Image segmentation

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.

- 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.

- The intuitive notion about an edge in a digital image is that it occurs at the boundary between two pixels when the respective grey level values of these adjacent pixels are significantly different.

- Edges are generally found because of changes in some physical and surface properties such as illumination, geometry (orientation, depth) and reflectance.

- Since image edges characterize object boundaries, these are useful for segmentation and identification of objects in a scene.

- Edge detection plays an important role in interpretation of digital images

- Segmentation of nontrivial (small) images is one of the most difficult tasks in image processing.
- Segmentation accuracy determines the eventual success or failure of computerized analysis procedures.
- Accurate care should be taken to improve the probability of accurate segmentation.
    - Eg: Infrared imaging by military to detect objects in motion.

Fundamentals

A more formal definition – Let R represent the entire image. Segmentation is a process that divides R into n subregions $R_1, R_2, \ldots, R_n$ such that:

1. $\bigcup_{i=1}^{n} R_i = R$.
2. R_i is a connected set for each $i = 1, 2, \ldots, n$.
3. $R_i \cap R_j = \emptyset$ for all i and j, $j \neq i$.
4. $Q(R_i) = TRUE$ for each $i = 1, 2, \ldots, n$.
5. $Q(R_i \cup R_j) = FALSE$ for any adjacent regions R_i and R_j.

Here $Q(R_k)$ is a predicate that indicates some property over the region.

1) Indicates segmentation must be complete. ie every pixel must be in region
2) Point in a region should be connected in some specified sense.(4 or 8 connected)
3) Regions must be disjoint
4) Properties must be satisfied by the pixel in the region - means if all pixels in Ri have same intensity level.
5) Two adjoint regions Ri and Rj must be different in the sense of predicate.

Segmentation is to partition an image into regions that satisfy the preceding conditions.
Assumptions are:

- Boundaries of regions are sufficiently different from each other and also from background
- Boundary detection is based on local discontinuities.
- Edge based segmentation is principal approach
- Region based approaches are based on partitioning an image into regions.{that are similar according to set of predefined criteria}.

- Segmentation methods are based on detecting sharp, local changes in intensity.

- Isolated points, Lines and edges are 3 types of image features by which you can segment.

- Edge pixels are pixels at which the intensity of an image function changes abruptly. Edges are sets of connected edge pixels.

- Edge detectors are local image processing methods designed to detect edge pixels.

- A line may be viewed as an edge segment in which the intensity of the background on either side of the line is either much higher or much lower than the line pixel intensity

Properties of First and Second order derivative

- 1^{st} and 2^{nd} order derivatives are well suited for edge and line detection.

- 1^{st} order derivative non-zero at the onset and along the entire ramp while 2^{nd} order derivative is non-zero only at the onset and end of ramp.

- For isolated noise point, 2^{nd} order derivative response is much stronger than 1^{st} order derivative.

- For both ramp and step edges, 2^{nd} order derivative has opposite signs.

- If edge transition from light to dark then negative value, similarly if edge transition from dark to light then positive value.(Transition of edges can be determined..).

Derivatives Properties

- First derivative generally produce thicker edges in an image
- Second derivative has a very strong response to fine details and noise
- Second derivative sign can be used to determine transition direction.

Derivatives

- We will use derivatives (first and second) in order to discover discontinuities.
- If we observe a line within an image we can consider it as a one-dimensional function f(x). We will now define the first derivative simply as the difference between two adjacent pixels.

$$\frac{\partial f}{\partial x} = f'(x) = f(x+1) - f(x)$$

$$\frac{\partial^2 f}{\partial x^2} = f''(x) = f(x+1) - 2f(x) + f(x-1)$$

Detection of isolated point

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

$$\frac{\partial^2 f(x,y)}{\partial x^2} = f(x+1,y) - 2f(x,y) + f(x-1,y)$$

$$\frac{\partial^2 f(x,y)}{\partial y^2} = f(x,y+1) - 2f(x,y) + f(x,y-1)$$

$$\nabla^2 f(x,y) = f(x+1,y) + f(x-1,y) + f(x,y+1) + f(x,y-1) - 4f(x,y)$$

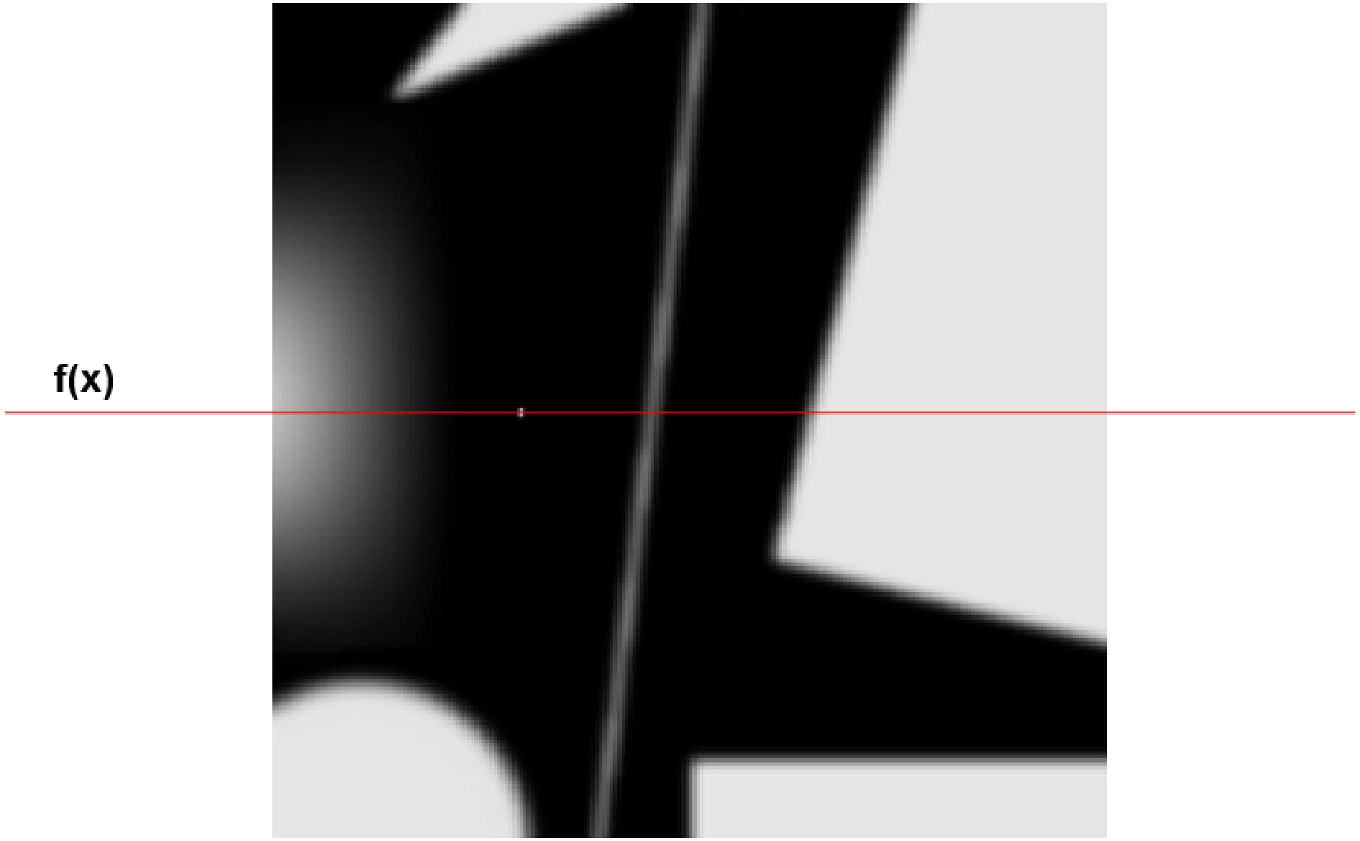
- 2^{nd} Order derivative produces 2 values for every edge in an image.

- An imaginary straight line joining the extreme positive and negative values of the second derivative would cross zero near the midpoint of the edge. (**zero-crossing property**)

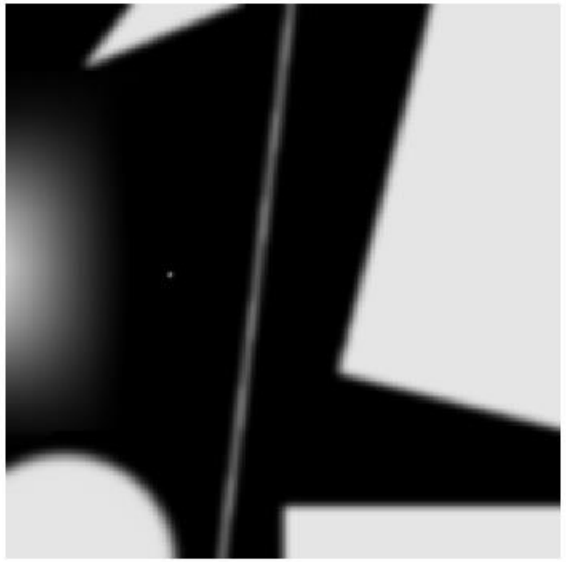
- The zero-crossing based methods search for zero crossings in a second-order derivative expression computed from the image in order to find edges.

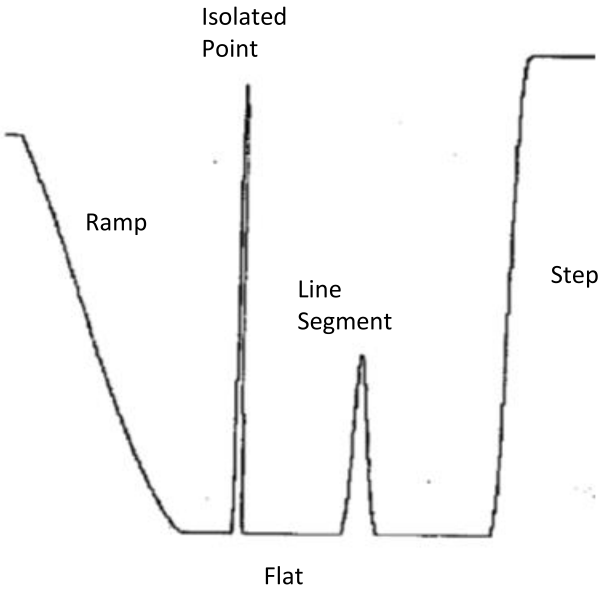
- This is quite useful for locating the centers of thick edges

Example



Example





We can notice that the 2nd order derivative is much more aggressive than a 1st order derivative in enhancing sharp changes. In both ramp and step edges 2nd order derivative has opposite signs(+ve to –Ve / -ve to +ve). Sign determines transition from dark to light or light to dark.

5	5	4	3	2	1	0	0	0	6	0	0	0	0	1	3	1	0	0	0	0	7	7	7	7	Image Strip
	-1	-1	-1	-1	-1	0	0	6	-6	0	0	0	1	2	-2	-1	0	0	0	7	0	0	0		I derivative
	-1	0	0	0	0	1	0	6	-12	6	0	0	1	1	-4	1	1	0	0	7	-7	0	0		2 derivative

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.

$$R = w_1 z_1 + w_2 z_2 + \ldots + w_9 z_9 = \sum_{i=1}^{9} w_i z_i$$

To compute the sum of the product of mask coefficients

$$G(x,y) = \begin{cases} 1 & \text{if } |R(x,y)| \geq T \\ 0 & \text{otherwise} \end{cases}$$

FIGURE 10.1 A general 3 × 3 mask.

