# Image Segmentation

Chapter 10

# Image segmentation:

- Segmentation refers to the process of partitioning a image into multiple regions.
- 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.

# Fundamentals

- Let represent the entire spatial region occupied by an image. We may view image segmentation as a process that partitions into sub regions, *R1, R2, R3,…,Rn* such that

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

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

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

**(d)** $Q(R_i) = \text{TRUE}$ for $i = 1, 2, \ldots, n.$

**(e)** $Q(R_i \cup R_j) = \text{FALSE}$ for any adjacent regions $R_i$ and $R_j$.

# Fundamentals

(a) segmentation must be complete → $\bigcup_{i=1}^{n} R_i = R$

– all pixels must belong to a region

(b) pixels in a region must be connected → $R_i$ is a connected set

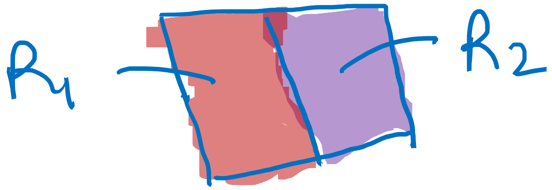
(c) Regions must be disjoint → $R_i \cap R_j = \phi$

(d) states that pixels in a region must all share the same property

– The logic predicate Q(Ri) over a region must return TRUE for each point in that region → All pixels in the Region must adhere to the logic

(e) indicates that regions are different in the sense of the predicate Q

$Q(R_i \cup R_j) = \text{FALSE}$




Q = Color of pixel

$Q(R_2) = \text{True}$

$Q(R_1 \cup R_2) = \text{FALSE}$

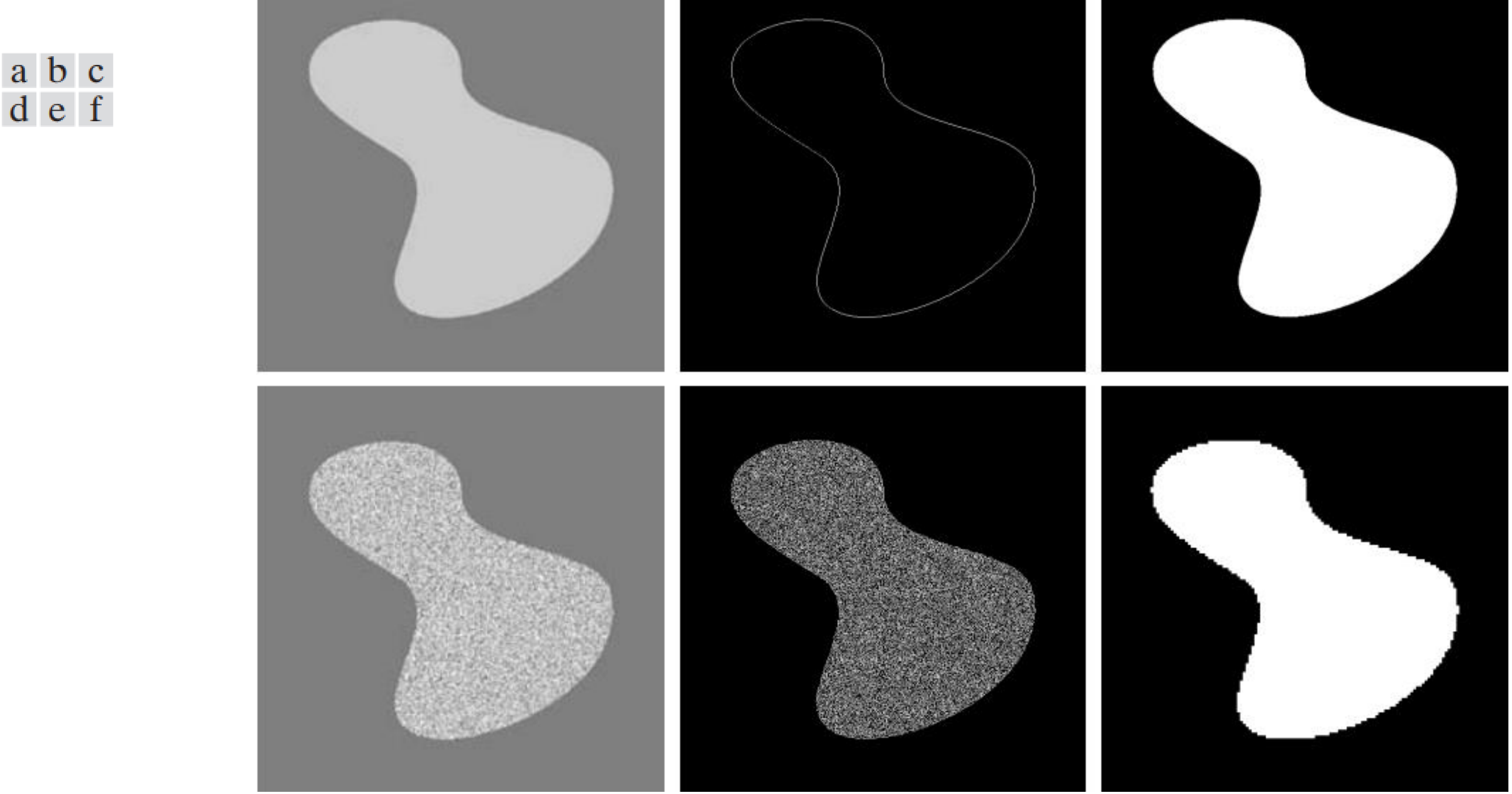
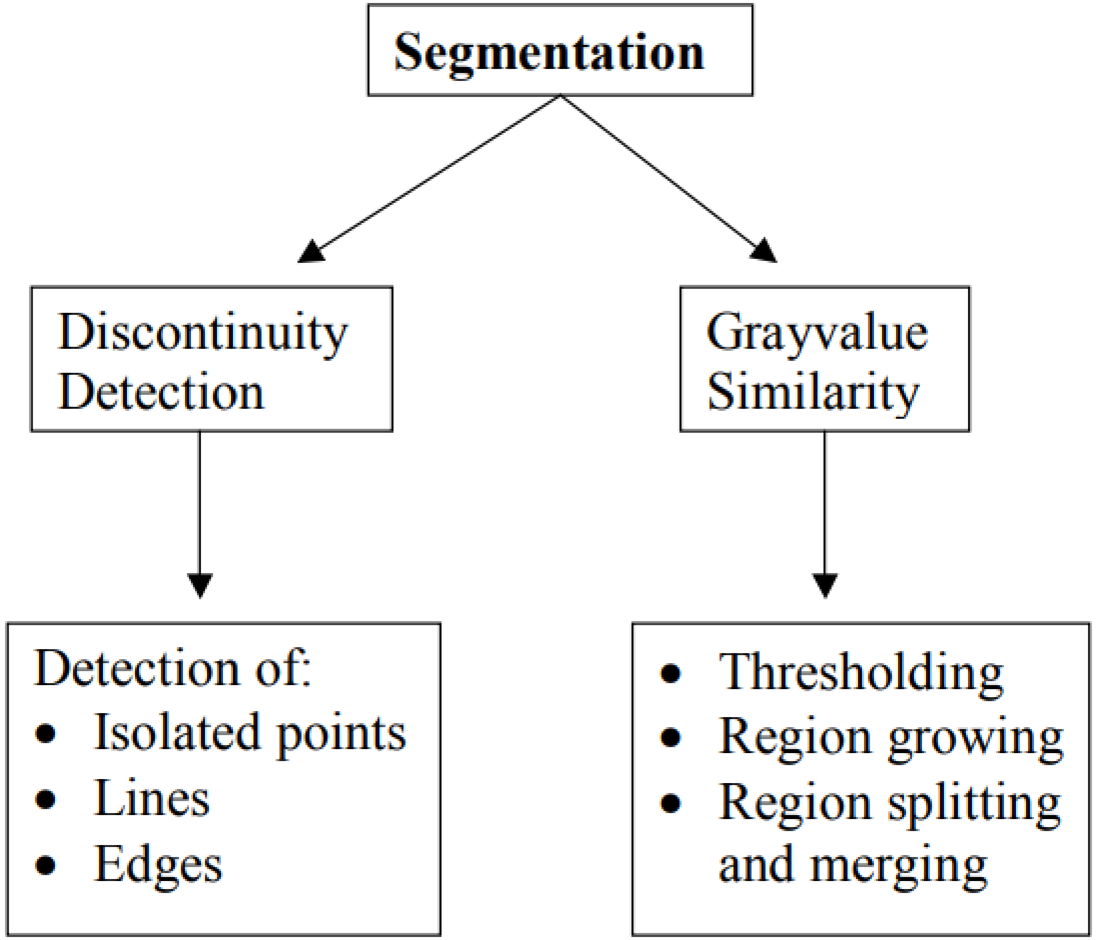


**FIGURE 10.1** (a) Image containing a region of constant intensity. (b) Image showing the boundary of the inner region, obtained from intensity discontinuities. (c) Result of segmenting the image into two regions. (d) Image containing a textured region. (e) Result of edge computations. Note the large number of small edges that are connected to the original boundary, making it difficult to find a unique boundary using only edge information. (f) Result of segmentation based on region properties.


Segmentation
Discontinuity Detection
Grayvalue Similarity
Detection of:
Isolated points
Lines
Edges
Thresholding
Region growing
Region splitting and merging

# Detection of Discontinuities

- There are three kinds of discontinuities of intensity: points, lines and edges.
- Three types of images features in which we are interested are:
  - Edges: edge pixels are pixels at which the intensity of an image function changes abruptly, and edges are sets of connected edge pixels.
  - Lines: line may be viewed as a edge segment in which the intensity of the background on either side of the line is either much higher or much lower than the intensity of the line pixels.
  - Isolated point: It can be viewed as a line whose length and width are equal to one pixel.

# Background (Revisit)

# Foundation (Revisit from Chapter 3)

- The derivatives of a digital function are defined in terms of differences.
- There are various ways to define these differences. However, we require that
- any definition we use for a *first derivative*
  (1) must be zero in areas of constant intensity;
  (2) must be nonzero at the onset of an intensity step or ramp; and
  (3) must be nonzero along ramps.
- Similarly, any definition of a second derivative
  (1) must be zero in constant areas;
  (2) must be nonzero at the onset *and* end of an intensity step or ramp; and
  (3) must be zero along ramps of constant slope.
- Because we are dealing with digital quantities whose values are finite, the maximum possible intensity change also is finite, and the shortest distance over which that change can occur is between adjacent pixels.

