Set 4: Intensity and Filtering

Lecturer Arto Kaarna

Lappeenranta University of Technology (LUT)
School of Engineering Science (LENS)
Machine Vision and Pattern Recognition (MVPR)

Arto.Kaarna@lut.fi

http://www.lut.fi/web/en/school-of-engineering-science/research/machine-vision-and-pattern-recognition

Contents

- Intensity Transformations
- Histograms
- Spatial Filtering
- Spatial Enhancement

Spatial Domain

- Spatial domain refers to image plane itself, operations applied to pixel values directly

$$g(x, y) = T[f(x, y)]$$

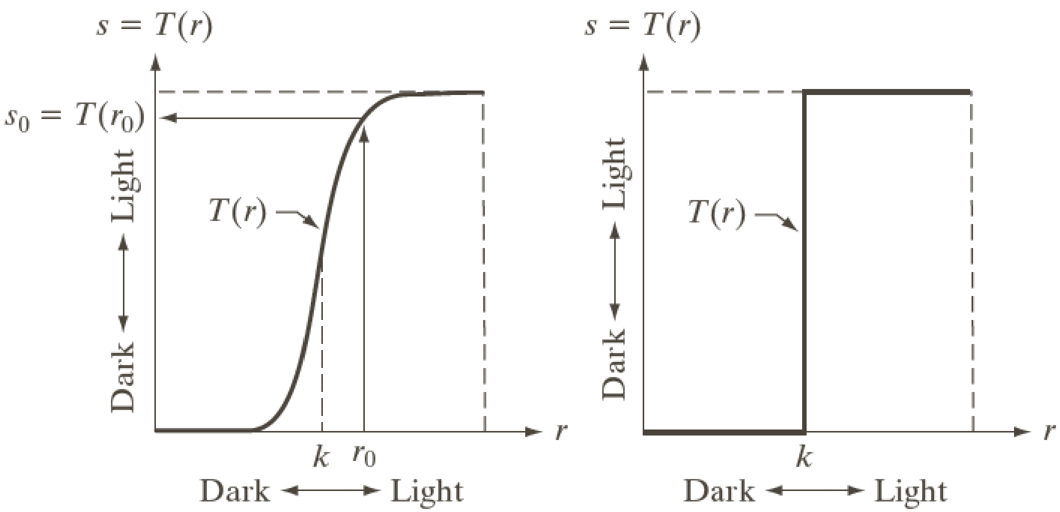
  - Efficient computationally, less processing
- Intensity transformations on single pixels
  - Contrast manipulation
  - Image thresholding
- Spatial filtering
  - Image enhancement
  - Image sharpening, pixels in a neighbourhood