Detection of isolated points can be implemented by Laplacian mask. When the location (x,y) on which mask is centred, if the absolute value of the response of the mask at that point exceeds a specific threshold. Such points are labelled 1 otherwise 0 (binary image). Here g is the output image, T is a non-negative threshold, R given by the equation.

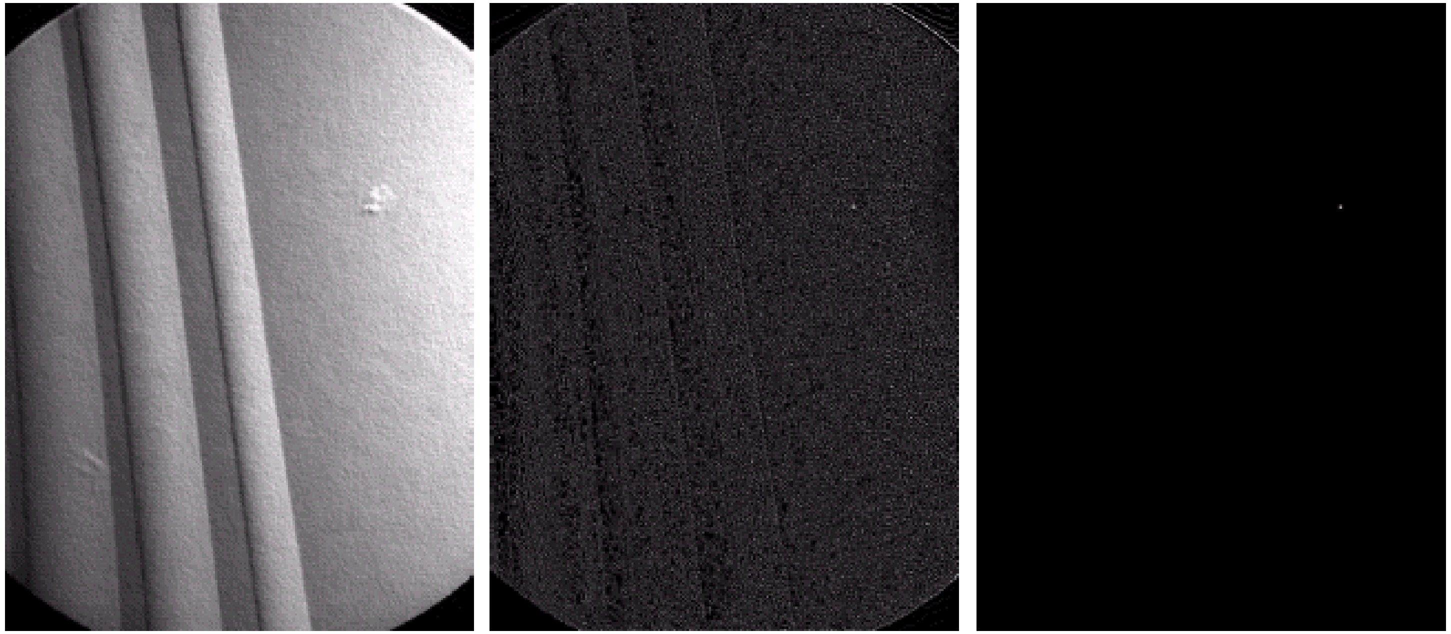
Detection of Discontinuities Point Detection

$$|R| \geq T$$

where T : a nonnegative threshold

- Point is detected when absolute value of the response of the mask at that point exceeds threshold value.
- It is based on abrupt intensity changes at a single-pixel locations that are surrounded by homogeneous background in the area of detector mask.
- When this is not satisfied, other methods can be used which is suitable.

−1	−1	−1



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

- We can use the Laplacian also for detection of line, since it is sensitive to sudden changes and thin lines.
- We must note that since the second derivative changes its sign on a line it creates a “double line effect” and it must be handled.
- Second derivative can have negative results and we need to scale the results

Detection of Discontinuities Line Detection

- To find one pixel wide line in an image.

For digital images the only three point straight lines are 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

Suppose that an image is filtered with the four masks, if at a given point in the image, |Ri| > |Rj| for all j#k, that point is said to be more likely associated with a line in the direction of mask i.

- When a 3 X 3 mask is centred on a line of constant intensity of 5 pixels wide, the response will be zero.
- We have assumed that lines are thin with respect to the detector.
- Lines that do not satisfy this assumption are treated as regions.
- Here predefined direction of each mask is weighted with a larger co-efficient(ie. 2) compare to other directions.
- The coefficients of each mask sum is zero, indicating that zero response in areas of constant intensity.

- Laplacian image contains negative values. So scaling is required (due to the existance of both negative and positive pixels in the result after convolving with Laplacian mask)
- When we consider both negative and positive values, it doubles the thickness of the lines.
- A More suitable approach is take only positive values, If required we have to use positive threshold values to eliminate.
- Often interest lies in detecting lines in specified direction (ie. 45 or 90 degree)
- To extract lines at 45 degree then 2^{nd} mask is required to convolve with the image. By this maximum response occurs at image in which 45 degree line passed through the middle of the row of mask.

Double Edge Effect - Example

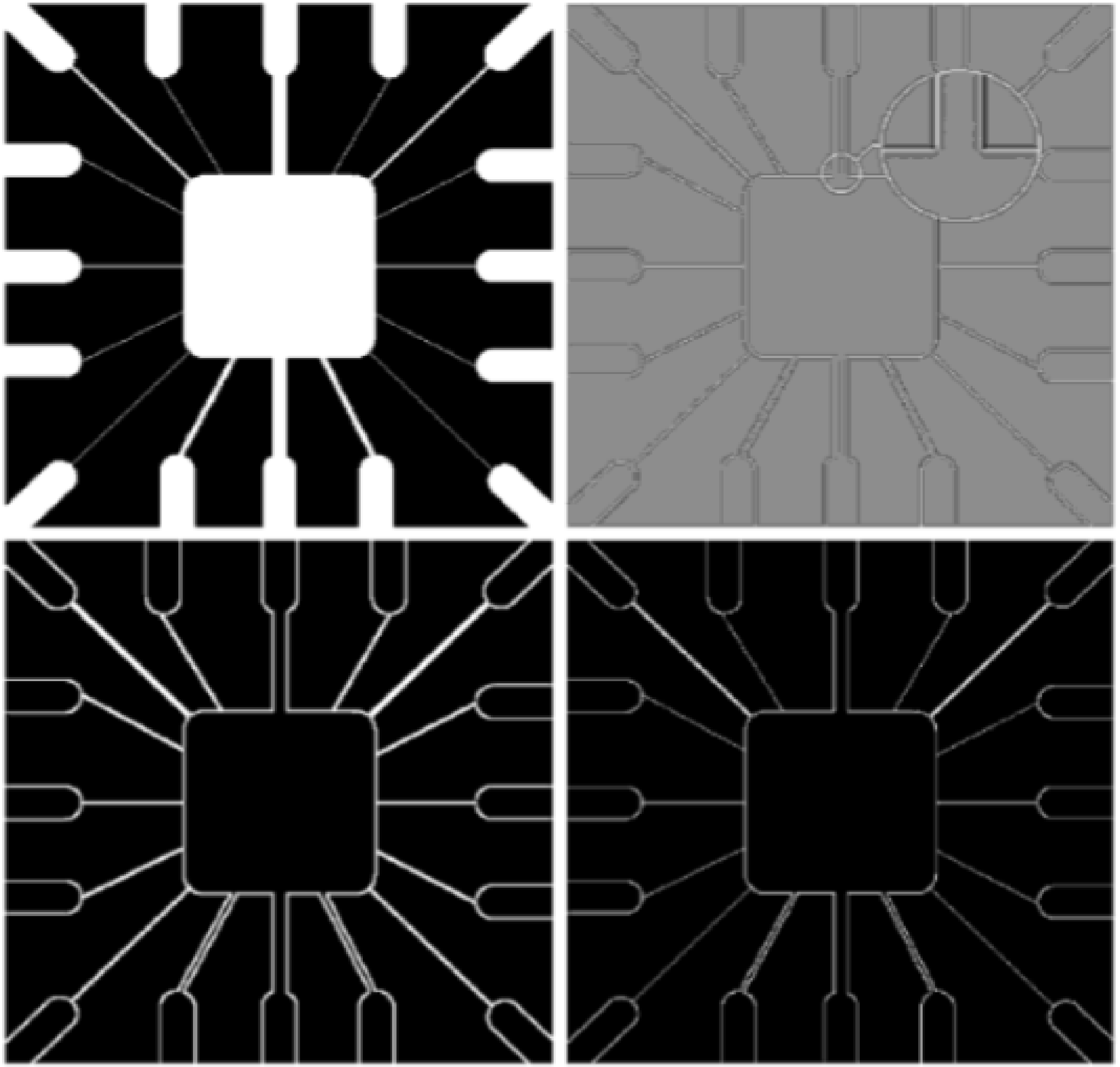
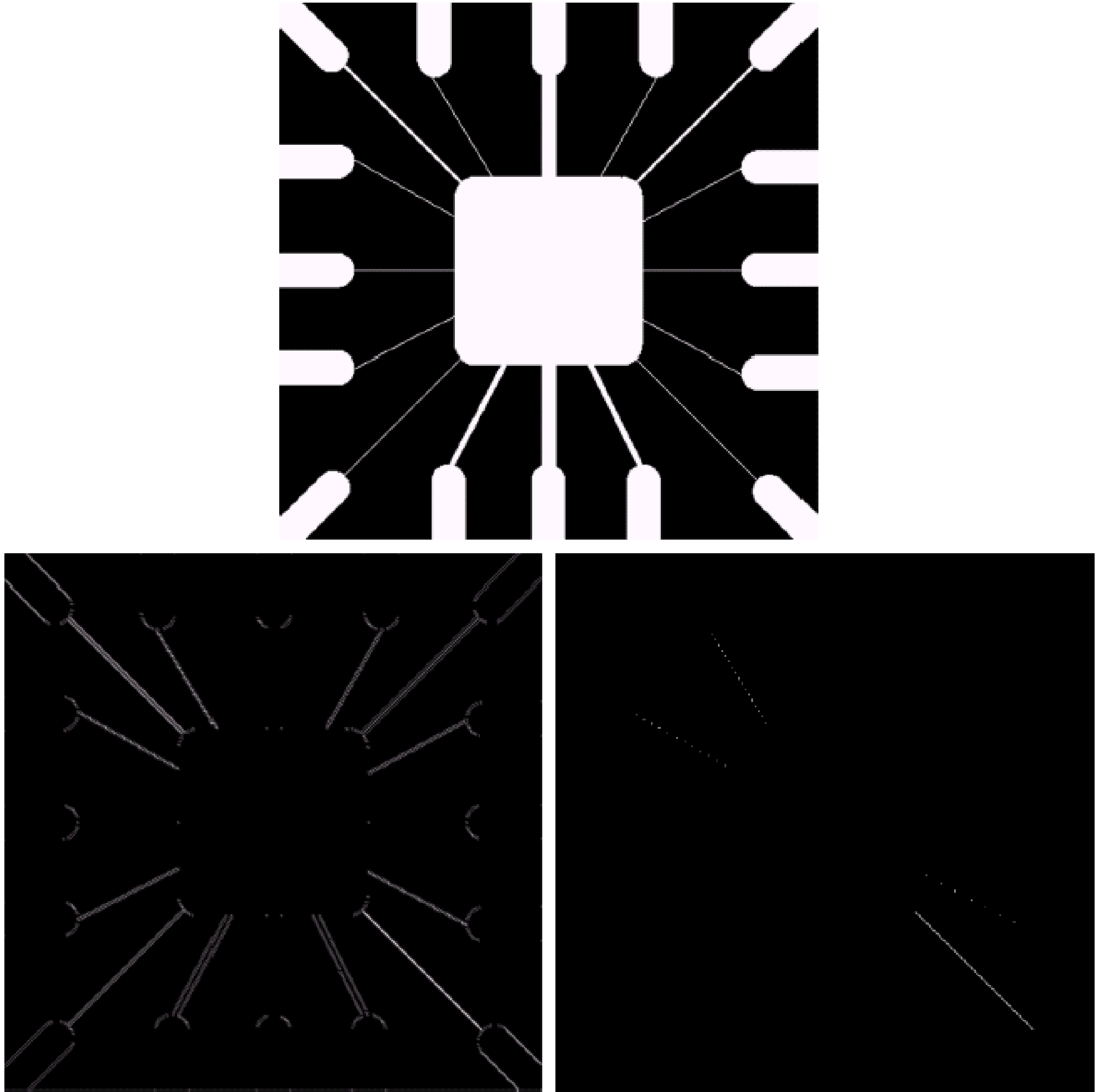


