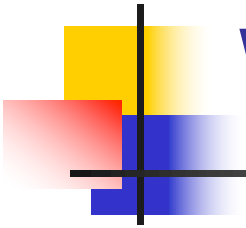
Εισαγωγή στην Κατάτμηση Εικόνας

Segmentation

- Segmentation is to partition an image into its constituent regions or objects. Segmentation should stop when the objects of interest in an application have been isolated.

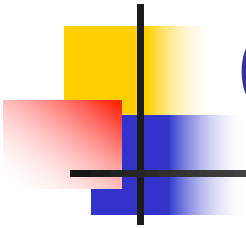
- Segmentation is an initial and vital step in pattern recognition, i.e., a series of processes aimed at overall image understanding.



Why Segment?

- Image simplification: an image may contain millions of pixels but only few regions
- Similar pixels within a region tend to belong to the same class of object
- Can provide the shape of an object
- Regions are a higher level description of image content
- To generate input for higher classification algorithms, e.g. Bayesian classifiers or Neural Networks
- Semantic description of image content allows for efficient approximate matching in Content Based Image Retrieval systems

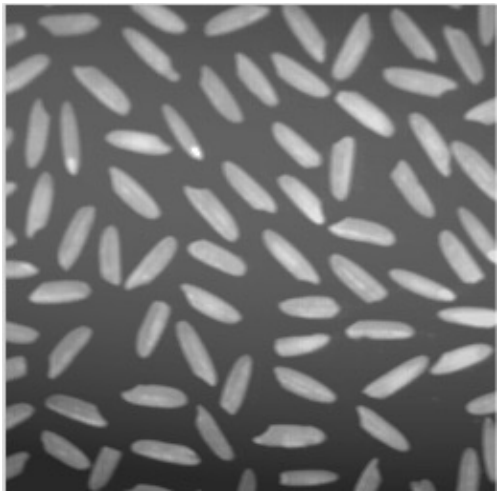
Common ways to segment

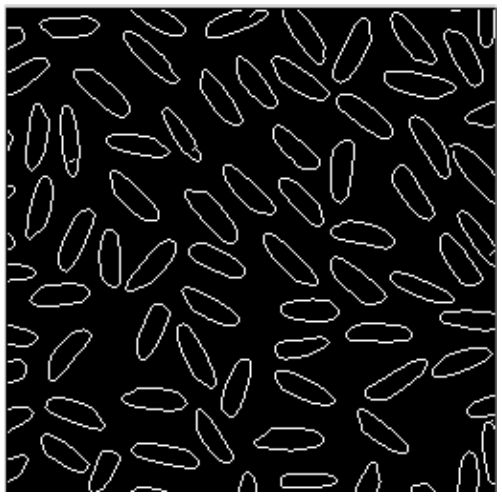


- Edge (or discontinuity) based methods:
  - Link up edges to form hopefully meaningful object boundaries
- Region based methods:
  - Region growing
  - Region splitting and merging
- Methods based on thresholding
- Clustering methods

Edge based segmentation

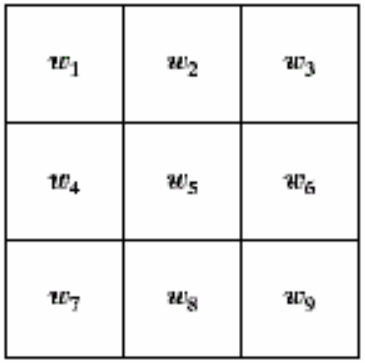
- This approach identifies the strong edges in an image.
- Hopefully these correspond to object boundaries.
- Edges are extended or deleted so as to produce closed boundaries.
- Termed "Edge Relaxation".
- Only good for simple images.
- Shape can then be used for recognition.



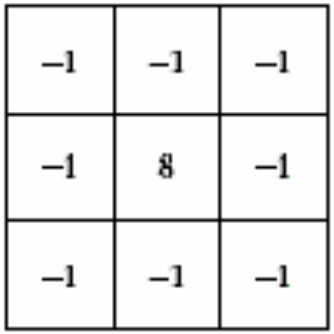


Detection of Discontinuities

- detect the three basic types of gray-level discontinuities
  - points , lines , edges
- the common way is to run a mask through the image



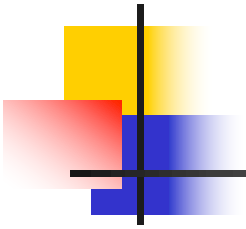
Point Detection



- a point has been detected at the location on which the mask is centered if

$$|R| \geq T$$

- where
  - T is a nonnegative threshold
  - R is the sum of products of the coefficients with the gray levels contained in the region encompassed by the mark.