Intensity Transformations

- Smallest neighbourhood is *1x1*, i.e. the pixel itself

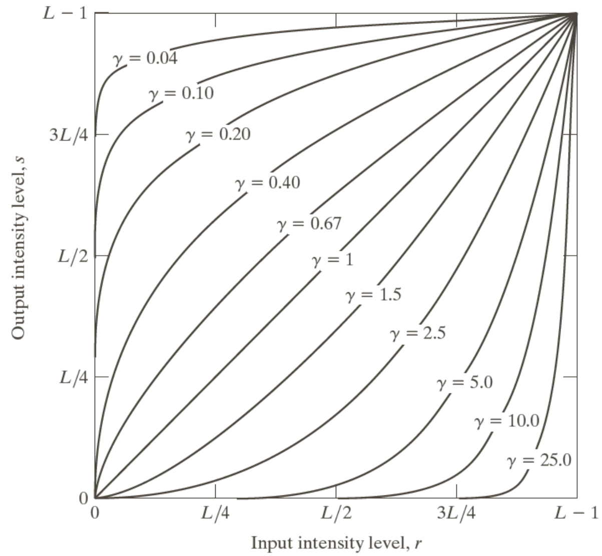
$$s = T(r)$$

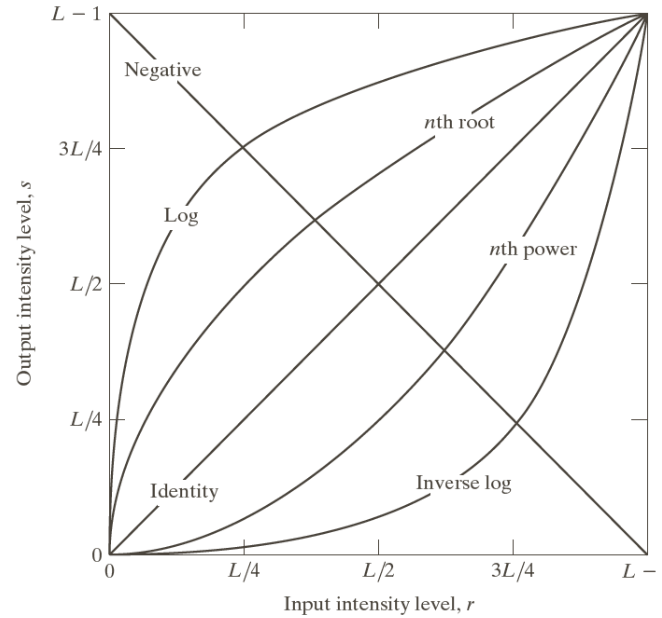
  - e.g. contrast stretching, thresholding



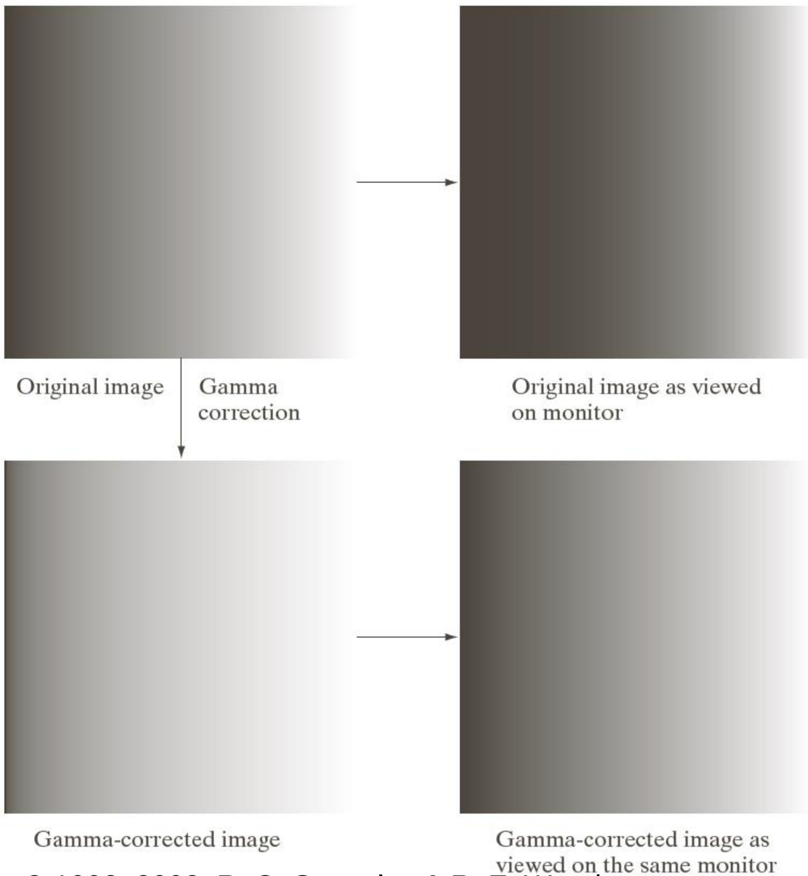
Intensity Transformations

- Negative image: $s = L - 1 - r$
- Log transformation: $s = c\log(1 + r)$
  - When is this needed?
- Power law: $s = cr^{\gamma}$ or $s = c(r + \varepsilon)^{\gamma}$

