected and have some similarity within the region and some difference between adjacent regions. The goal is usually to find individual objects in an image. For the most part there are fundamentally two kinds of approaches to segmentation: discontinuity and similarity.

- Similarity may be due to pixel intensity, color or texture.
- Differences are sudden changes (discontinuities) in any of these, but especially sudden changes in intensity along a boundary line, which is called an edge.

Detection of Discontinuities:

There are three kinds of discontinuities of intensity: points, lines and edges. The most common way to look for discontinuities is to scan a small mask over the image. The mask determines which kind of discontinuity to look for.

$$R = w_1z_1 + w_2z_2 + \dots + w_9z_9 = \sum_{i=1}^{9} w_i z_i$$

FIGURE 10.1 A general 3 × 3 mask.

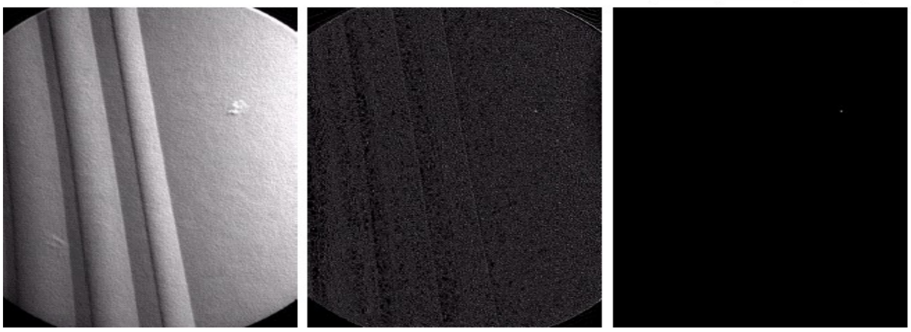
w_1	w_2	w_3
w_4	w_5	w_6
w_7	w_8	w_9

Point Detection: Whose gray value is significantly different from its background

$$|R| \geq T$$

where T: a nonnegative threshold

−1	−1	−1
−1	8	−1
−1	−1	−1



a
b c d

FIGURE 10.2 (a) Point detection mask. (b) X-ray image of a turbine blade with a porosity. (c) Result of point detection. (d) Result of using Eq. (10.1-2). (Original image courtesy of X-TEK Systems Ltd.)

Line Detection:

- Only slightly more common than point detection is to find a one pixel wide line in an image.
- For digital images the only three point straight lines are only horizontal, vertical, or diagonal (+ or –45).

FIGURE 10.3 Line masks.

Horizontal

−1	−1	−1
2	2	2
−1	−1	−1

+45°

−1	−1	2
−1	2	−1
2	−1	−1

Vertical

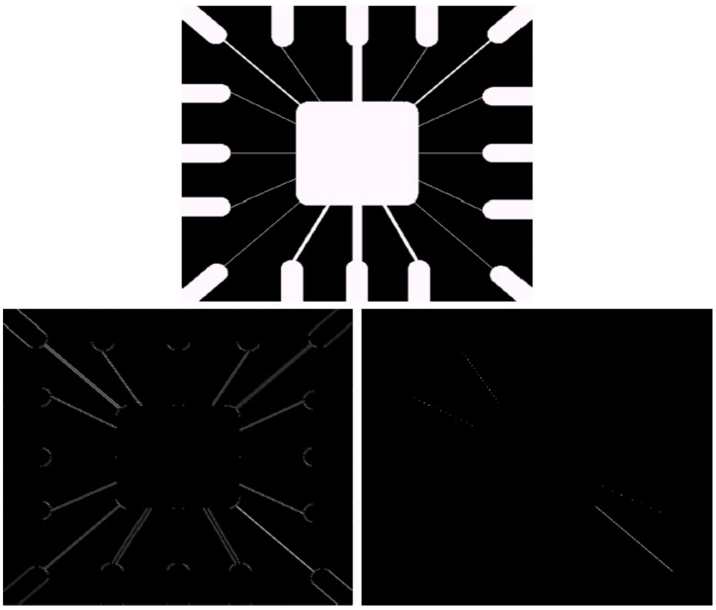
−1	2	−1
−1	2	−1
−1	2	−1

−45°

2	−1	−1
−1	2	−1
−1	−1	2

Preferred direction is weighted by with a larger coefficient

- The coefficients in each mask sum to zero response of constant gray level areas
- Compare values of individual masks (run all masks) or run only the mask of specified direction

