Lecture 6:

Image Enhancement in Spatial Domain

Spatial Filtering: Order statistic filtering and Image Sharpening

Local enhancement using statistical parameters from histogram

Local enhancement can be based on statistical properties of the gray levels in a block instead of using full histogram

Examples:

- Mean gives the average brightness of the image
- Variance (σ^2) and its square root the standard deviation gives the deviation of intensities on average from the mean value (average contrast)

Local enhancement using statistical parameters

S_{xy} : a neighborhood (subimage) of size N_{Sxy}; a block centered at (x,y)

$m_{S_{xy}}$: gray-level mean in S_{xy} $$m_{S_{xy}} = \frac{1}{N_{S_{xy}}} \sum_{(s,t)\in S_{xy}} f(s,t)$$

$\sigma^2_{S_{xy}}$: gray-level variance in S_{xy} $$\sigma^2_{S_{xy}} = \frac{1}{N_{S_{xy}}} \sum_{(s,t)\in S_{xy}} \left(f(s,t) - m_{S_{xy}}\right)^2$$

$\sigma_{S_{xy}}$: standard deviation, square root of variance $\sigma^2_{S_{xy}}$

M_G : global mean of $f(x,y)$

D_G : global standard deviation of $f(x,y)$

Local enhancement using statistical parameters

The statistical parameters can be used in various ways:

- For the direct calculation of transformation function (adaptive transformation function) for example

$$g(x,y) = A_{S_{xy}}[f(x,y) - m_{S_{xy}}] + m_{S_{xy}},$$

where $A_{S_{xy}}$ is the local gain factor, $A_{S_{xy}} = \dfrac{kM_G}{\sigma_{S_{xy}}}, \quad 0 < k < 1$

- Using them in defining ranges for different transfer functions for example

$$g(x,y) = \begin{cases} E.f(x,y) & \text{if } m_{S_{xy}} \le k_0 M_G \text{ AND } k_1 D_G \le \sigma_{S_{xy}} \le k_2 D_G \\ f(x,y) & \text{otherwise} \end{cases}$$

where E, k_0, k_1, k_2, are specified parameters

Local enhancement with adaptive transformation function

Original moon image

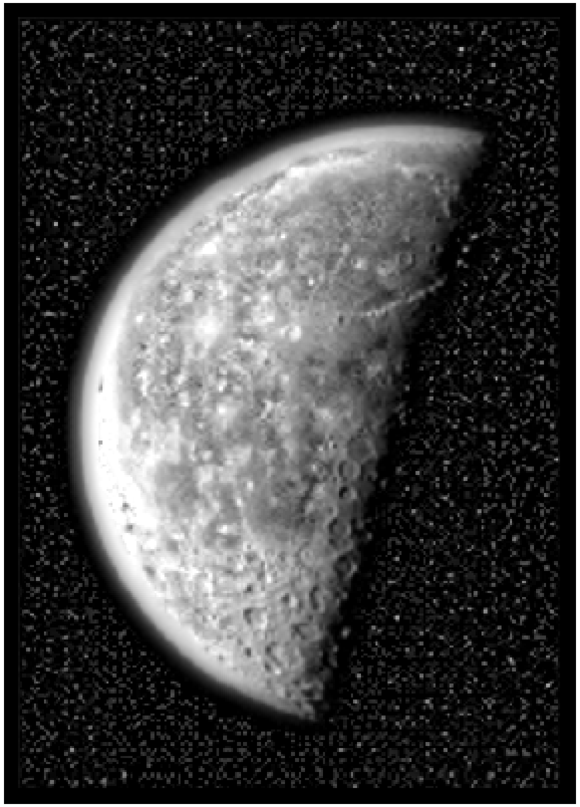


Image enhanced using adaptive transformation, window size: 15x15, k = 0.5

Histogram equalized image

Local enhancement using statistical parameters

Defining ranges for different transformation functions

$$g(x,y) = \begin{cases} E.f(x,y) & \text{if } m_{S_{xy}} \le k_0 M_G \text{ AND } k_1 D_D \le \sigma_{S_{xy}} \le k_2 D_G \\ f(x,y) & \text{otherwise} \end{cases}$$

FIGURE 3.24 SEM image of a tungsten filament and support, magnified approximately 130×.

Local enhancement using statistical parameters

Image formed from the local means

Image formed from the local standard deviations

Image formed from the multiplication constants selected for different mean/std ranges

$E = 4.0,\ k_0 = 0.4,\ k_1 = 0.02,$ and $k_2 = 0.4,$

Local enhancement using statistical parameters

Input microscopic image

Enhanced output image

Mathematical/logical operations on images

- Addition
  - *Averaging images for noise removal*
- Subtraction
  - *Removal of background from images*
  - *Image enhancement*
  - *Image matching*
  - *Moving/displaced object tracking*
- Multiplication
  - *Superimposing of texture on an image*
  - *Convolution and correlation of images*
- And and or operations
  - *To remove the unnecessary area of an image through mask operations*

Image averaging for noise reduction

A noisy image can be represented by

$$g(x, y) = f(x, y) + \eta(x, y),$$

where $\eta(x, y)$ denotes the noise in the image

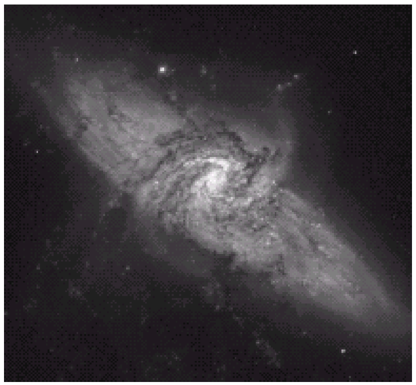
Since the noise is random and the content $f(x, y)$ is fixed,

The noise can be removed by taking more noisy images of the same object and averaging them out

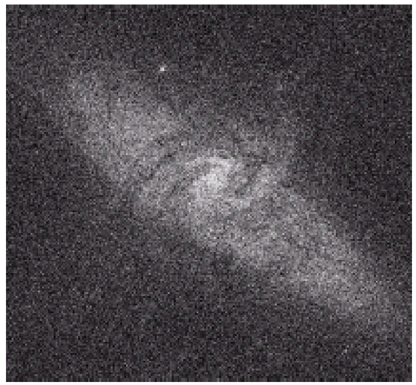
$$\overline{g}(x, y) = \frac{1}{K}\sum_{i=1}^{K} g_i(x, y),$$

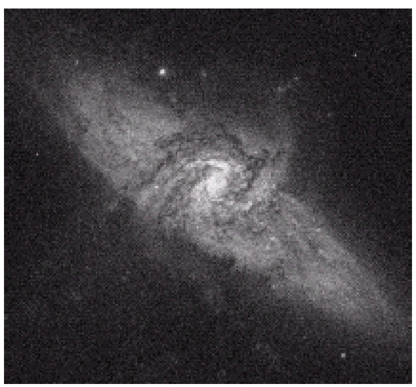
Image averaging for noise reduction

Original image

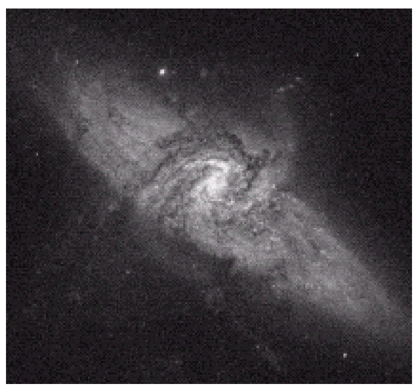


Noisy image

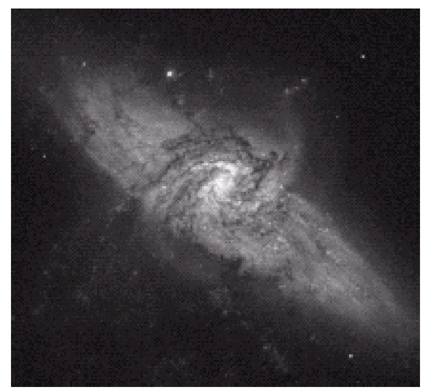




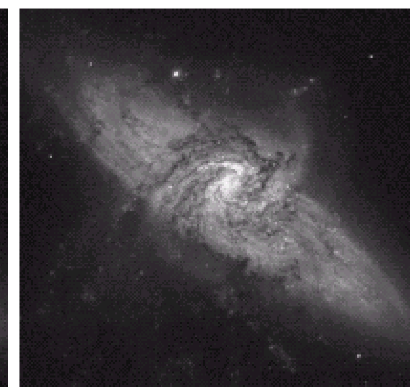
Result of averaging using 8 noise samples