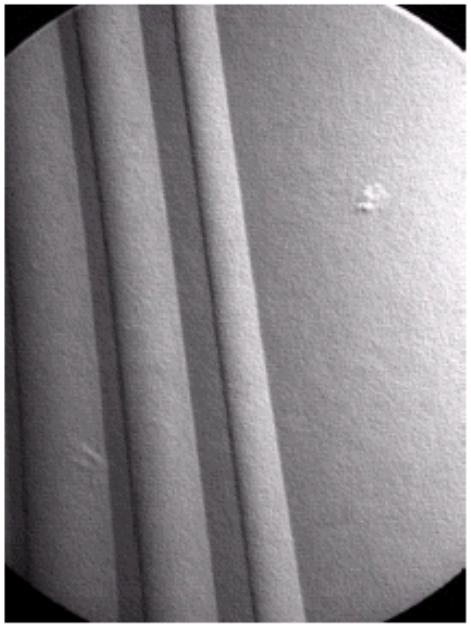
Point Detection

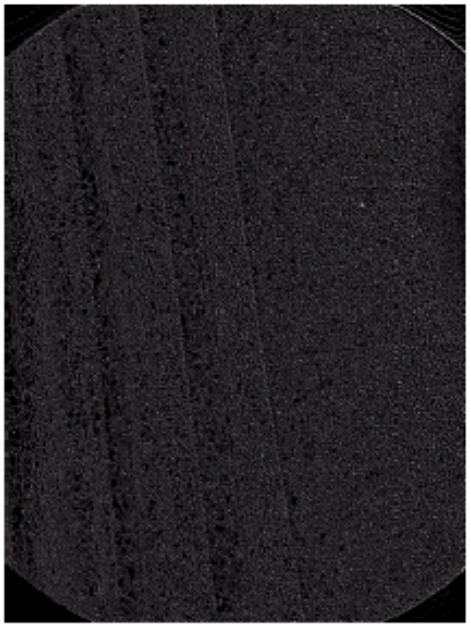
- Note that the mask is the same as the Laplacian mask
- The only differences that are considered of interest are those large enough (determined by T) to be considered isolated points.