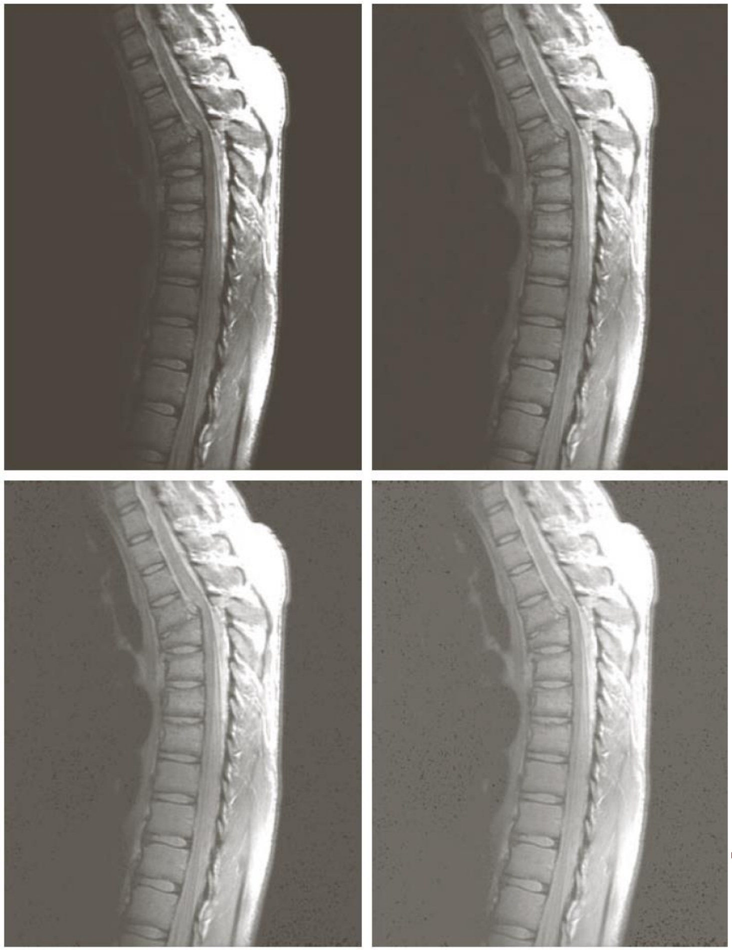




Gamma Correction

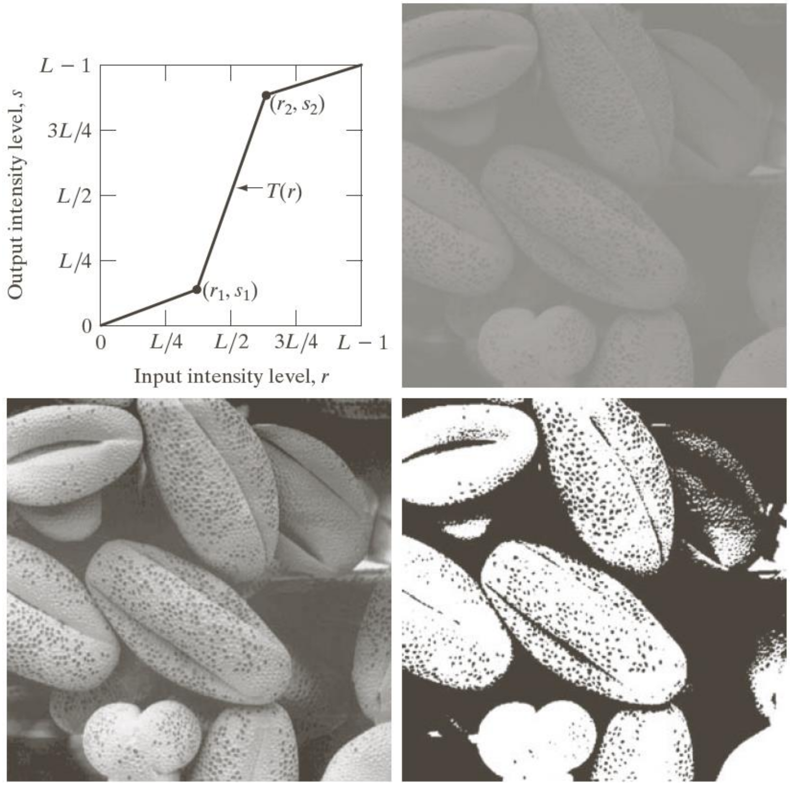


Original, $\gamma = 0.6, 0.4, 0.3$

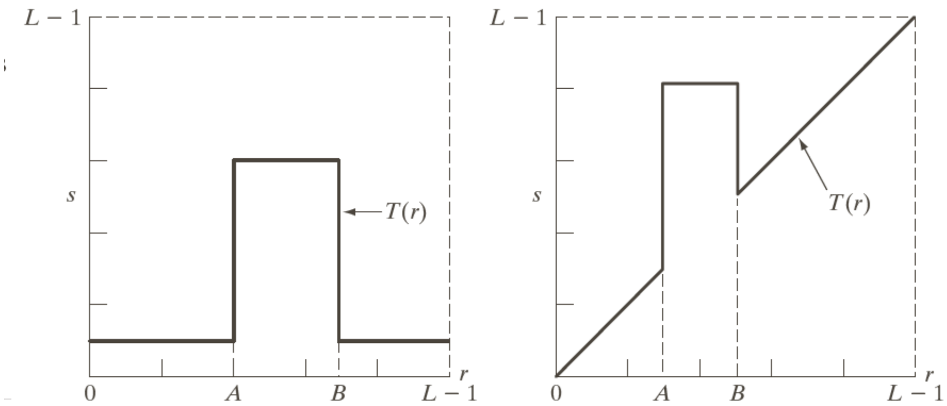


Piecewise-linear Transformations

- Contrast stretching and slicing the intensity levels