Using 16 noise samples



Using 64 noise samples

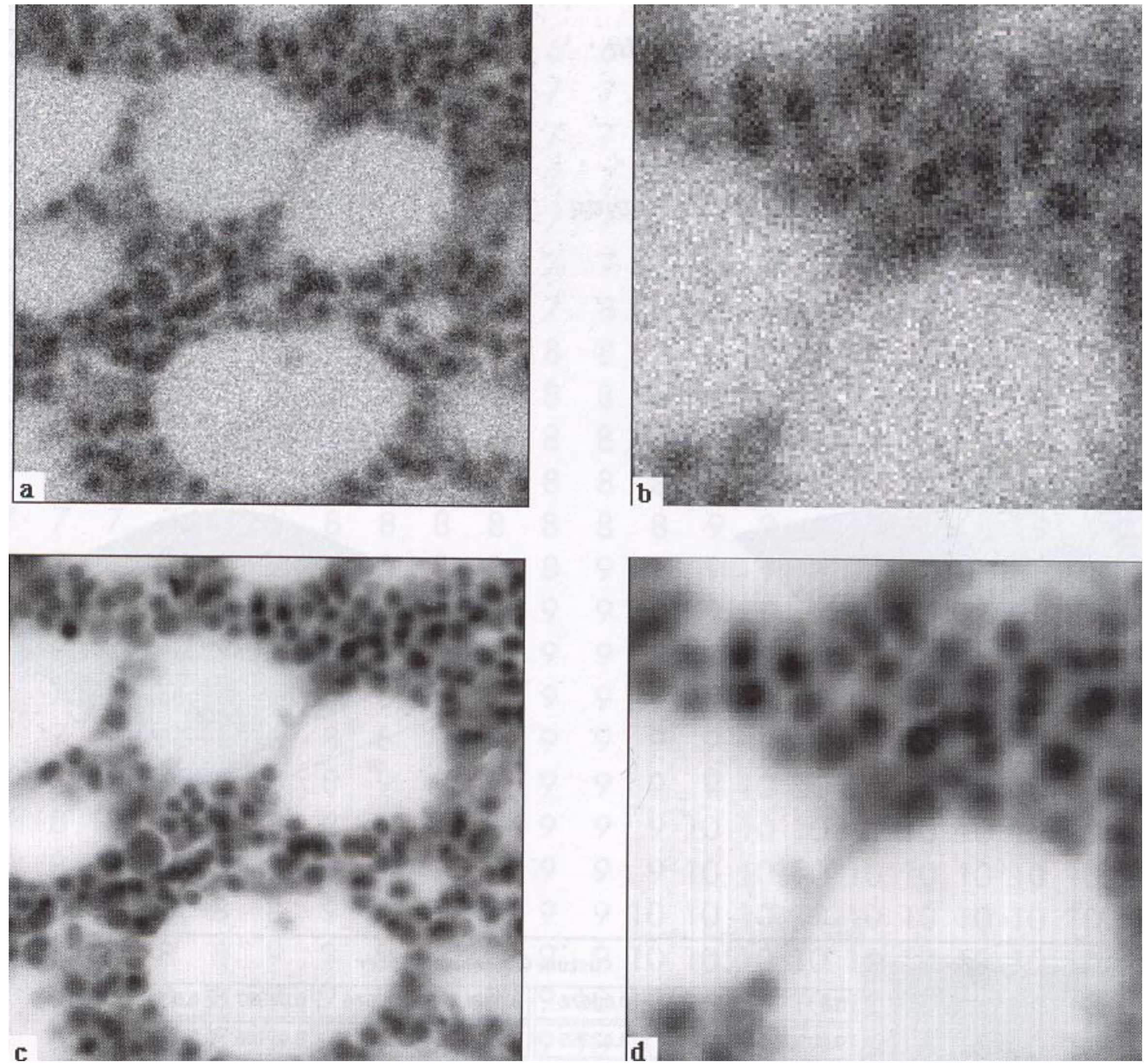


Using 128 noise samples

Image averaging for noise reduction

Noisy image

Noise reduction by averaging 256 samples



Examples of image subtraction

Original image

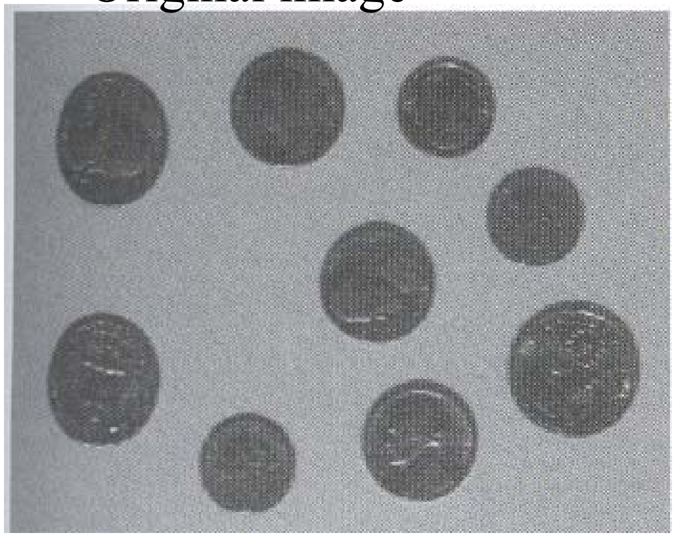
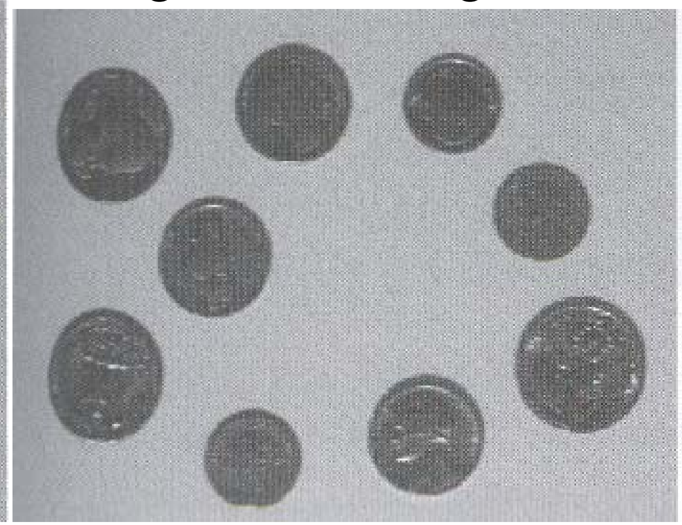
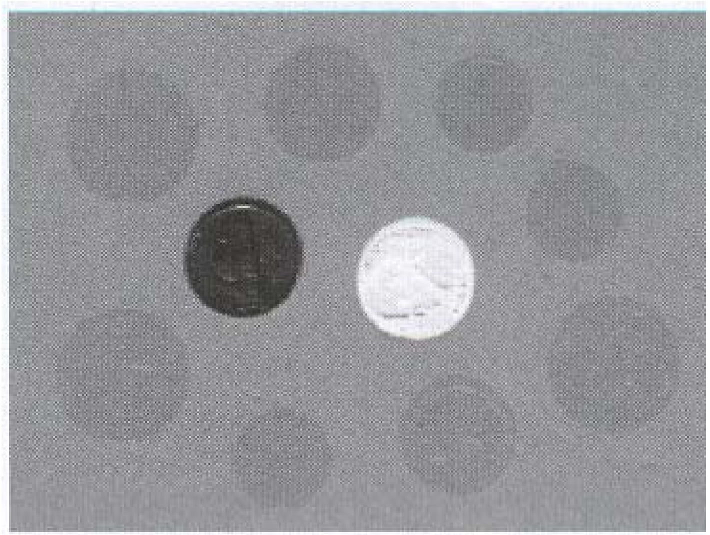