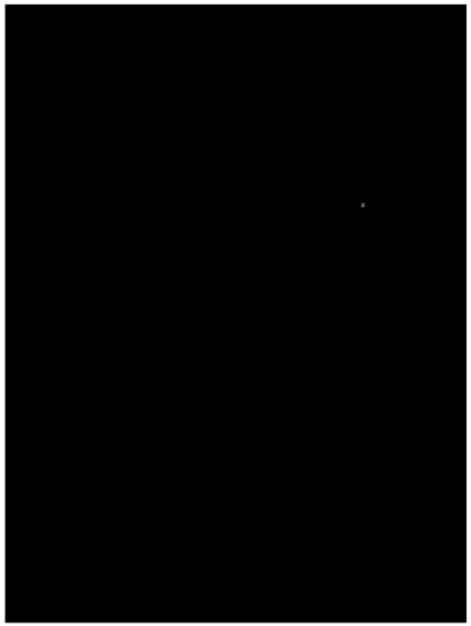
$$|R| \geq T$$

Example

−1	−1	−1
−1	8	−1
−1	−1	−1





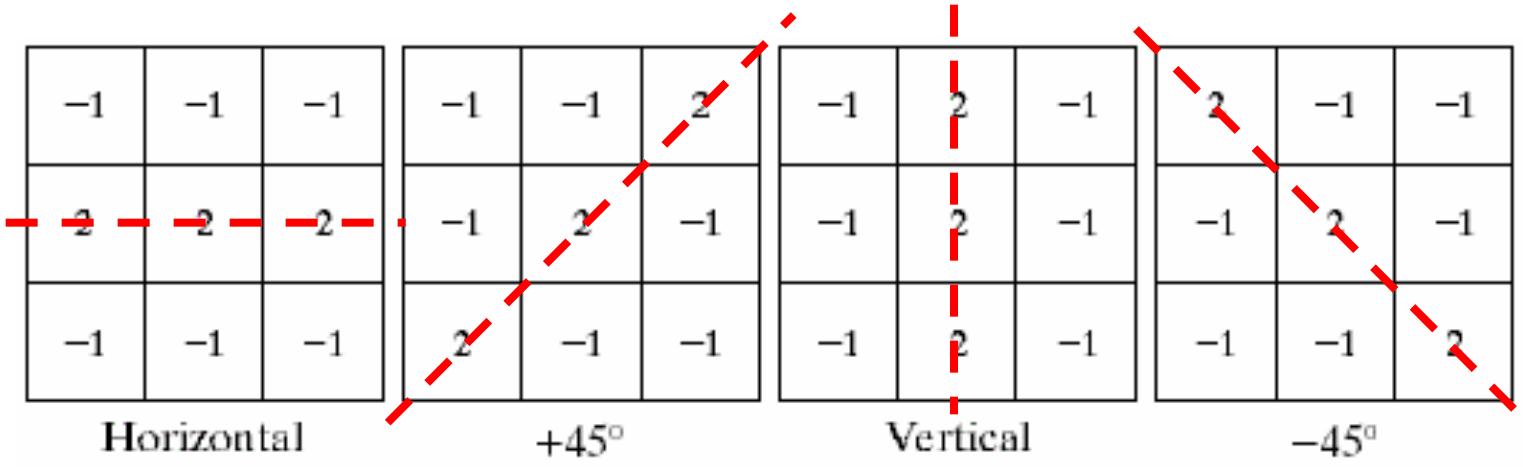


a
b c d

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

Line masks



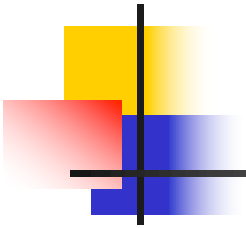
- Horizontal mask will have max response when a line of 1-pixel width passes through the middle row of the mask and when the background is constant. Similarly for the other masks.
- Note: the preferred direction of each mask is weighted with a larger coefficient (i.e.,2) than other possible directions.
- To detect dark lines we need $R<T$ and $T<0$; to detect bright lines we set $R>T>0$; to detect all lines, set $|R|>T>0$.

Line Detection

- Apply every mask on the image
- let R_1, R_2, R_3, R_4 denote the response of the horizontal, +45 degrees, vertical and -45 degrees masks, respectively.
- if, at a certain point in the image