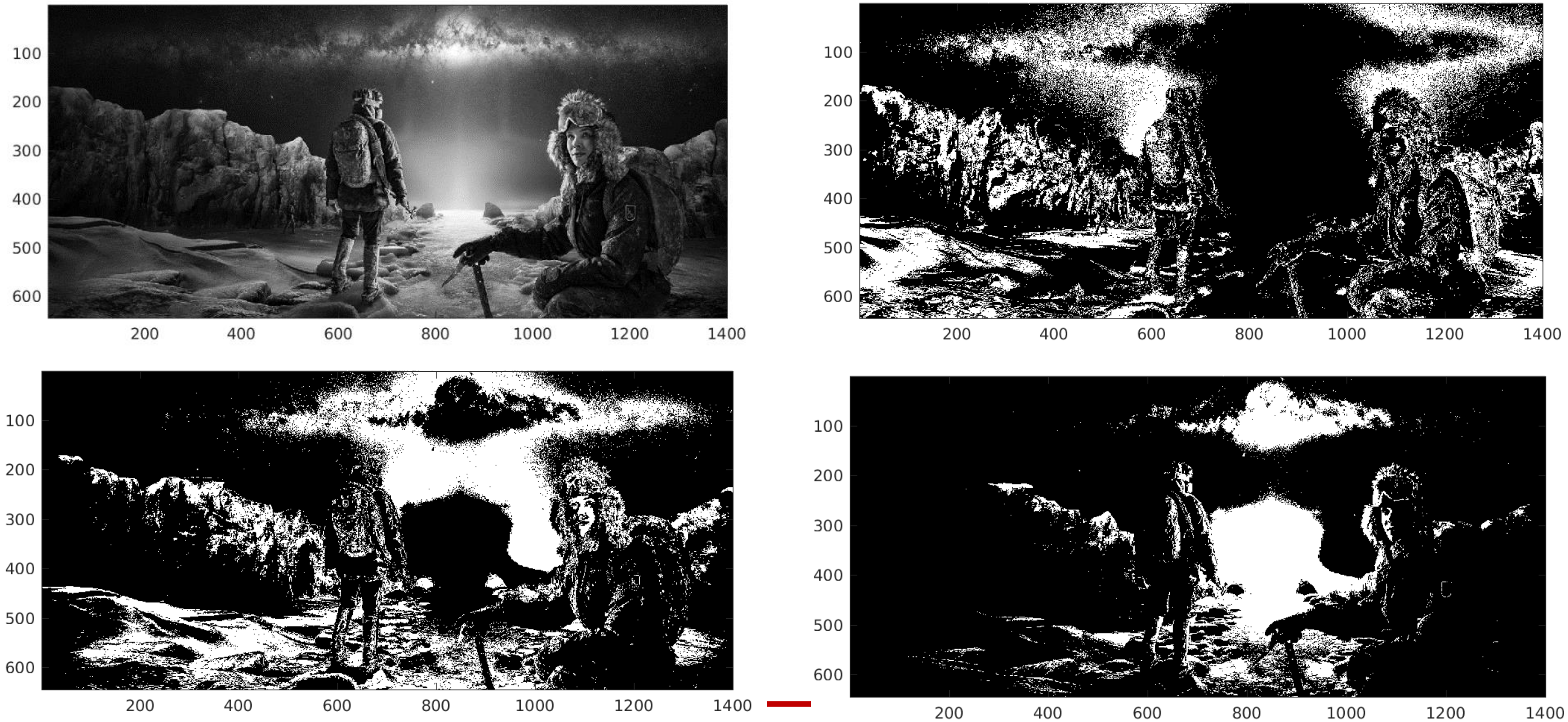


Highlighting a range of intensities



Piecewise-linear Transformations

- Processing bit-planes: with 8 bit-planes, 256 gray levels
- E.g. highest bitplanes: original, bitplanes 6,7, and 8.