# Foundation (Revisit from Chapter 3)

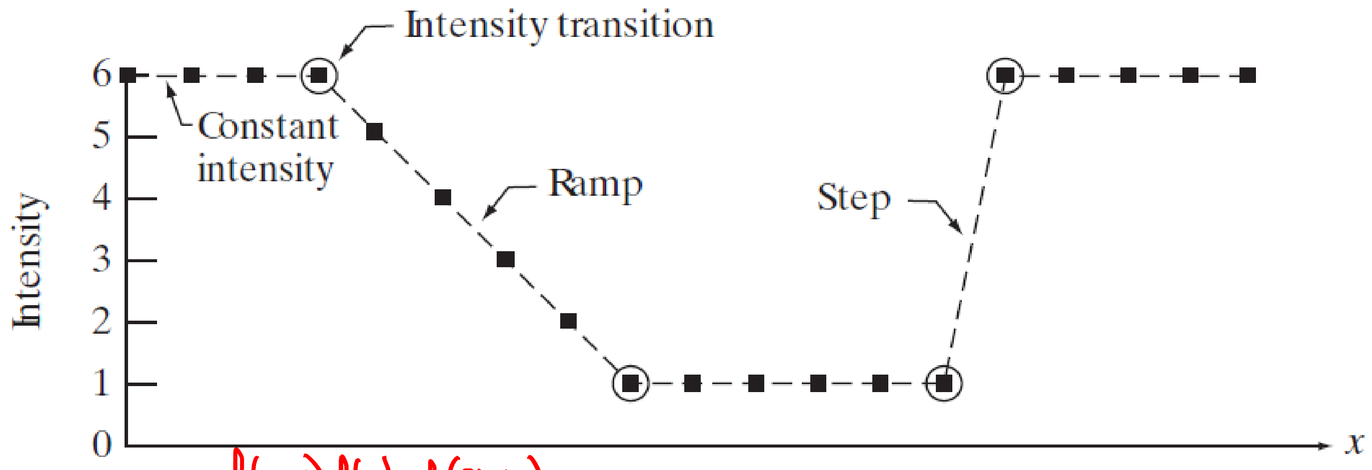
$$\frac{\partial f}{\partial x}(x) = f(x+1) - f(x)$$

first-order derivative

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

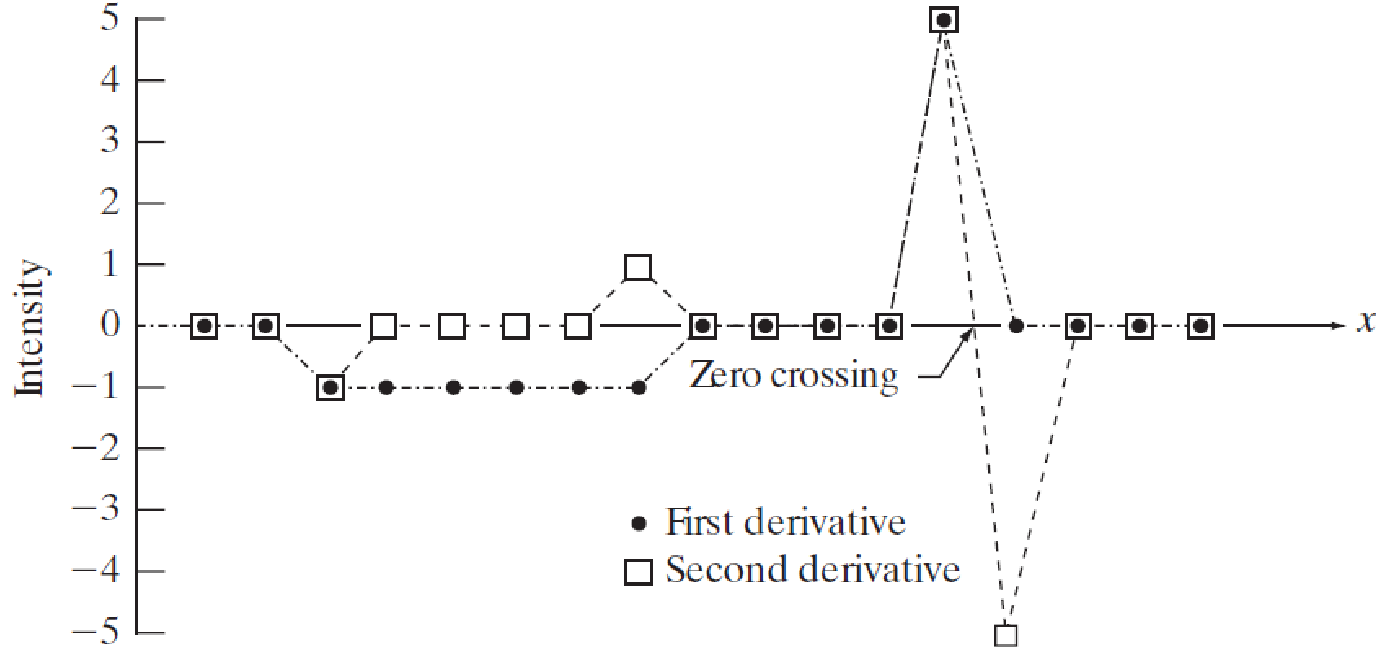
$$\frac{\partial^2 f}{\partial x^2} = f(x+1) + f(x-1) - 2f(x)$$

second-order derivative




f(x-1) f(x) f(x+1)

| | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Scan line | 6 | 6 | 6 | 6 | 5 | 4 | 3 | 2 | 1 | 1 | 1 | 1 | 1 | 1 | 6 | 6 | 6 | 6 | 6 → x |
| 4 − 5   1st derivative | | 0 | 0 | −1 | −1 | −1 | −1 | −1 | 0 | 0 | 0 | 0 | 0 | 5 | 0 | 0 | 0 | 0 | |
| 4+6−2×5   2nd derivative | | 0 | 0 | −1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 5 | −5 | 0 | 0 | 0 | |




a
b
c

**FIGURE 3.36** Illustration of the first and second derivatives of a 1-D digital function representing a section of a horizontal intensity profile from an image. In (a) and (c) data points are joined by dashed lines as a visualization aid.

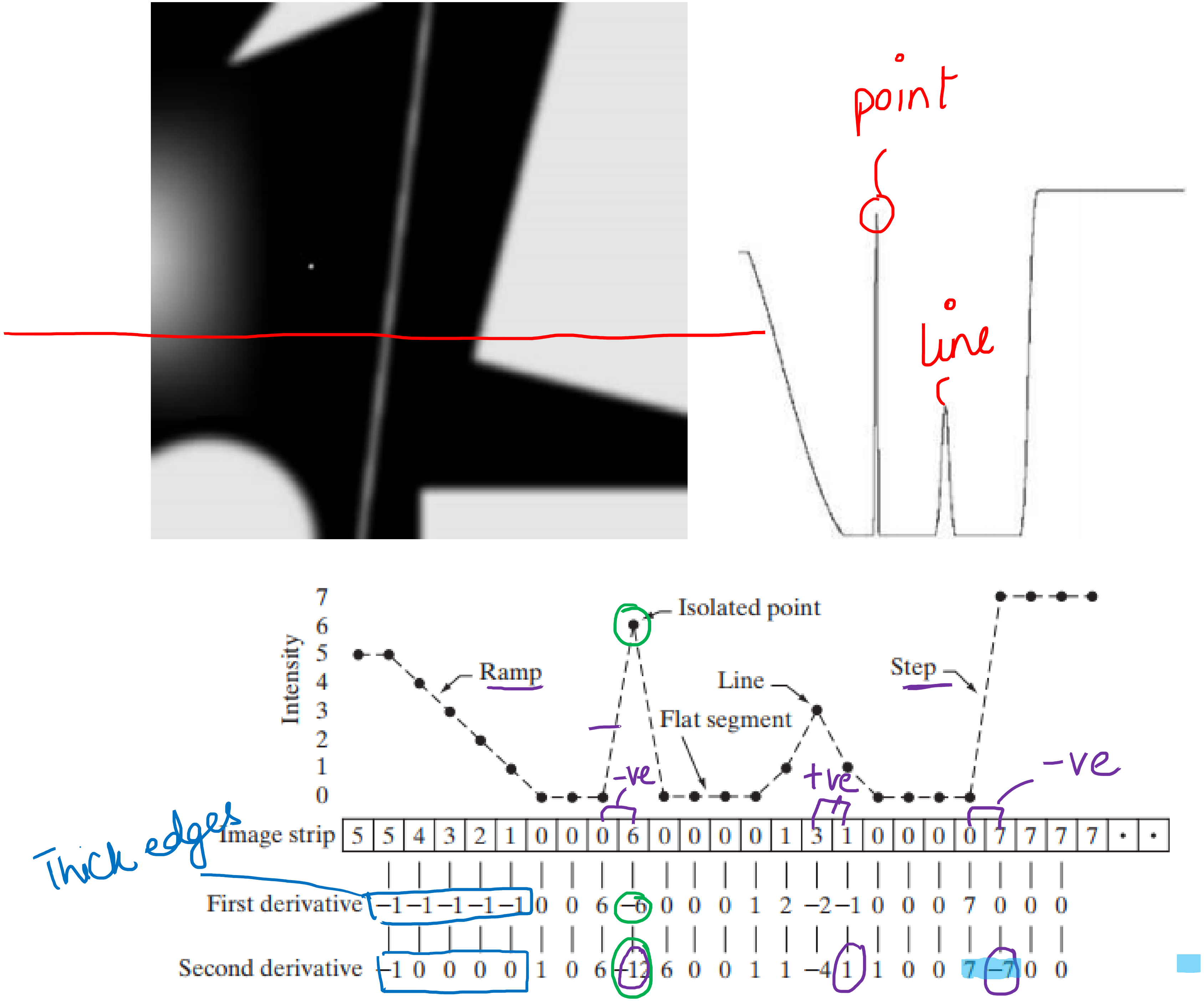



**FIGURE 10.2** (a) Image. (b) Horizontal intensity profile through the center of the image, including the isolated noise point. (c) Simplified profile (the points are joined by dashes for clarity). The image strip corresponds to the intensity profile, and the numbers in the boxes are the intensity values of the dots shown in the profile. The derivatives were obtained using Eqs. (10.2-1) and (10.2-2).

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

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

- Double edges

- We see that:
  - First order derivative produce thicker edges in an image
  - Second order derivative have high response to detail (points, noise...)
  - Second order derivative produce double-edge response at ramps/steps.
  - The sign of the second derivative can be used to determine transitions dark→light (positive) or light →dark (negative)

# Detection of discontinuities

- 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.