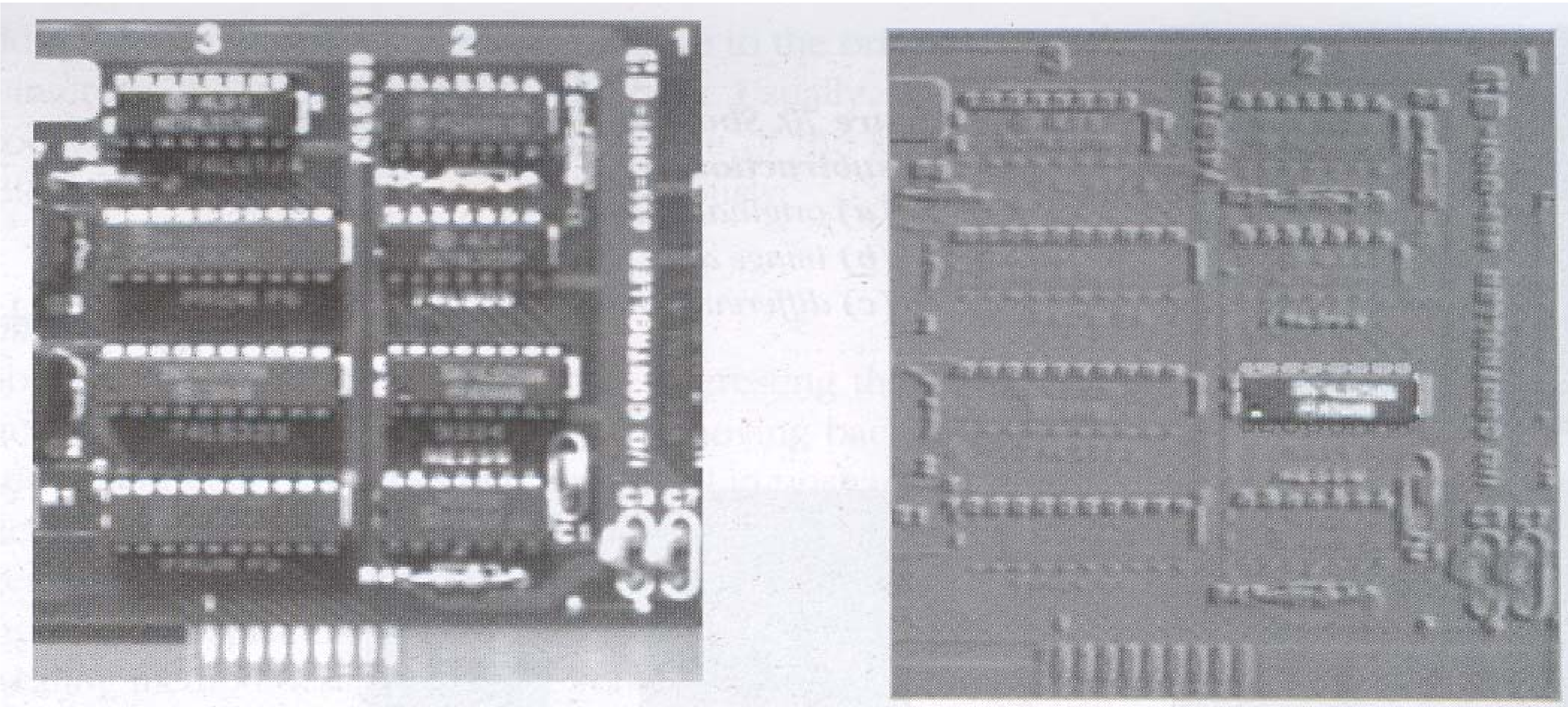
Image after moving one coin





Difference image after pixel by pixel subtraction of second image from first image

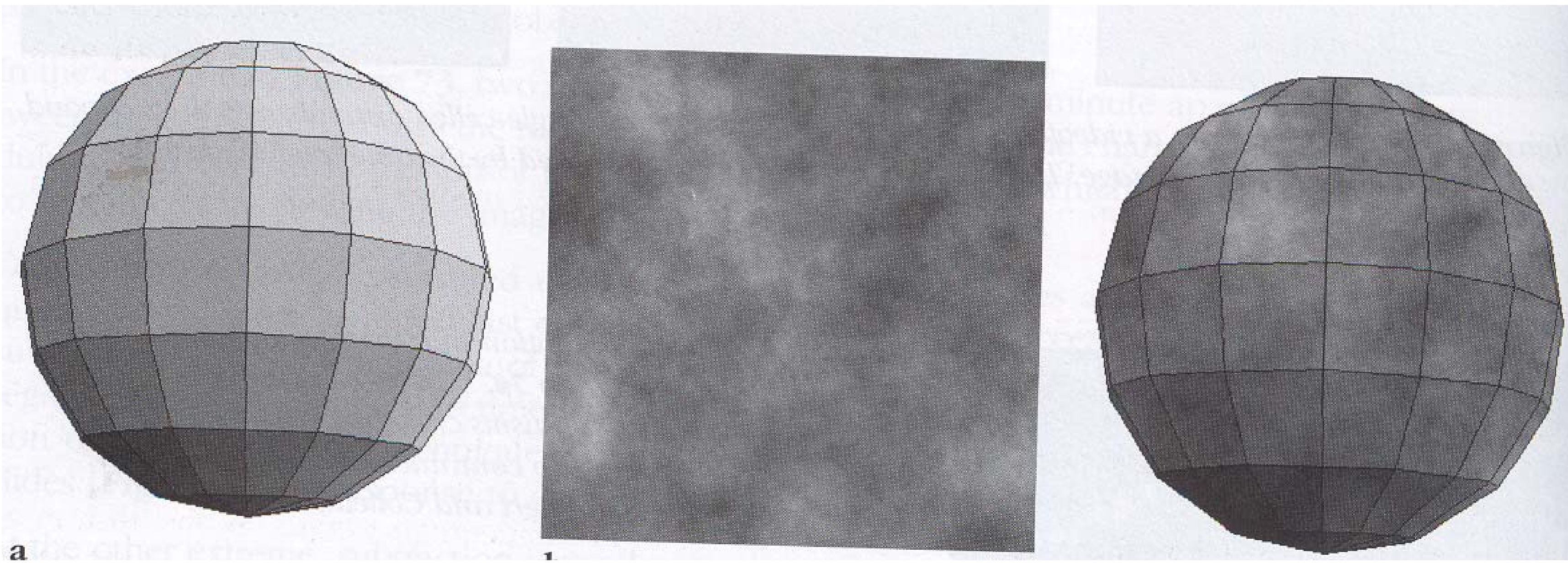
Examples of image subtraction



Difference of images from quality control: a missing chip in PCB is detected by subtracting the master image from image of each sample

