Linear Spatial Filtering

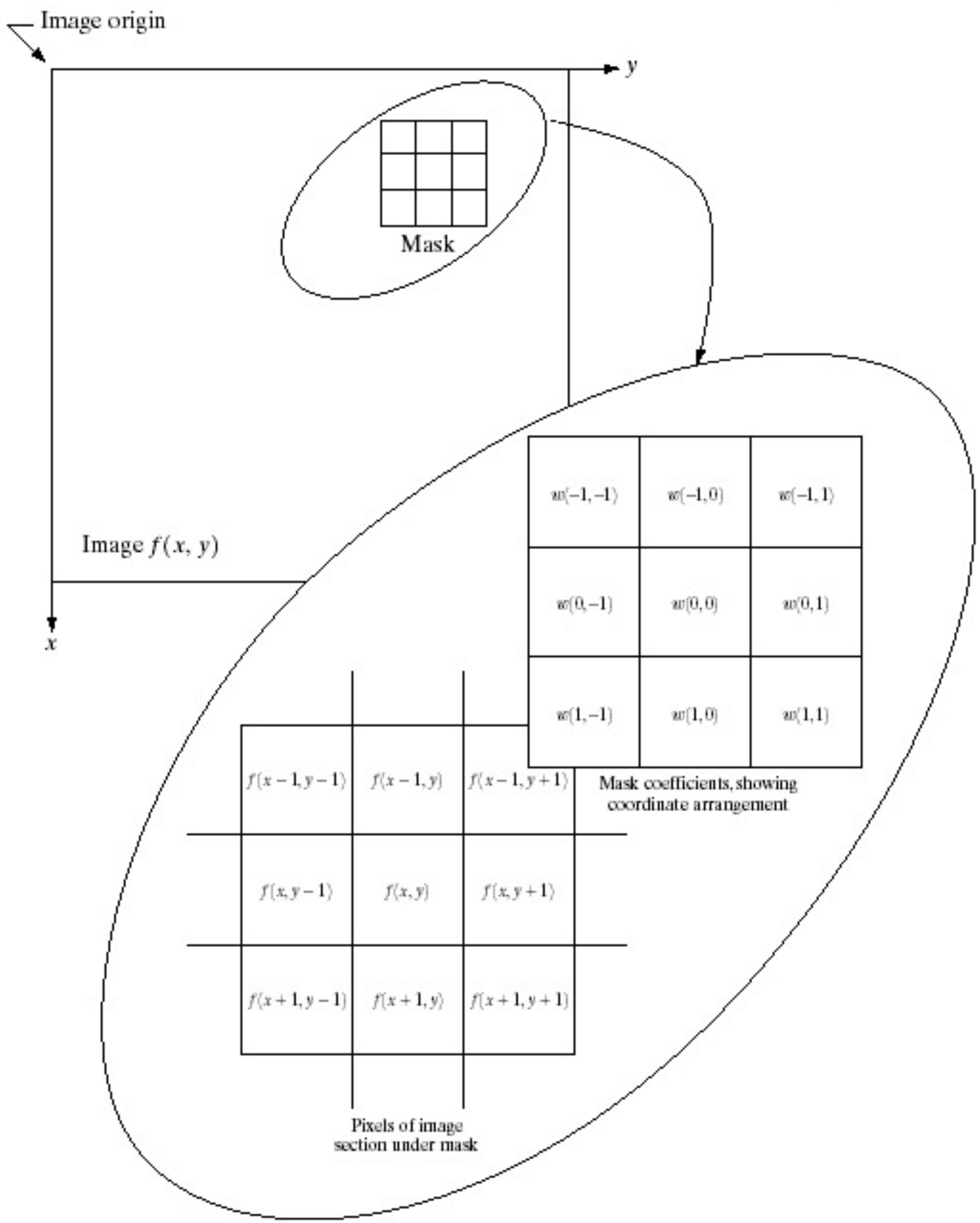


FIGURE 3.32 The mechanics of spatial filtering. The magnified drawing shows a 3×3 mask and the image section directly under it; the image section is shown displaced out from under the mask for ease of readability.

Filter Output $R(x,y) = w(-1,-1)f(x-1,y-1) + w(-1,0)f(x-1,y) + \dots + w(0,0)f(x,y) + \dots$

$$+ w(1,0)f(x+1,y) + w(1,1)f(x+1,y+1)$$

Filter Output $R(x,y) = w(-1,-1)f(x-1,y-1) + w(-1,0)f(x-1,y) + \dots + w(0,0)f(x,y) + \dots$
$+ w(1,0)f(x+1,y) + w(1,1)f(x+1,y+1)$

Let mask be of size $m \times n$ where $m = 2a+1$ and $n = 2b+1$ where a and b are nonnegative integers so that m and n are odd integers.

Filter Output $R(x,y) = w(-1,-1)f(x-1,y-1) + w(-1,0)f(x-1,y) + \cdots + w(0,0)f(x,y) + \cdots$
$+ w(1,0)f(x+1,y) + w(1,1)f(x+1,y+1)$

Let mask be of size $m \times n$ where $m = 2a+1$ and $n = 2b+1$ where a and b are nonnegative integers so that m and n are odd integers.

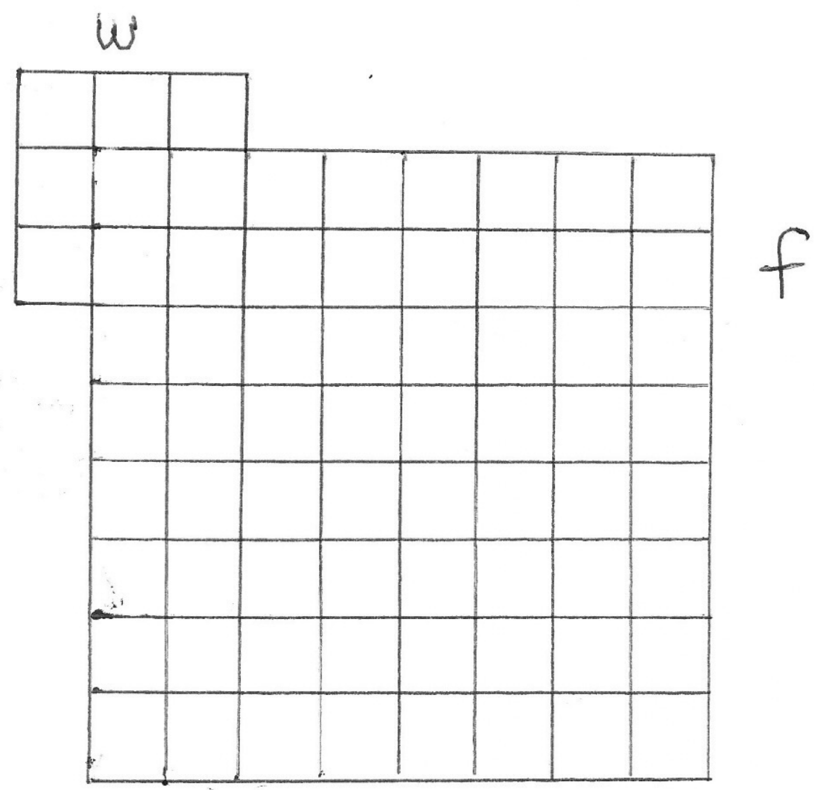
For an image f of size $M \times N$ and a filter mask w of size $m \times n$, the output image is given by

$$g(x,y) = \sum_{s=-a}^{a} \sum_{t=-b}^{b} w(s,t) f(x+s, y+t)$$

for $x = 0, 1, \cdots, M-1$ and $y = 0, 1, \cdots, N-1$

This computation is called convolution.

Borders – the output $g(x,y)$ cannot be computed properly for locations (x,y) near the border of the input image f because part of the mask will extend beyond the edge of the input image



Solutions

1) Only compute $g(x,y)$ for (x,y) locations where all terms in the convolution sum can be computed. The output image g will be smaller than the input image f.

Solutions

1) Only compute $g(x,y)$ for (x,y) locations where all terms in the convolution sum can be computed. The output image g will be smaller than the input image f.

2) Compute the convolution sum for every location (x,y) in the input image but only use the terms in the convolution sum that can be computed. The output image g will be the same size as the input image f but pixels near the boundary of g will be the result of using a partial mask.

3) Pad the input image by replicating rows and columns before filtering. The output image g will be the same size as the input image f (before padding), but pixels near the boundary of g will be the result of processing some replicated (not real) pixel values.

$$\begin{array}{cccccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & & \cdots \\ 2 & 3 & 4 & 5 & 6 & 7 & 8 & 1 & & \\ 3 & 4 & 5 & 6 & 7 & 8 & 9 & 1 & & f \\ & & \vdots & & & & & & & \end{array}$$

3) Pad the input image by replicating rows and columns before filtering. The output image g will be the same size as the input image f (before padding), but pixels near the boundary of g will be the result of processing some replicated (not real) pixel values.

padded f

$$
\begin{array}{ccccccccccc}
1 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & & \\
1 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & & \\
2 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 1 & & \cdots \\
3 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 1 & & f \\
 & & & \vdots & & & & & & &
\end{array}
$$

Smoothing (blurring, averaging, lowpass) Linear Filters

Smoothing linear filters are defined by masks that compute an average or weighted average of the pixels in the input image under the mask.

Smoothing (blurring, averaging, lowpass) Linear Filters

Smoothing linear filters are defined by masks that compute an average or weighted average of the pixels in the input image under the mask.

Ex

$$\underbrace{\frac{1}{9} \times \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}}_{M_A} \qquad \frac{1}{16} \times \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$$

Smoothing (blurring, averaging, lowpass) Linear Filters

Smoothing linear filters are defined by masks that compute an average or weighted average of the pixels in the input image under the mask.

Ex

$$\frac{1}{9} \times \begin{array}{|c|c|c|}\hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline \end{array} \qquad \frac{1}{16} \times \begin{array}{|c|c|c|}\hline 1 & 2 & 1 \\ \hline 2 & 4 & 2 \\ \hline 1 & 2 & 1 \\ \hline \end{array}$$

M_A

The coefficients of smoothing filters are usually all nonnegative. The coefficients are usually normalized to sum to one so that the filter does not make the image brighter or darker.

Smoothing filters reduce noise and blur edges

Ex Noise reduction using M_A

Input

100	100	100	100	100
100	100	100	100	100
100	100	190	100	100
100	100	100	100	100
100	100	100	100	100

Smoothing filters reduce noise and blur edges

Ex Noise reduction using M_A

Input

100	100	100	100	100
100	100	100	100	100
100	100	190	100	100
100	100	100	100	100
100	100	100	100	100

Output

100	100	100	100	100
100	110	110	110	100
100	110	110	110	100
100	110	110	110	100
100	100	100	100	100

Smoothing filters reduce noise and blur edges

Ex Noise reduction using M_A

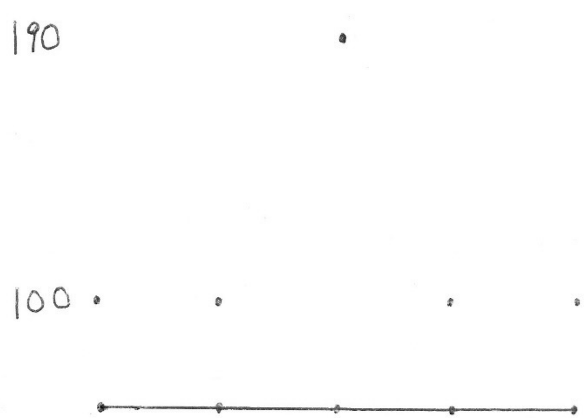
Input

100	100	100	100	100
100	100	100	100	100
100	100	190	100	100
100	100	100	100	100
100	100	100	100	100

Output

100	100	100	100	100
100	110	110	110	100
100	110	110	110	100
100	110	110	110	100
100	100	100	100	100

Middle Row Input



Smoothing filters reduce noise and blur edges

Ex Noise reduction using M_A

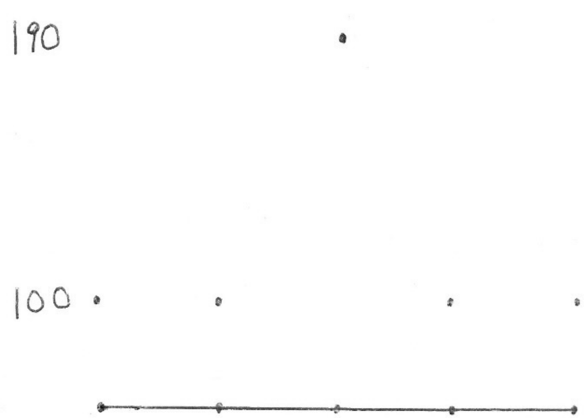
Input

100	100	100	100	100
100	100	100	100	100
100	100	190	100	100
100	100	100	100	100
100	100	100	100	100