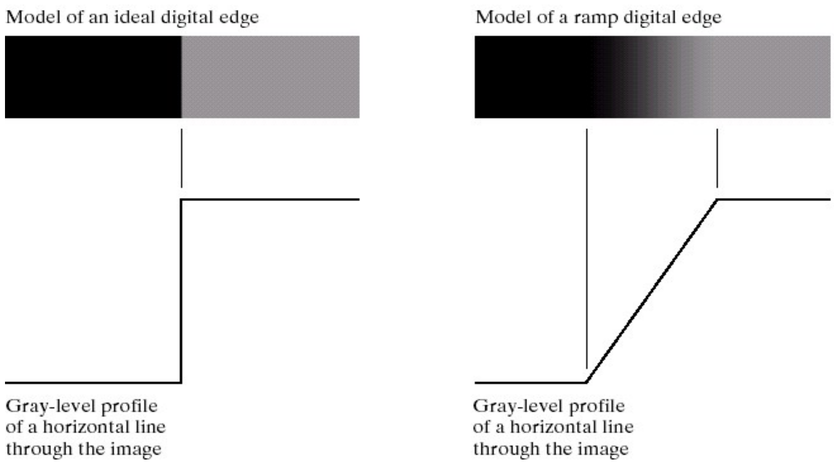


a
b c
FIGURE 10.4 Illustration of line detection. (a) Binary wire-bond mask. (b) Absolute value of result after processing with −45° line detector. (c) Result of thresholding image (b).

Edge Detection:

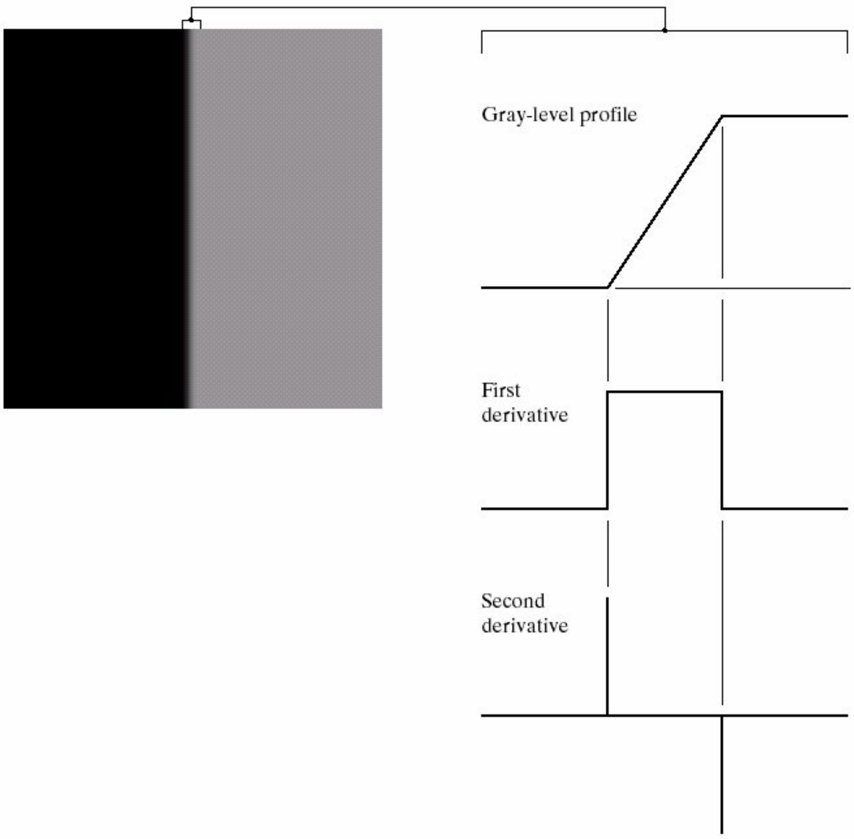
Edge is a set of connected pixels that lie on the boundary between two regions

- 'Local' concept in contrast to 'more global' boundary concept
- To be measured by grey-level transitions
- Ideal and blurred edges



a b

FIGURE 10.5 (a) Model of an ideal digital edge. (b) Model of a ramp edge. The slope of the ramp is proportional to the degree of blurring in the edge.



a b

FIGURE 10.6 (a) Two regions separated by a vertical edge. (b) Detail near the edge, showing a gray-level profile, and the first and second derivatives of the profile.

- First derivative can be used to detect the presence of an edge (if a point is on a ramp).
- The sign of the second derivative can be used to determine whether an edge pixel lie on the dark or light side of an edge
- Second derivative produces two value per edge
- Zero crossing near the edge midpoint
- Non-horizontal edges – define a profile perpendicular to the edge direction

Edges in the presense of noise

- Derivatives are sensitive to (even fairly little) noise
- Consider image smoothing prior to the use of derivatives

Edge definition again

- Edge point – whose first derivative is above a pre-specified threshold
- Edge – connected edge points
- Derivatives are computed through gradients (1st) and Laplacians (2nd)