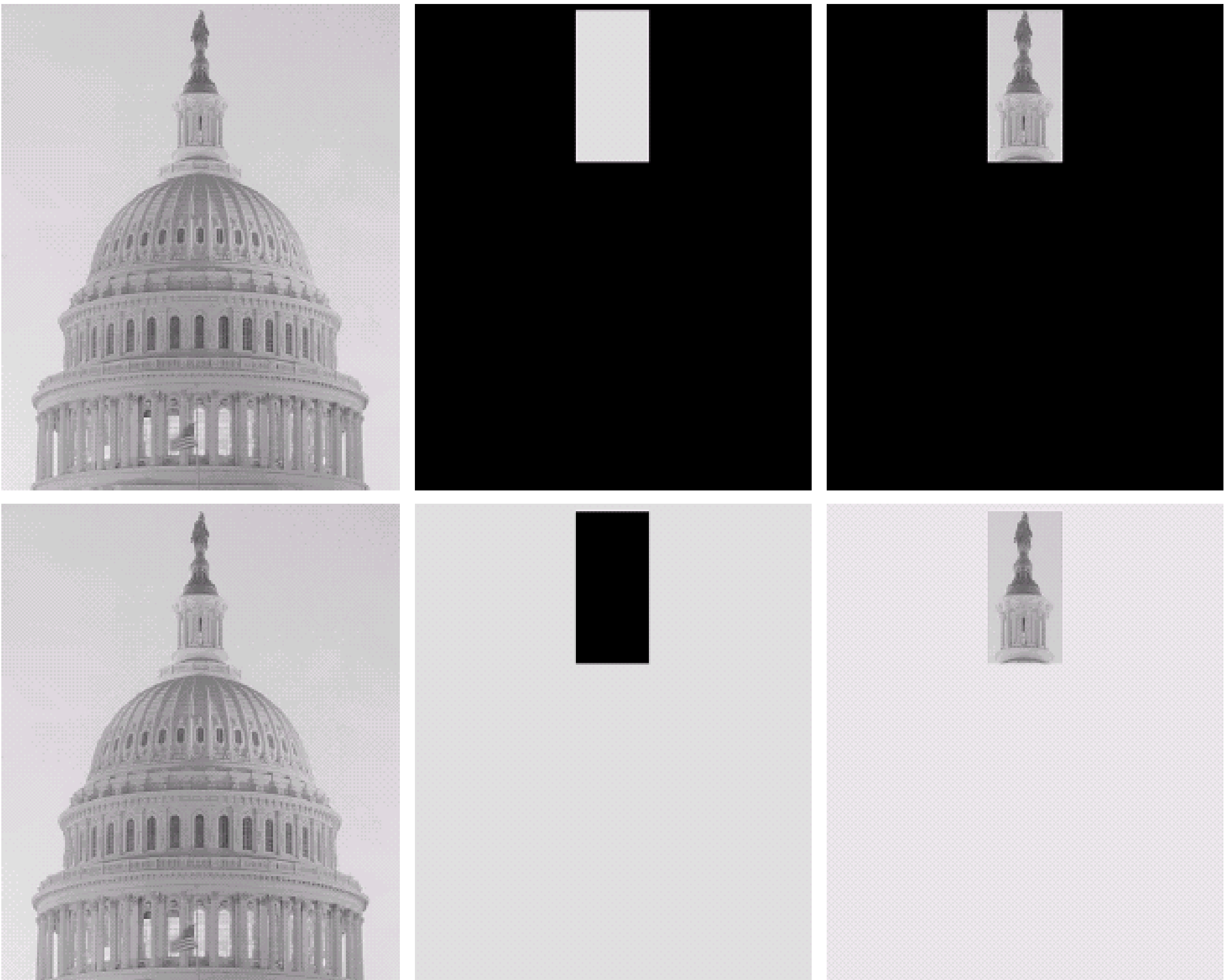
Examples of image Multiplication

Multiplication of images can be used for superimposing texture on an image



Smooth spherical surface image | Texture to be superimposed | output image

Example of logical operations using masks



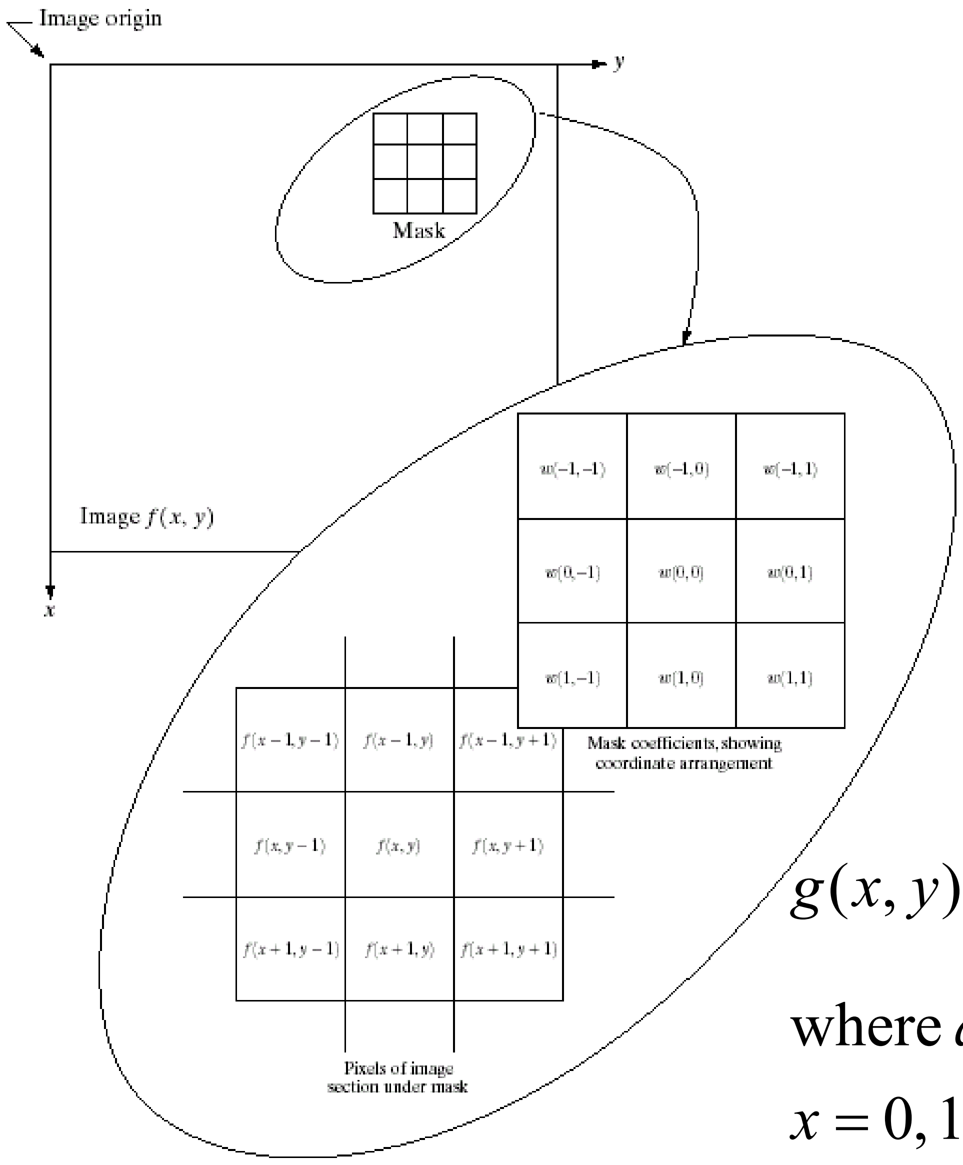
a	b	c
d	e	f

FIGURE 3.27 (a) Original image. (b) AND image mask. (c) Result of the AND operation on images (a) and (b). (d) Original image. (e) OR image mask. (f) Result of operation OR on images (d) and (e).