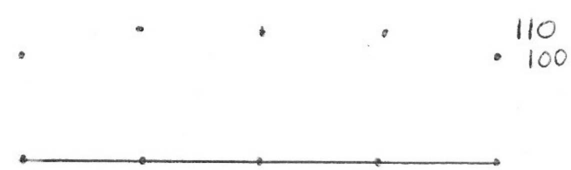
Output

100	100	100	100	100
100	110	110	110	100
100	110	110	110	100
100	110	110	110	100
100	100	100	100	100

Middle Row Input



Middle Row Output



Ex Edge blur due to M_A

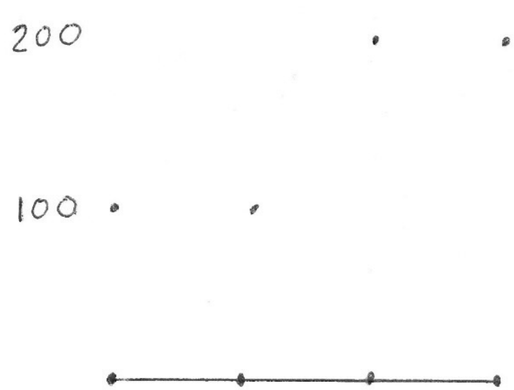
Input

100	100	200	200
100	100	200	200
100	100	200	200
100	100	200	200

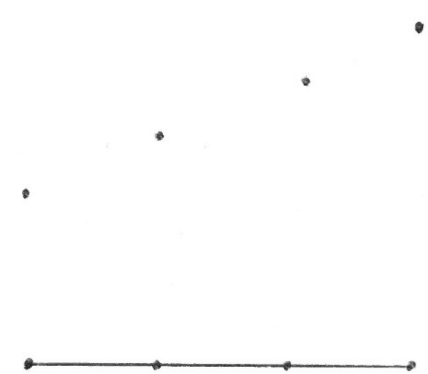
Output

100	133	167	200
100	133	167	200
100	133	167	200
100	133	167	200

Input Row



Output Row



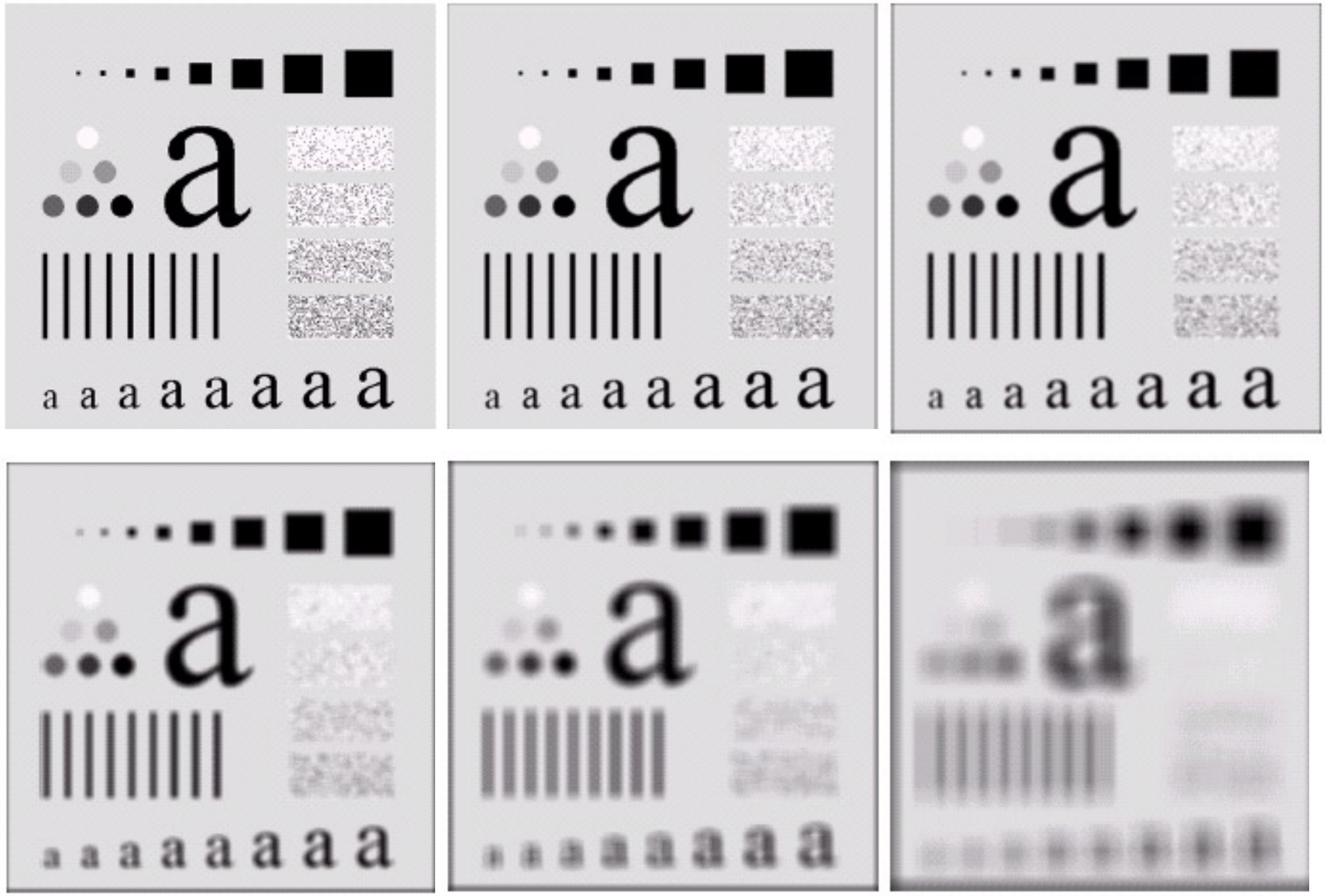
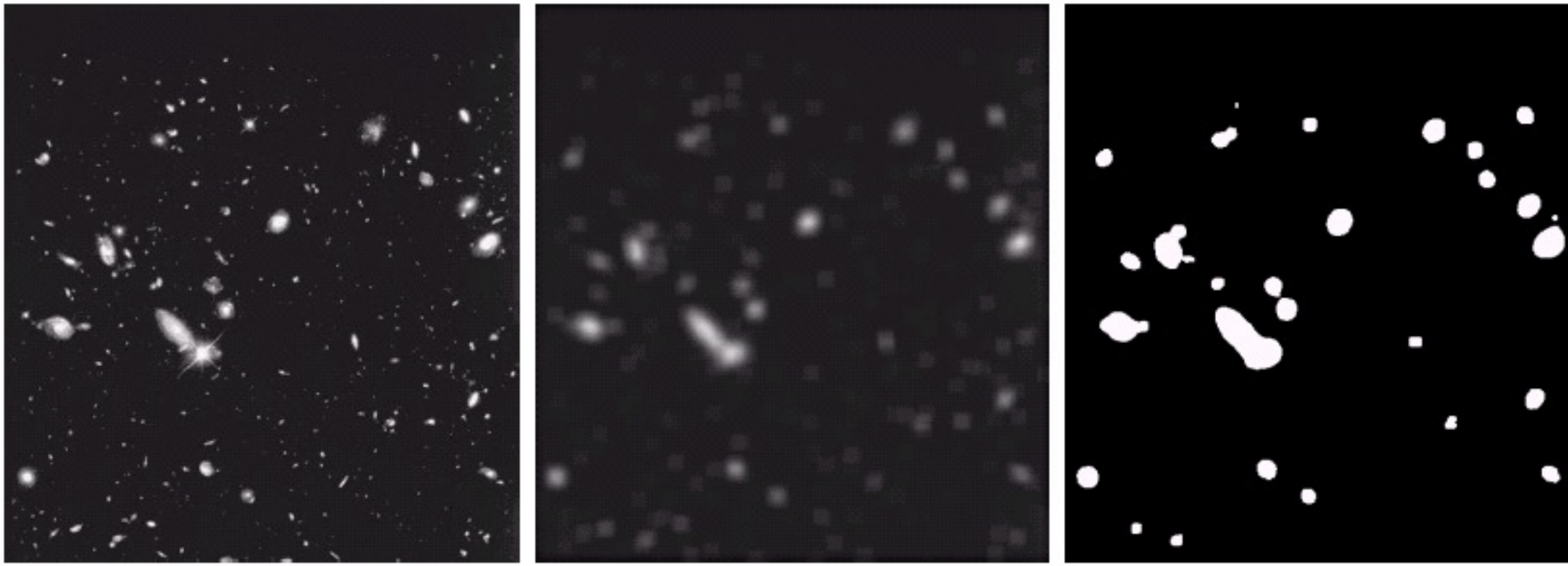


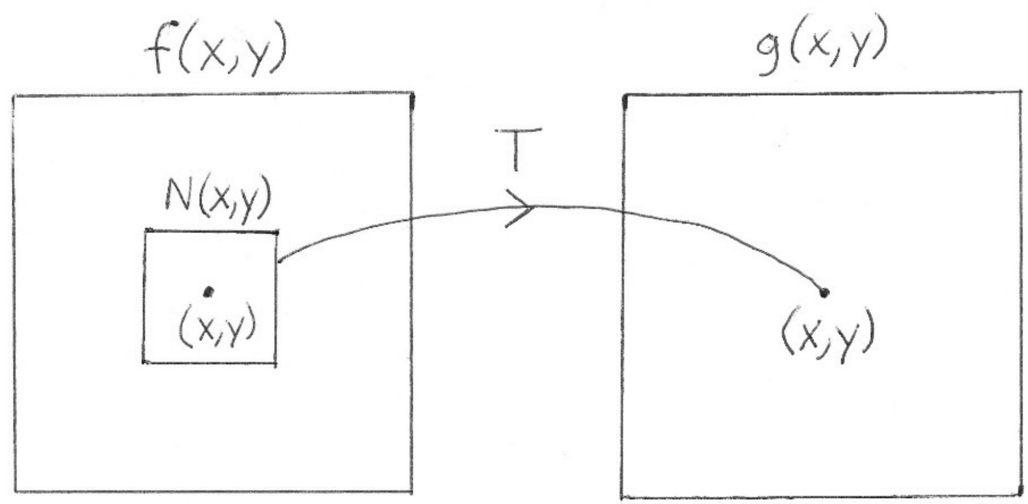
FIGURE 3.35 (a) Original image, of size 500 × 500 pixels. (b)–(f) Results of smoothing with square averaging filter masks of sizes $n = 3, 5, 9, 15,$ and 35, respectively. The black squares at the top are of sizes 3, 5, 9, 15, 25, 35, 45, and 55 pixels, respectively; their borders are 25 pixels apart. The letters at the bottom range in size from 10 to 24 points, in increments of 2 points; the large letter at the top is 60 points. The vertical bars are 5 pixels wide and 100 pixels high; their separation is 20 pixels. The diameter of the circles is 25 pixels, and their borders are 15 pixels apart; their gray levels range from 0% to 100% black in increments of 20%. The background of the image is 10% black. The noisy rectangles are of size 50 × 120 pixels.



a b c

FIGURE 3.36 (a) Image from the Hubble Space Telescope. (b) Image processed by a 15 × 15 averaging mask. (c) Result of thresholding (b). (Original image courtesy of NASA.)

Smoothing Nonlinear Median Filter - the output $g(x,y)$ at location (x,y) is the median of the pixel values in the input $f(x,y)$ in a neighborhood $N(x,y)$ of (x,y).