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

**FIGURE 10.1** A general 3 × 3 mask.

| | | |
|---|---|---|
| $w_1$ | $w_2$ | $w_3$ |
| $w_4$ | $w_5$ | $w_6$ |
| $w_7$ | $w_8$ | $w_9$ |

# Detection of Isolated Points (Using Second Order Derivative – Laplacian)

- Isotropic filters: rotation invariant
- Simplest isotropic second-order derivative operator: Laplacian

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

2-D Laplacian operation

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

x-direction

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

y-direction

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

| | | |
|---|---|---|
| 0 | 1 | 0 |
| 1 | −4 | 1 |
| 0 | 1 | 0 |

| | | |
|---|---|---|
| 1 | 1 | 1 |
| 1 | −8 | 1 |
| 1 | 1 | 1 |

| | | |
|---|---|---|
| 0 | −1 | 0 |
| −1 | 4 | −1 |
| 0 | −1 | 0 |

| | | |
|---|---|---|
| −1 | −1 | −1 |
| −1 | 8 | −1 |
| −1 | −1 | −1 |

a b
c d

**FIGURE 3.37**
(a) Filter mask used to implement Eq. (3.6-6). (b) Mask used to implement an extension of this equation that includes the diagonal terms. (c) and (d) Two other implementations of the Laplacian found frequently in practice.

# Point Detection

- The only differences that are considered of interest are those large enough (as Determined by T) to be considered isolated points.

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

$$|R| \geq T$$

where $T$ : a nonnegative threshold

Mask:

| | | |
|---|---|---|
| -1 | -1 | -1 |
| -1 | 8 | -1 |
| -1 | -1 | -1 |

Isolated point: response value 8

| | | |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 0 | 0 | 0 |

Point which is part of an line: response value 6

| | | |
|---|---|---|
| 0 | 1 | 0 |
| 0 | 1 | 0 |
| 0 | 1 | 0 |

Suppose image matrix is

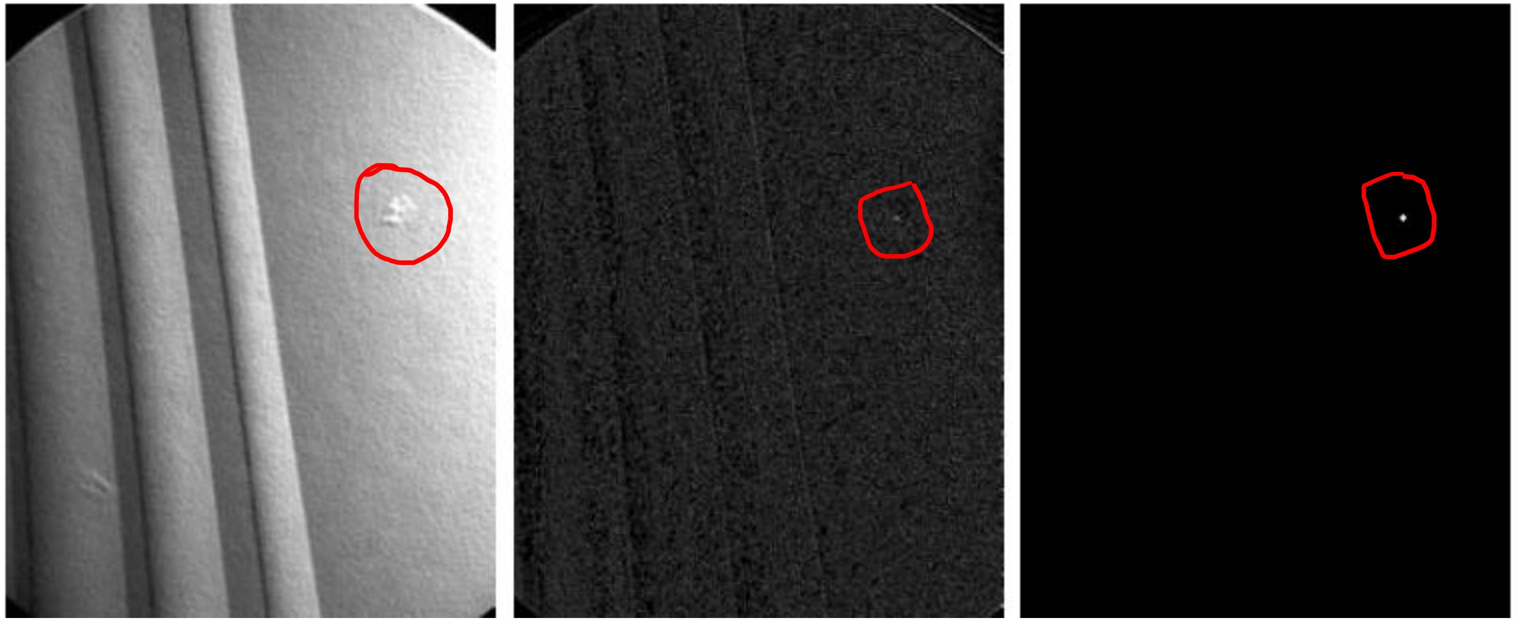
| | | |
|---|---|---|
| 100 | 100 | 100 |
| 100 | 10 | 100 |
| 100 | 100 | 100 |

if 100 then $\boxed{R = 0}$

$$R = -100 - 100 - 100 - 100 + 80 - 100 - 100 - 100 - 100$$
$$= -720 \quad |R| = 720$$

point detection

| | | |
|---|---|---|
| 1 | 1 | 1 |
| 1 | −8 | 1 |
| 1 | 1 | 1 |



a
b c d

**FIGURE 10.4** (a) Point detection (Laplacian) mask. (b) X-ray image of turbine blade with a porosity. The porosity contains a single black pixel. (c) Result of convolving the mask with the image. (d) Result of using Eq. (10.2-8) showing a single point (the point was enlarged to make it easier to see). (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°).

| | | |
|---|---|---|
| −1 | −1 | −1 |
| 2 | 2 | 2 |
| −1 | −1 | −1 |

Horizontal

| | | |
|---|---|---|
| 2 | −1 | −1 |
| −1 | 2 | −1 |
| −1 | −1 | 2 |

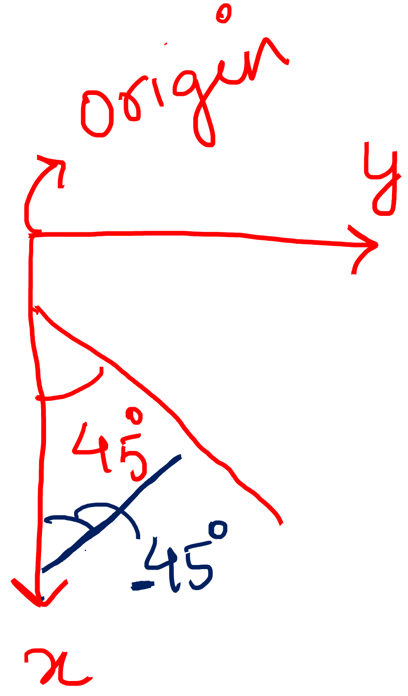
+45°

| | | |
|---|---|---|
| −1 | 2 | −1 |
| −1 | 2 | −1 |
| −1 | 2 | −1 |

Vertical

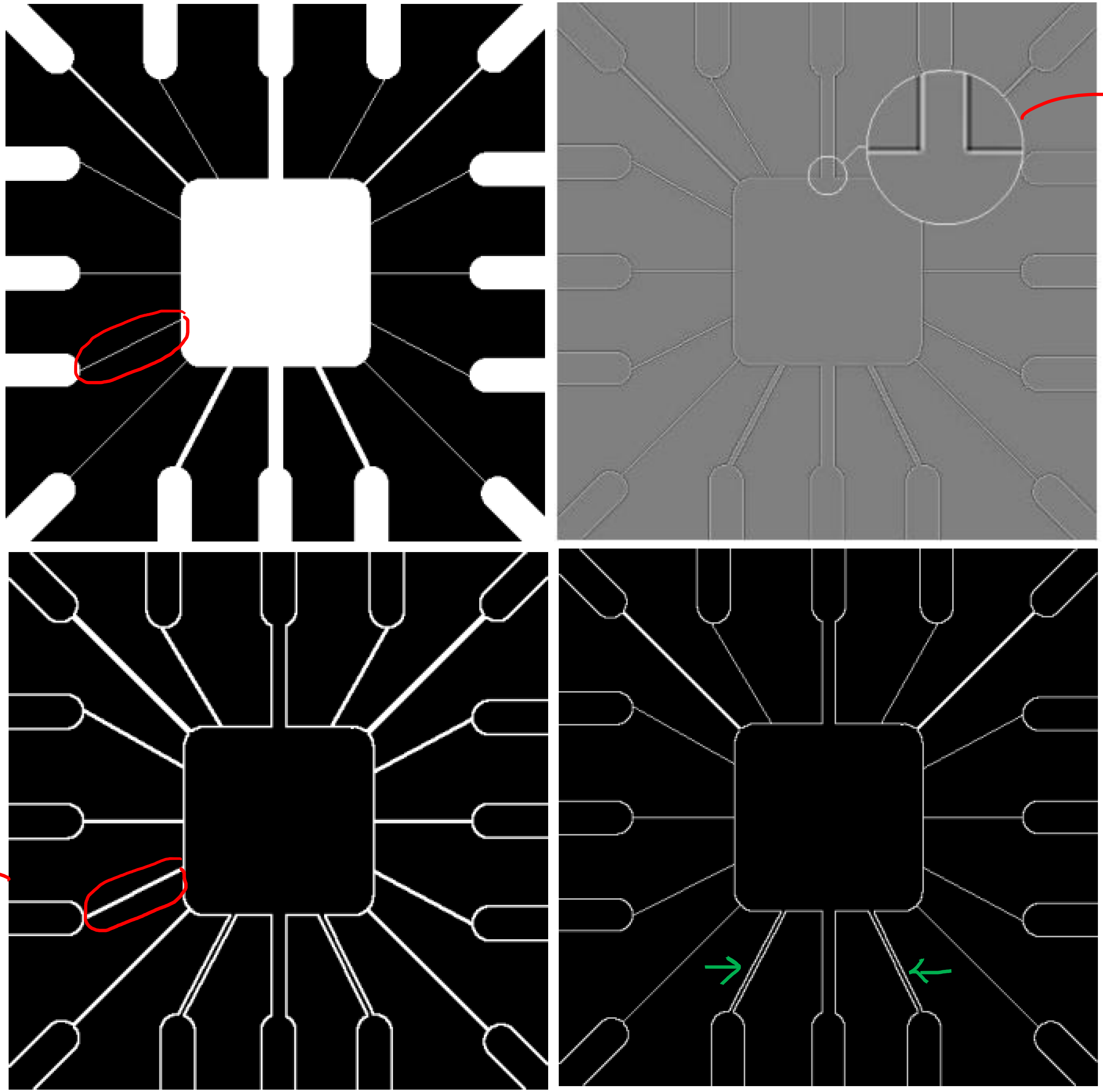
| | | |
|---|---|---|
| −1 | −1 | 2 |
| −1 | 2 | −1 |
| 2 | −1 | −1 |

−45°



3x3 filter is placed on 5 pixel wide line

R = zero



a b
c d

**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.

→ double edge effect (second order derivative)

→ use only +ve values

doubles the thickness of lines

Zero valley for lines wider than mask

# Line detection:

- Apply every masks on the image
- let R1, R2, R3, R4 denotes the response of the horizontal, +45 degree, vertical and -45 degree masks, respectively.
- if, at a certain point in the image

$$|R_i| > |R_j|, \text{ for all } j \neq i,$$

that point is said to be more likely associated with a line in the direction of

mask i.