Local enhancement through spatial filtering

- The output intensity value at (x,y) depends not only on the input intensity value at (x,y) but also on the specified number of neighboring intensity values around (x,y)
- Spatial masks (also called window, filter, kernel, template) are used and convolved over the entire image for local enhancement (spatial filtering)
- The size of the masks determines the number of neighboring pixels which influence the output value at (x,y)
- The values (coefficients) of the mask determine the nature and properties of enhancing technique

Local enhancement through spatial filtering



The mechanics of spatial filtering

For an image of size M x N and a mask of size m x n

The resulting output gray level for any coordinates x and y is given by

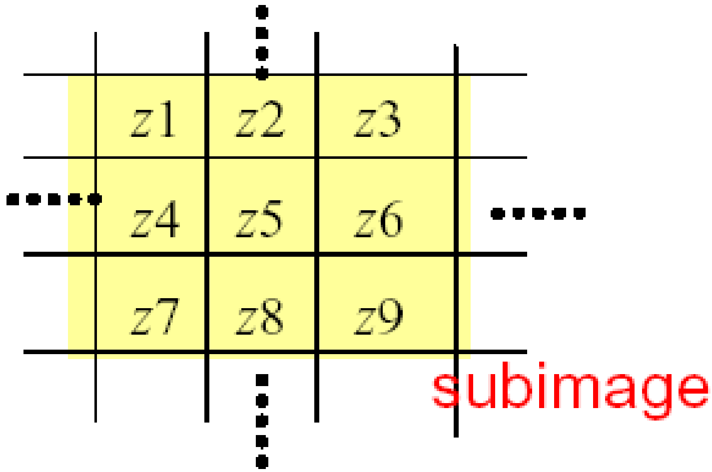
$$g(x,y)=\sum_{s=-a}^{a}\sum_{t=-b}^{b}w(s,t)f(x+s,y+t)$$

where $a=(m-1)/2, \quad b=(n-1)/2$

$x=0,1,2,\ldots\ldots,M-1, \quad y=0,1,2,\ldots\ldots,N-1,$

Basics of spatial filtering

- Given the 3×3 mask with coefficients: $w_1, w_2, \ldots, w_9$
- The mask cover the pixels with gray levels: $z_1, z_2, \ldots, z_9$



w1	w2	w3
w4	w5	w6
w7	w8	w9

Mask coefficients

$$z_5 \leftarrow z = w_1 z_1 + w_2 z_2 + \cdots + w_9 z_9 = \sum_{i=1}^{9} w_i z_i$$

- z gives the output intensity value for the processed image (to be stored in a new array) at the location of z_5 in the input image

Basics of spatial filtering

Mask operation near the image border

Problem arises when part of the mask is located outside the image plane; to handle the problem:

1. Discard the problem pixels (e.g. $512x512_{input}$ $510x510_{output}$ if mask size is 3x3)
2. Zero padding: expand the input image by padding zeros ($512x512_{input}$ $514x514_{output}$)
   - *Zero padding is not good create artificial lines or edges on the border;*
3. We normally use the gray levels of border pixels to fill up the expanded region (for 3x3 mask). For larger masks a border region equal to half of the mask size is mirrored on the expanded region.

Mask operation near the image border

102	102	130	143	123	115	…	…
102	102	130	143	123	115	…	…
93	93	…	…	…			
98	98	…					
82	82	…					
65	65						
…	…						
…	…						

Expanded area

Original image size (shaded area)

Spatial filtering for Smoothing

- For blurring/noise reduction;
- Blurring is usually used in preprocessing steps,

  e.g., to remove small details from an image prior to object extraction,

  or to bridge small gaps in lines or curves
- Equivalent to Low-pass spatial filtering in frequency domain because smaller (high frequency) details are removed based on neighborhood averaging (averaging filters)

Implementation: The simplest form of the spatial filter for averaging is a square mask (assume $m \times m$ mask) with the same coefficients $1/m^2$ to preserve the gray levels (averaging).

Applications: Reduce noise; smooth false contours

Side effect: Edge blurring

Smoothing filters

Consider the output pixel is positioned at the center

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

Box filter all coefficients are equal

Weighted average give more (less) weight to pixels near (away from) the output location

Consider mask size:

$$m\times n$$

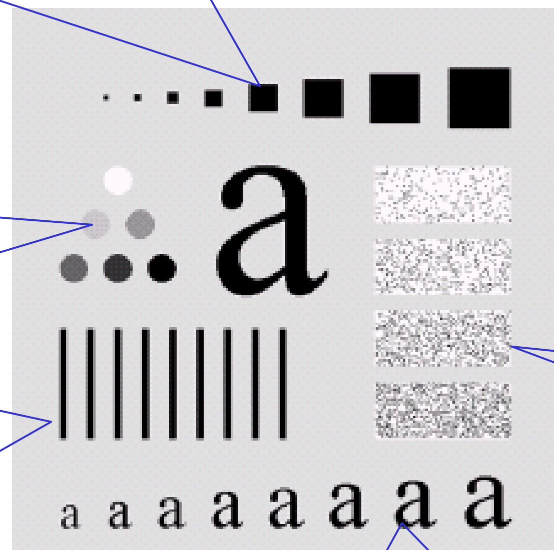
$$w_i=\frac{1}{mn}, i=1,\ldots,mn$$

Spatial filtering for Smoothing (example)

Squares of sizes: 3, 5, 9, 15, 25, 35, 45, and 55 pixels; borders are 25 pixels apart

Circles:
Diameter 25 pixels,
separation 15 pixels

Vertical bars:
5 pixels wide,
100 pixels high,
separation 20 pixels



Noisy rectangles:
50x120 pixels

Letter sizes: range from 10 to 24 points, in increments of 2 points

Spatial filtering for Smoothing (example)