Ex Noise reduction using 3x3 median filter

Input

100	100	100	100	100
100	100	100	100	100
100	100	190	100	100
100	100	100	100	100
100	100	100	100	100

Ex Noise reduction using 3x3 median filter

Input

100	100	100	100	100
100	100	100	100	100
100	100	190	100	100
100	100	100	100	100
100	100	100	100	100

Output

100	100	100	100	100
100	100	100	100	100
100	100	100	100	100
100	100	100	100	100
100	100	100	100	100

Ex Noise reduction using 3x3 median filter

Input

100	100	100	100	100
100	100	100	100	100
100	100	190	100	100
100	100	100	100	100
100	100	100	100	100

Output

100	100	100	100	100
100	100	100	100	100
100	100	100	100	100
100	100	100	100	100
100	100	100	100	100

Ex Edge processing using 3x3 median filter

Input

100	100	200	200
100	100	200	200
100	100	200	200
100	100	200	200

Ex Noise reduction using 3x3 median filter

Input

100	100	100	100	100
100	100	100	100	100
100	100	190	100	100
100	100	100	100	100
100	100	100	100	100

Output

100	100	100	100	100
100	100	100	100	100
100	100	100	100	100
100	100	100	100	100
100	100	100	100	100

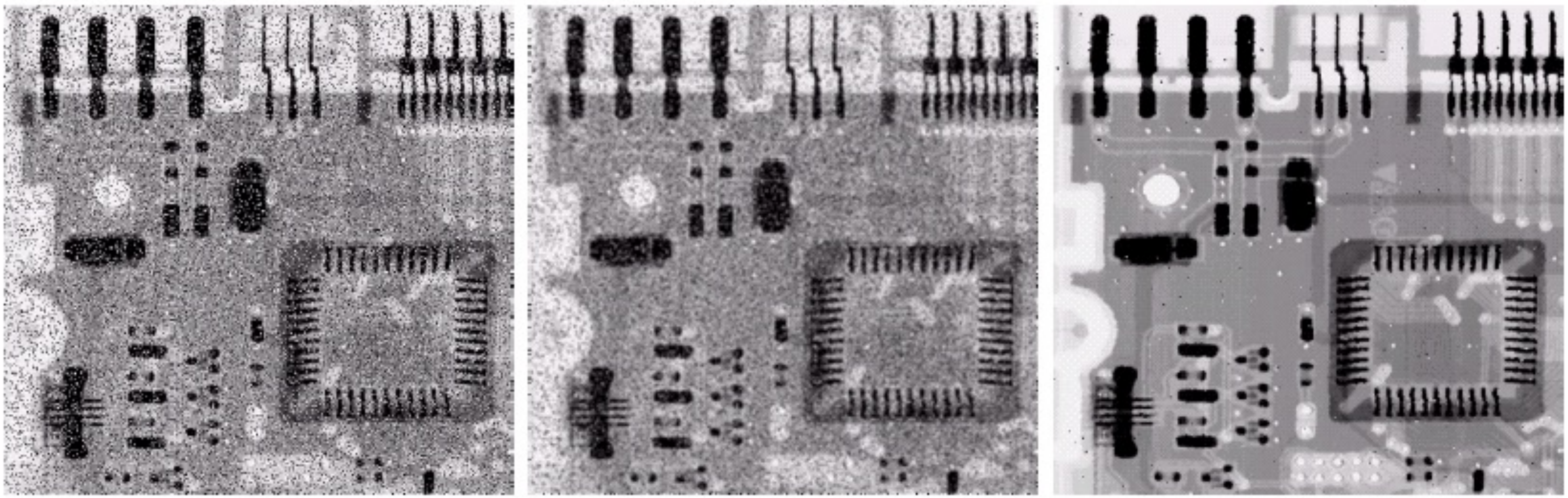
Ex Edge processing using 3x3 median filter

Input

100	100	200	200
100	100	200	200
100	100	200	200
100	100	200	200

Output

100	100	200	200
100	100	200	200
100	100	200	200
100	100	200	200



a b c

FIGURE 3.37 (a) X-ray image of circuit board corrupted by salt-and-pepper noise. (b) Noise reduction with a 3×3 averaging mask. (c) Noise reduction with a 3×3 median filter. (Original image courtesy of Mr. Joseph E. Pascente, Lixi, Inc.)

Sharpening (differencing, highpass) Linear Filters

Sharpening filter – the magnitude of the output image $g(x,y)$ tends to be large at locations (x,y) where there is large pixel-to-pixel change in gray level in the input image. The magnitude of the output image is small at locations (x,y) where the input image is nearly constant.

Sharpening (differencing, highpass) Linear Filters

Sharpening filter – the magnitude of the output image $g(x,y)$ tends to be large at locations (x,y) where there is large pixel-to-pixel change in gray level in the input image. The magnitude of the output image is small at locations (x,y) where the input image is nearly constant.

Derivatives are useful for sharpening.

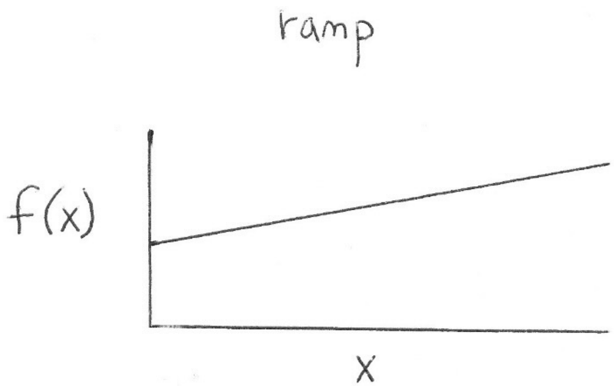
Discrete approximation to first derivative

$$\left.\frac{\partial f}{\partial x}\right|_{x} = f(x+1) - f(x)$$

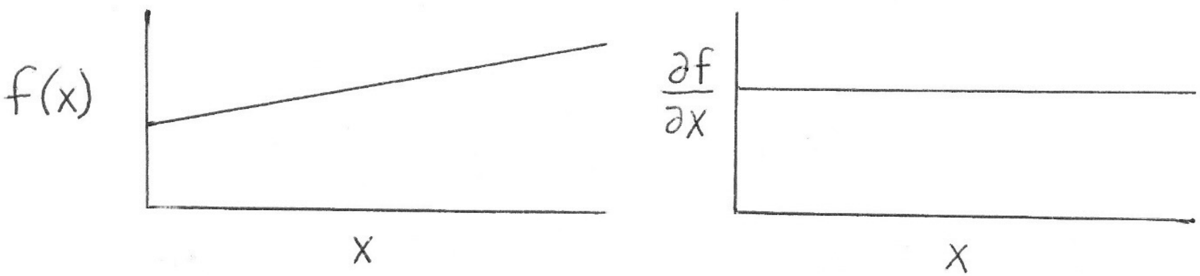
Discrete approximation to second derivative

$$\left.\frac{\partial^2 f}{\partial x^2}\right|_{x} = \left.\frac{\partial f}{\partial x}\right|_{x} - \left.\frac{\partial f}{\partial x}\right|_{x-1} = f(x+1) - f(x) - \left[f(x) - f(x-1)\right]$$

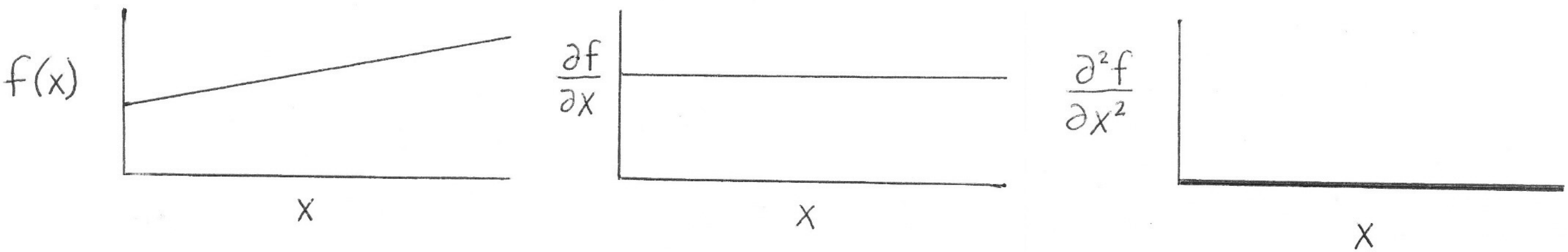
$$= f(x+1) - 2f(x) + f(x-1)$$


ramp
f(x)
x

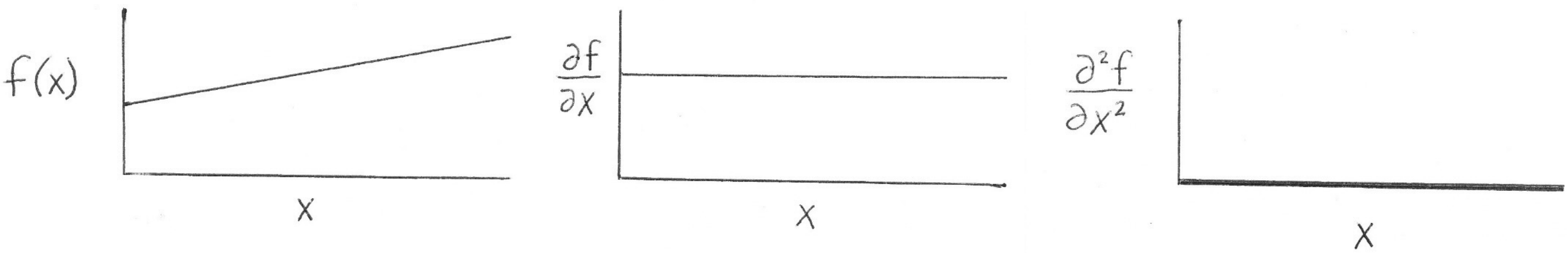
ramp



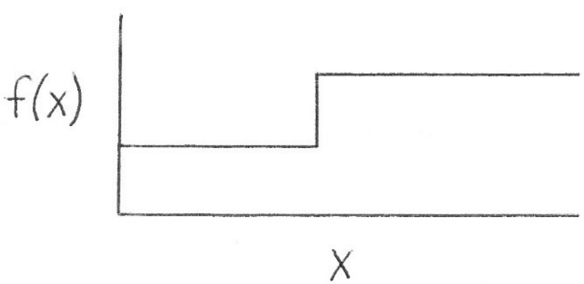
ramp



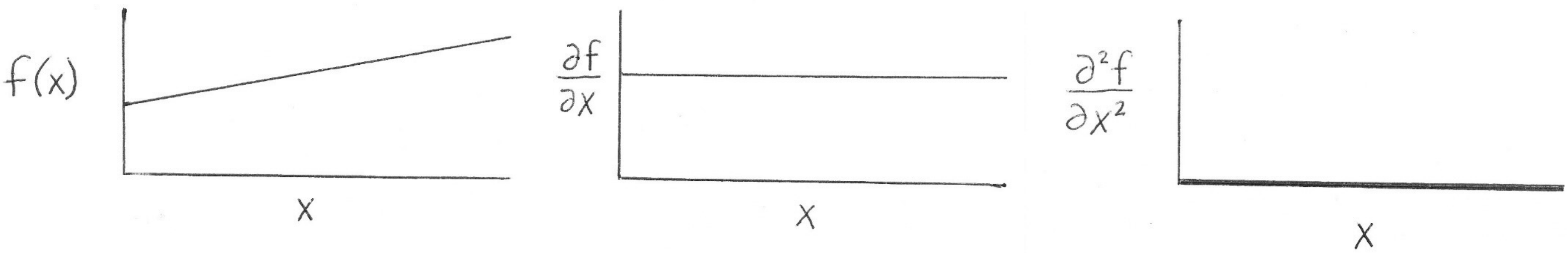
ramp


f(x)
x
$\frac{\partial f}{\partial x}$
x
$\frac{\partial^2 f}{\partial x^2}$
x

step

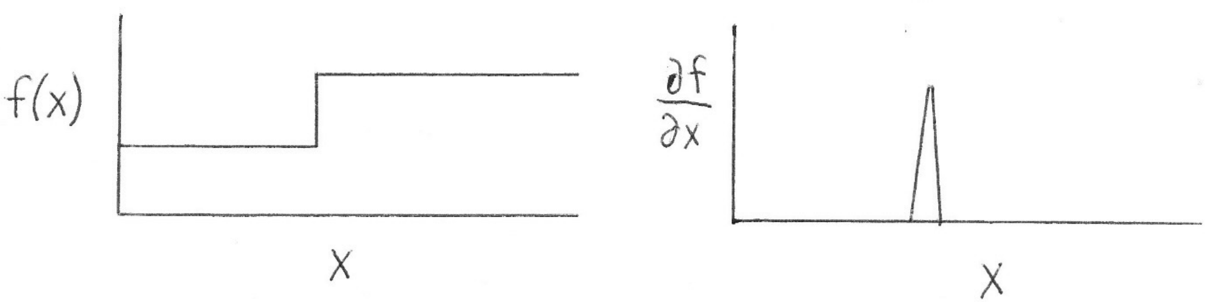

f(x)
x

ramp



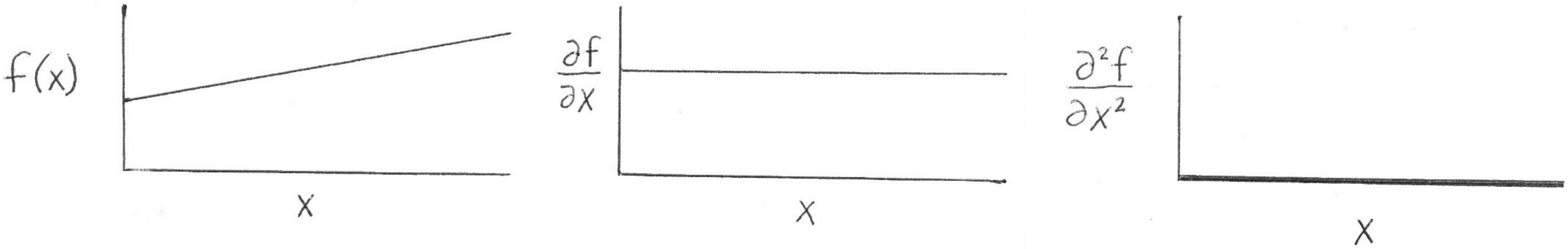
$f(x)$
x
$\frac{\partial f}{\partial x}$
x
$\frac{\partial^2 f}{\partial x^2}$
x