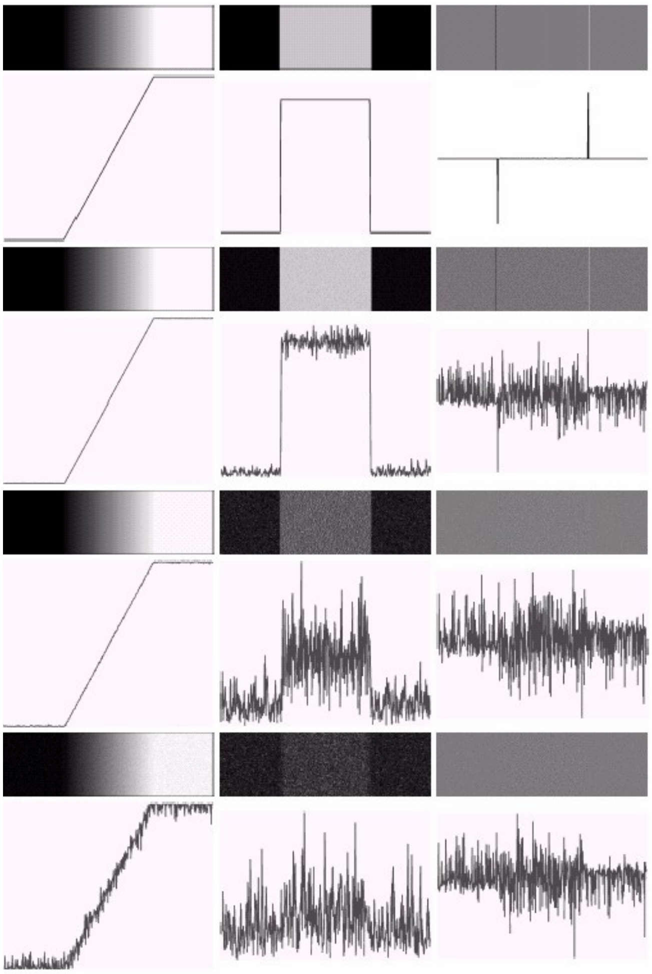


FIGURE 10.7 First column: images and gray-level profiles of a ramp edge corrupted by random Gaussian noise of mean 0 and σ = 0.0, 0.1, 1.0, and 10.0, respectively. Second column: first-derivative images and gray-level profiles. Third column: second-derivative images and gray-level profiles.

a
b
c
d

Edge Detection Gradient Operators:

Gradient

– **Vector** pointing to the direction of maximum rate of change of *f* at coordinates (*x*,*y*)

$$\nabla \mathbf{f} = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix}$$

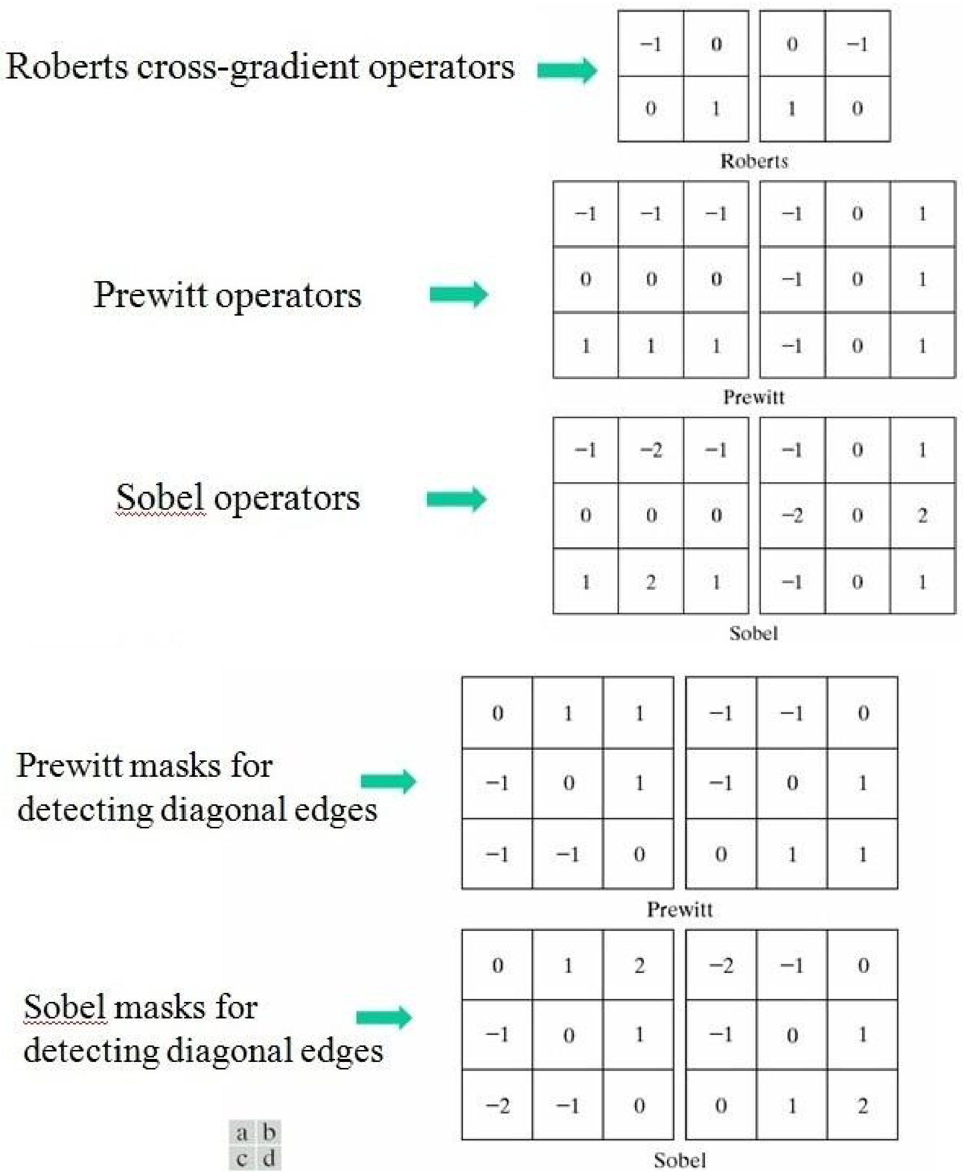
– **Magnitude**: gives the quantity of the increase (some times referred to as *gradient* too)