Line is assumed 1 pixel thick.

| | | |
|---|---|---|
| -1 | -1 | 2 |
| -1 | 2 | -1 |
| 2 | -1 | -1 |

Mask
detects line
at $-45°$

| | | | |
|---|---|---|---|
| 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 0 |
| 1 | 0 | 0 | 0 |

image

| | | | |
|---|---|---|---|
| | | -2 | 6 |
| | -2 | 6 | -2 |
| -2 | 6 | -2 | |
| 6 | -2 | | |

$R = 6$ line is detected

Presence of negative value in edge detection:

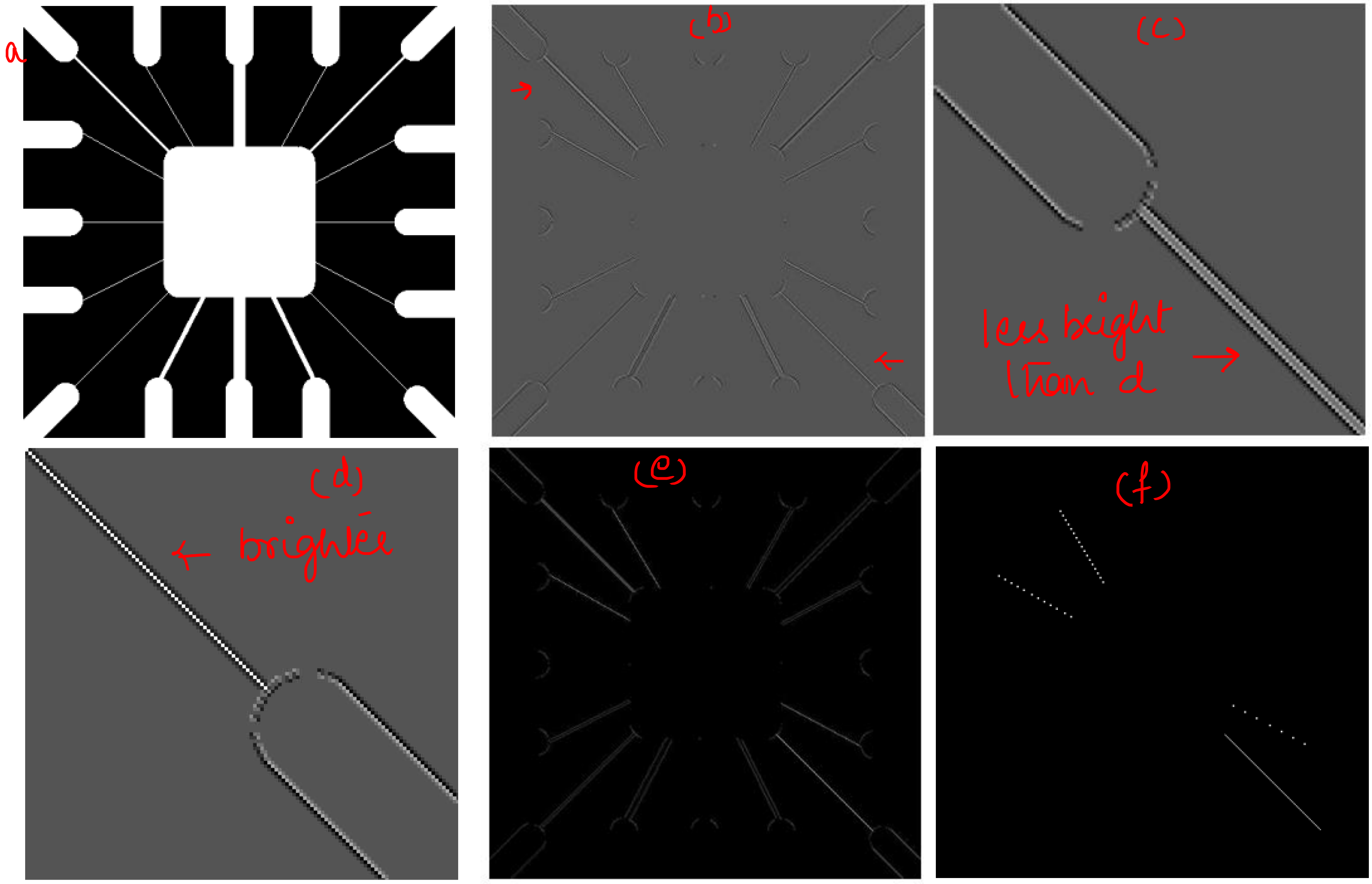
| | | |
|---|---|---|
| -1 | -1 | 2 |
| -1 | 2 | -1 |
| 2 | -1 | -1 |

One pixel thick line➔ respeonse value 6

| | | |
|---|---|---|
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |

Three pixel thick line➔ response value 2

| | | |
|---|---|---|
| 0 | 1 | 1 |
| 1 | 1 | 1 |
| 1 | 1 | 0 |




a b c
d e f

b → processing $\bar{c}$ $+45^\circ$ mask

c & d magnified section

Mask detects 1 pixel wide line

e) +ve values only

f) $|R| \geq T$

# Edge Models

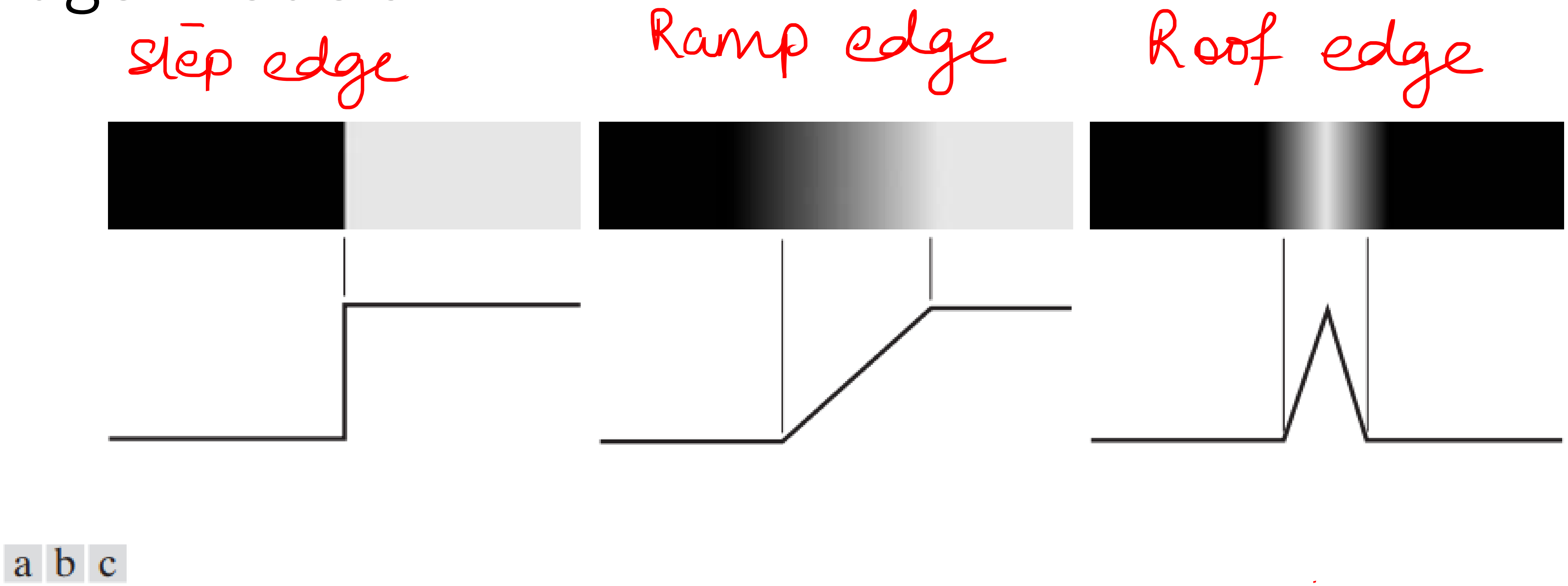
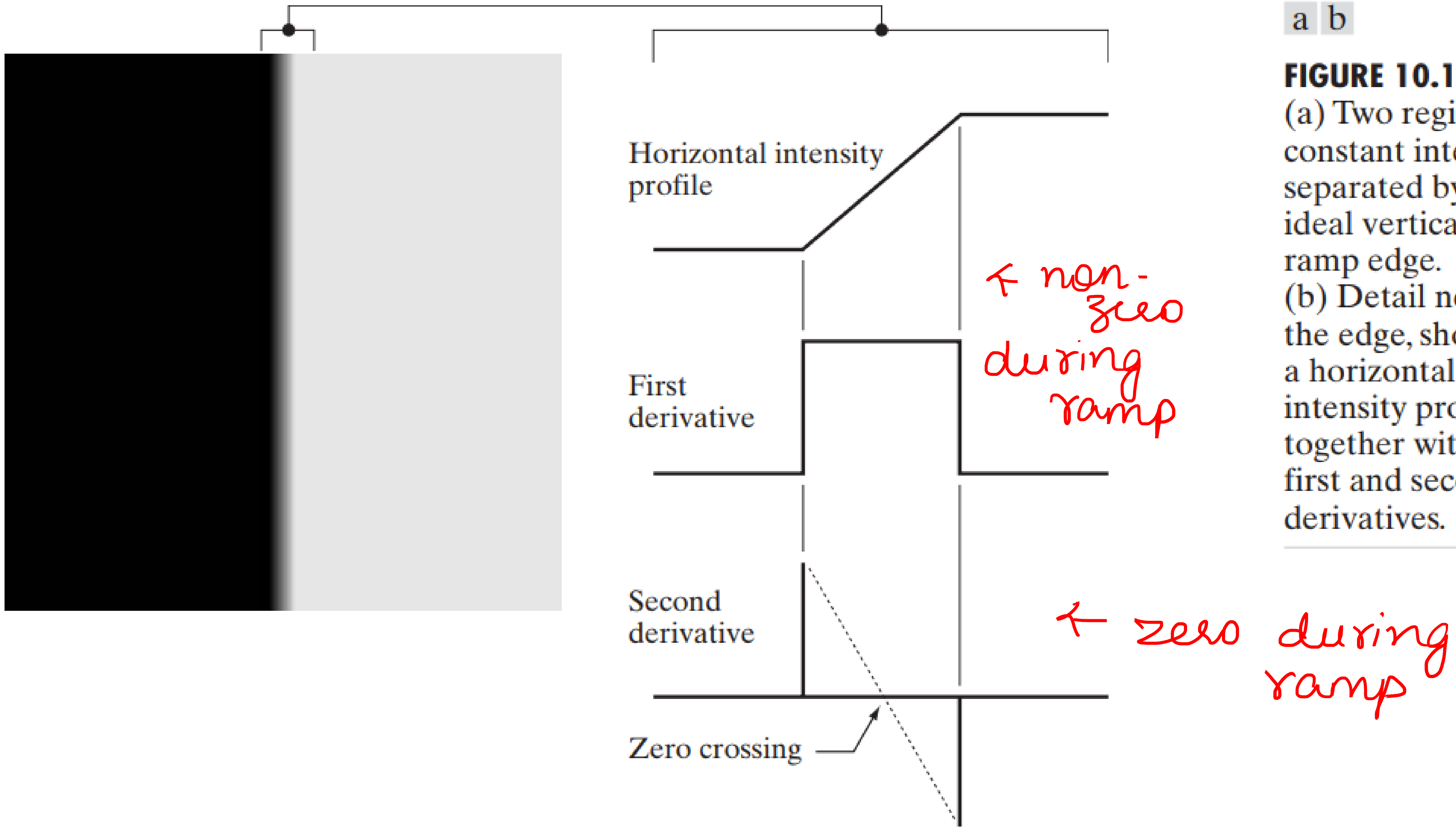