Histogram Processing

- Histogram collects data from the pixel values

$$h(r_k) = n_k$$

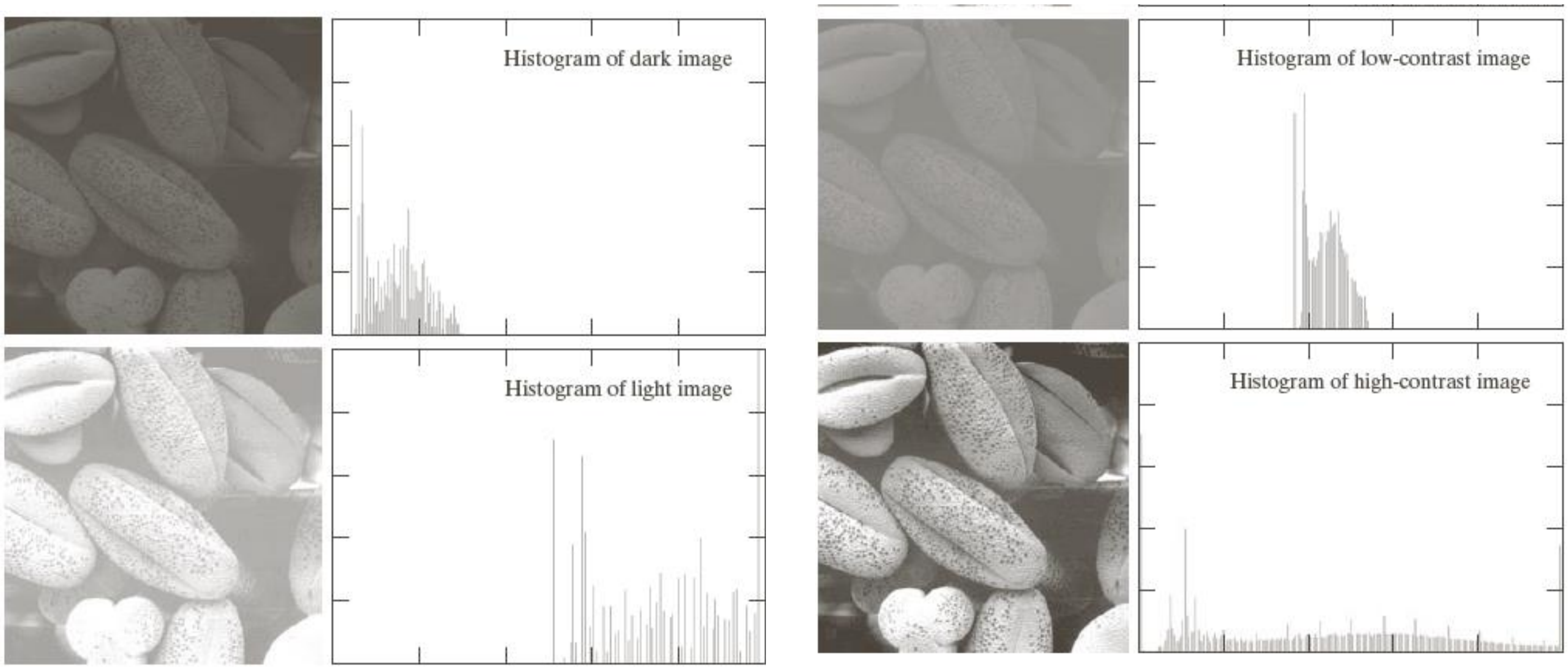
- r_k is the *k*th intensity value and n_k is the number of pixels with that value
- Normalized values are mostly used as a histogram

$$p_r(r_k) = \frac{n_k}{MN}, \qquad k = 0,1,2,\ldots,L-1$$

- Applications
  - Image enhancement, intensity transformations
  - Image compression and segmentation

Histogram Processing

- Examples on histograms



Histogram Processing

- Histogram equalization tries to find a constant p_r

$$s_k = T(r_k) = (L-1)\sum_{j=0}^{k} p_r(r_j) = \frac{(L-1)}{MN}\sum_{j=0}^{k} n_j$$

$$k = 0,1,2, \ldots L-1$$

- For the transform *T* there is also an inverse transform

$$r_k = T^{-1}(s_k),\ k = 0,1,2, \ldots L-1$$

- Why this is important? How this was obtained?