Original image
size: 500 x 500

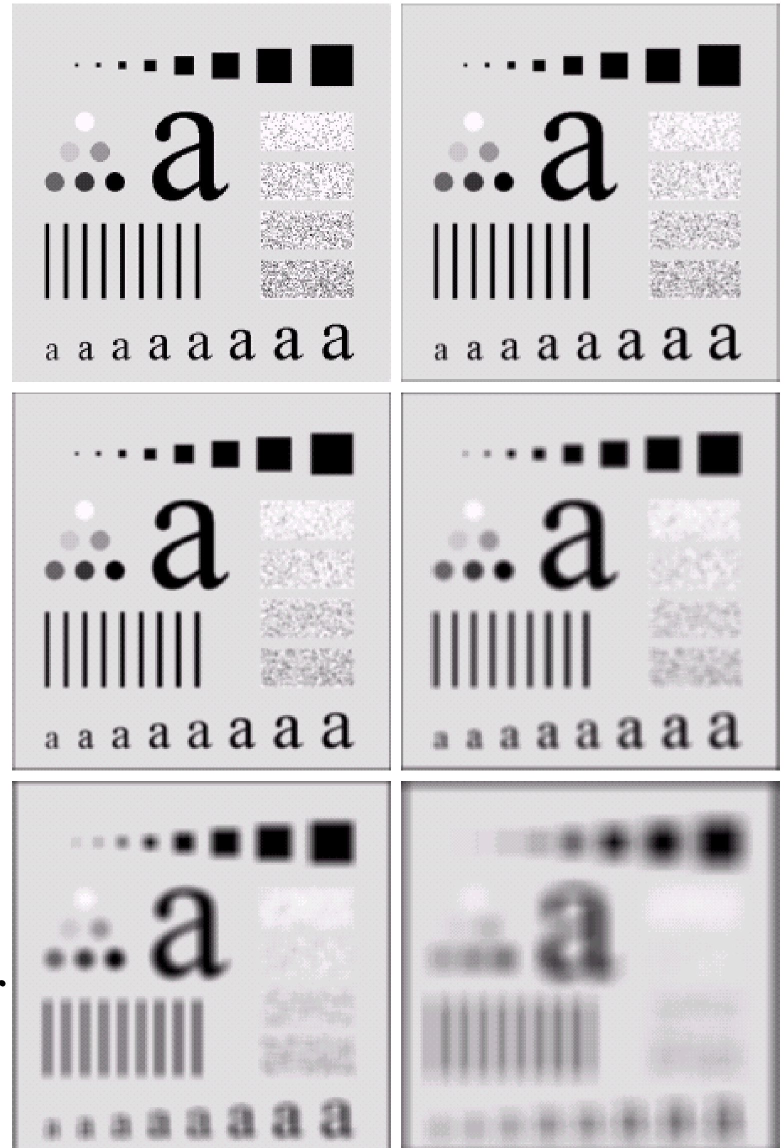
Smoothed by
3 x 3 box filter

Smoothed by
5 x 5 box filter

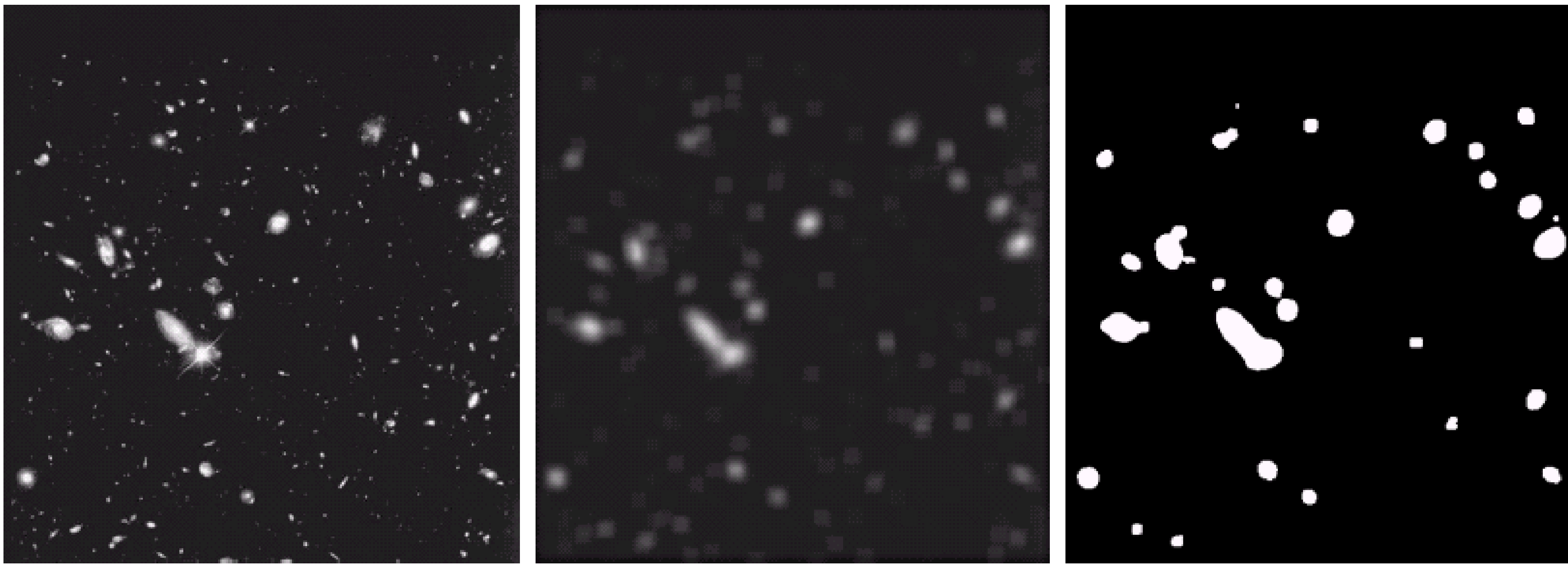
Smoothed by
9 x 9 box filter

Smoothed by
15 x 15 box filter

Smoothed by
35 x 35 box filter



Spatial filtering for Smoothing (example)



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.)

Order-statistics filtering

- Nonlinear spatial filters
- Output is based on order of gray levels in the masked area (sub-image)
- Examples: Median filtering, Max & Min filtering

Median filtering

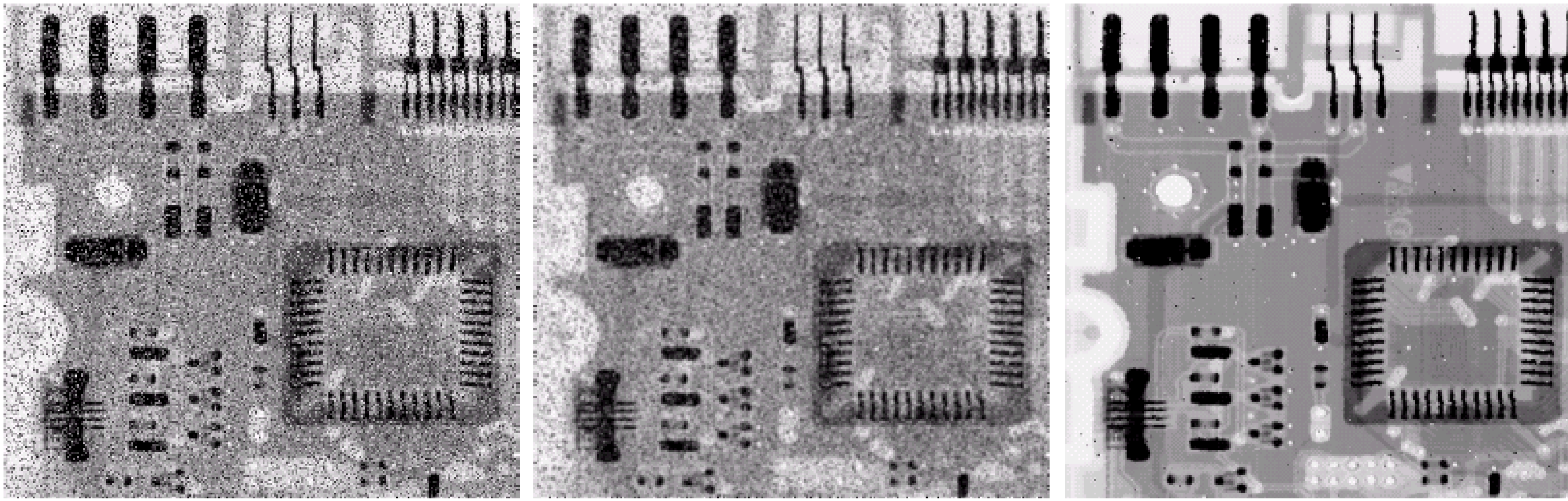
- Assigns the mid value of all the gray levels in the mask to the center of mask;
- Particularly effective when
  - *the noise pattern consists of strong, spiky components (impulse noise, salt-and-pepper*
  - *edges are to be preserved*
  - *Force points with distinct gray levels to be more like their neighbors*

Median Filtering

10	20	20
20	15	20
20	25	100

Output = ? **20**

Median Filtering (example)



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.)

Spatial filtering for image sharpening

Background: to highlight fine detail in an image or to enhance blurred detail

Applications: electronic printing, medical imaging, industrial inspection, autonomous target detection (smart weapons)......