$$\nabla f = \text{mag}(\nabla \mathbf{f}) = \left[G_x^2 + G_y^2\right]^{\frac{1}{2}}$$

– **Direction**: perpendicular to the direction of the edge at (x,y)

$$\alpha(x, y) = \tan^{-1}\left(\frac{G_x}{G_y}\right)$$

Partial derivatives computed through 2x2 or 3x3 masks. Sobel operators introduce some smoothing and give more importance to the center point



Roberts

−1	0	0	−1
0	1	1	0

Prewitt

−1	−1	−1	−1	0	1
0	0	0	−1	0	1
1	1	1	−1	0	1

Sobel

−1	−2	−1	−1	0	1
0	0	0	−2	0	2
1	2	1	−1	0	1

Prewitt

0	1	1	−1	−1	0
−1	0	1	−1	0	1
−1	−1	0	0	1	1

Sobel

0	1	2	−2	−1	0
−1	0	1	−1	0	1
−2	−1	0	0	1	2

FIGURE 10.9 Prewitt and Sobel masks for detecting diagonal edges.

Laplacian

– Second-order derivative of a 2-D function

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

– Digital approximations by proper masks

FIGURE 10.13 Laplacian masks used to implement Eqs. (10.1-14) and (10.1-15), respectively.

0	−1	0
−1	4	−1
0	−1	0

−1	−1	−1
−1	8	−1
−1	−1	−1

- Complementary use for edge detection
- Cons: Laplacian is very sensible to noise; double edges
- Pros: Dark or light side of the edge; zero crossings are of better use
- Laplacian of Gaussian (LoG): preliminary smoothing to find edges through zero crossings.
- Consider the function:

A Gaussian function

$$h(r) = -e^{-\frac{r^2}{2\sigma^2}} \quad \text{where } r^2 = x^2 + y^2$$

and σ : the standard deviation

The Laplacian of h is

$$\nabla^2 h(r) = -\left[\frac{r^2 - \sigma^2}{\sigma^4}\right] e^{-\frac{r^2}{2\sigma^2}}$$

The Laplacian of a Gaussian (LoG)

The Laplacian of a Gaussian sometimes is called the Mexican hat function. It also can be computed by smoothing the image with the Gaussian smoothing mask, followed by application of the Laplacian mask.

Edge Linking and Boundary Detection

Local Processing:

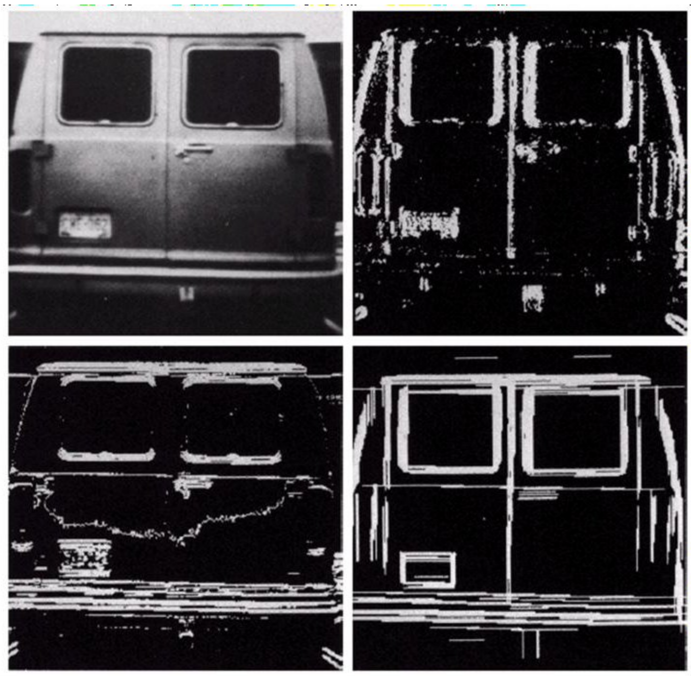
- Two properties of edge points are useful for edge linking:
  - the strength (or magnitude) of the detected edge points
  - their directions (determined from gradient directions)
- This is usually done in local neighborhoods.
- Adjacent edge points with similar magnitude and direction are linked.