FIGURE 10.5
(a) Original image.
(b) Laplacian image; the magnified section shows the positive/negative double-line effect characteristic of the Laplacian.
(c) Absolute value of the Laplacian.
(d) Positive values of the Laplacian.

Detection of Discontinuities Line Detection



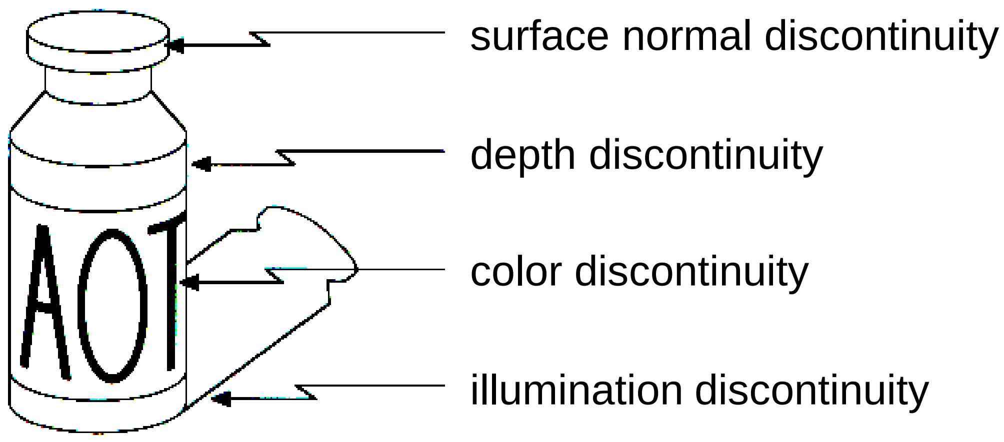
a
b c

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).

What Causes Intensity Changes?

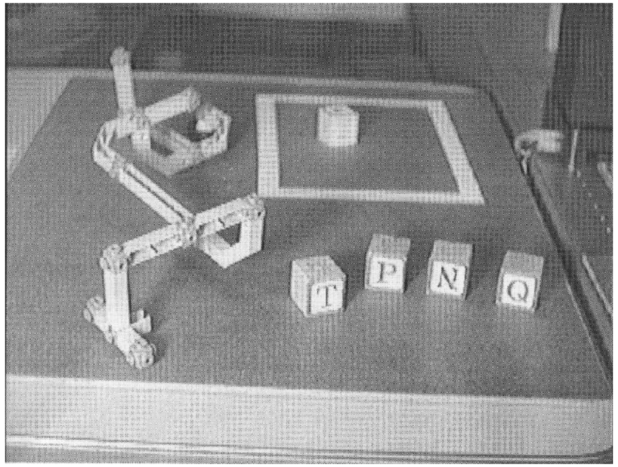
- Geometric events
  - surface orientation (boundary) discontinuities
  - depth discontinuities
  - color and texture discontinuities

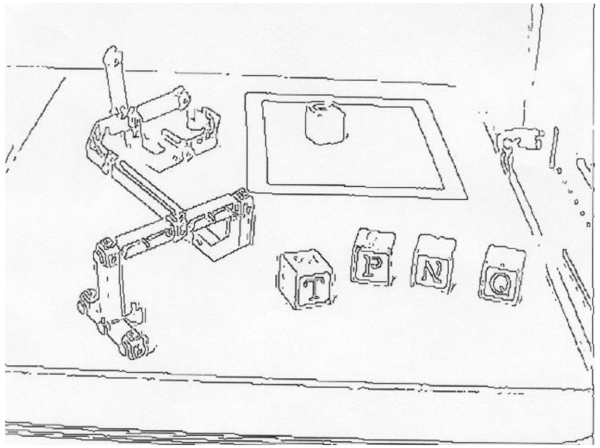
- Non-geometric events
  - illumination changes
  - specularities
  - shadows
  - inter-reflections



Goal of Edge Detection

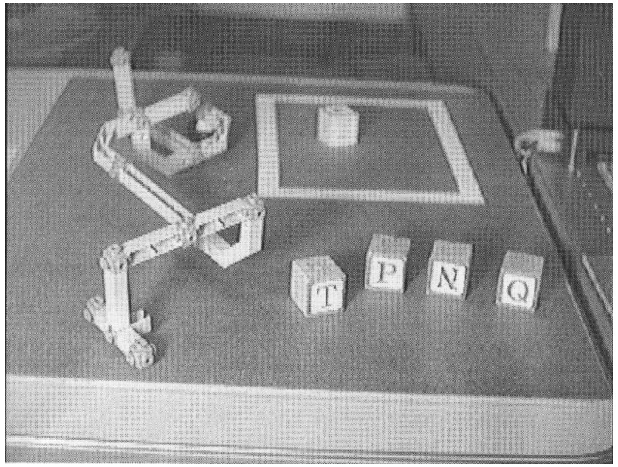
- Produce a line "drawing" of a scene from an image of that scene.

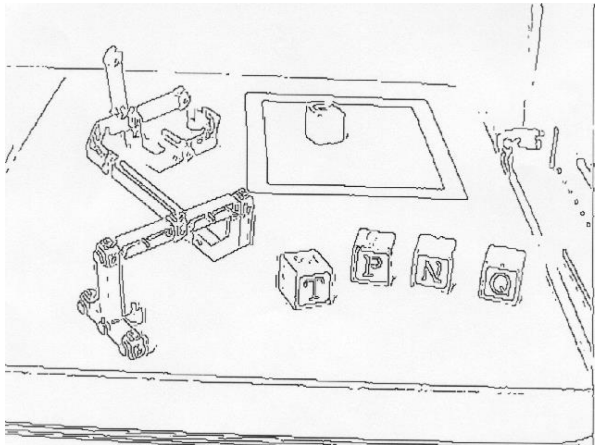




Why is Edge Detection Useful?

- Important features can be extracted from the edges of an image (e.g., corners, lines, curves).
- These features are used by higher-level computer vision algorithms (e.g., recognition).





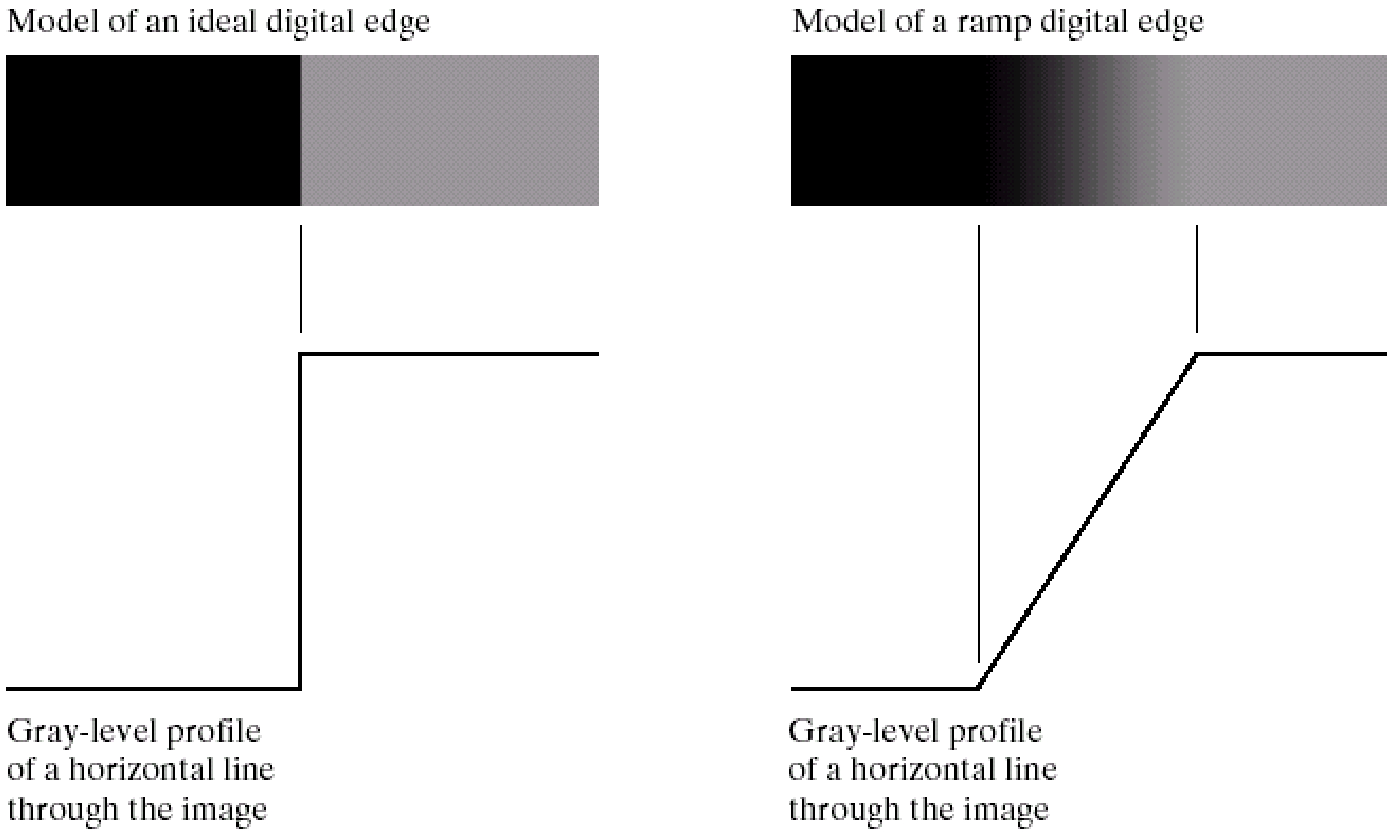
Edge Models

- Edge detection is the approach used most frequently for segmenting images based on abrupt changes in intensity.
- Edge models are classified according to the intensity profiles.
- ***Step Edge:*** Transition between two intensity levels occurring ideally over a distance of 1 pixel (10.5 (a))
  - This occurs in computer generated images. (Eg: Solid Modelling & Animation)
- ***Ramp Edge:*** In practice, digital images have edges that are blurred and noisy.
  - In such cases edges are closely modelled as having an intensity ramp profile.(10.5(b))
  - Edge point can be any point in the ramp & edge segment would be set of such points that are connected.
  - Degree of blurring determined by limitations in the focusing mechanism. (Eg: Lens of optical images)
  - In such situations edges are more closely modelled as intensity ramp profile.