**FIGURE 10.8**
From left to right, models (ideal representations) of a step, a ramp, and a roof edge, and their corresponding intensity profiles.

# Edge Models

- Step edge: A *step edge* involves a transition between two intensity levels occurring ideally over the distance of 1 pixel.
  - Ideal one! Possible in computer synthesized images. Noisy images has gradual change in intensities.-ramp model of edge.
- Ramp edge: The slope of the ramp is inversely proportional to the degree of blurring in the edge.
- Roof edges : are models of lines through a region, with the base (width) of a roof edge being determined by the thickness and sharpness of the line.
  - In the limit, when its base is 1 pixel wide, a roof edge is really nothing more than a 1-pixel-thick line running through a region in an image.
  - Roof edges arise, for example, in range imaging, when thin objects (such as pipes) are closer to the sensor than their equidistant background (such as walls). The pipes appear brighter and thus create an image similar to the model in Fig. 10.8(c).




a b

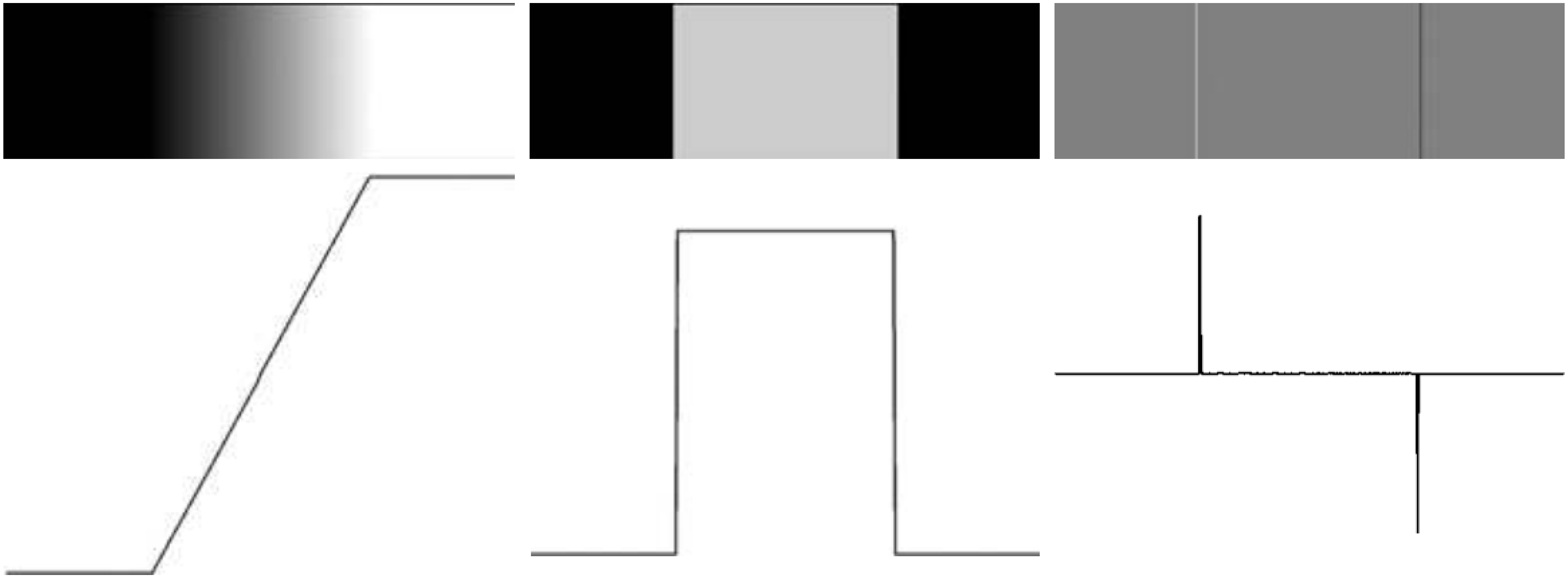
**FIGURE 10.10** (a) Two regions of constant intensity separated by an ideal vertical ramp edge. (b) Detail near the edge, showing a horizontal intensity profile, together with its first and second derivatives.

- We conclude from these observations that the
  - *magnitude* of the first derivative can be used to detect the presence of an edge at a point in an image.
  - Similarly, the *sign* of the second derivative can be used to determine whether an edge pixel lies on the dark or light side of an edge.
  - second derivative around an edge:

    (1) it produces two values for every edge in an image (an undesirable feature);

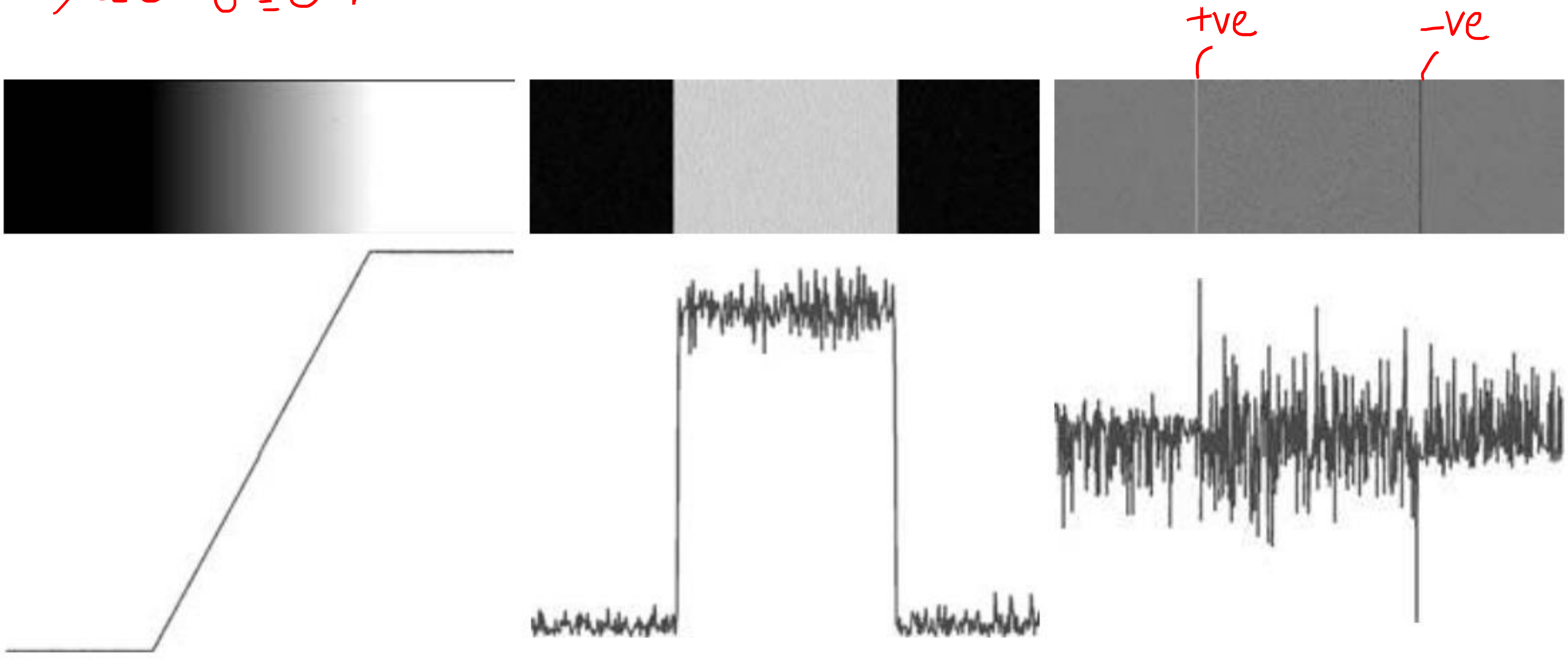
    (2) its zero crossings can be used for locating the centers of thick edges, as we show later in this section.