For example, an edge pixel with coordinates (x_0,y_0) in a predefined neighborhood of (x,y) is similar to the pixel at (x,y) if

Strength of the gradient vector response $|\nabla f(x,y) - \nabla f(x_0,y_0)| \le E$, E: a nonnegative threshold

Gradient vector direction $|\alpha(x,y) - \alpha(x_0,y_0)| \le A$, A: a nonegative angle threshold

Both magnitude and angle criteria should be satisfied



a b
c d

FIGURE 10.16 (a) Input image. (b) G_y component of the gradient. (c) G_x component of the gradient. (d) Result of edge linking. (Courtesy of Perceptics Corporation.)

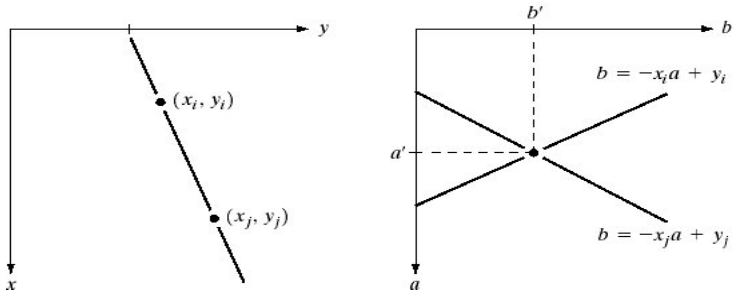
In this example, we can find the license plate candidate after edge linking process.

Global Processing via the Hough Transform:

Hough transform: a way of finding edge points in an image that lie along a straight line.

Example: *xy*-plane v.s. *ab*-plane (parameter space)

$$y_i = ax_i + b$$



a b

FIGURE 10.17 (a) *xy*-plane. (b) Parameter space.

The Hough transform consists of finding all pairs of values of () and ✕ which satisfy the equations that pass through (*x*,*y*). These are accumulated in what is basically a 2-dimensional histogram. When plotted these pairs of () and ✕ will look like a sine wave. The process is repeated for all appropriate (*x*,*y*) locations.

Thresholding:

The range of intensity levels covered by objects of interest is different from the background.

$$g(x, y) = \begin{cases} 1 & \text{if } f(x, y) > T \\ 0 & \text{if } f(x, y) \le T \end{cases}$$

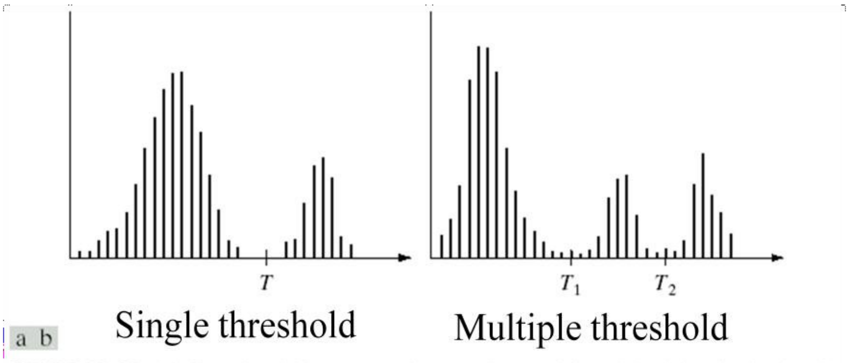


FIGURE 10.26 (a) Gray-level histograms that can be partitioned by (a) a single threshold, and (b) multiple thresholds.