step

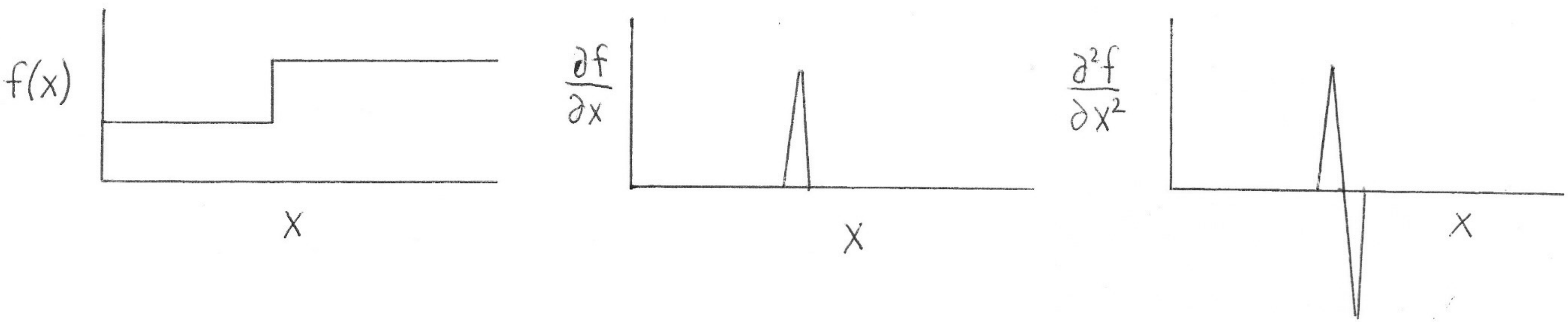


$f(x)$
x
$\frac{\partial f}{\partial x}$
x

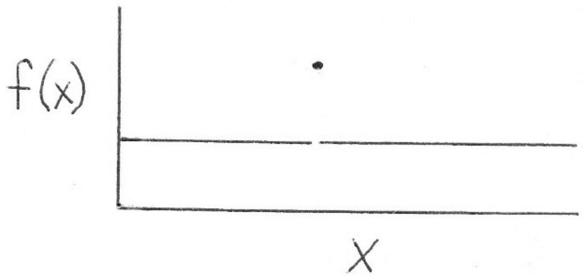
ramp


f(x)
x
$\frac{\partial f}{\partial x}$
x
$\frac{\partial^2 f}{\partial x^2}$
x

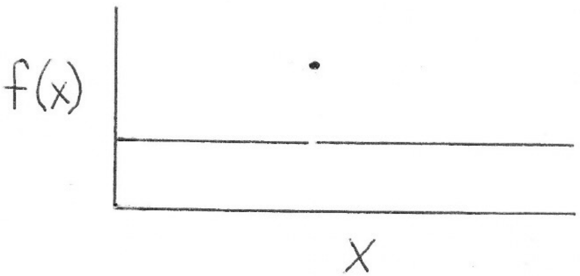
step

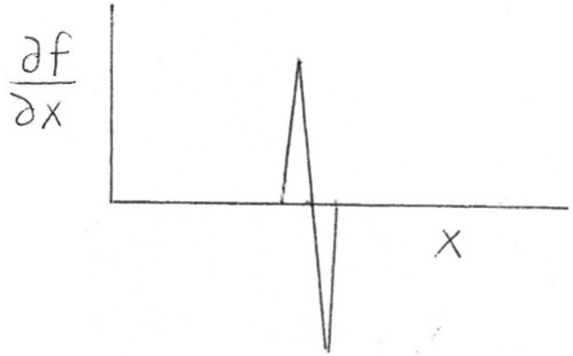

f(x)
x
$\frac{\partial f}{\partial x}$
x
$\frac{\partial^2 f}{\partial x^2}$
x

isolated point

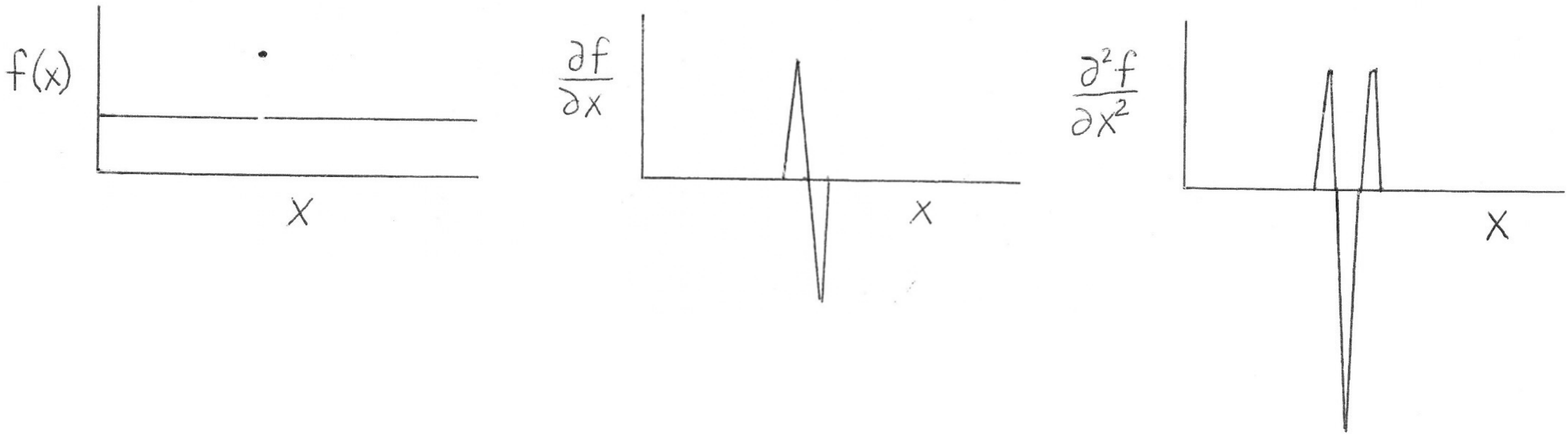


isolated point

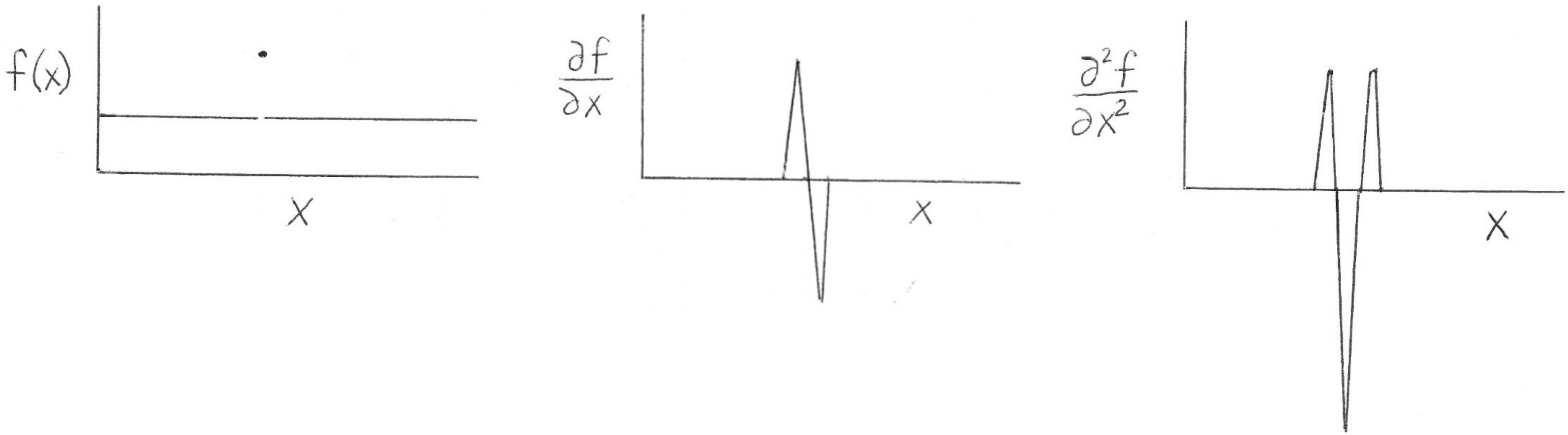




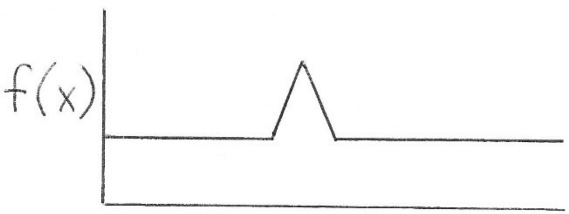
isolated point



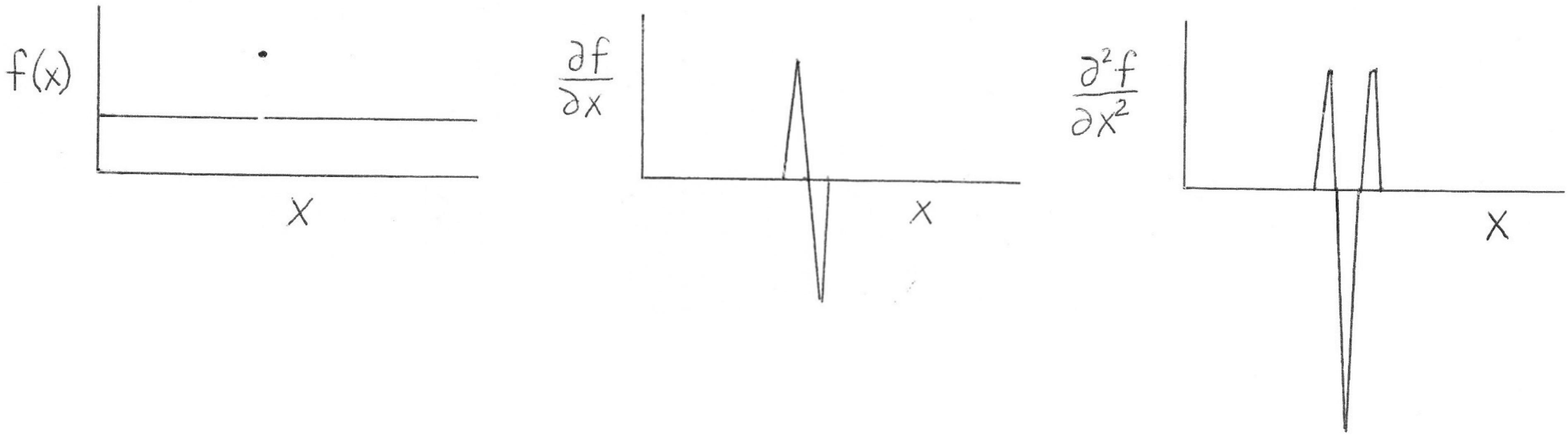