$$|R_i| > |R_j|,$$

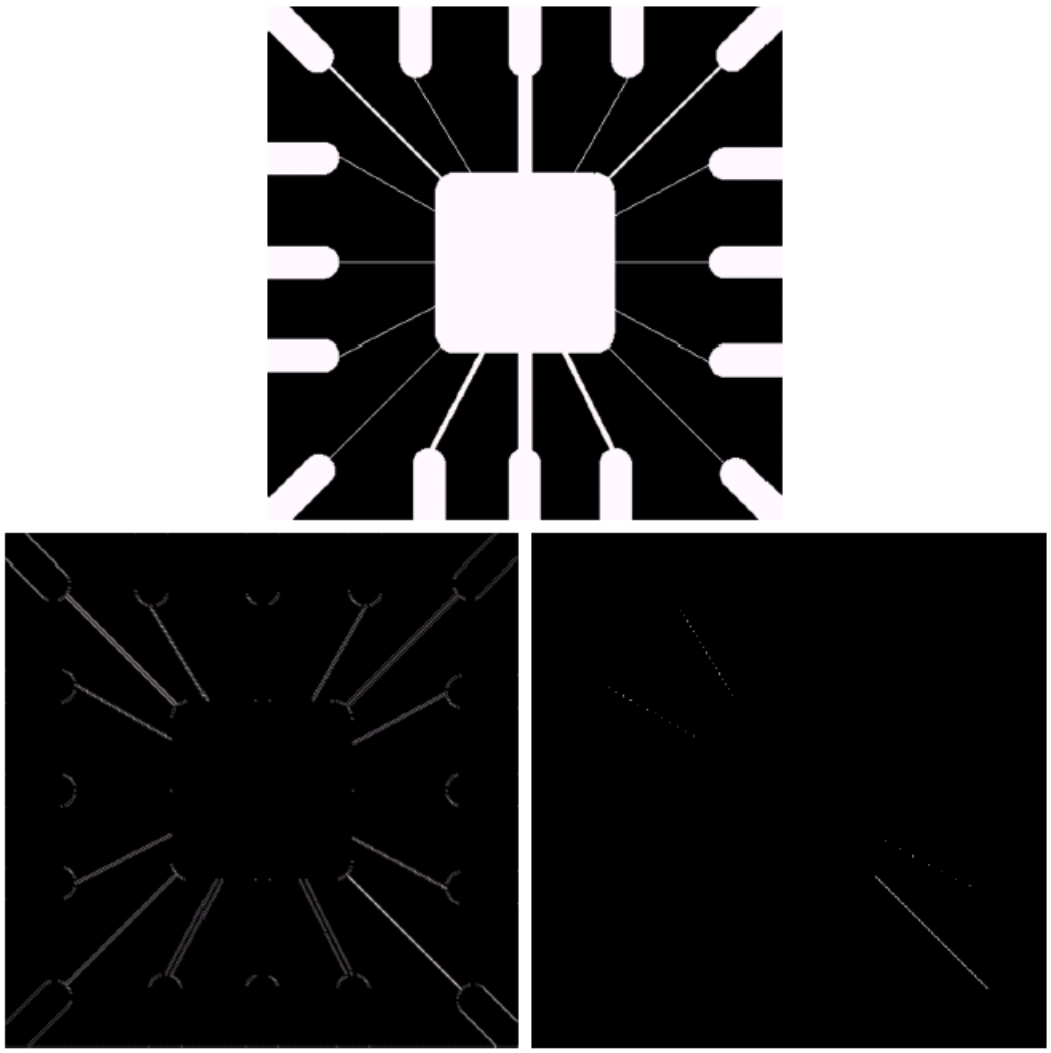
for all $j \neq i$, that point is said to be more likely associated with a line in the direction of mask i.



Line Detection

- Alternatively, if we are interested in detecting all lines in an image in the direction defined by a given mask, we simply run the mask through the image and threshold the absolute value of the result.
- The points that are left are the strongest responses, which, for lines one pixel thick, correspond closest to the direction defined by the mask.

Example



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

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

a
b c
d e
f g

FIGURE 10.8
A 3 × 3 region of an image (the z's are gray-level values) and various masks used to compute the gradient at point labeled z_5.

−1	0	0	−1
0	1	1	0

Roberts

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

Prewitt

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

Sobel

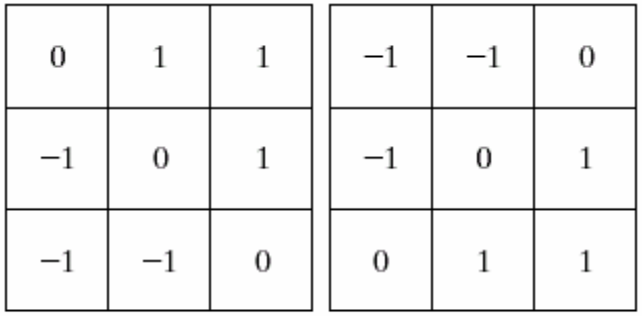
$$\nabla f = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix}$$