Illumination:

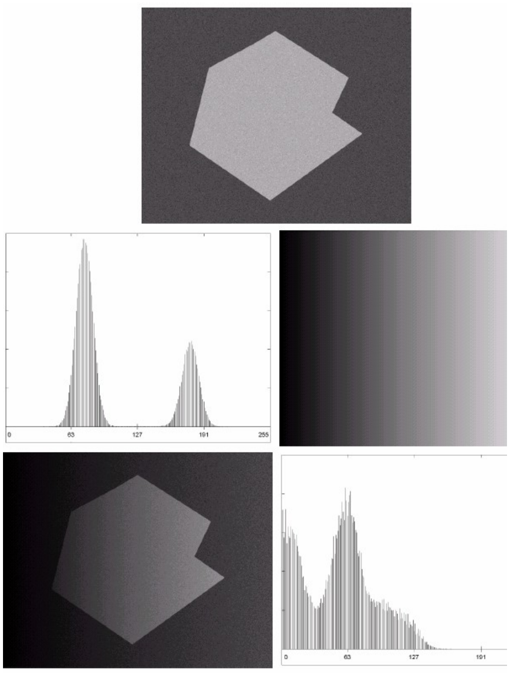


Fig. a) Computer Generated Reflectance Function b) Histogram of Reflectance Function c) Computer Generated Illumination Function d) Product of a and c e) Histogram of Product Image.

- Image is a product of reflectance and illuminance
- Reflection nature of objects and backaground
- Poor (nonlinear) illumination could impede the segmentation
- The final histogram is a result of convolution of the histogram of the log reflectance and log illuminance functions
- Normalization if the illuminance function is known

Basic Global Thresholding:

Threshold midway between maximum and minimum gray levels

– Appropriate for industrial inspection applications with controllable illumination

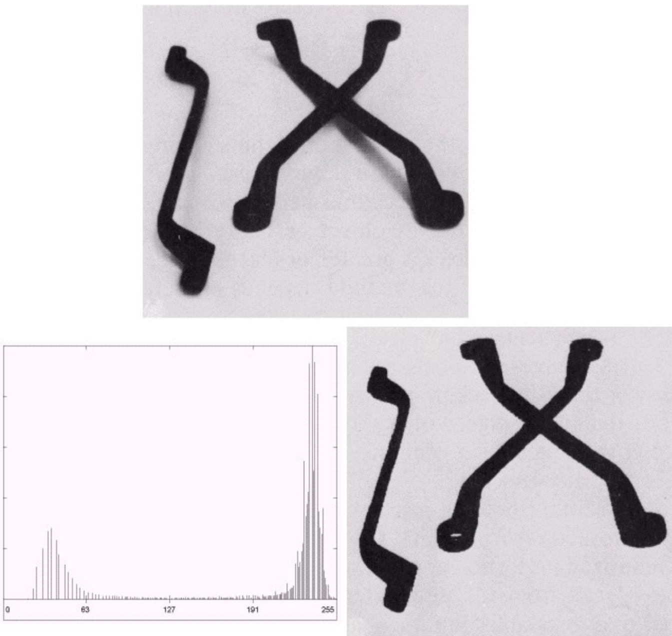
– Automatic algorithm

• Segment with initial T into regions $G1$ and $G2$

• Compute the average gray level $m1$ and $m2$

• Compute new T=0.5($m1$+$m2$)

• Repeat until reach an acceptably small change of T in successive iterations



a
b c

FIGURE 10.28 (a) Original image. (b) Image histogram. (c) Result of global thresholding with T midway between the maximum and minimum gray levels.

Region Based Segmentation:

Edges and thresholds sometimes do not give good results for segmentation. Region-based segmentation is based on the connectivity of similar pixels in a region.

- Each region must be uniform.
- Connectivity of the pixels within the region is very important.

There are two main approaches to region-based segmentation: region growing and region splitting.

Region Growing:

- Let R represent the entire image region.
- Segmentation is a process that partitions R into subregions, $R_1, R_2, \ldots, R_n$, such that

(a) $\bigcup_{i=1}^{n} R_i = R$

(b) R_i is a connected region, $i = 1,2,\ldots,n$

(c) $R_i \cap R_j = \emptyset$ for all i and $j, i \neq j$

(d) $P(R_i) =$ TRUE for $i = 1,2,\ldots,n$

(e) $P(R_i \cup R_j) =$ FALSE for any adjacent regions R_i and R_j

where $P(R_k)$: a logical predicate defined over the points in set R_k

For example: $P(R_k)$=TRUE if all pixels in R_k have the same gray level.