- ***Roof edge:*** Here lines are passed through region. Width of roof edge depends on thickness & sharpness of line.
    - Eg: Satellite images, where thin features such as roads can be modelled.

- When its base is 1 pixel wide, a roof edge is really nothing more than 1 pixel thick line running through the region in an image.

Detection of Discontinuities Line Detection

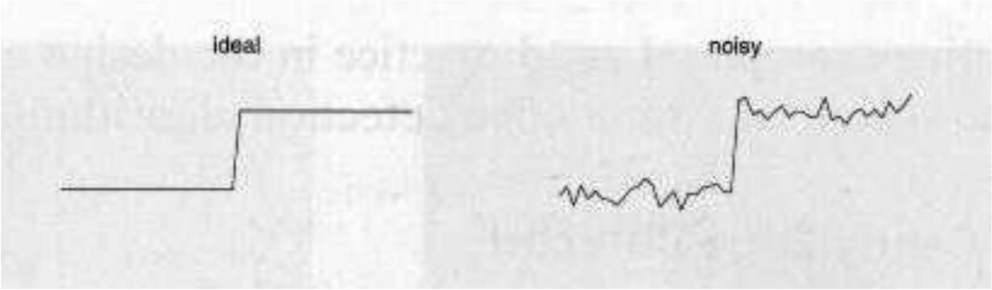


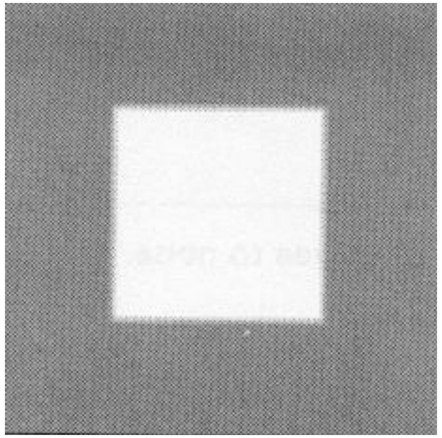
a b

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.

Modeling Intensity Changes

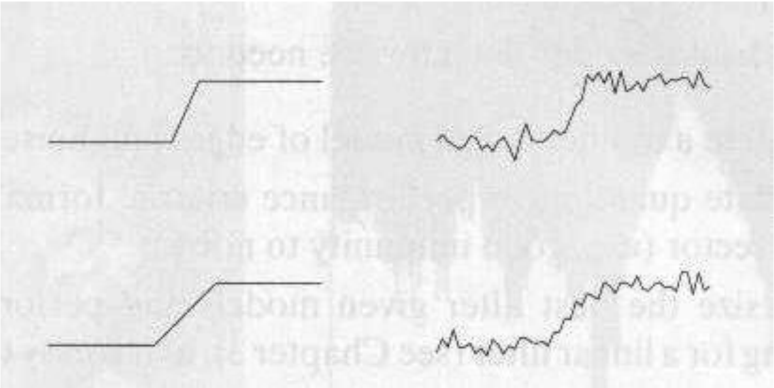
- **Step edge:** the image intensity abruptly changes from one value on one side of the discontinuity to a different value on the opposite side.

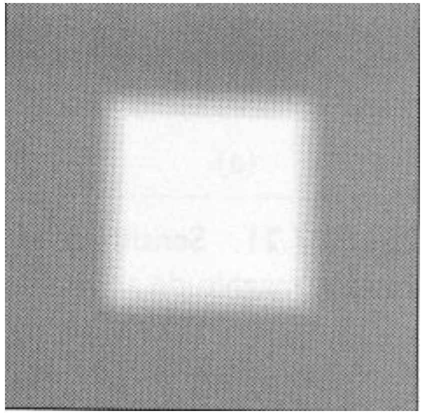




Modeling Intensity Changes (cont'd)

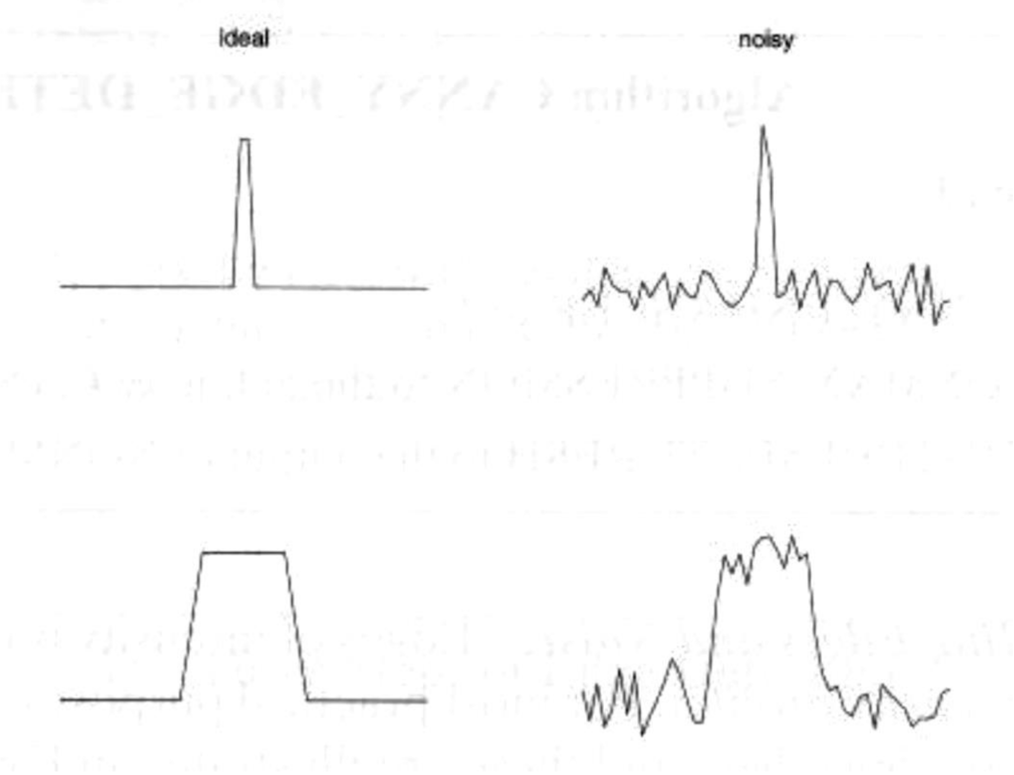
- **Ramp edge:** a step edge where the intensity change is not instantaneous but occur over a finite distance.





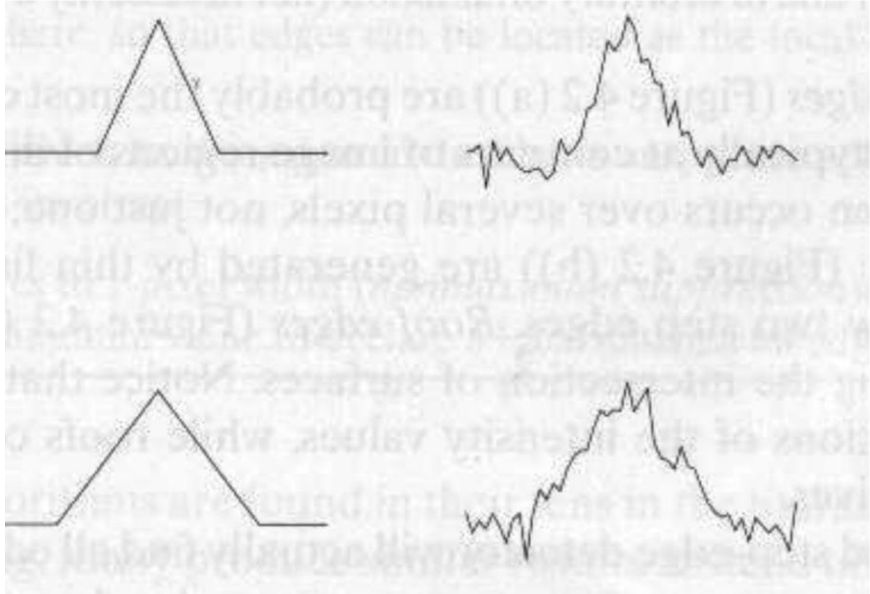
Modeling Intensity Changes (cont'd)

- **Ridge edge:** the image intensity abruptly changes value but then returns to the starting value within some short distance (i.e., usually generated by lines).



Modeling Intensity Changes (cont'd)

- **Roof edge: a** roof edge where the intensity change is not instantaneous but occur over a finite distance (i.e., usually generated by the intersection of two surfaces).



Main Steps in Edge Detection

(1) Smoothing: suppress as much noise as possible, without destroying true edges.

(2) Enhancement: apply differentiation to enhance the quality of edges (i.e., sharpening).

Main Steps in Edge Detection (cont'd)

(3) Thresholding: determine which edge pixels should be discarded as noise and which should be retained (i.e., threshold edge magnitude).