no noise



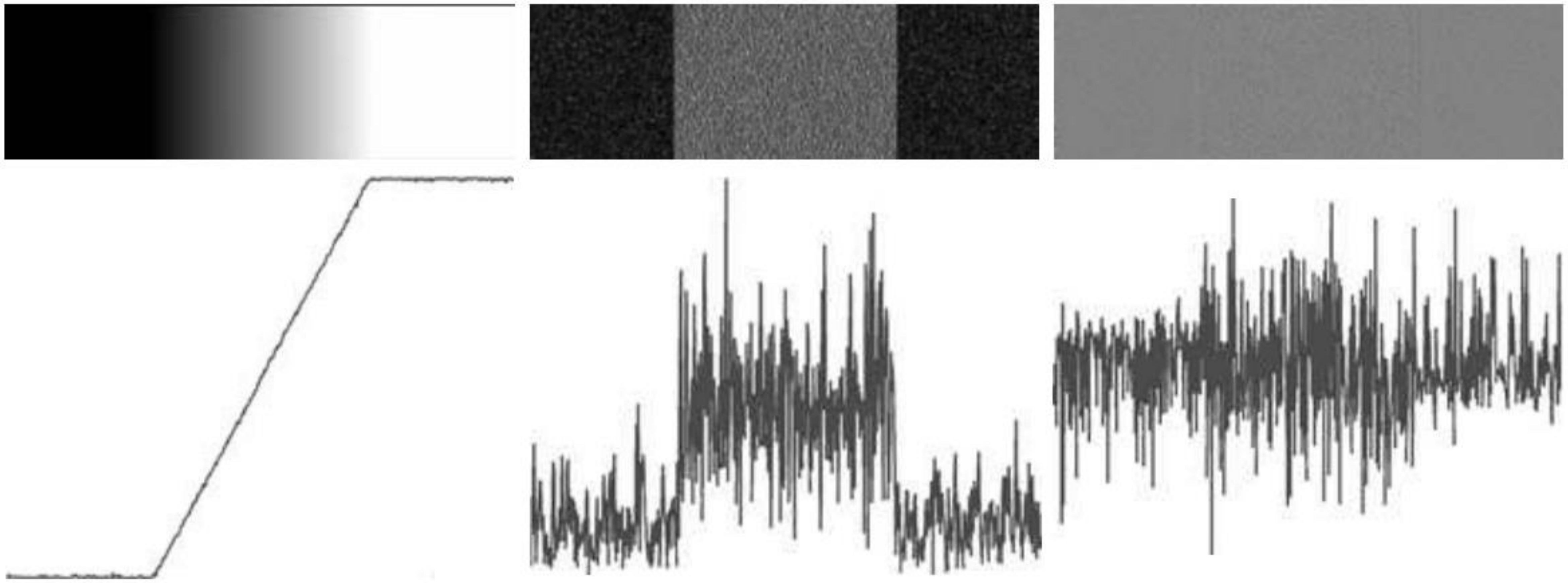
Gaussian noise

$\mu = 0 \quad \sigma = 0.1$



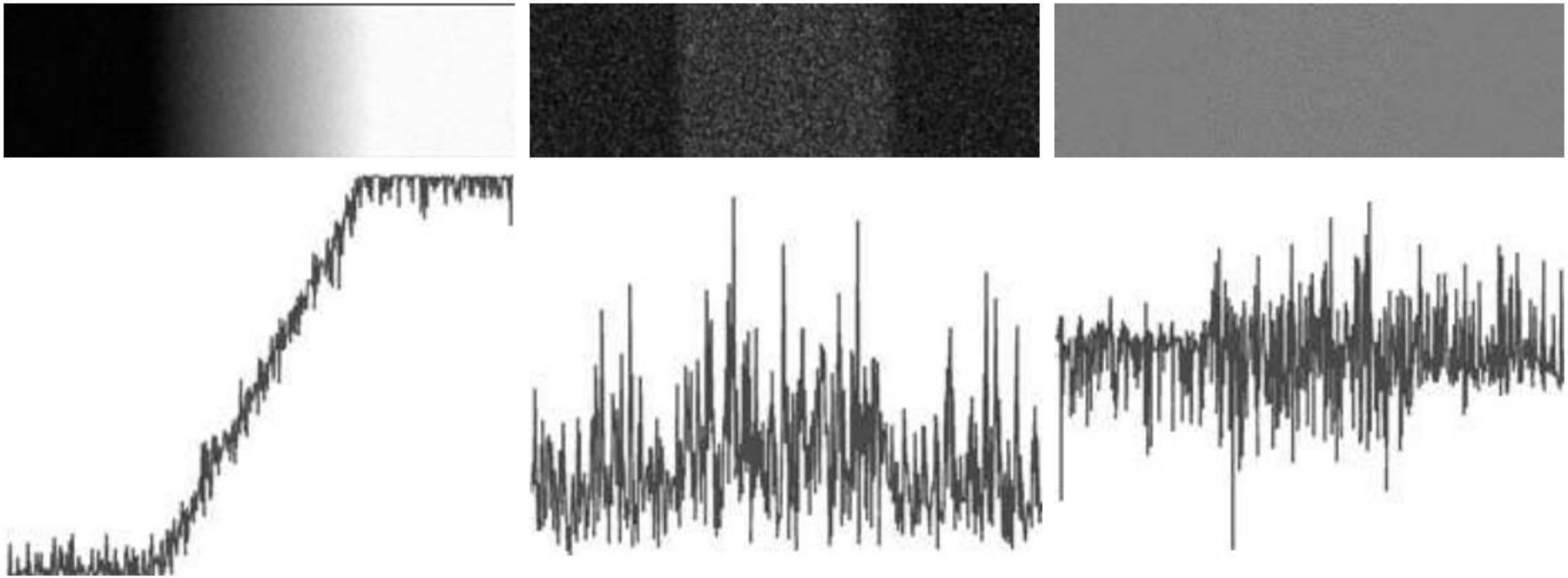
AGN

$\mu = 0 \quad \sigma = 1.0$



AGN

$\mu = 0 \quad \sigma = 10.0$



- We conclude this section by noting that there are three fundamental steps performed in edge detection:

1. *Image smoothing for noise reduction.* The need for this step is simply illustrated by the results in the second and third columns of previous figures.
2. *Detection of edge points.* As mentioned earlier, this is a local operation that extracts from an image all points that are potential candidates to become edge points.
3. *Edge localization.* The objective of this step is to select from the candidate edge points only the points that are true members of the set of points comprising an edge.

# Basic Edge Detection

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

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

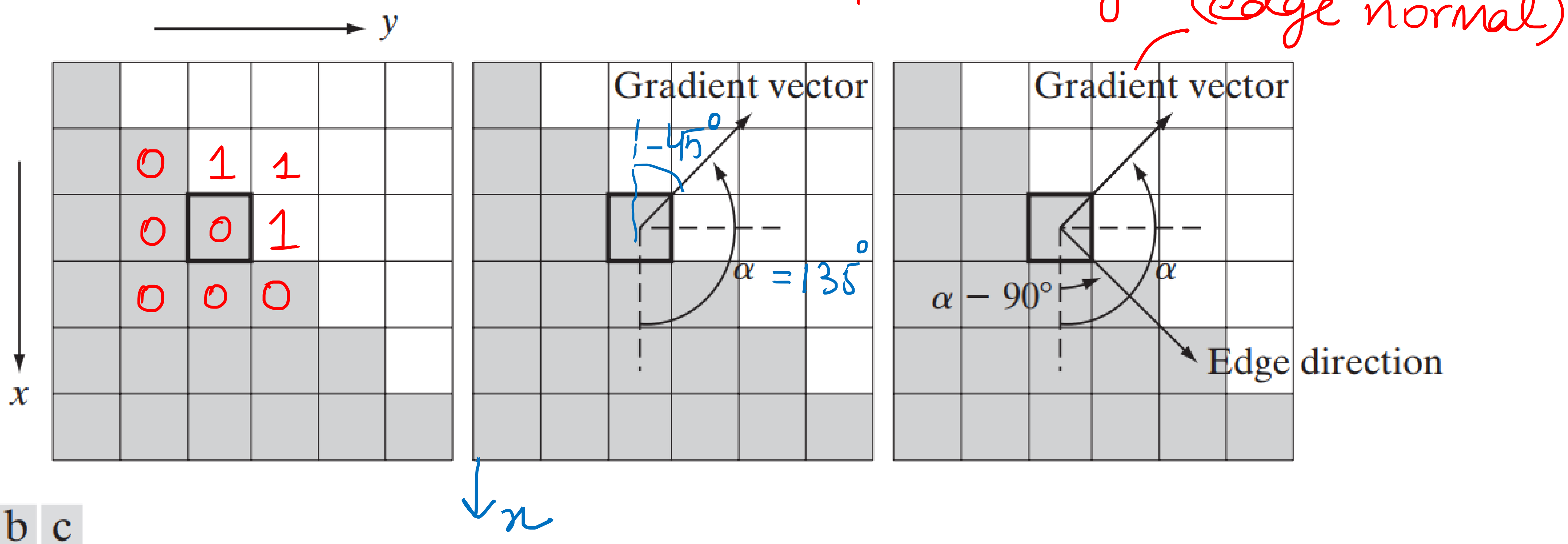
  - The magnitude of this vector:

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

  - The direction of this vector:

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

edge unit normal $= \dfrac{\text{Gradient vector}}{\text{Gradient Mag}}$




a b c

**FIGURE 10.12** Using the gradient to determine edge strength and direction at a point. Note that the edge is perpendicular to the direction of the gradient vector at the point where the gradient is computed. Each square in the figure represents one pixel.

# Detection of edges

To get partial derivative in x direction

Subtract the pixels in the top row of the neighborhood from the pixels in the bottom row

| | | |
|---|---|---|
| 0 | 1 | 1 |
| 0 | 0 | 1 |
| 0 | 0 | 0 |

We get

$$g_x = \frac{\partial f}{\partial x} = (0-0)+(0-1)+(0-1) = -2$$

# Detection of edges

To get partial derivative in y direction

Subtract the pixels in the left column from the pixels in the right column.

| | | |
|---|---|---|
| 0 | 1 | 1 |
| 0 | 0 | 1 |
| 0 | 0 | 0 |

We get

$$g_x = \frac{\partial f}{\partial y} = (1-0)+(1-0)+(0-0) = 2$$

# Detection of edges

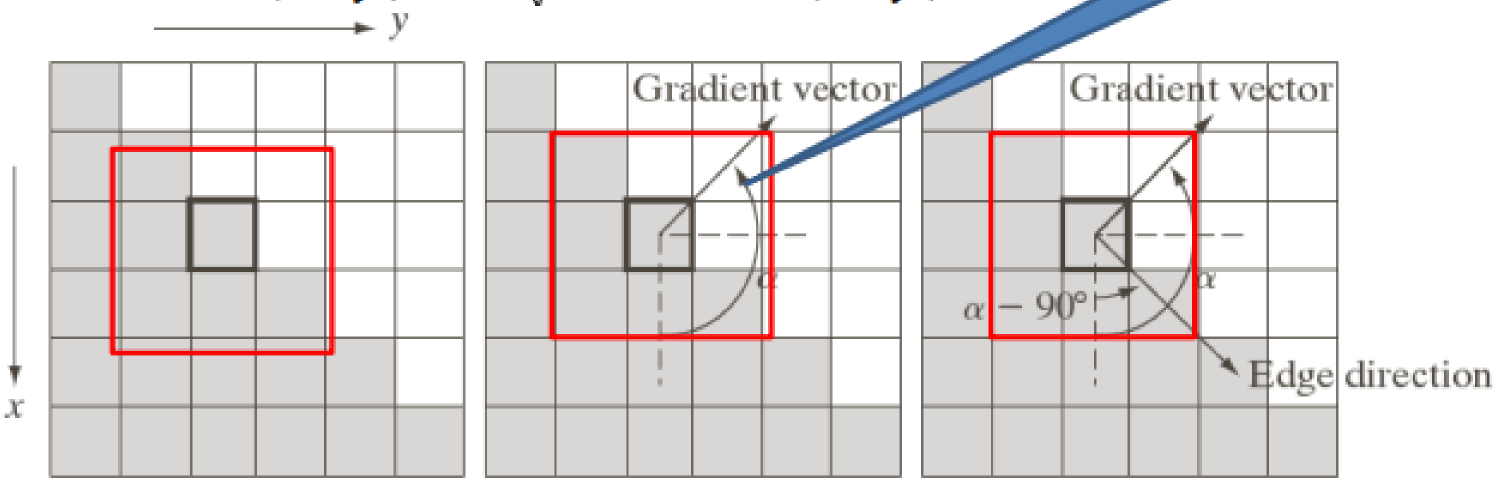
At the point of question, we get

Edge normal
If normalized to unit length then
Edge unit normal

$$\nabla f = \begin{bmatrix} g_x \\ g_y \end{bmatrix} = \begin{bmatrix} \dfrac{\partial f}{\partial x} \\ \dfrac{\partial f}{\partial y} \end{bmatrix} = \begin{bmatrix} -2 \\ 2 \end{bmatrix}$$