a b
c d

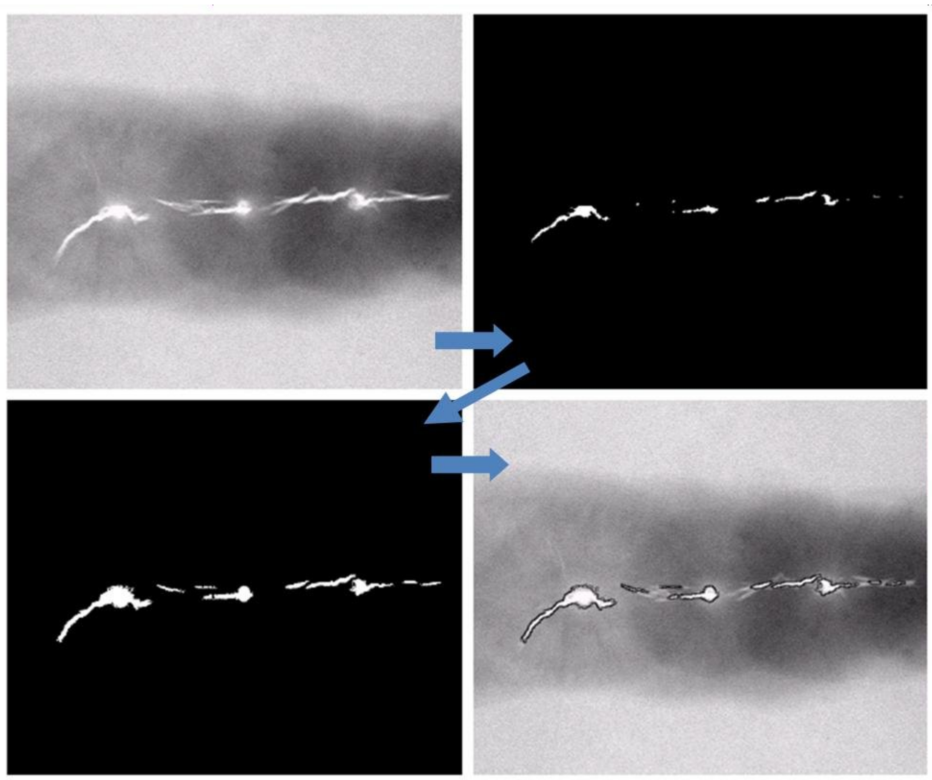


FIGURE 10.40 (a) Image showing defective welds. (b) Seed points. (c) Result of region growing. (d) Boundaries of segmented defective welds (in black). (Original image courtesy of X-TEK Systems, Ltd.).

Fig. 10.41 shows the histogram of Fig. 10.40 (a). It is difficult to segment the defects by thresholding methods. (Applying region growing methods are better in this case.)

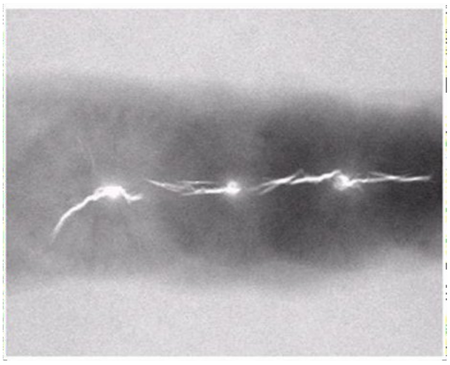


Figure 10.40(a)

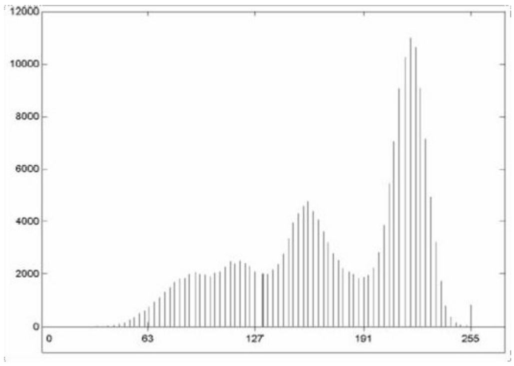


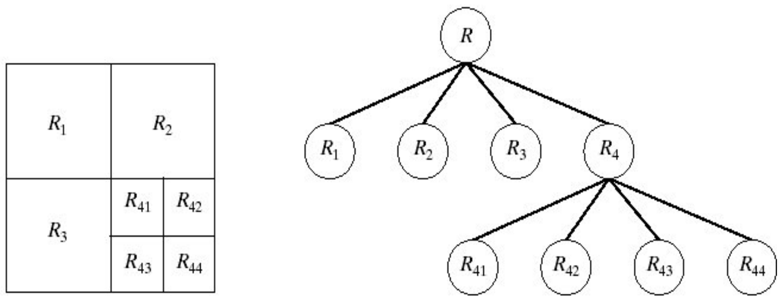
Figure 10.41

Region Splitting:

- First there is a large region (possible the entire image).
- Then a predicate (measurement) is used to determine if the region is uniform.
- If not, then the method requires that the region be split into two regions.
- Then each of these two regions is independently tested by the predicate (measurement).
- This procedure continues until all resulting regions are uniform.
- The main problem with region splitting is determining where to split a region.
- One method to divide a region is to use a quadtree structure.
- Quadtree: a tree in which nodes have exactly four descendants.

a b

FIGURE 10.42
(a) Partitioned image.
(b) Corresponding quadtree.