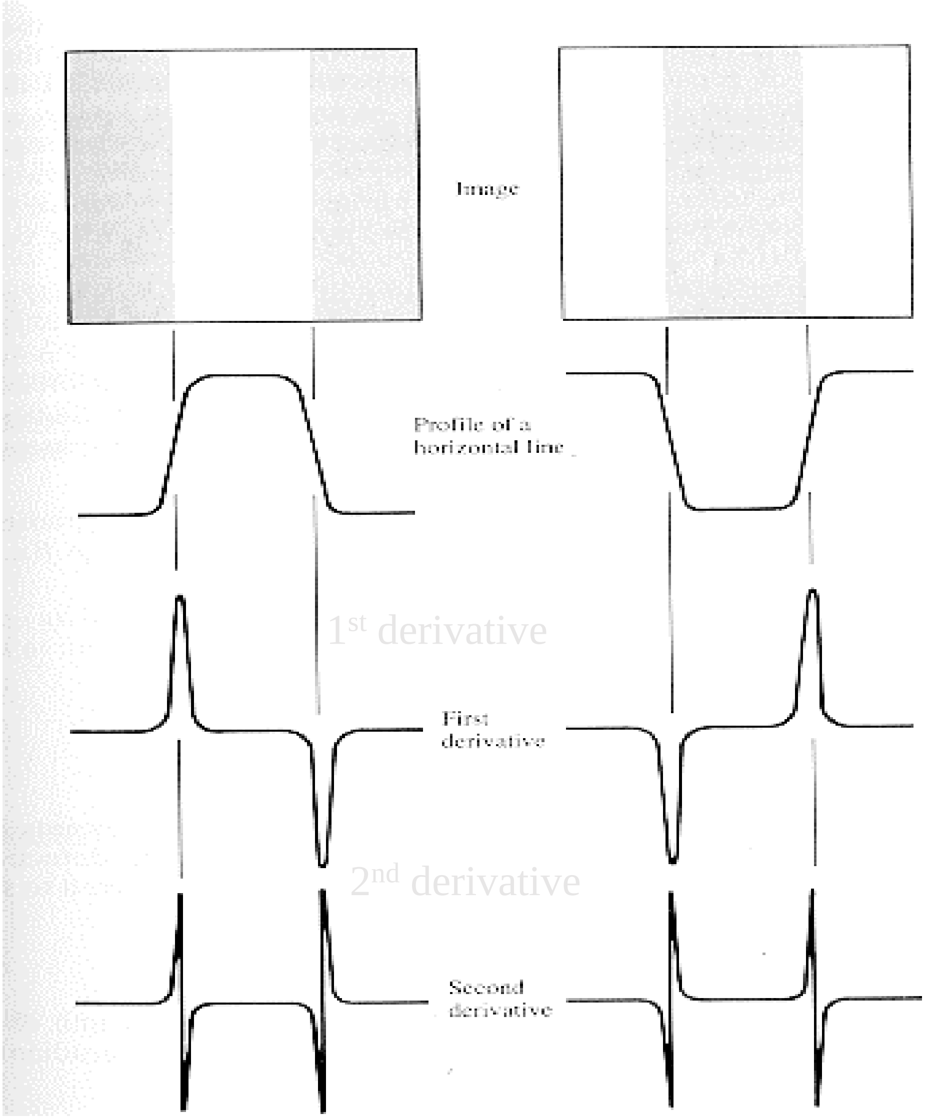
(4) Localization: determine the exact edge location.

Edge Detection Using Derivatives

- Often, points that lie on an edge are detected by:

  (1) Detecting the local **maxima** or **minima** of the first derivative.

  (2) Detecting the **zero-crossings** of the second derivative.



Edge Detection Using First Derivative (Gradient)

- Magnitude of the first derivative can be used to detect the presence of edge sign can be used to detect a point is in darker or brighter side

- The first derivative of an image can be computed using the gradient:

$$grad(f) = \begin{pmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{pmatrix}$$

- Masks are used to obtain the gradient components at every pixel

- Partial derivatives are used to estimate edge strength and direction

Image Gradient and its Properties

- First-order derivatives:
  - The gradient of an image $f(x,y)$ at location (x,y) is defined as the vector:

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

- The tool of choice for finding edge strength and direction at location (x,y) of an image, f is the gradient defined as vector.

- This vector has important geometrical property that it points in the direction of the greatest rate of change of f at location (x,y).

Detection of Discontinuities Gradient Operators

- The magnitude (length) of vector is denoted as M(x,y). It is the value of the rate of change in the direction of the gradient vector

- The gradient is a vector which has magnitude and direction:

  - The magnitude of this vector: $$\nabla f = \mathrm{mag}(\nabla \mathbf{f}) = \left[G_x^2 + G_y^2\right]^{1/2}$$

- **Magnitude:** indicates edge strength.

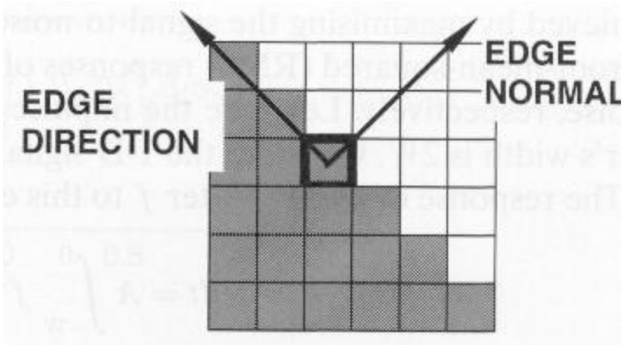
Detection of Discontinuities Gradient Operators

- The direction of the gradient vector is given by the angle measured with respect to x axis.

- **Direction:** indicates edge direction.
  - i.e., perpendicular to edge direction

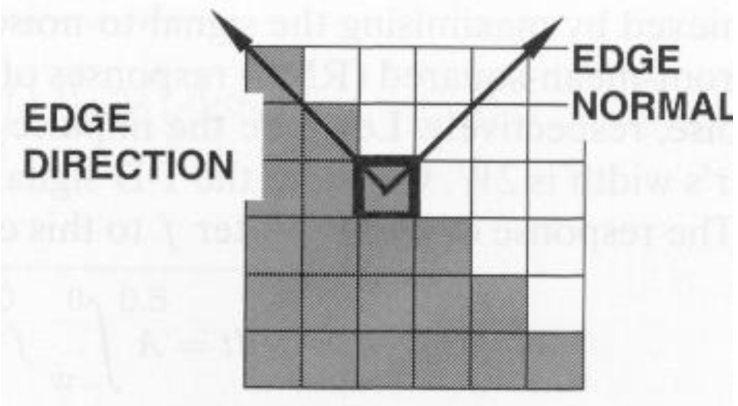
  - The direction of this vector:

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



Edge Descriptors

- **Edge direction:** perpendicular to the direction of maximum intensity change (i.e., edge normal)
- **Edge strength:** related to the local image contrast along the normal.
- **Edge position:** the image position at which the edge is located.



Gradient Operators

- Obtaining the gradient of an image requires computing the partial derivative at every pixel location in the image
- We are dealing with digital quantities, so a digital approximation of the partial derivatives over a neighbourhood about the point if required.

$$\Delta_x f(x, y) = f(x+1, y) - f(x,y)$$

$$\Delta_y f(x, y) = f(x, y+1) - f(x,y)$$

- Theses two equations can be implemented for all pertinent values of x and y by filtering f(x,y) with the 2D mask.

Detection of Discontinuities Gradient Operators

Roberts cross-gradient operators

−1	0		0	−1
0	1		1	0

Roberts

Prewitt operators

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

Prewitt

Sobel operators

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

Sobel

Detection of Discontinuities Gradient Operators

Prewitt masks for detecting diagonal edges

0	1	1
−1	0	1
−1	−1	0

−1	−1	0
−1	0	1
0	1	1

Prewitt

Sobel masks for detecting diagonal edges

0	1	2
−1	0	1
−2	−1	0

−2	−1	0
−1	0	1
0	1	2

Sobel

a b
c d

FIGURE 10.9 Prewitt and Sobel masks for detecting diagonal edges.

The coefficients of all the masks in the above fig. sum to zero, thus giving a response of zero in areas of constant intensity (as expected of a derivative operator)

Detection of Discontinuities Gradient Operators: 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, $|G_x| + |G_y|$.

$$\nabla f \approx |G_x| + |G_y|$$

Detection of Discontinuities Gradient Operators: Example

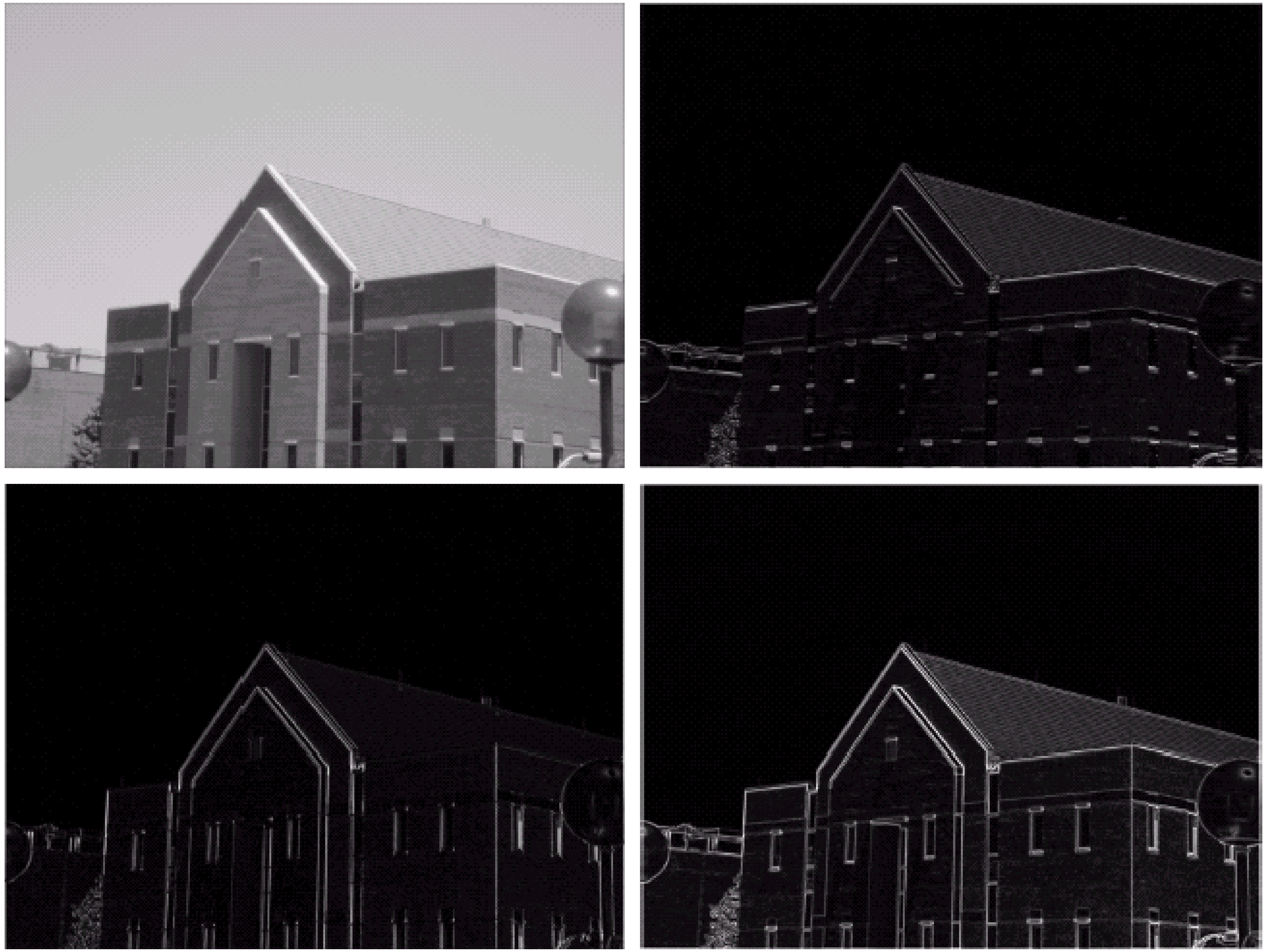


FIGURE 10.11 Same sequence as in Fig. 10.10, but with the original image smoothed with a 5×5 averaging filter.