Histogram Processing

- Discrete form is derived from the continuous case of PDFs for $p_r(r)$ and $p_s(s)$ and from the CDF on the right hand side as

$$s = T(r) = (L-1)\int_0^r p_r(w)dw$$

- which finally will lead to

$$p_s(s) = p_r(r)\left|\frac{1}{(L-1)p_r(r)}\right| = \frac{1}{L-1}$$

Histogram Processing

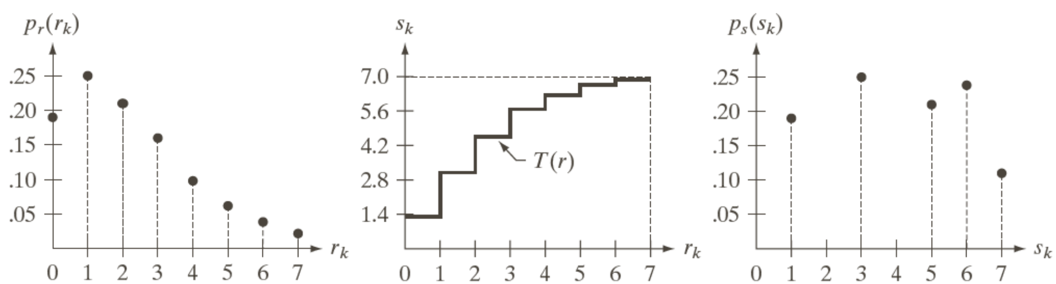
- Example on histogram equalization

$$s_0 = T(r_0)$$
$$= (8-1)\sum_{j=0}^{0} p_r(r_j)$$
$$= 7p_0(r_o) = 1.33 = 1$$

$$s_{1-7} = 3{,}5{,}6{,}7{,}7{,}7$$

r_k	n_k	$p_r(r_k) = n_k/MN$
$r_0 = 0$	790	0.19
$r_1 = 1$	1023	0.25
$r_2 = 2$	850	0.21
$r_3 = 3$	656	0.16
$r_4 = 4$	329	0.08
$r_5 = 5$	245	0.06
$r_6 = 6$	122	0.03
$r_7 = 7$	81	0.02

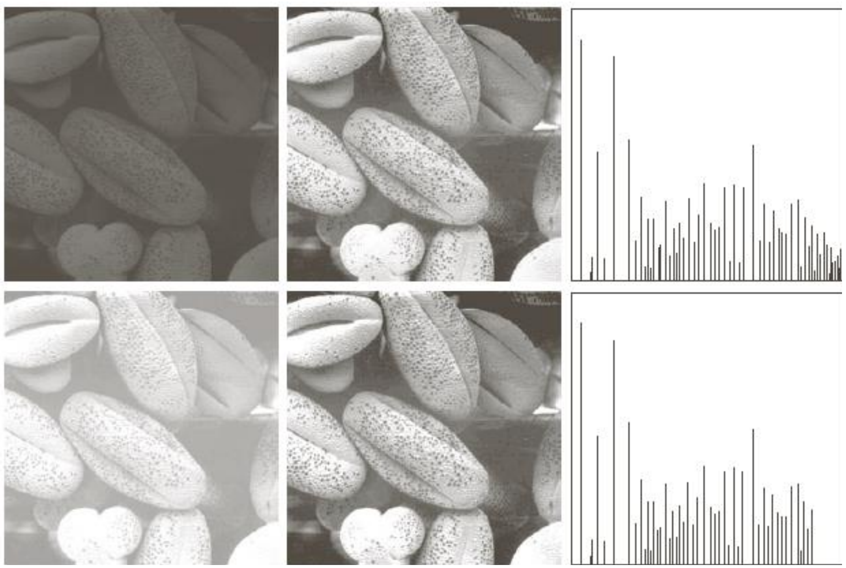
TABLE 3.1 Intensity distribution and histogram values for a 3-bit, 64 × 64 digital image.