isolated point



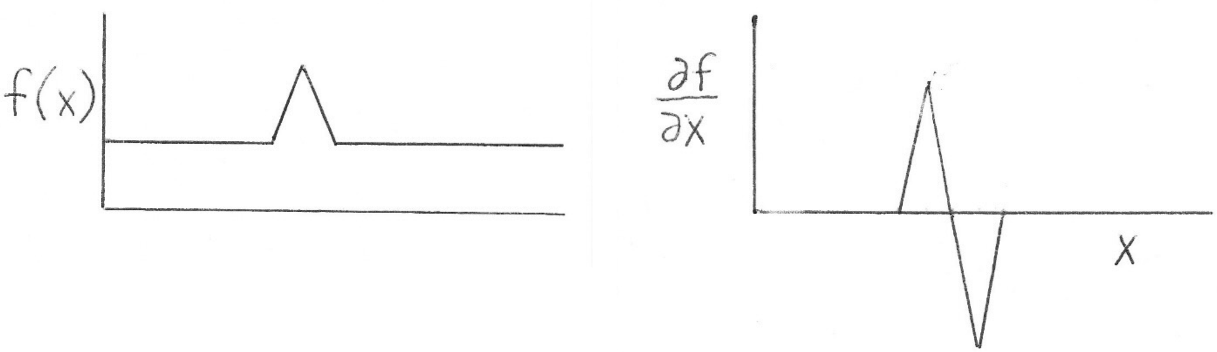
thin line



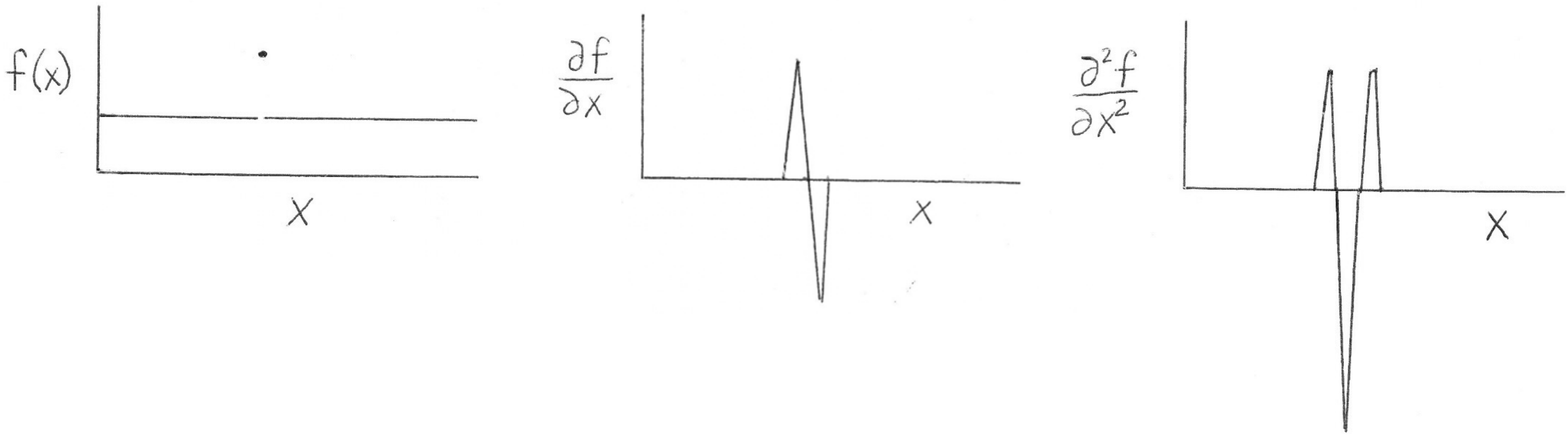
isolated point



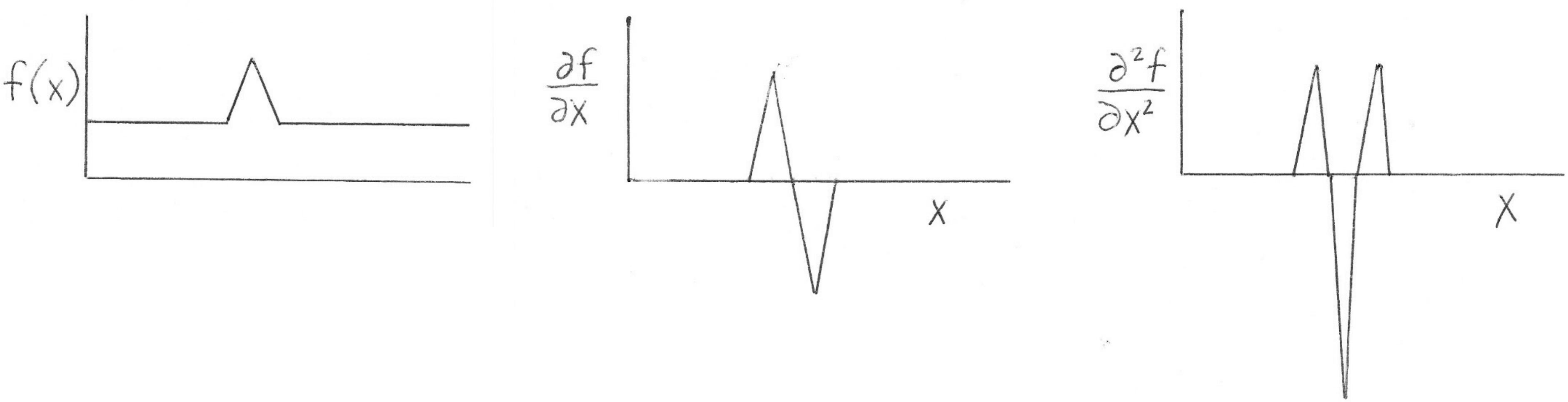
thin line

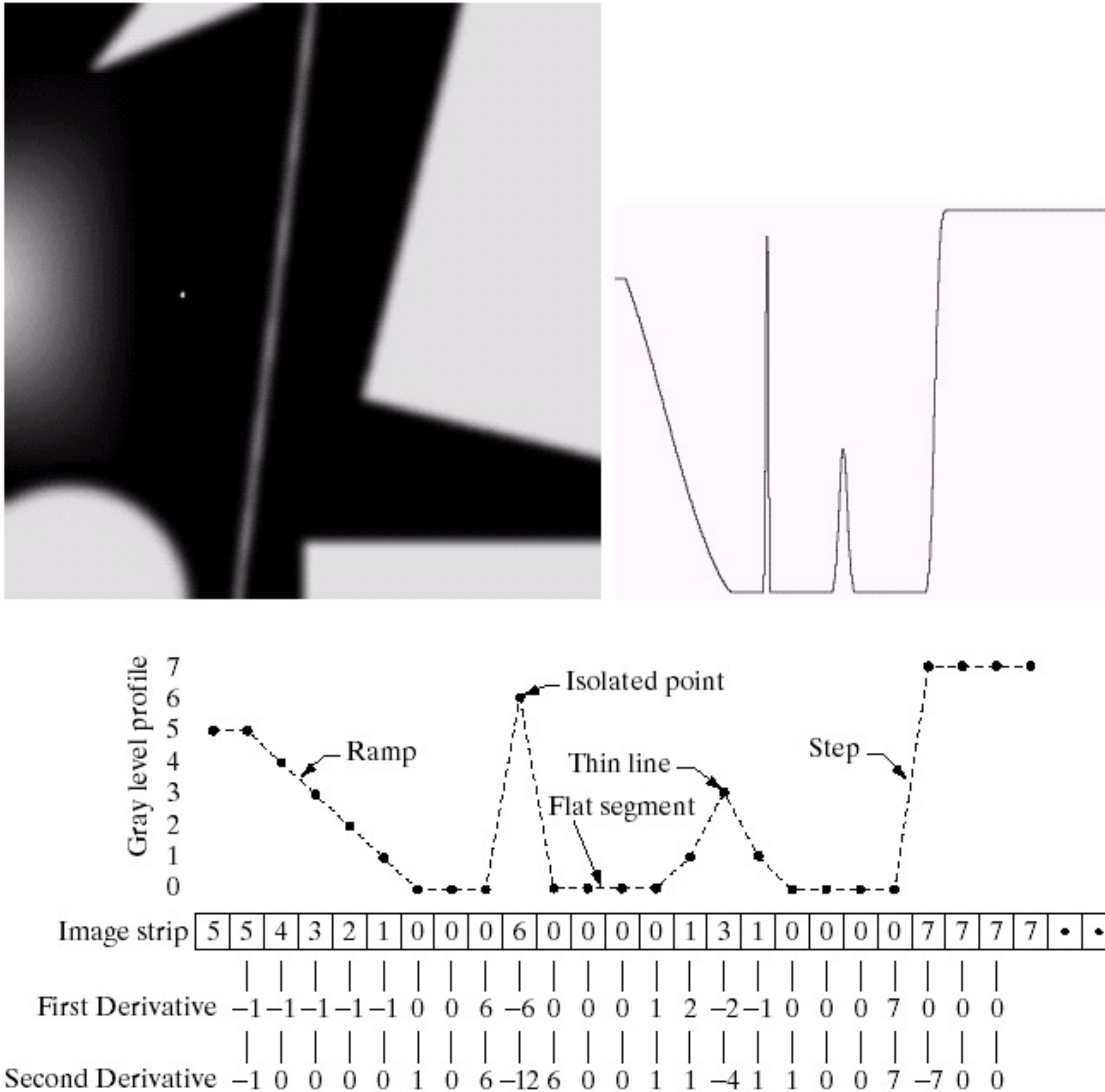


isolated point



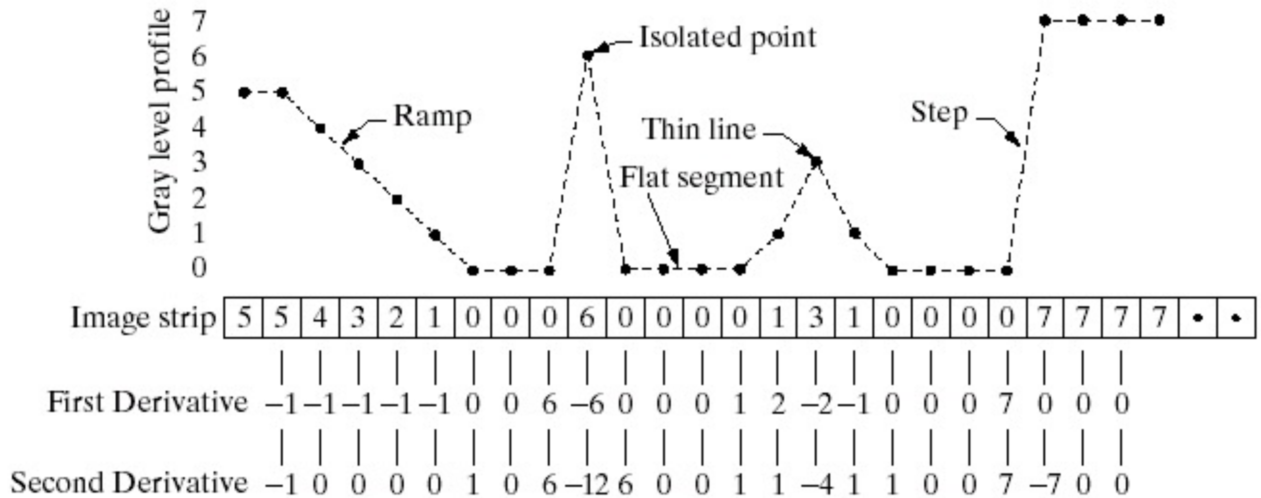
thin line



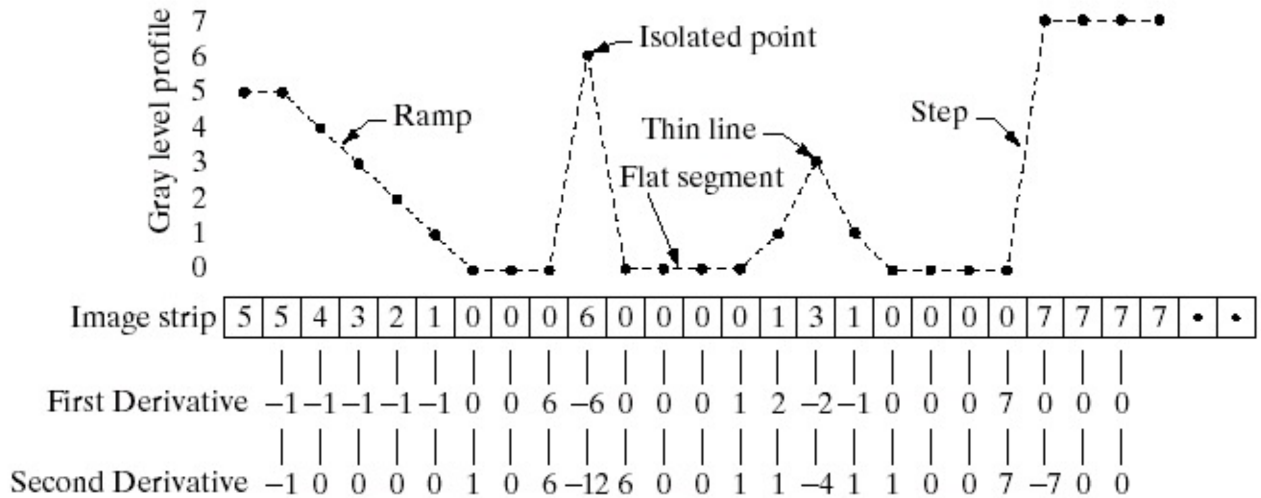


a b
c

FIGURE 3.38
(a) A simple image. (b) 1-D horizontal gray-level profile along the center of the image and including the isolated noise point. (c) Simplified profile (the points are joined by dashed lines to simplify interpretation).