Which is same as 135º measured from the +ve axis

$$M(x,y) = 2\sqrt{2} \quad and \quad \alpha(x,y) = -45^{o}$$

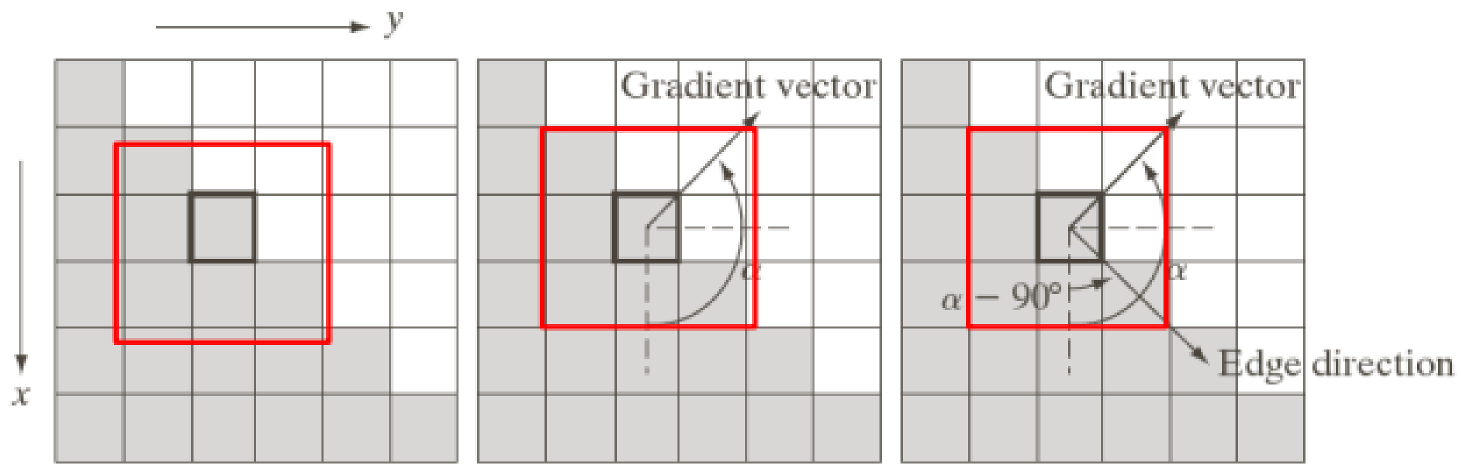


# Detection of edges

Edge at a point is orthogonal to the gradient vector at that point.

So direction of the angle of the edge in this example is (α-90)=45

All edge points in the example shown below have same gradient so the entire segment is in the same direction.



# Gradient Operator

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

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

a b

**FIGURE 10.13** One-dimensional masks used to implement Eqs.

| |
|---|
| −1 |
| 1 |

| | |
|---|---|
| −1 | 1 |

image →

| | | |
|---|---|---|
| $z_1$ | $z_2$ | $z_3$ |
| $z_4$ | $z_5$ | $z_6$ |
| $z_7$ | $z_8$ | $z_9$ |

$$\frac{\partial f}{\partial x} = z_9 - z_5 \qquad \frac{\partial f}{\partial y} = (z_8 - z_6)$$

Roberts cross-gradient operators →

| | |
|---|---|
| −1 | 0 |
| 0 | 1 |

| | |
|---|---|
| 0 | −1 |
| 1 | 0 |

Roberts

$$\frac{\partial f}{\partial x} = (z_7 + z_8 + z_9) - (z_1 + z_2 + z_3)$$

$$\frac{\partial f}{\partial y} = (z_3 + z_6 + z_9) - (z_1 + z_4 + z_7)$$

Prewitt operators →

| | | |
|---|---|---|
| −1 | −1 | −1 |
| 0 | 0 | 0 |
| 1 | 1 | 1 |

| | | |
|---|---|---|
| −1 | 0 | 1 |
| −1 | 0 | 1 |
| −1 | 0 | 1 |

Prewitt

$$\frac{\partial f}{\partial x} = (z_7 + 2z_8 + z_9) - (z_1 + 2z_2 + z_3)$$

$$\frac{\partial f}{\partial y} = (z_3 + 2z_6 + z_9) - (z_1 + 2z_4 + z_7)$$

Sobel operators →

(provides image smoothing)

(better noise suppression)

| | | |
|---|---|---|
| −1 | −2 | −1 |
| 0 | 0 | 0 |
| 1 | 2 | 1 |

| | | |
|---|---|---|
| −1 | 0 | 1 |
| −2 | 0 | 2 |
| −1 | 0 | 1 |

Sobel

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

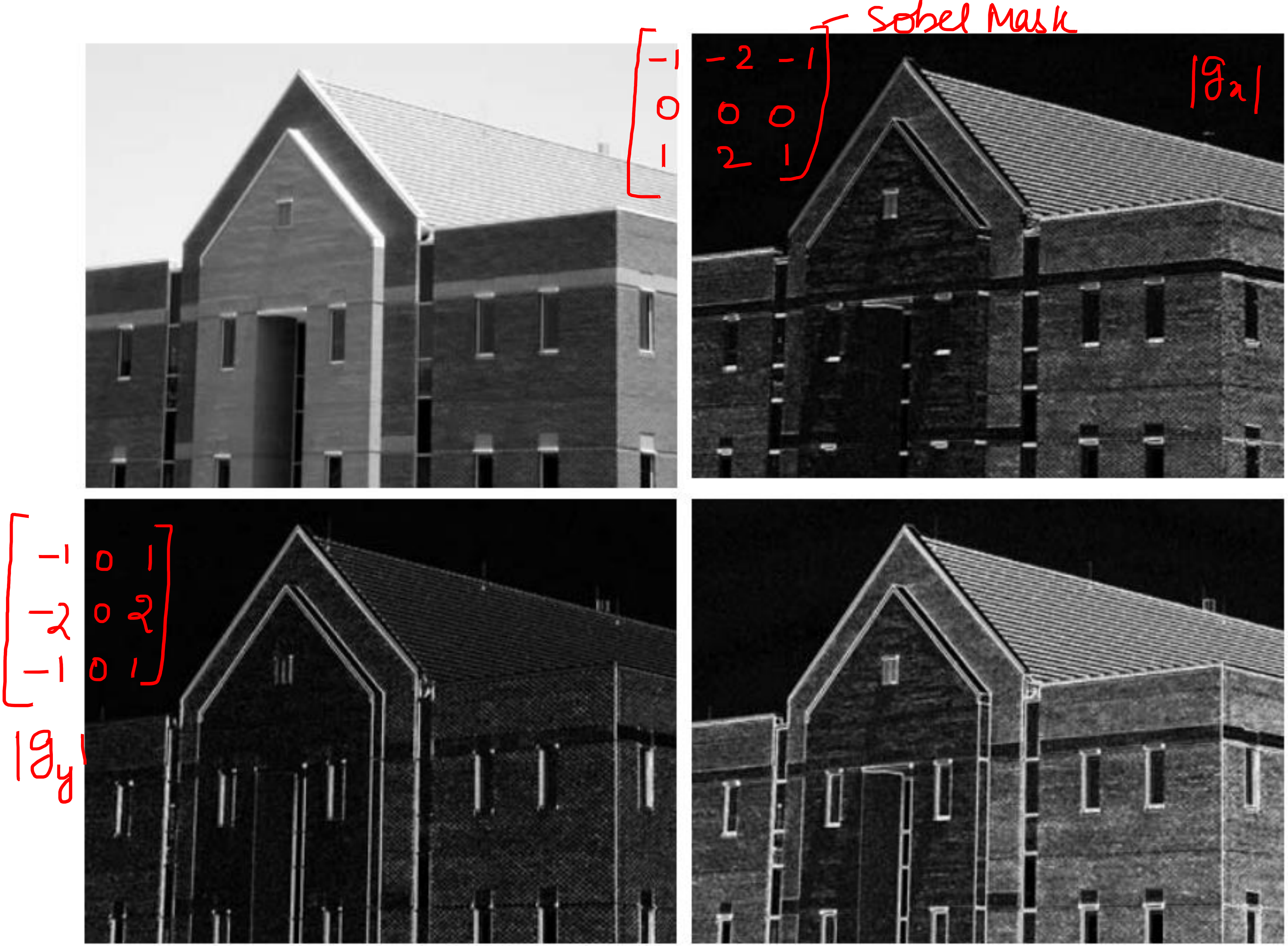
a b
c d

**FIGURE 10.9** Prewitt and Sobel masks for detecting diagonal edges.

- Using any of mask two partial derivatives $g_x$ and $g_y$ is obtained.
- Used to determine the magnitude and direction of the edge.

$$M(x, y) \approx |g_x| + |g_y|$$

- Terminology → Edge map: when referring to an image whose principle features are edges such as gradient magnitude images.




| | |
|---|---|
| a | b |
| c | d |

**FIGURE 10.16** (a) Original image of size $834 \times 1114$ pixels, with intensity values scaled to the range [0, 1]. (b) $|g_x|$, the component of the gradient in the $x$-direction, obtained using the Sobel mask in Fig. 10.14(f) to filter the image. (c) $|g_y|$, obtained using the mask in Fig. 10.14(g). (d) The gradient image, $|g_x| + |g_y|$.