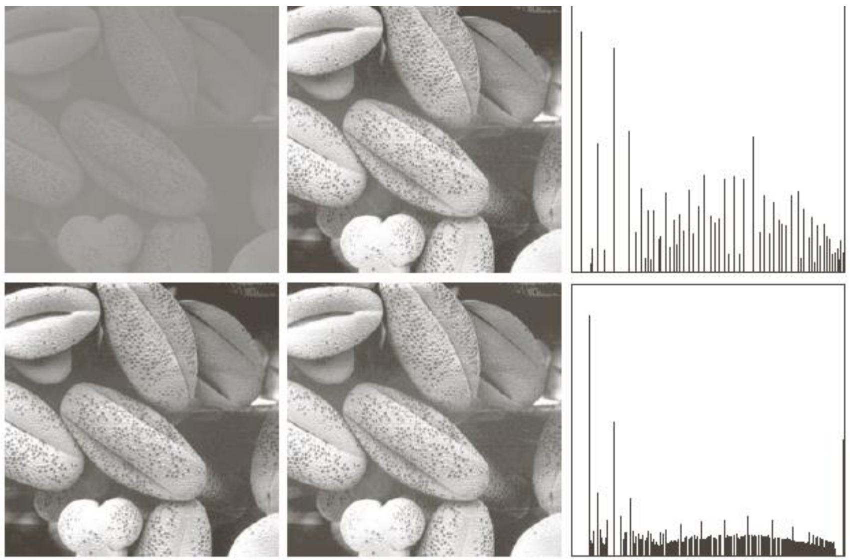


a b c

FIGURE 3.19 Illustration of histogram equalization of a 3-bit (8 intensity levels) image. (a) Original histogram. (b) Transformation function. (c) Equalized histogram.

Histogram Processing





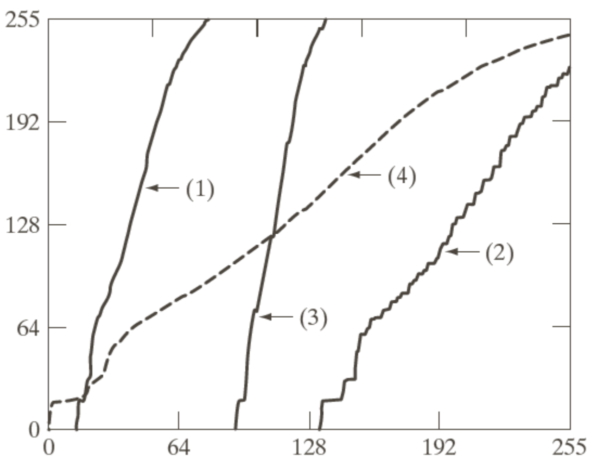


FIGURE 3.21 Transformation functions for histogram equalization. Transformations (1) through (4) were obtained from the histograms of the images (from top to bottom) in the left column of Fig. 3.20 using Eq. (3.3-8).

Histogram Processing

- Histogram matching enables a transformation where the target histogram can be designed (non-flat)

1. Obtain $p_r(r)$ from the input image and values for s.

$$s = T(r) = (L-1)\int_0^r p_r(w)dw \qquad \text{(equalization)}$$

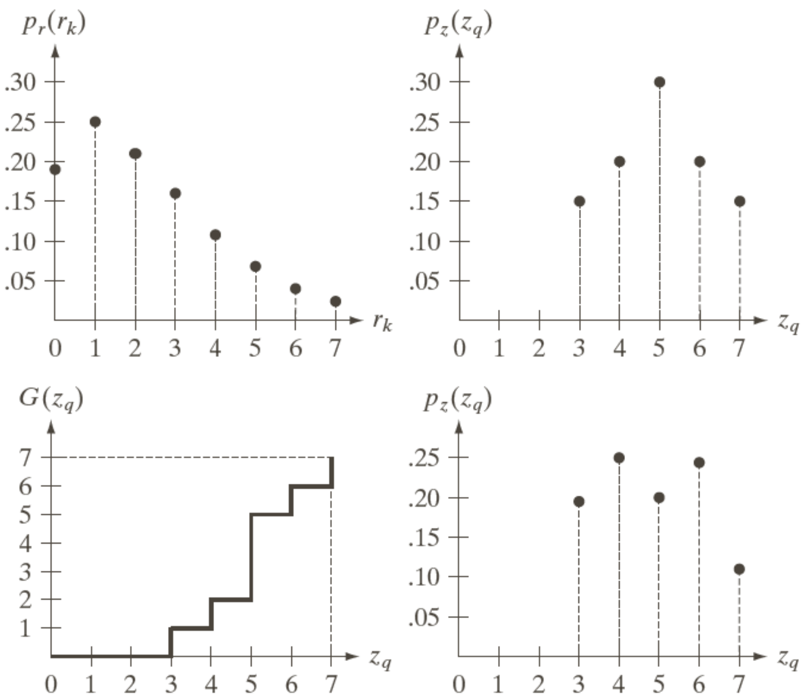
2. Specify the form for the output histogram and the corresponding transformation function

$$G(z) = (L-1)\int_0^z p_z(t)dt = s$$

3. Obtain inverse transformation $z = G^{-1}(s)$
4. Obtain the output image with the values z. (Map s to z.)

Histogram Processing

1. $s_{0-7} = 1,3,5,6,7,7,7,7$
2. $G(z_0) = 7\sum_{j=0}^{0} p_z(z_j) = 0.0$, etc.
3. Round G, construct tables.