Segmentation by Morphological Watersheds:

- The concept of watersheds is based on visualizing an image in three dimensions: two spatial coordinates versus gray levels.
- In such a topographic interpretation, we consider three types of points:

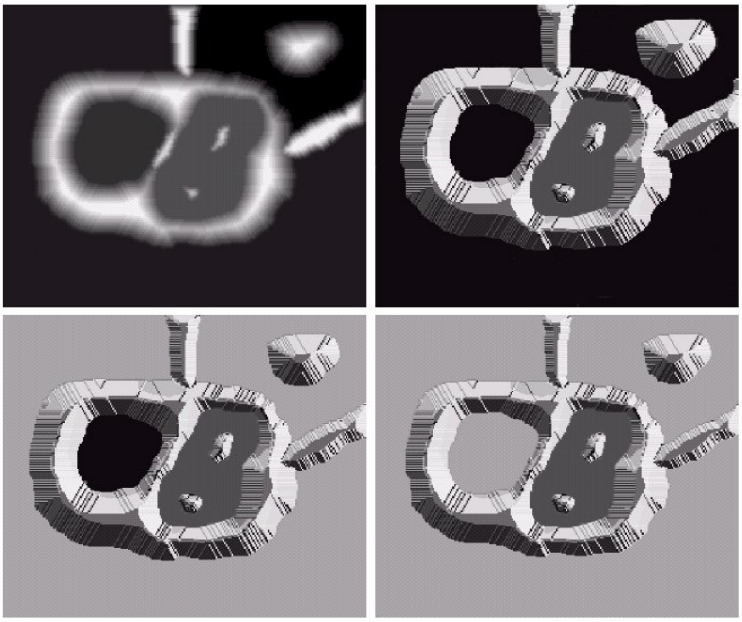
  (a) Points belonging to a regional minimum

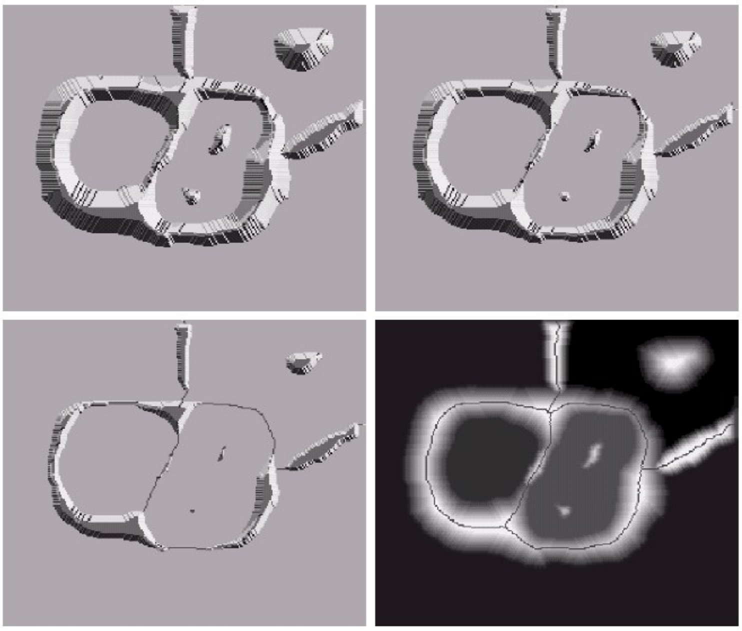
  (b) Points at which a drop of water would fall with certainty to a single minimum

  (c) Points at which water would be equally likely to fall to more than one such minimum.
- The principal objective of segmentation algorithms based on these concepts is to find the watershed lines.

a b
c d

FIGURE 10.44 (a) Original image. (b) Topographic view. (c)–(d) Two stages of flooding.





e f
g h

FIGURE 10.44 *(Continued)* (e) Result of further flooding. (f) Beginning of merging of water from two catchment basins (a short dam was built between them). (g) Longer dams. (h) Final watershed (segmentation) lines. (Courtesy of Dr. S. Beucher, CMM/Ecole des Mines de Paris.)

PREVIOUS QUESTIONS

1. With necessary figures, explain the opening and closing operations.
2. Explain the following morphological algorithms i) Boundary extraction ii) Hole filling.
3. Explain the following morphological algorithms i) Thinning ii) Thickening
4. What is Hit-or-Miss transformation? Explain.
5. Discuss about Grey-scale morphology.
6. Write a short notes on Geometric Transformation
7. Explain about edge detection using gradient operator.
8. What is meant by edge linking? Explain edge linking using local processing
9. Explain edge linking using Hough transform.
10. Describe Watershed segmentation Algorithm
11. Discuss about region based segmentation.
12. Explain the concept of Thresholding in image segmentation and discuss its merits and limitations.