Prewitt Operator

- Setting *c* = 1, we get the Prewitt operator:

$$M_x = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix} \qquad M_y = \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}$$

M_x and M_y are approximations at (i, j)

Sobel Operator

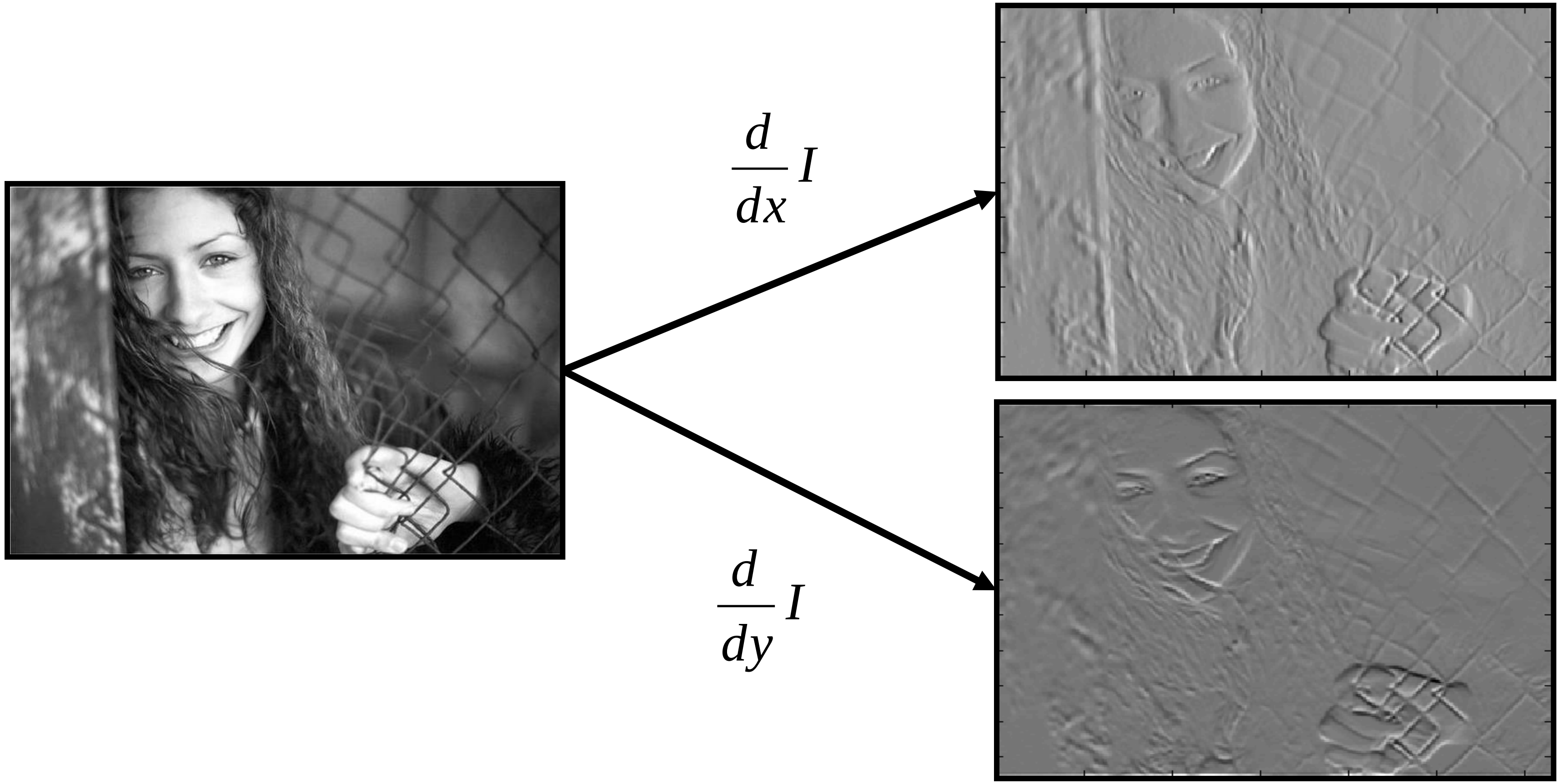
- Setting c = 2, we get the Sobel operator:

$$M_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \qquad M_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

M_x and M_y are approximations at (i, j)

- Better in noise supression

Another Example



Another Example (cont'd)

$$\nabla = \sqrt{\left(\frac{d}{dx}I\right)^2 + \left(\frac{d}{dy}I\right)^2}$$

$$\nabla \geq Threshold = 100$$

Detection of Discontinuities Gradient Operators

- Second-order derivatives: (The Laplacian)
  - The Laplacian of an 2D function $f(x,y)$ is defined as

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

  - Two forms in practice:

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

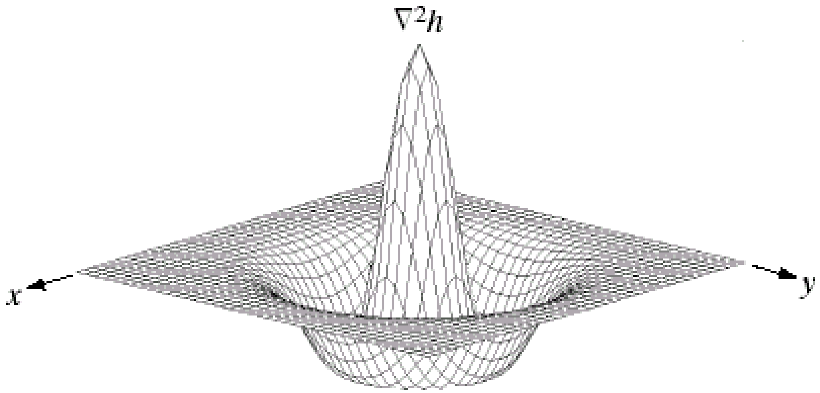
−1	−1	−1
−1	8	−1
−1	−1	−1

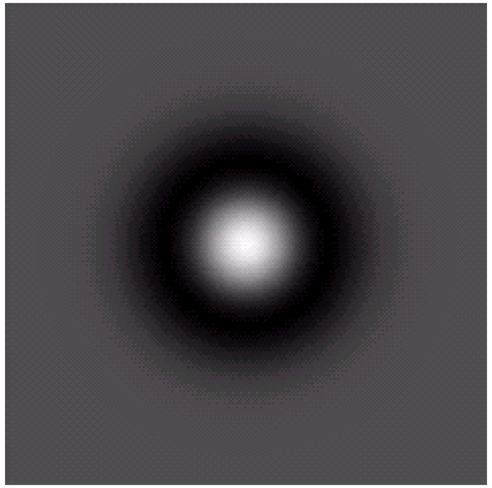
The Laplacian

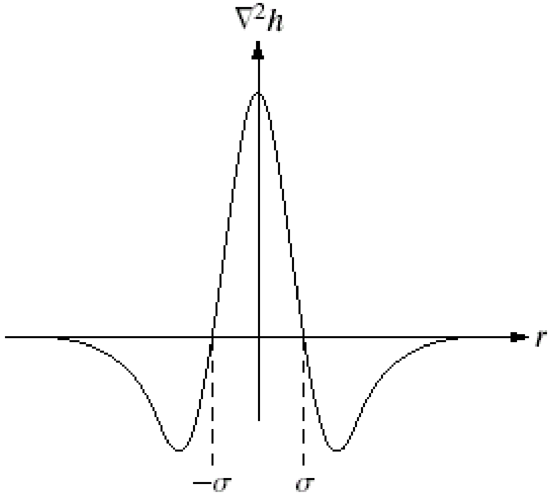
- Advantages:
  - Identifies a minute change/info.
  - Easy to access depth info.
  - Used in edge detection when image quality is smooth/dull
  - Zero cross is used to identify a point lies in dark/bright
- Dis-Advantages:
  - Sensitive to noise
  - May generate double edges
- May overcome by using LoG

- EXTRA

Detection of Discontinuities Gradient Operators





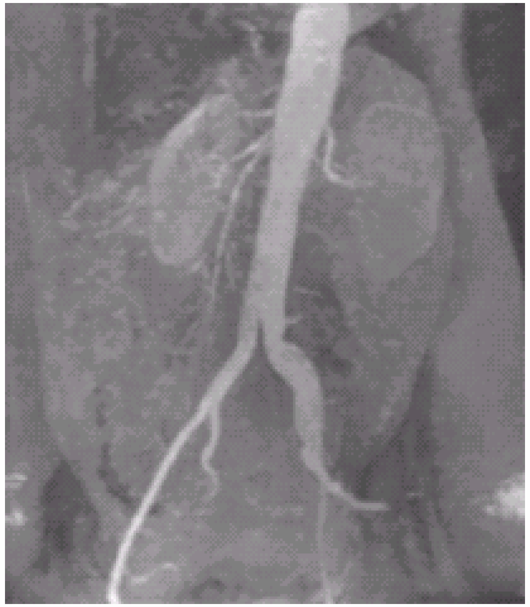


0	0	–1	0	0
0	–1	–2	–1	0
–1	–2	16	–2	–1
0	–1	–2	–1	0
0	0	–1	0	0

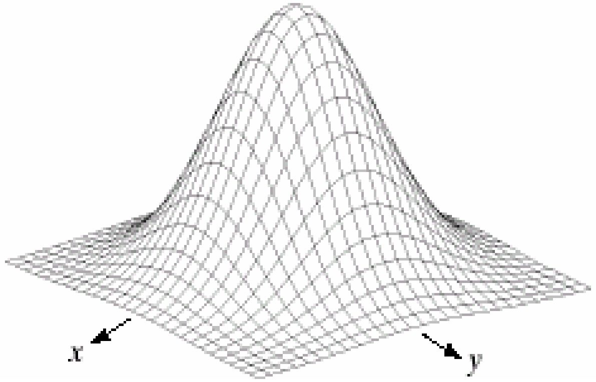
a	b
c	d

FIGURE 10.14 Laplacian of a Gaussian (LoG). (a) 3-D plot. (b) Image (black is negative, gray is the zero plane, and white is positive). (c) Cross section showing zero crossings. (d) 5 × 5 mask approximation to the shape of (a).