$$G = |G_x| + |G_y| \text{ or}$$

$$G = \sqrt{G_x^2 + G_y^2}$$

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

Detection of diagonal edges



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

Prewitt

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

Sobel

a b
c d

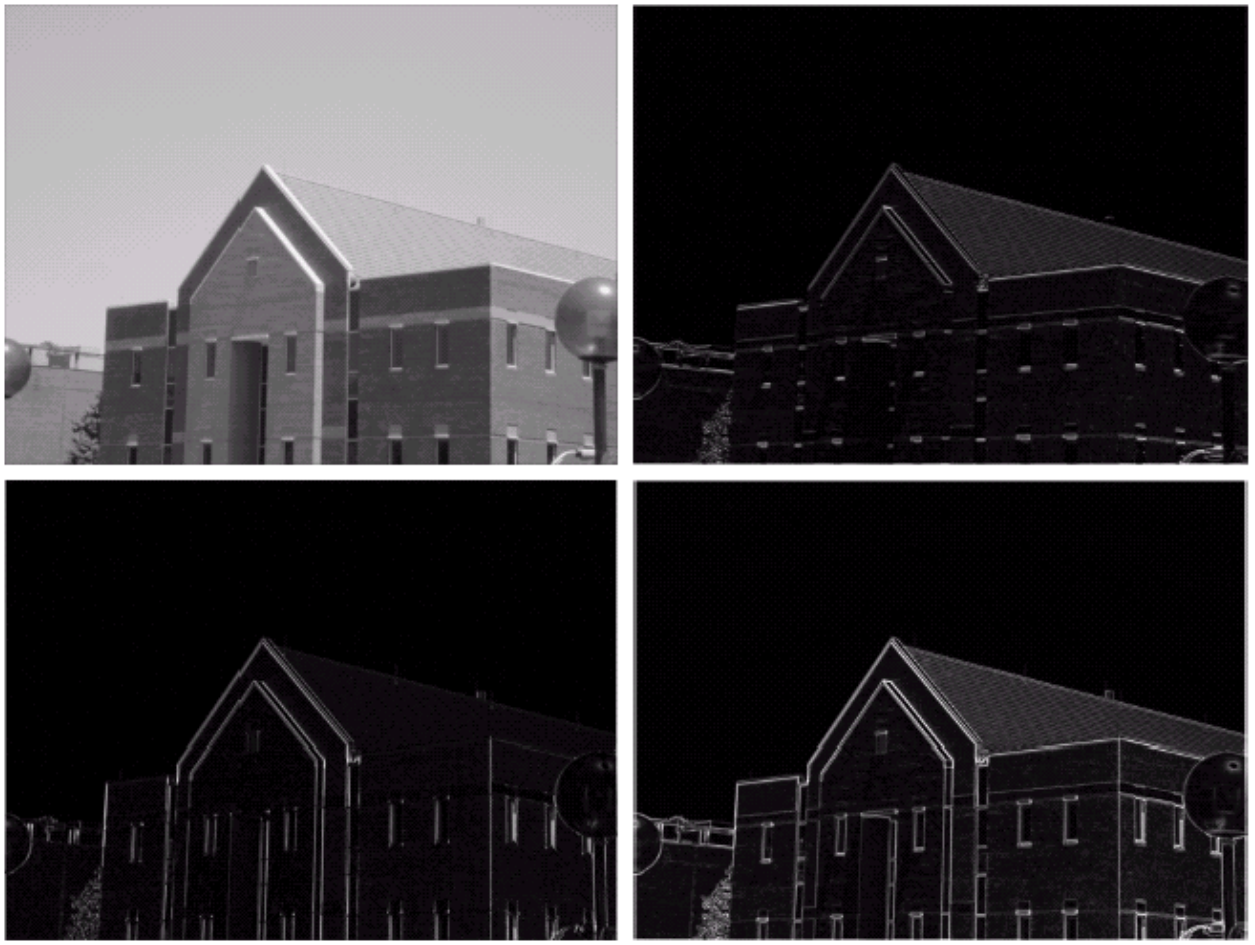
FIGURE 10.9 Prewitt and Sobel masks for detecting diagonal edges.

Edge Detection: Example

a b
c d

FIGURE 10.10
(a) Original image. (b) $|G_x|$, component of the gradient in the x-direction. (c) $|G_y|$, component in the y-direction. (d) Gradient image, $|G_x| + |G_y|$.

Edge Detection: Example



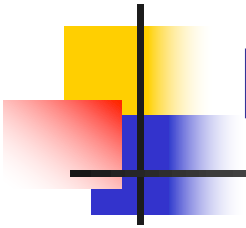
a b
c d

FIGURE 10.11
Same sequence as in Fig. 10.10, but with the original image smoothed with a 5×5 averaging filter.

Edge Detection: Example

a b

FIGURE 10.12 Diagonal edge detection. (a) Result of using the mask in Fig. 10.9(c). (b) Result of using the mask in Fig. 10.9(d). The input in both cases was Fig. 10.11(a).



Edge Linking & Boundary Detection

- The edge detection algorithms are followed by linking procedures to assemble edge pixels into meaningful edges.
- Basic approaches
  - Local Processing
  - Global Processing via the Hough Transform

Βιβλιογραφία

Οι παρούσες διαφάνειες έχουν δημιουργηθεί από τον Καθηγητή κ. Ν. Βασιλά για το μάθημα «Επεξεργασία Εικόνας», ακαδημαϊκό έτος 2017-2018.