For a step edge, the absolute value of the first derivative has a single maximum. For a step edge, the second derivative has a zero-crossing.



For a step edge, the absolute value of the first derivative has a single maximum. For a step edge, the second derivative has a zero-crossing.

The absolute value of the second derivative has a larger peak than the absolute value of the first derivative in response to fine detail such as an isolated point or a thin line.

A 2-D spatial filter $f(x,y)$ is rotation invariant (isotropic) if for any image $I(x,y)$ the output of the filter at location (x,y) does not change as the image is rotated about (x,y).

A 2-D spatial filter $f(x,y)$ is rotation invariant (isotropic) if for any image $I(x,y)$ the output of the filter at location (x,y) does not change as the image is rotated about (x,y).

For a linear filter, this is equivalent to saying that the filter $f(x,y)$ does not change as we rotate it about its center.

A 2-D spatial filter $f(x,y)$ is rotation invariant (isotropic) if for any image $I(x,y)$ the output of the filter at location (x,y) does not change as the image is rotated about (x,y).

For a linear filter, this is equivalent to saying that the filter $f(x,y)$ does not change as we rotate it about its center.

2nd Derivative Filters

The lowest order isotropic differential operator on a 2-D function $f(x,y)$ is the Laplacian

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

Use 1-D discrete approximation

$$\left.\frac{\partial^2 f}{\partial x^2}\right|_{x} = f(x+1) - 2f(x) + f(x-1)$$

to get

$$\left.\frac{\partial^2 f}{\partial x^2}\right|_{x,y} = f(x+1,y) - 2f(x,y) + f(x-1,y) \qquad \text{fix } y$$

Use 1-D discrete approximation

$$\left.\frac{\partial^2 f}{\partial x^2}\right|_x = f(x+1) - 2f(x) + f(x-1)$$

to get

$$\left.\frac{\partial^2 f}{\partial x^2}\right|_{x,y} = f(x+1,y) - 2f(x,y) + f(x-1,y) \qquad \text{fix } y$$

$$\left.\frac{\partial^2 f}{\partial y^2}\right|_{x,y} = f(x,y+1) - 2f(x,y) + f(x,y-1) \qquad \text{fix } x$$

Use 1-D discrete approximation

$$\left.\frac{\partial^2 f}{\partial x^2}\right|_x = f(x+1) - 2f(x) + f(x-1)$$

to get

$$\left.\frac{\partial^2 f}{\partial x^2}\right|_{x,y} = f(x+1,y) - 2f(x,y) + f(x-1,y) \qquad \text{fix } y$$

$$\left.\frac{\partial^2 f}{\partial y^2}\right|_{x,y} = f(x,y+1) - 2f(x,y) + f(x,y-1) \qquad \text{fix } x$$

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

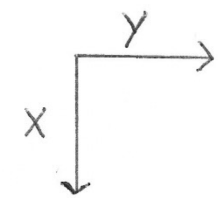
$$= f(x+1,y) + f(x-1,y) + f(x,y+1) + f(x,y-1) - 4f(x,y)$$

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

Laplacian can be implemented with the mask

0	1	0
1	-4	1
0	1	0

$$\begin{array}{ccc} f(x-1,y-1) & f(x-1,y) & f(x-1,y+1) \\ f(x,y-1) & f(x,y) & f(x,y+1) \\ f(x+1,y-1) & f(x+1,y) & f(x+1,y+1) \end{array}$$

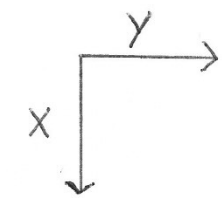


$$\nabla^2 f = f(x+1,y) + f(x-1,y) + f(x,y+1) + f(x,y-1) - 4f(x,y)$$

Laplacian can be implemented with the mask

0	1	0
1	-4	1
0	1	0

$$\begin{array}{ccc} f(x-1,y-1) & f(x-1,y) & f(x-1,y+1) \\ f(x,y-1) & f(x,y) & f(x,y+1) \\ f(x+1,y-1) & f(x+1,y) & f(x+1,y+1) \end{array}$$



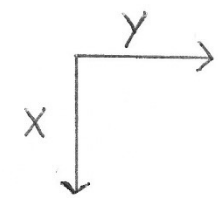
This filter gives isotropic results for rotations in multiples of $90°$

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

Laplacian can be implemented with the mask

0	1	0
1	-4	1
0	1	0

$$\begin{array}{ccc} f(x-1,y-1) & f(x-1,y) & f(x-1,y+1) \\ f(x,y-1) & f(x,y) & f(x,y+1) \\ f(x+1,y-1) & f(x+1,y) & f(x+1,y+1) \end{array}$$



This filter gives isotropic results for rotations in multiples of $90°$

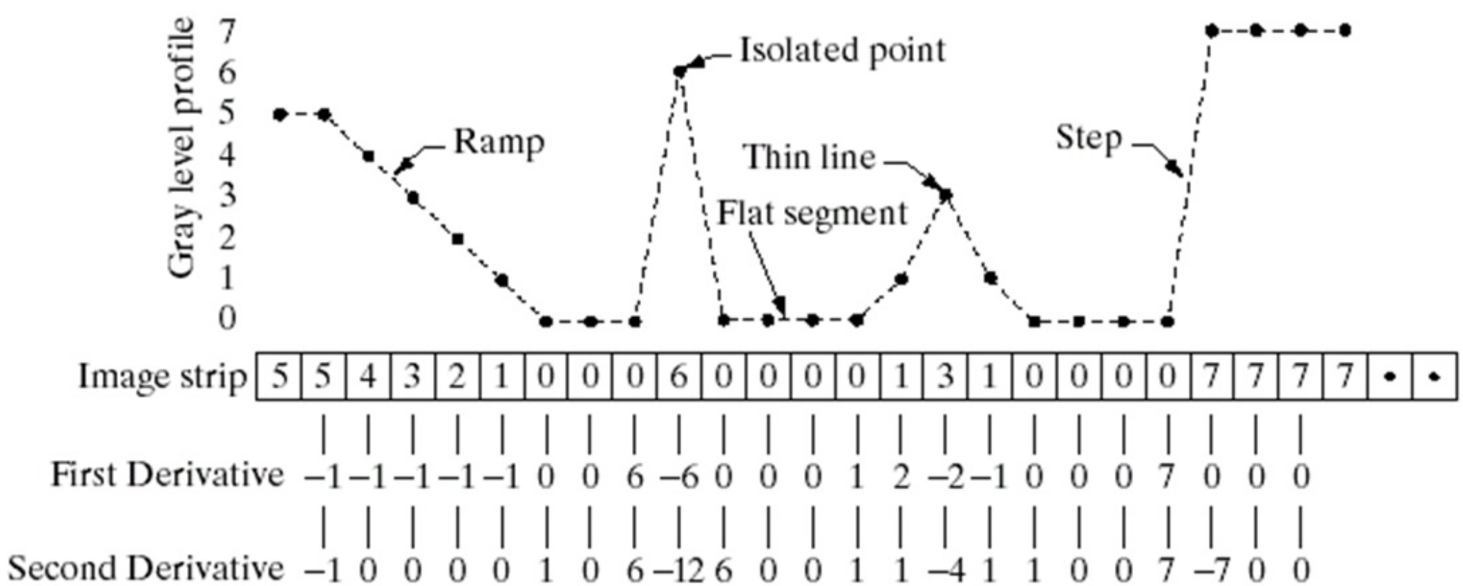
Since the Laplacian is isotropic, we can compute the discrete approximations using diagonal neighbors to get the mask

1	0	1
0	-4	0
1	0	1

Combining the two masks gives another Laplacian mask

$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix} + \begin{bmatrix} 1 & 0 & 1 \\ 0 & -4 & 0 \\ 1 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

The magnitude of the output of a Laplacian filter will be small for constant or slowly changing regions and large for step edges, isolated points, or thin lines.



We can create a sharpened version of the input image by combining the input and Laplacian images.

We can create a sharpened version of the input image by combining the input and Laplacian images.

Subtracting the Laplacian image from the input image $f(x,y)$

$$g(x,y) = f(x,y) - \nabla^2 f(x,y)$$

produces an image $g(x,y)$ for which the contrast for step edges, isolated points, and thin lines increases.

Image strip	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	•	•
First Derivative		−1	−1	−1	−1	−1	0	0	6	−6	0	0	0	1	2	−2	−1	0	0	0	7	0	0	0			
Second Derivative		−1	0	0	0	0	1	0	6	−12	6	0	0	1	1	−4	1	1	0	0	7	−7	0	0			

isolated point

	image f	0	6	0
	2nd Derivative $\nabla^2 f$	6	-12	6
	$f - \nabla^2 f$	-6	18	-6

Image strip	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	•	•
First Derivative		−1	−1	−1	−1	−1	0	0	6	−6	0	0	0	1	2	−2	−1	0	0	0	7	0	0	0			
Second Derivative		−1	0	0	0	0	1	0	6	−12	6	0	0	1	1	−4	1	1	0	0	7	−7	0	0			