Foundation:

- Blurring/smoothing is performed by spatial averaging (equivalent to integration)
- Sharpening is performed by noting only the gray level changes in the image that is the differentiation

Spatial filtering for image sharpening

Operation of Image Differentiation

- Enhance edges and discontinuities (magnitude of output gray level >>0)
- De-emphasize areas with slowly varying gray-level values (output gray level: 0)

Mathematical Basis of Filtering for Image Sharpening

- First-order and second-order derivatives
- Approximation in discrete-space domain
- Implementation by mask filtering

First and second order derivatives

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

$f(x)$	$f(x+1)$

Position for the output pixel

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

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

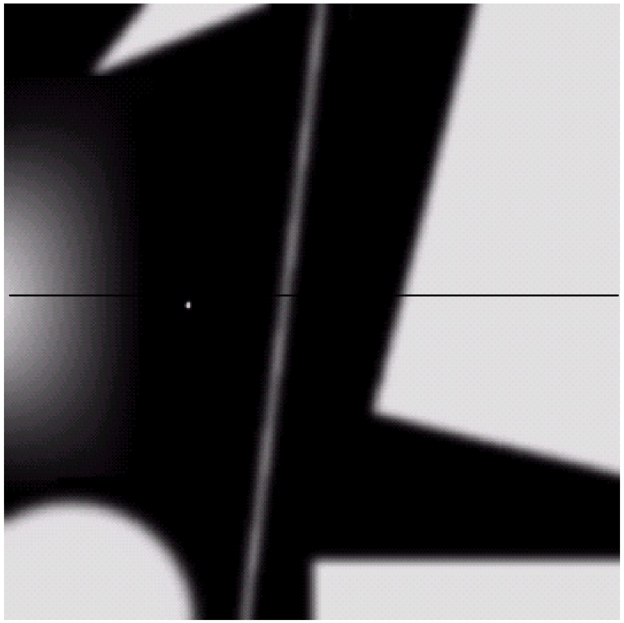
$$= f(x+2) - 2f(x+1) + f(x)$$

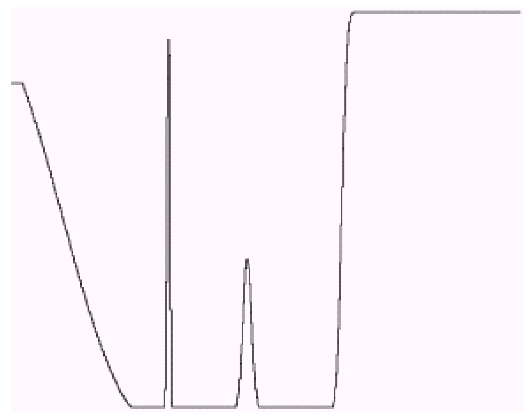
$f(x)$	$f(x+1)$	$f(x+2)$

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

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

Example for discrete derivatives





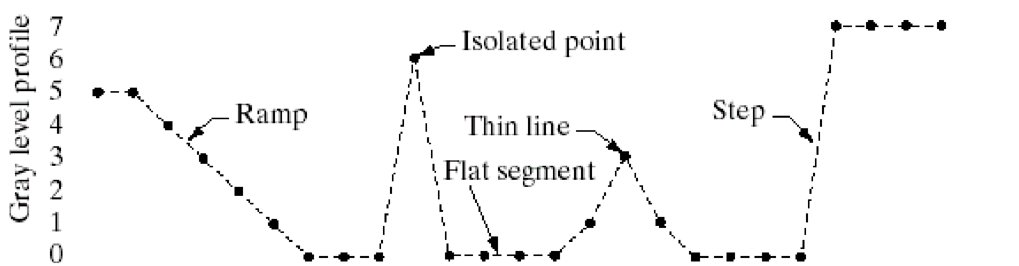


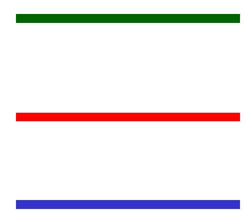
Image strip	5	5	4	3	2	1	0	0	0	6	0	0	0	0	1	3	1	0	0	0	0	7	7	7	7	•	•
First Derivative		−1	−1	−1	−1	−1	0	0	6	−6	0	0	0	1	2	−2	−1	0	0	0	7	0	0	0		
Second Derivative		−1	0	0	0	0	1	0	6	−12	6	0	0	1	1	−4	1	1	0	0	7	−7	0	0		

Various situations encountered for derivatives

$$f' = \frac{\partial f}{\partial x} \qquad f'' = \frac{\partial^2 f}{\partial x^2}$$

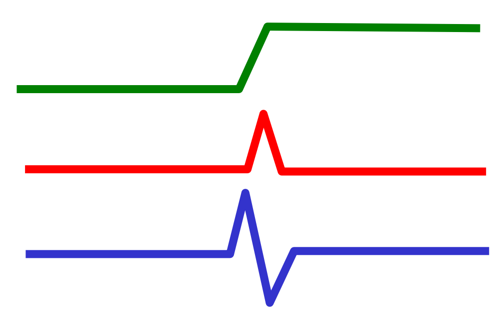
- Flat segment→ $(f')=0$; $(f'')=0$

f	0	0	0	0	0	
f'		0	0	0	0	
f''		0	0	0		



- Step→ (f'):{0,+,0}; (f''):{0,+,–,0}

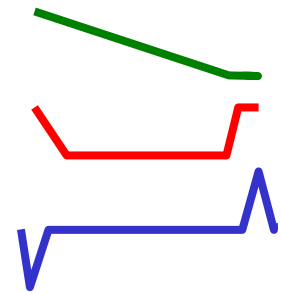
f	0	0	0	7	7	7	7	
f'		0	0	7	0	0	0	0
f''		0	7	–7	0	0	0	



Various situations encountered for derivatives

•Ramp→ (f')≈constant; (f'')=0

f	5	4	3	2	1	0	0	
f'	0	–1	–1	–1	–1	–1	0	0
f''	–1	0	0	0	0	1	0	

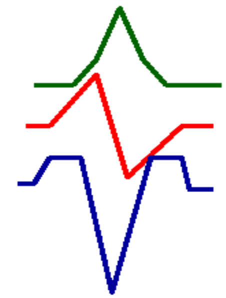


- *Ramps or steps* in the 1D profile normally characterize the edges in an image
- f'' is nonzero at the onset and end of the ramp: produce thin (double) edges
- f' is nonzero along the entire ramp produce thick edges

Various situations encountered for derivatives

- Thin lines

f	0	0	1	3	1	0	0	
f'	0	0	1	2	–2	–1	0	0
f''	0	1	1	–4	1	1	0	



- Isolated point

f	0	0	0	6	0	0	0	
f'		0	0	6	–6	0	0	0
f''		0	6	–12	6	0	0	

f'' responses much stronger than f' around the point

f'' enhances fine detail (including noise) much more than f'

Comparison between *f″* and *f´*

- ***f´*** generally produce thicker edges in an image
- ***f″*** have a stronger response to file detail
- ***f´*** generally have a stronger response to a gray-level step
- ***f″*** produces a double response at step changes in gray level
- ***f″*** responses given similar changes in gray-level values line > point > step
- For image enhancement, ***f″*** is generally better suited than ***f´***
- Major application of ***f´*** is for edge extraction; ***f´*** used together with ***f″*** results in impressive enhancement effect

Laplacian for image enhancement

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

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

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

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

y →

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

x ↓

0	1	0
1	–4	1
0	1	0

Laplacian for image enhancement

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

Isotropic for rotations in increments of 90°

$$+$$

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

Isotropic for rotations in increments of 45°

Laplacian for image enhancement

(a)

0	1	0
1	−4	1
0	1	0

(b)

1	1	1
1	−8	1
1	1	1

(c)

0	−1	0
−1	4	−1
0	−1	0

(d)

−1	−1	−1
−1	8	−1
−1	−1	−1

a	b
c	d

FIGURE 3.39 (a) Filter mask used to implement the digital Laplacian, as defined in Eq. (3.7-4). (b) Mask used to implement an extension of this equation that includes the diagonal neighbors. (c) and (d) Two other implementations of the Laplacian.