Detection of Discontinuities Gradient Operators: Example



Sobel gradient



Gaussian smooth function

–1	–1	–1
–1	8	–1
–1	–1	–1

Laplacian mask

Detection of Discontinuities Gradient Operators: Example

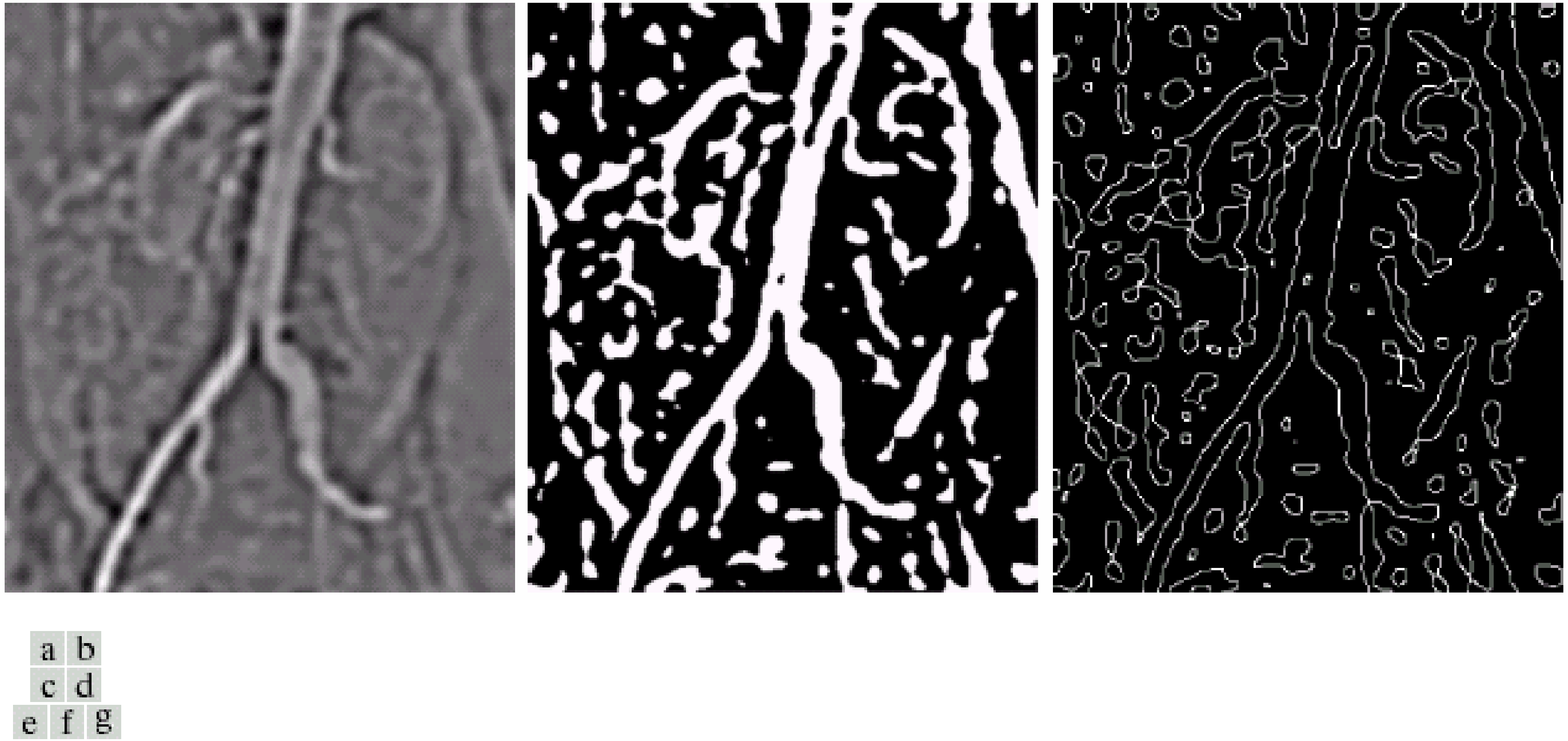
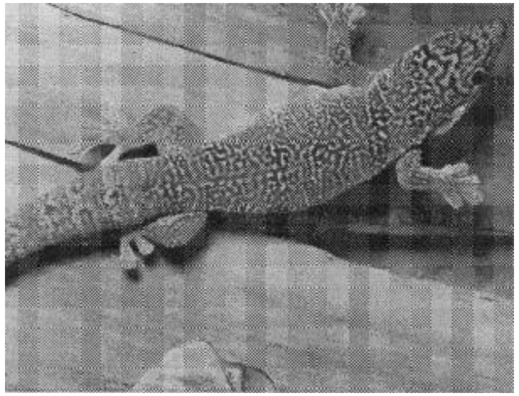


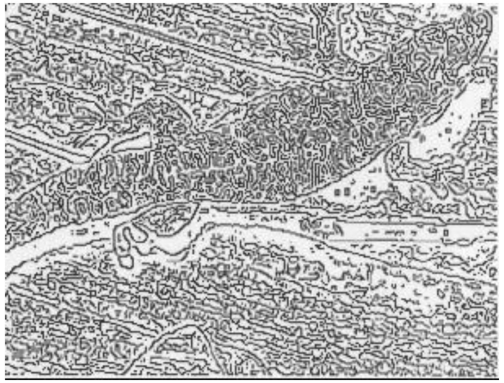
FIGURE 10.15 (a) Original image. (b) Sobel gradient (shown for comparison). (c) Spatial Gaussian smoothing function. (d) Laplacian mask. (e) LoG. (f) Thresholded LoG. (g) Zero crossings. (Original image courtesy of Dr. David R. Pickens, Department of Radiology and Radiological Sciences, Vanderbilt University Medical Center.)

Practical Issues

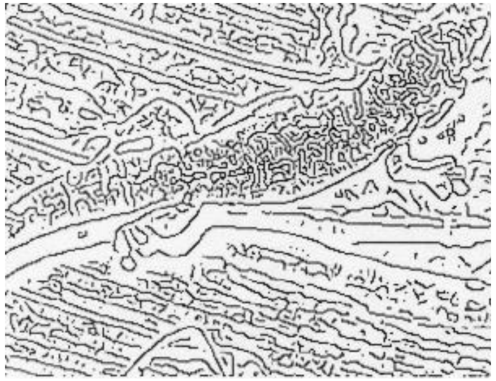
- Noise suppression-localization tradeoff.
  - Smoothing depends on mask size (e.g., depends on σ for Gaussian filters).
  - Larger mask sizes reduce noise, but worsen localization (i.e., add uncertainty to the location of the edge) and vice versa.



smaller mask



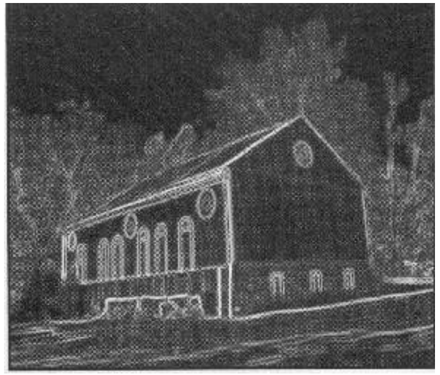
larger mask



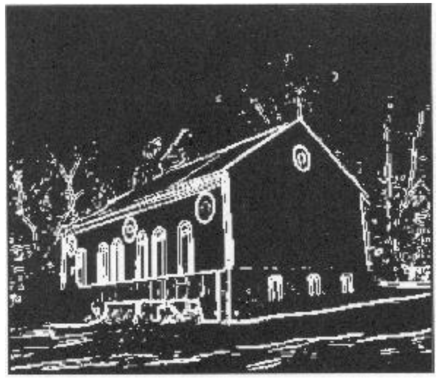
Practical Issues (cont'd)

- Choice of threshold.

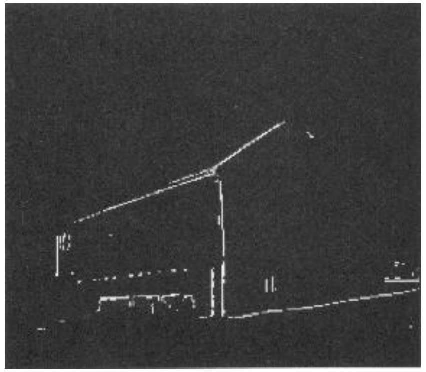
gradient magnitude



low threshold

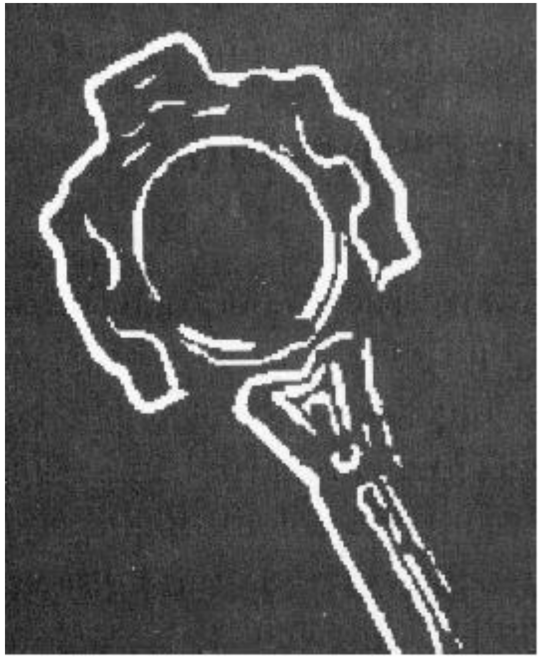


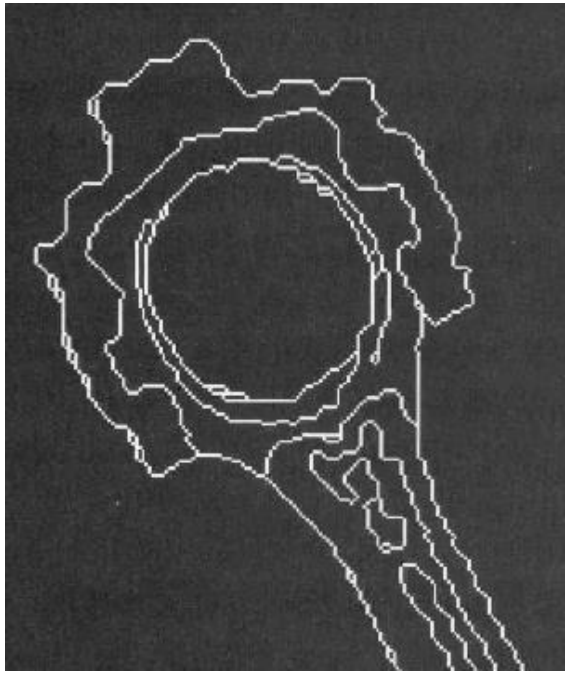
high threshold



Practical Issues (cont'd)

- Edge thinning and linking.

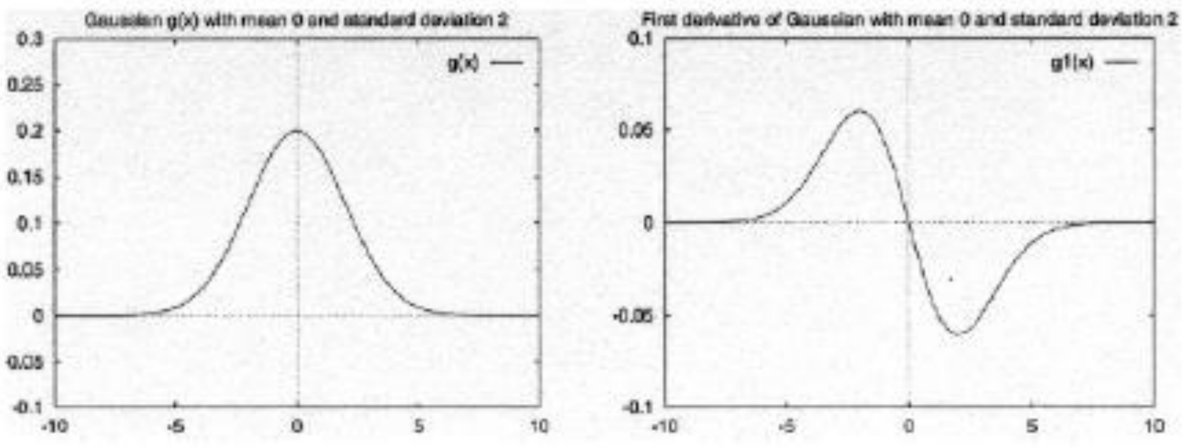




Canny edge detector

- Canny has shown that the **first derivative of the Gaussian** closely approximates the operator that optimizes the product of <u>*signal-to-noise* ratio</u> and <u>localization</u>.