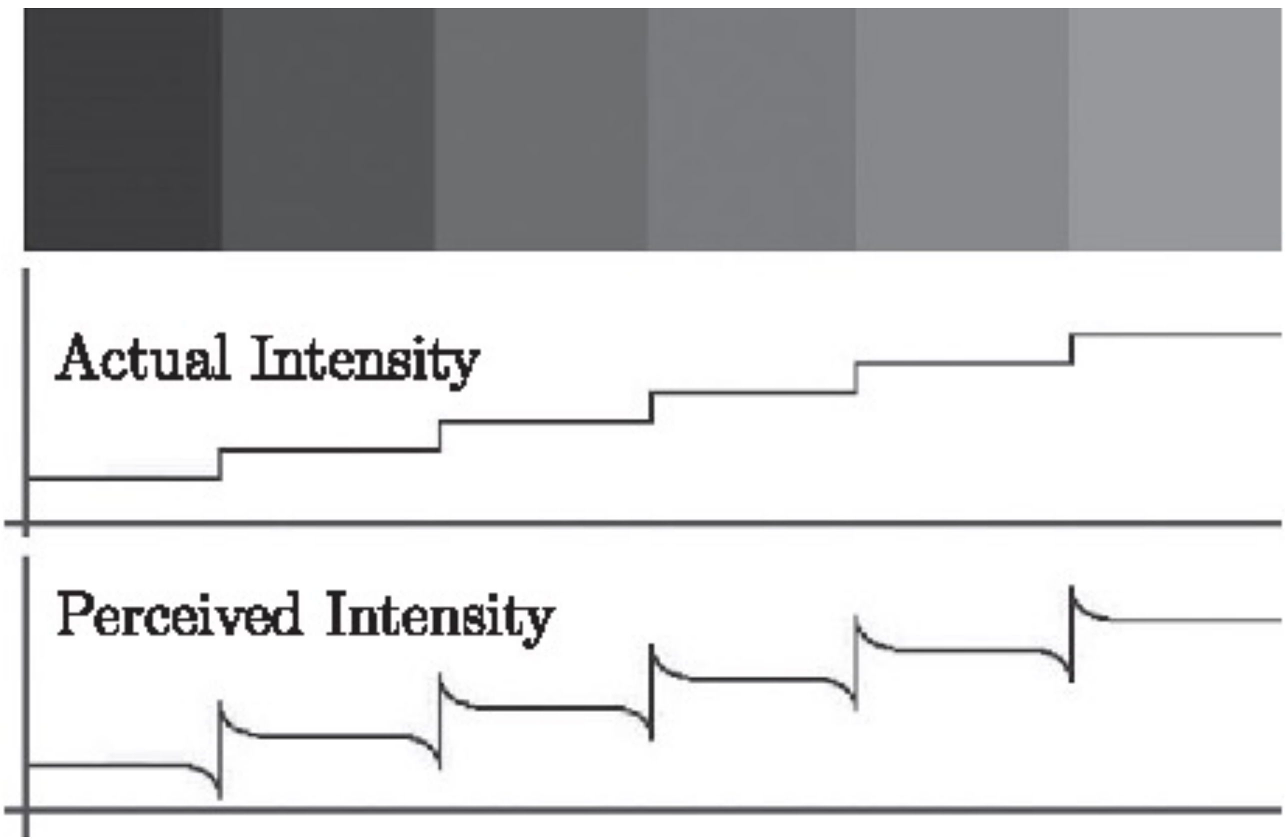
isolated point

image f	0	6	0
2nd Derivative $\nabla^2 f$	6	-12	6
$f - \nabla^2 f$	-6	18	-6

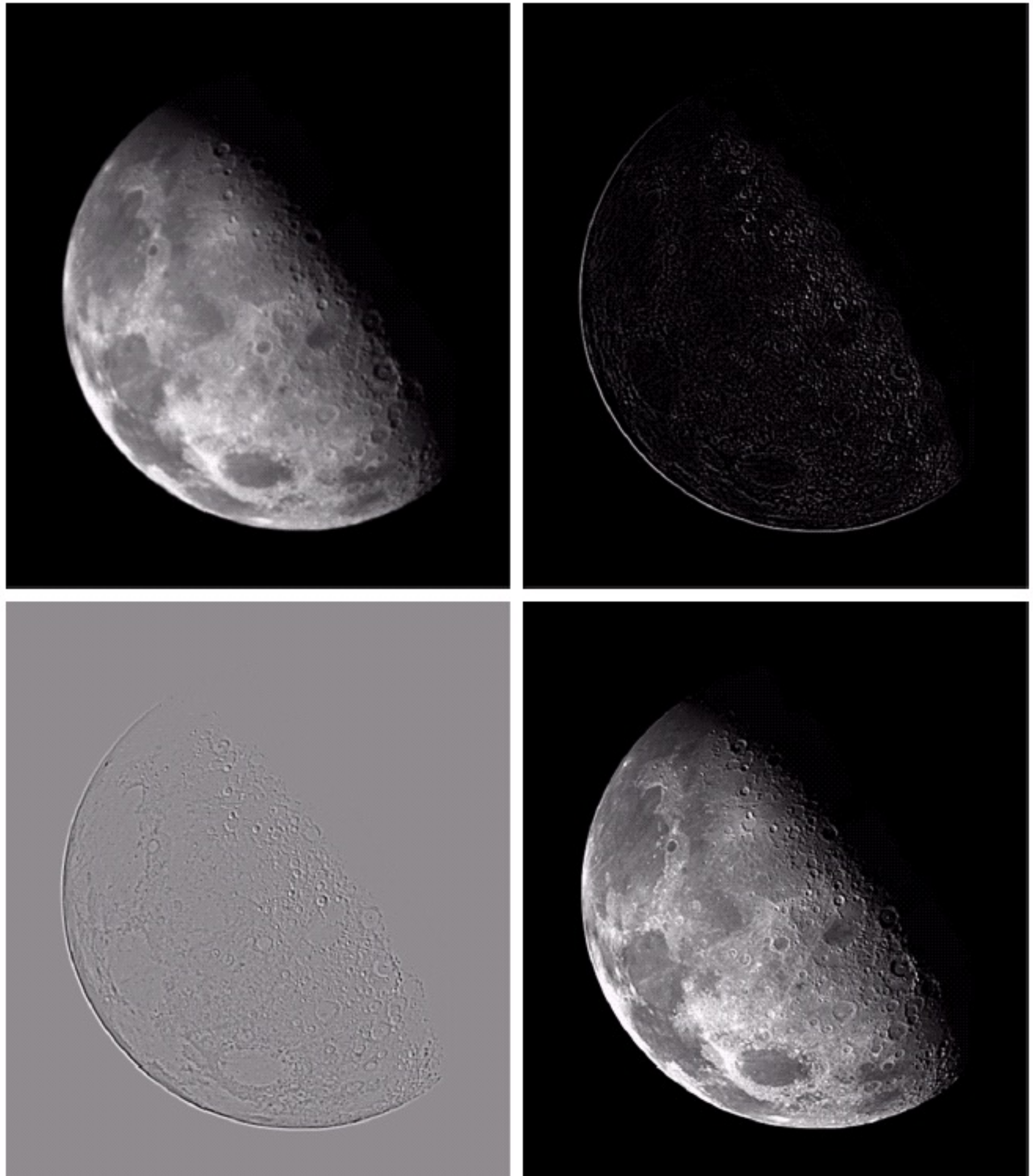
step edge

image f	0	0	7	7
2nd Derivative $\nabla^2 f$	0	7	-7	0
$f - \nabla^2 f$	0	-7	14	7

This exaggeration of contrast across edges is implemented by the human vision system and known as Mach bands.



Mach bands



a b
c d

FIGURE 3.40 (a) Image of the North Pole of the moon. (b) Laplacian-filtered image. (c) Laplacian image scaled for display purposes. (d) Image enhanced by using Eq. (3.7-5). (Original image courtesy of NASA.)

Can implement $f(x,y) - \nabla^2 f(x,y)$ with a single mask

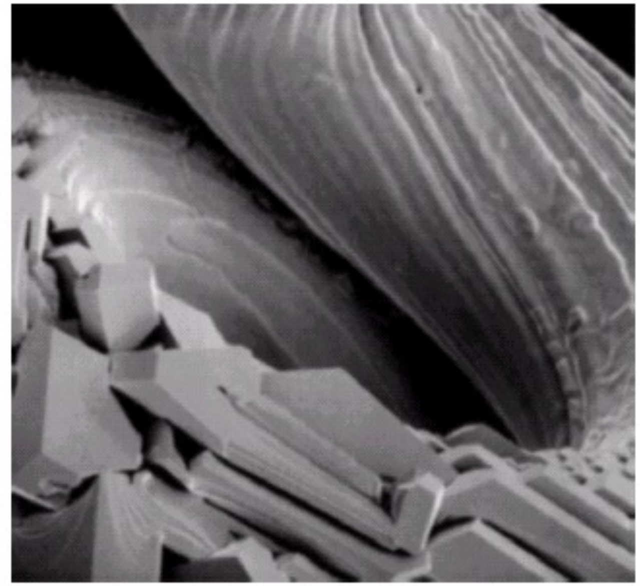
$$\begin{array}{|c|c|c|}\hline 0 & 0 & 0 \\ \hline 0 & 1 & 0 \\ \hline 0 & 0 & 0 \\ \hline \end{array} - \begin{array}{|c|c|c|}\hline 0 & 1 & 0 \\ \hline 1 & -4 & 1 \\ \hline 0 & 1 & 0 \\ \hline \end{array} = \begin{array}{|c|c|c|}\hline 0 & -1 & 0 \\ \hline -1 & 5 & -1 \\ \hline 0 & -1 & 0 \\ \hline \end{array}$$

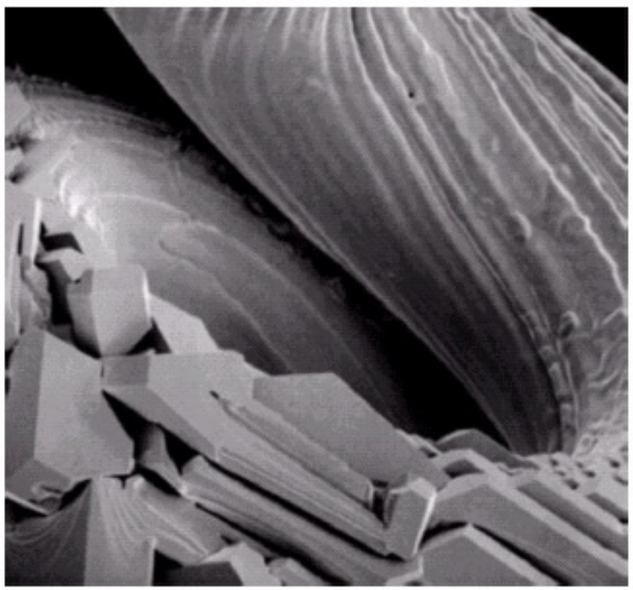
or

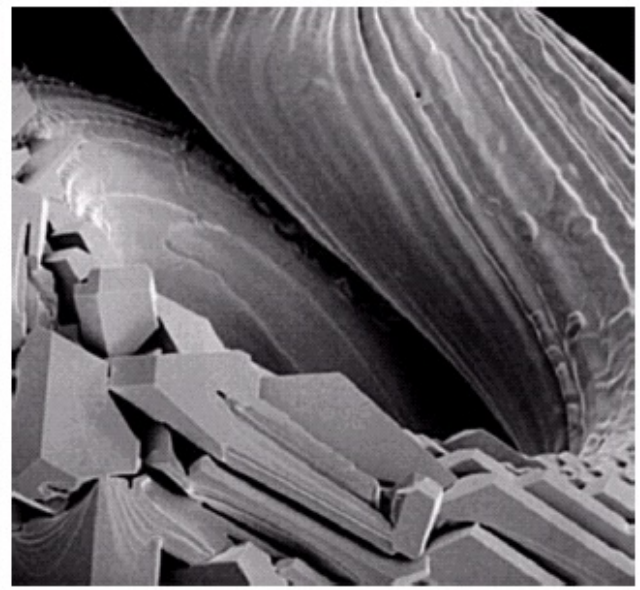
$$\begin{array}{|c|c|c|}\hline 0 & 0 & 0 \\ \hline 0 & 1 & 0 \\ \hline 0 & 0 & 0 \\ \hline \end{array} - \begin{array}{|c|c|c|}\hline 1 & 1 & 1 \\ \hline 1 & -8 & 1 \\ \hline 1 & 1 & 1 \\ \hline \end{array} = \begin{array}{|c|c|c|}\hline -1 & -1 & -1 \\ \hline -1 & 9 & -1 \\ \hline -1 & -1 & -1 \\ \hline \end{array}$$

0	−1	0
−1	5	−1
0	−1	0

−1	−1	−1
−1	9	−1
−1	−1	−1







a b c
d e

FIGURE 3.41 (a) Composite Laplacian mask. (b) A second composite mask. (c) Scanning electron microscope image. (d) and (e) Results of filtering with the masks in (a) and (b), respectively. Note how much sharper (e) is than (d). (Original image courtesy of Mr. Michael Shaffer, Department of Geological Sciences, University of Oregon, Eugene.)

1st Derivative Filters

gradient $\quad \nabla f = \begin{bmatrix} \dfrac{\partial f}{\partial x} \\[2ex] \dfrac{\partial f}{\partial y} \end{bmatrix}$

1st Derivative Filters

gradient $\quad \nabla f = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix}$

The components of the gradient vector are linear operators but not isotropic

1st Derivative Filters

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

The components of the gradient vector are linear operators but not isotropic

gradient magnitude $\|\nabla f\| = \sqrt{\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2}$

1st Derivative Filters

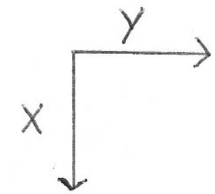
gradient $\nabla f = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix}$

The components of the gradient vector are linear operators but not isotropic

gradient magnitude $\|\nabla f\| = \sqrt{\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2}$

The gradient magnitude is isotropic, but a nonlinear operator

Use previous discrete approximation $\begin{array}{cc} f(x,y) & f(x,y+1) \\ f(x+1,y) & f(x+1,y+1) \end{array}$



$$\left.\frac{\partial f}{\partial x}\right|_{(x,y)} = f(x+1,y) - f(x,y) \qquad \left.\frac{\partial f}{\partial x}\right|_{(x,y+1)} = f(x+1,y+1) - f(x,y+1)$$

$$\left.\frac{\partial f}{\partial y}\right|_{(x,y)} = f(x,y+1) - f(x,y) \qquad \left.\frac{\partial f}{\partial y}\right|_{(x+1,y)} = f(x+1,y+1) - f(x+1,y)$$

Approximate $\frac{\partial f}{\partial x}$ by $\left.\frac{\partial f}{\partial x}\right|_{(x,y)} + \left.\frac{\partial f}{\partial x}\right|_{(x,y+1)}$ $\quad \begin{array}{|c|c|} \hline -1 & -1 \\ \hline 1 & 1 \\ \hline \end{array}$

Approximate $\frac{\partial f}{\partial y}$ by $\left.\frac{\partial f}{\partial y}\right|_{(x,y)} + \left.\frac{\partial f}{\partial y}\right|_{(x+1,y)}$ $\quad \begin{array}{|c|c|} \hline -1 & 1 \\ \hline -1 & 1 \\ \hline \end{array}$

$$\underset{\text{(SGM)}}{\text{Squared Gradient Magnitude}} = \left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2$$

$$\frac{\partial f}{\partial x} \quad \boxed{\begin{array}{|c|c|} \hline -1 & -1 \\ \hline 1 & 1 \\ \hline \end{array}} \qquad \frac{\partial f}{\partial y} \quad \boxed{\begin{array}{|c|c|} \hline -1 & 1 \\ \hline -1 & 1 \\ \hline \end{array}}$$

$$\begin{array}{cc} f(x,y) & f(x,y+1) \\ f(x+1,y) & f(x+1,y+1) \end{array} = \begin{array}{cc} z_5 & z_6 \\ z_8 & z_9 \end{array}$$