Laplacian for image enhancement

To obtain the enhanced image

$$g(x,y)=\begin{cases} f(x,y)-\nabla^2 f(x,y), w_5<0 \\ f(x,y)+\nabla^2 f(x,y), w_5>0 \end{cases}$$

w_1	w_2	w_3
w_4	w_5	w_6
w_7	w_8	w_9

In this way, background tonality can be perfectly preserved
while details are enhanced

–1	–1	–1
–1	8	–1
–1	–1	–1

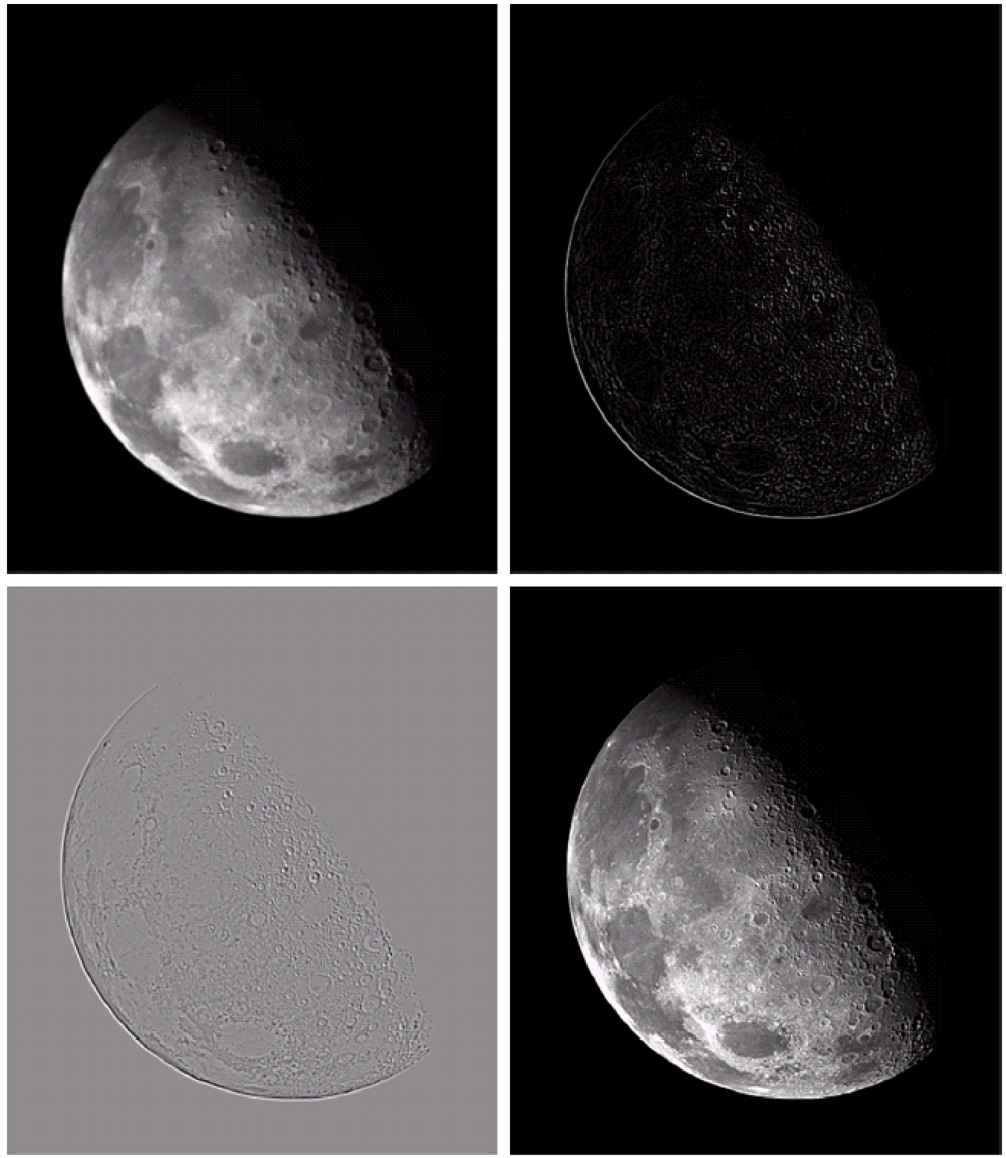
+

0	0	0
0	1	0
0	0	0

→

–1	–1	–1
–1	9	–1
–1	–1	–1

Laplacian for image enhancement (example)



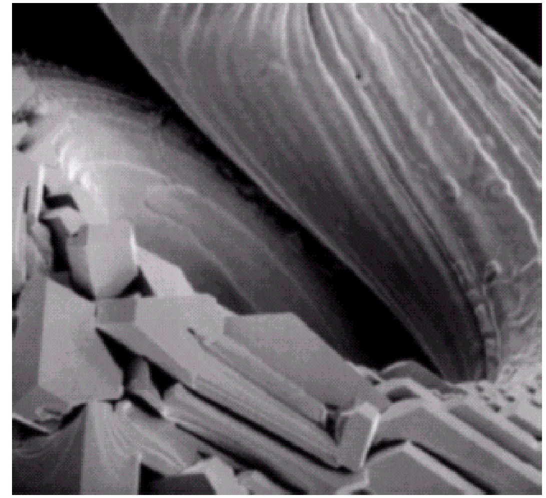
a b
c d

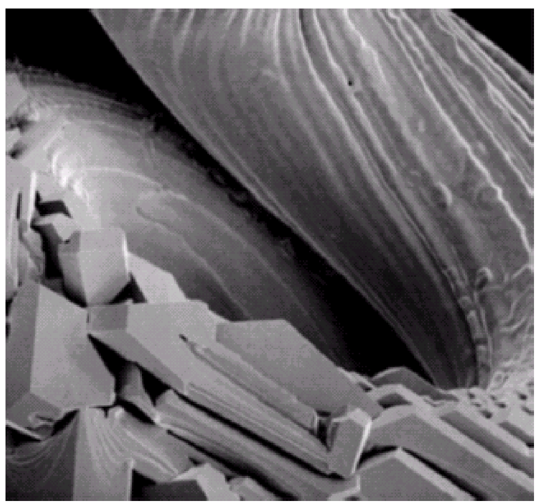
FIGURE 3.40
(a) Image of the North Pole of the moon.
(b) Laplacian-filtered image.
(c) Laplacian image scaled for display purposes.
(d) Image enhanced by using Eq. (3.7-5).
(Original image courtesy of NASA.)

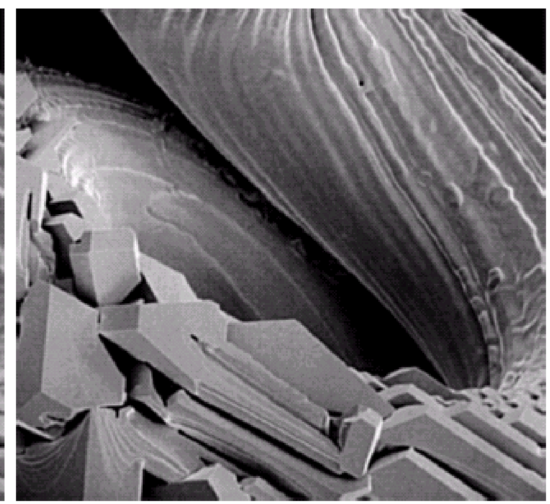
Laplacian for image enhancement (example)

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.)