  (i.e., analysis based on "step-edges" corrupted by "Gaussian noise“)



J. Canny, ***A Computational Approach To Edge Detection***, IEEE Trans. Pattern Analysis and Machine Intelligence, 8:679-714, 1986.

Canny edge detector - example

original image

Canny edge detector – example (cont’d)

Gradient magnitude

- Various other gradient based operators are given in the following figure

Operator	Row Gradient $(G_R(i,j))$	Column Gradient $(G_C(i,j))$
Roberts	$\begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$	$\begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}$
Prewitt	$\frac{1}{3}\begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix}$	$\frac{1}{3}\begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}$
Sobel	$\frac{1}{4}\begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix}$	$\frac{1}{4}\begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$
Frei-Chen	$\frac{1}{2+\sqrt{2}}\begin{bmatrix} 1 & 0 & -1 \\ \sqrt{2} & 0 & -\sqrt{2} \\ 1 & 0 & -1 \end{bmatrix}$	$\frac{1}{2+\sqrt{2}}\begin{bmatrix} -1 & -\sqrt{2} & -1 \\ 0 & 0 & 0 \\ 1 & \sqrt{2} & 1 \end{bmatrix}$

Various gradient-based operators

- A limitation common to edge gradient generation operators is that these are unable to detect edges accurately in a highly noisy environments. This problem can be alleviated by increasing the size of the window. As an example, a Prewitt-type 7 x 7 operator has a row gradient impulse response of the following form.

$$H_R = \frac{1}{21}\begin{bmatrix} 1 & 1 & 1 & 0 & -1 & -1 & -1 \\ 1 & 1 & 1 & 0 & -1 & -1 & -1 \\ 1 & 1 & 1 & 0 & -1 & -1 & -1 \\ 1 & 1 & 1 & 0 & -1 & -1 & -1 \\ 1 & 1 & 1 & 0 & -1 & -1 & -1 \\ 1 & 1 & 1 & 0 & -1 & -1 & -1 \\ 1 & 1 & 1 & 0 & -1 & -1 & -1 \end{bmatrix}$$

Use of masks

- Another way of finding out edge location is by using masks.

$$East \quad \begin{bmatrix} 5 & -3 & -3 \\ 5 & 0 & -3 \\ 5 & -3 & -3 \end{bmatrix} \qquad West \quad \begin{bmatrix} -3 & -3 & 5 \\ -3 & 0 & 5 \\ -3 & -3 & 5 \end{bmatrix}$$

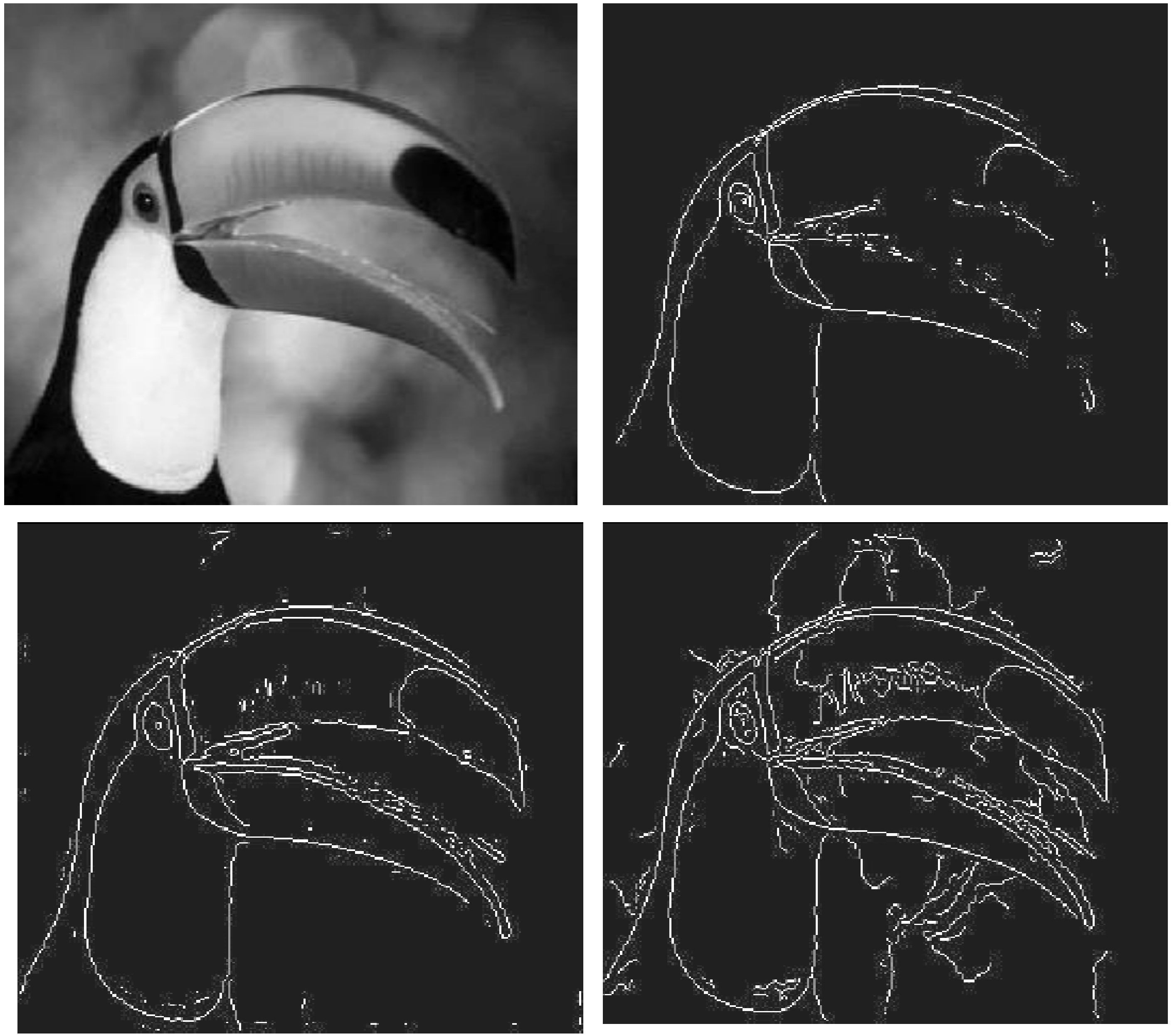
$$Northeast \quad \begin{bmatrix} -3 & -3 & -3 \\ 5 & 0 & -3 \\ 5 & 5 & -3 \end{bmatrix} \qquad Southwest \quad \begin{bmatrix} -3 & 5 & 5 \\ -3 & 0 & 5 \\ -3 & -3 & -3 \end{bmatrix}$$

$$North \quad \begin{bmatrix} -3 & -3 & -3 \\ -3 & 0 & -3 \\ 5 & 5 & 5 \end{bmatrix} \qquad South \quad \begin{bmatrix} 5 & 5 & 5 \\ -3 & 0 & -3 \\ -3 & -3 & -3 \end{bmatrix}$$

Sobel

LOG

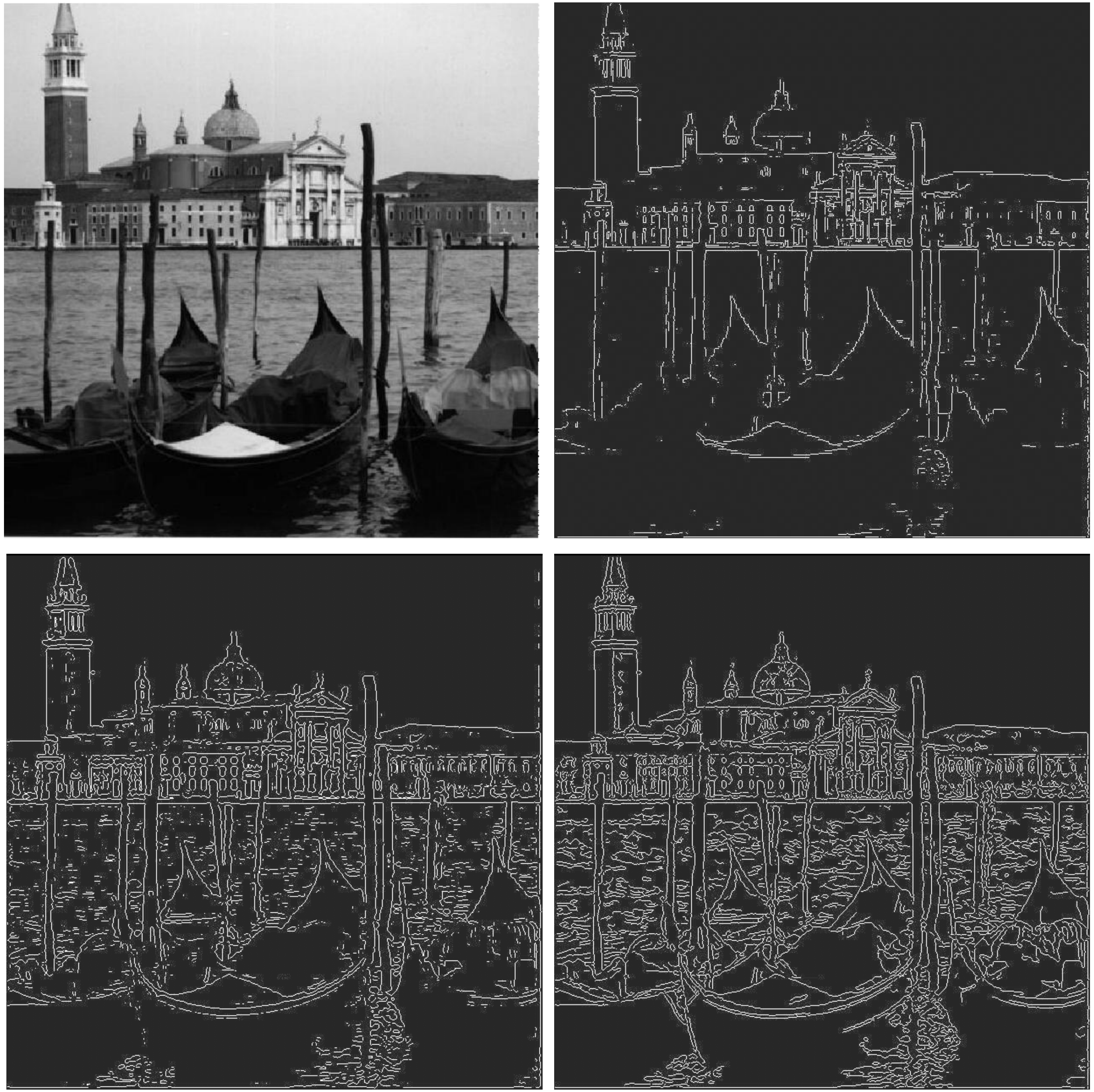
Canny



Sobel

LOG

Canny



Sobel

LOG

Canny