Squared Gradient Magnitude (SGM) $= \left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2$

$$\frac{\partial f}{\partial x} \quad \begin{array}{|c|c|} \hline -1 & -1 \\ \hline 1 & 1 \\ \hline \end{array} \qquad \frac{\partial f}{\partial y} \quad \begin{array}{|c|c|} \hline -1 & 1 \\ \hline -1 & 1 \\ \hline \end{array}$$

$$\begin{matrix} f(x,y) & f(x,y+1) \\ f(x+1,y) & f(x+1,y+1) \end{matrix} = \begin{matrix} z_5 & z_6 \\ z_8 & z_9 \end{matrix}$$

$$SGM = (z_9 - z_5 + z_8 - z_6)^2 + (z_9 - z_5 - (z_8 - z_6))^2$$

$$\text{Squared Gradient Magnitude (SGM)} = \left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2$$

$\dfrac{\partial f}{\partial x}$ $\begin{array}{|c|c|} \hline -1 & -1 \\ \hline 1 & 1 \\ \hline \end{array}$ $\dfrac{\partial f}{\partial y}$ $\begin{array}{|c|c|} \hline -1 & 1 \\ \hline -1 & 1 \\ \hline \end{array}$

$$\begin{matrix} f(x,y) & f(x,y+1) \\ f(x+1,y) & f(x+1,y+1) \end{matrix} = \begin{matrix} z_5 & z_6 \\ z_8 & z_9 \end{matrix}$$

$$SGM = (z_9 - z_5 + z_8 - z_6)^2 + (z_9 - z_5 - (z_8 - z_6))^2$$

$$= (d_1 + d_2)^2 + (d_1 - d_2)^2 = 2(d_1^2 + d_2^2)$$

$$\text{Squared Gradient Magnitude (SGM)} = \left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2$$

$$\frac{\partial f}{\partial x} \quad \begin{array}{|c|c|}\hline -1 & -1 \\ \hline 1 & 1 \\ \hline \end{array} \qquad \frac{\partial f}{\partial y} \quad \begin{array}{|c|c|}\hline -1 & 1 \\ \hline -1 & 1 \\ \hline \end{array}$$

$$\begin{matrix} f(x,y) & f(x,y+1) \\ f(x+1,y) & f(x+1,y+1) \end{matrix} = \begin{matrix} z_5 & z_6 \\ z_8 & z_9 \end{matrix}$$

$$\begin{aligned} SGM &= (z_9 - z_5 + z_8 - z_6)^2 + (z_9 - z_5 - (z_8 - z_6))^2 \\ &= (d_1 + d_2)^2 + (d_1 - d_2)^2 = 2(d_1^2 + d_2^2) \\ &= 2\left((z_9 - z_5)^2 + (z_8 - z_6)^2\right) \end{aligned}$$

Squared Gradient Magnitude (SGM) $= \left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2$

$\frac{\partial f}{\partial x}$ $\begin{array}{|c|c|}\hline -1 & -1 \\ \hline 1 & 1 \\ \hline\end{array}$ $\frac{\partial f}{\partial y}$ $\begin{array}{|c|c|}\hline -1 & 1 \\ \hline -1 & 1 \\ \hline\end{array}$

$$\begin{matrix} f(x,y) & f(x,y+1) \\ f(x+1,y) & f(x+1,y+1) \end{matrix} = \begin{matrix} z_5 & z_6 \\ z_8 & z_9 \end{matrix}$$

$$SGM = (z_9 - z_5 + z_8 - z_6)^2 + (z_9 - z_5 - (z_8 - z_6))^2$$

$$= (d_1 + d_2)^2 + (d_1 - d_2)^2 = 2(d_1^2 + d_2^2)$$

$$= 2\left((z_9 - z_5)^2 + (z_8 - z_6)^2\right)$$

Roberts Cross Operator $(f(x+1,y+1) - f(x,y))^2 + (f(x+1,y) - f(x,y+1))^2$

$$\text{Squared Gradient Magnitude (SGM)} = \left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2$$

$\frac{\partial f}{\partial x}$ $\begin{array}{|c|c|}\hline -1 & -1 \\ \hline 1 & 1 \\ \hline\end{array}$ $\frac{\partial f}{\partial y}$ $\begin{array}{|c|c|}\hline -1 & 1 \\ \hline -1 & 1 \\ \hline\end{array}$

$$\begin{matrix} f(x,y) & f(x,y+1) \\ f(x+1,y) & f(x+1,y+1) \end{matrix} = \begin{matrix} z_5 & z_6 \\ z_8 & z_9 \end{matrix}$$

$$SGM = (z_9 - z_5 + z_8 - z_6)^2 + (z_9 - z_5 - (z_8 - z_6))^2$$

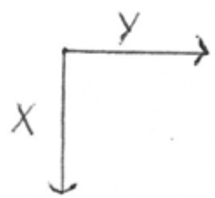
$$= (d_1 + d_2)^2 + (d_1 - d_2)^2 = 2(d_1^2 + d_2^2)$$

$$= 2((z_9 - z_5)^2 + (z_8 - z_6)^2)$$

Roberts Cross Operator $(f(x+1,y+1) - f(x,y))^2 + (f(x+1,y) - f(x,y+1))^2$

Since Roberts Cross uses 2x2 filter, the estimate of the SGM is in the middle of 4 pixels.

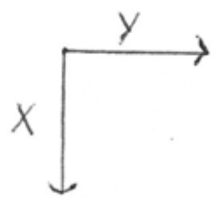
The most popular 3x3 first derivative operator is the Sobel operator



$$\frac{\partial f}{\partial x} \quad \begin{array}{|c|c|c|}\hline -1 & -2 & -1 \\ \hline 0 & 0 & 0 \\ \hline 1 & 2 & 1 \\ \hline \end{array} \qquad\qquad \frac{\partial f}{\partial y} \quad \begin{array}{|c|c|c|}\hline -1 & 0 & 1 \\ \hline -2 & 0 & 2 \\ \hline -1 & 0 & 1 \\ \hline \end{array}$$

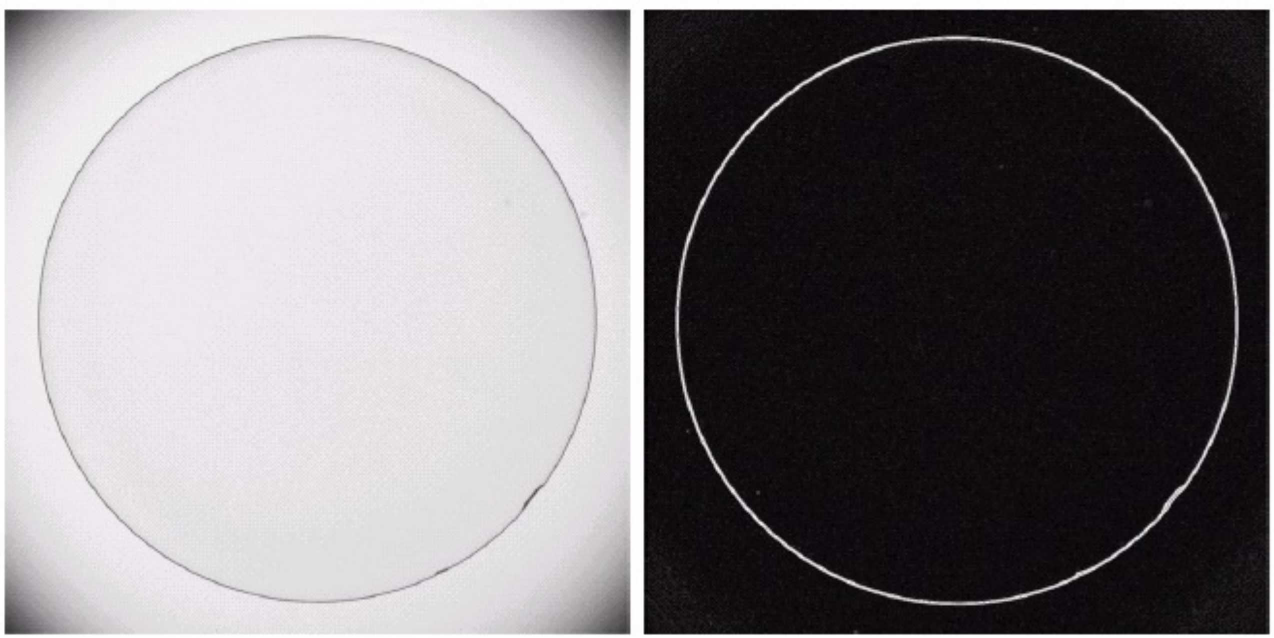
which can also be used to compute the SGM

The most popular 3x3 first derivative operator is the Sobel operator



$$\frac{\partial f}{\partial x} \quad \begin{array}{|c|c|c|}\hline -1 & -2 & -1 \\ \hline 0 & 0 & 0 \\ \hline 1 & 2 & 1 \\ \hline \end{array} \qquad \frac{\partial f}{\partial y} \quad \begin{array}{|c|c|c|}\hline -1 & 0 & 1 \\ \hline -2 & 0 & 2 \\ \hline -1 & 0 & 1 \\ \hline \end{array}$$

which can also be used to compute the SGM

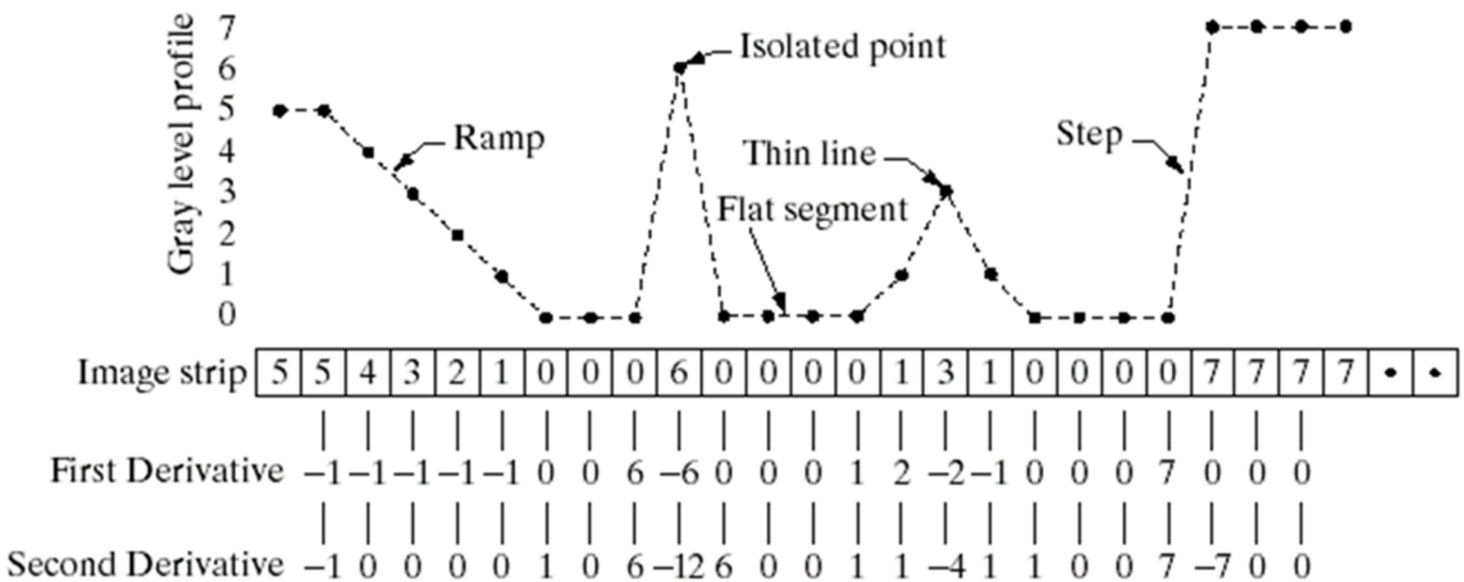


a b

FIGURE 3.45 Optical image of contact lens (note defects on the boundary at 4 and 5 o'clock). (b) Sobel gradient. (Original image courtesy of Mr. Pete Sites, Perceptics Corporation.)

$$\left|\frac{\partial f}{\partial x}\right| + \left|\frac{\partial f}{\partial y}\right|$$

The SGM can be used to find edges and is less sensitive to noise than the 2nd derivative Laplacian.



Threshold $|\text{1st derivative}| \geq 7 \Rightarrow$ only get step

Threshold $|\text{2nd derivative}| \geq 7 \Rightarrow$ get step and isolated point

We often use large maxima in the SGM to detect edges and zero-crossings of the 2nd derivative to localize edges.