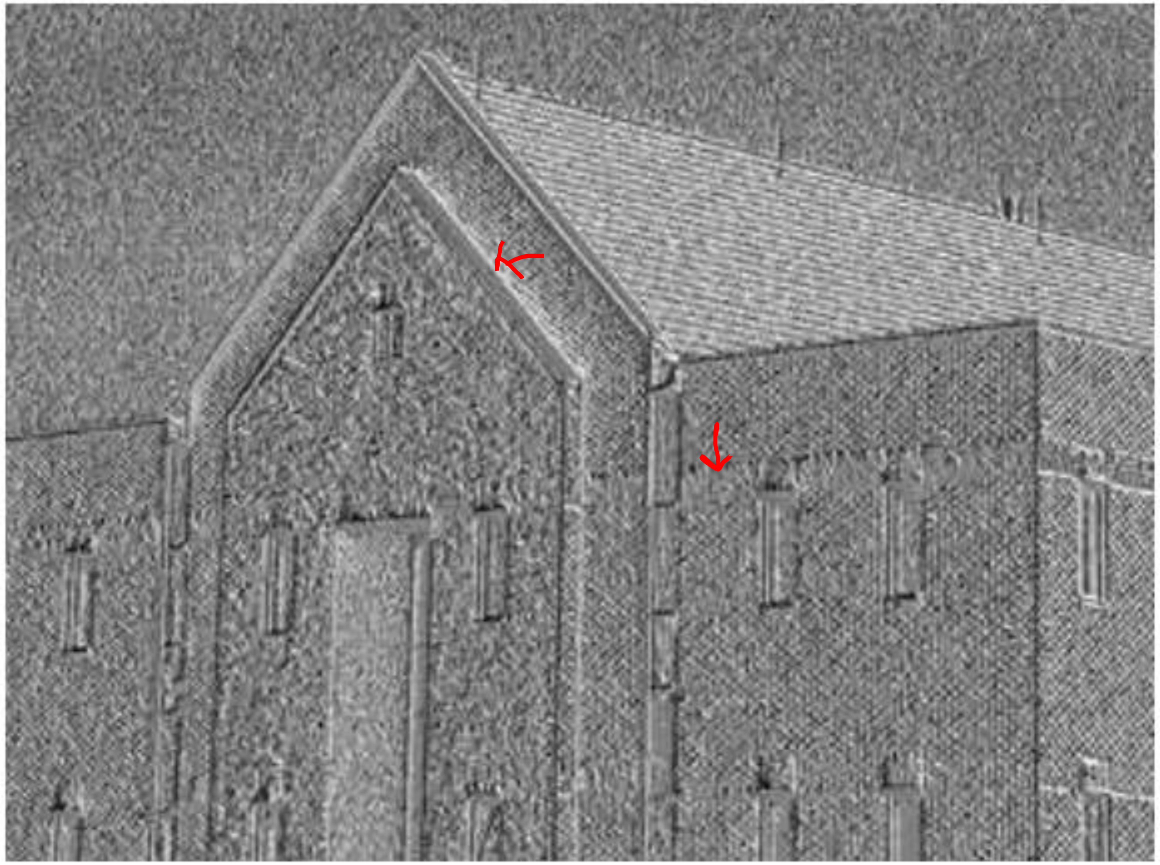
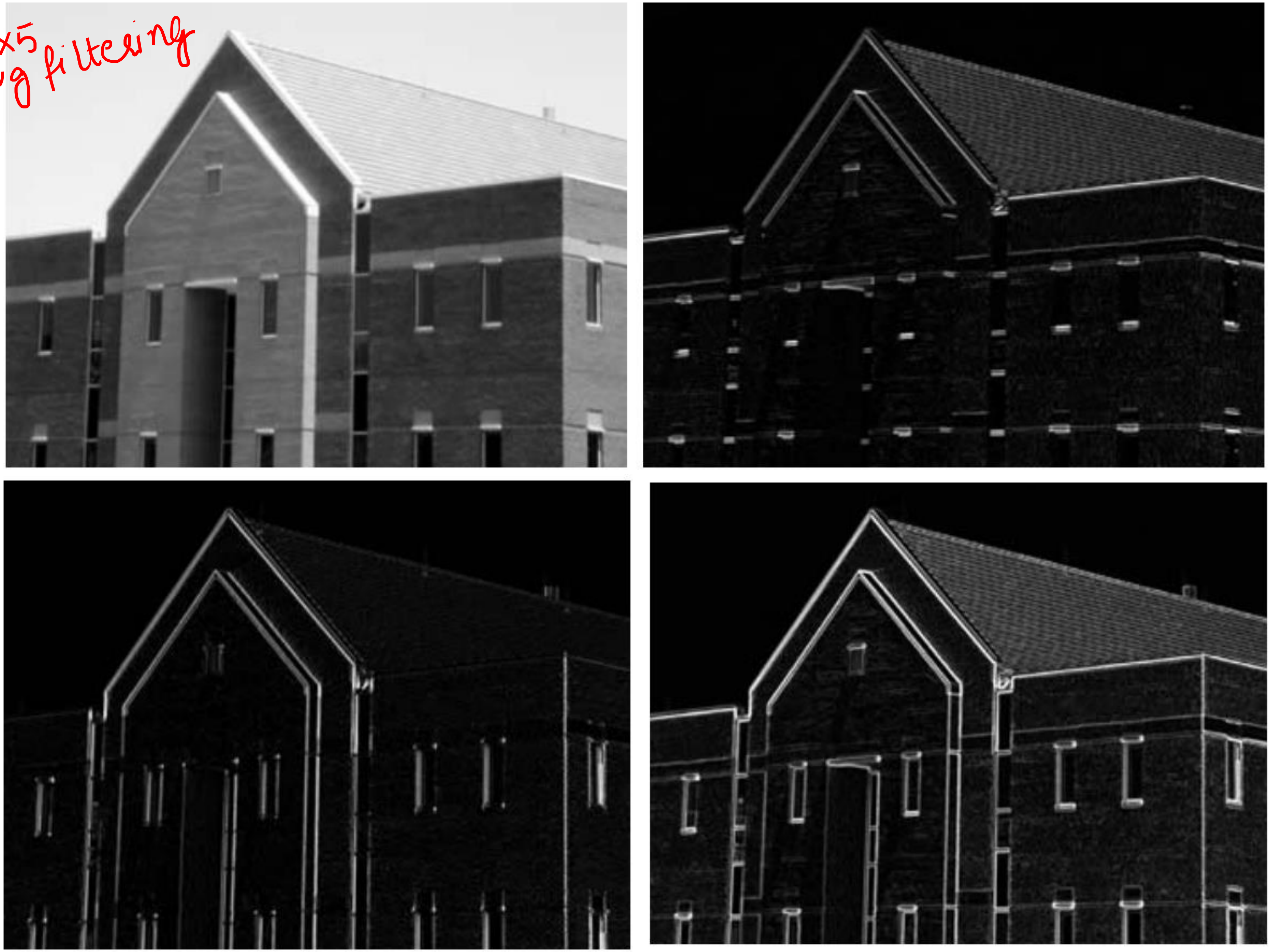


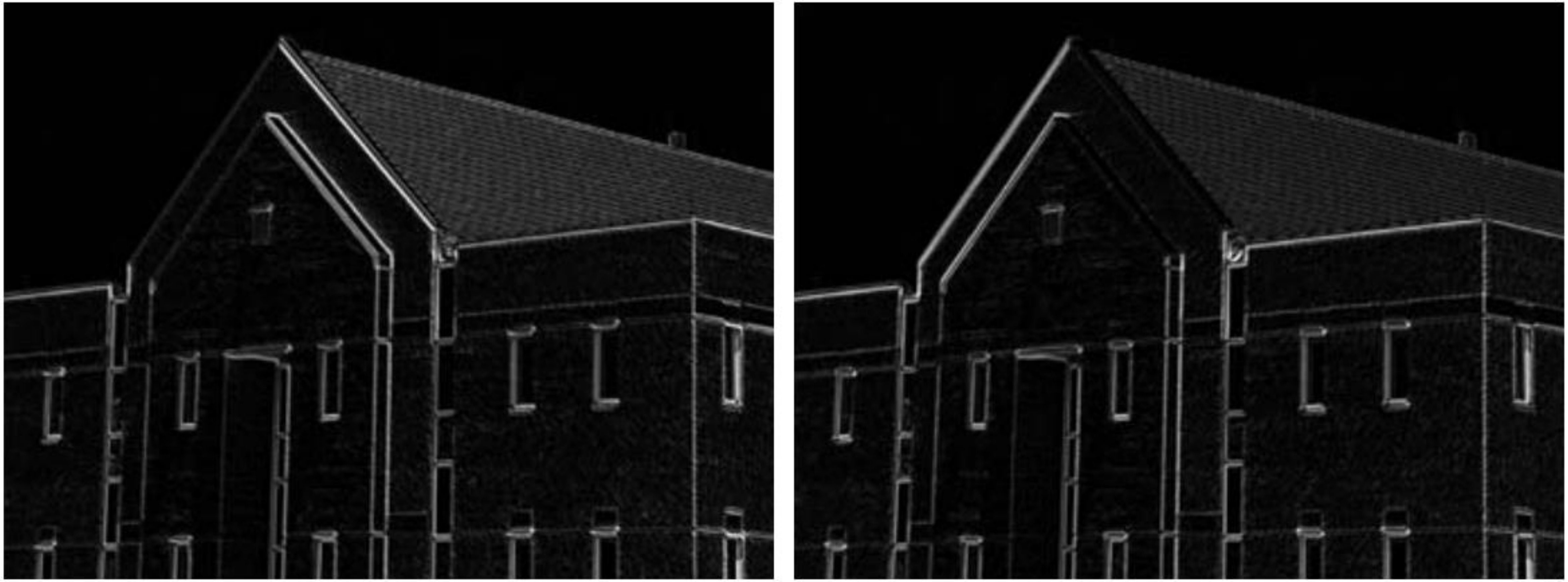
**FIGURE 10.17** Gradient angle image computed using Eq. (10.2-11). Areas of constant intensity in this image indicate that the direction of the gradient vector is the same at all the pixel locations in those regions.

3x5 avg filtering



a b
c d

**FIGURE 10.18** Same sequence as in Fig. 10.16, but with the original image smoothed using a $5 \times 5$ averaging filter prior to edge detection.



| | | |
|---|---|---|
| 0 | 1 | 2 |
| −1 | 0 | 1 |
| −2 | −1 | 0 |

| | | |
|---|---|---|
| −2 | −1 | 0 |
| −1 | 0 | 1 |
| 0 | 1 | 2 |

**FIGURE 10.19**
Diagonal edge detection. (a) Result of using the mask in Fig. 10.15(c). (b) Result of using the mask in Fig. 10.15(d). The input image in both cases was Fig. 10.18(a).