a b
c d

FIGURE 3.22 (a) Histogram of a 3-bit image. (b) Specified histogram. (c) Transformation function obtained from the specified histogram. (d) Result of performing histogram specification. Compare (b) and (d).

z_q	Specified $p_z(z_q)$	Actual $p_z(z_k)$
$z_0 = 0$	0.00	0.00
$z_1 = 1$	0.00	0.00
$z_2 = 2$	0.00	0.00
$z_3 = 3$	0.15	0.19
$z_4 = 4$	0.20	0.25
$z_5 = 5$	0.30	0.21
$z_6 = 6$	0.20	0.24
$z_7 = 7$	0.15	0.11

TABLE 3.2 Specified and actual histograms (the values in the third column are from the computations performed in the body of Example 3.8).

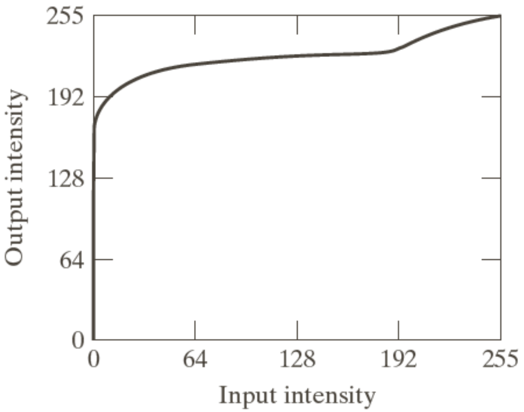
z_q	$G(z_q)$
$z_0 = 0$	0
$z_1 = 1$	0
$z_2 = 2$	0
$z_3 = 3$	1
$z_4 = 4$	2
$z_5 = 5$	5
$z_6 = 6$	6
$z_7 = 7$	7

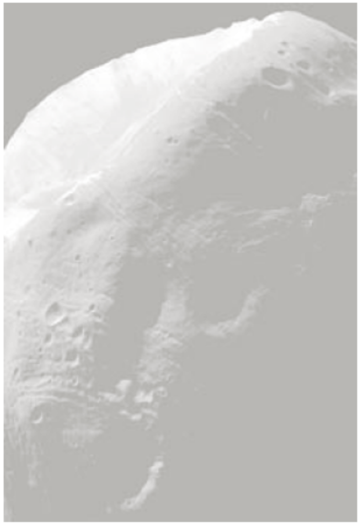
TABLE 3.3 All possible values of the transformation function G scaled, rounded, and ordered with respect to z.

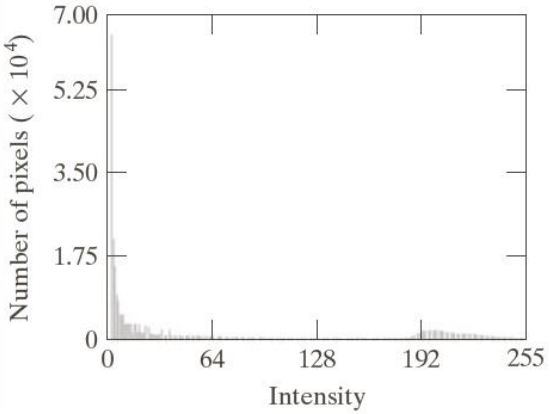
s_k	$\rightarrow$	z_q
1	$\rightarrow$	3
2	$\rightarrow$	4
5	$\rightarrow$	5
6	$\rightarrow$	6
7	$\rightarrow$	7

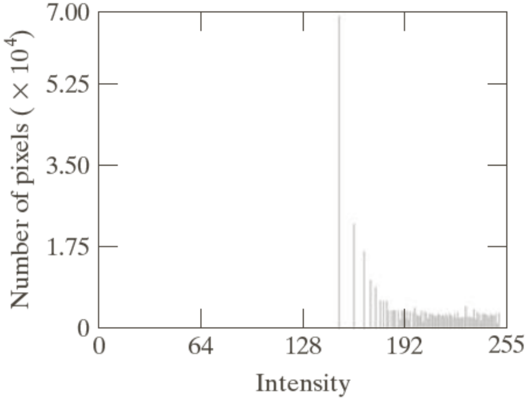
Histogram Processing

- Example: original image and the histogram
- Transformation function, new image and histogram.



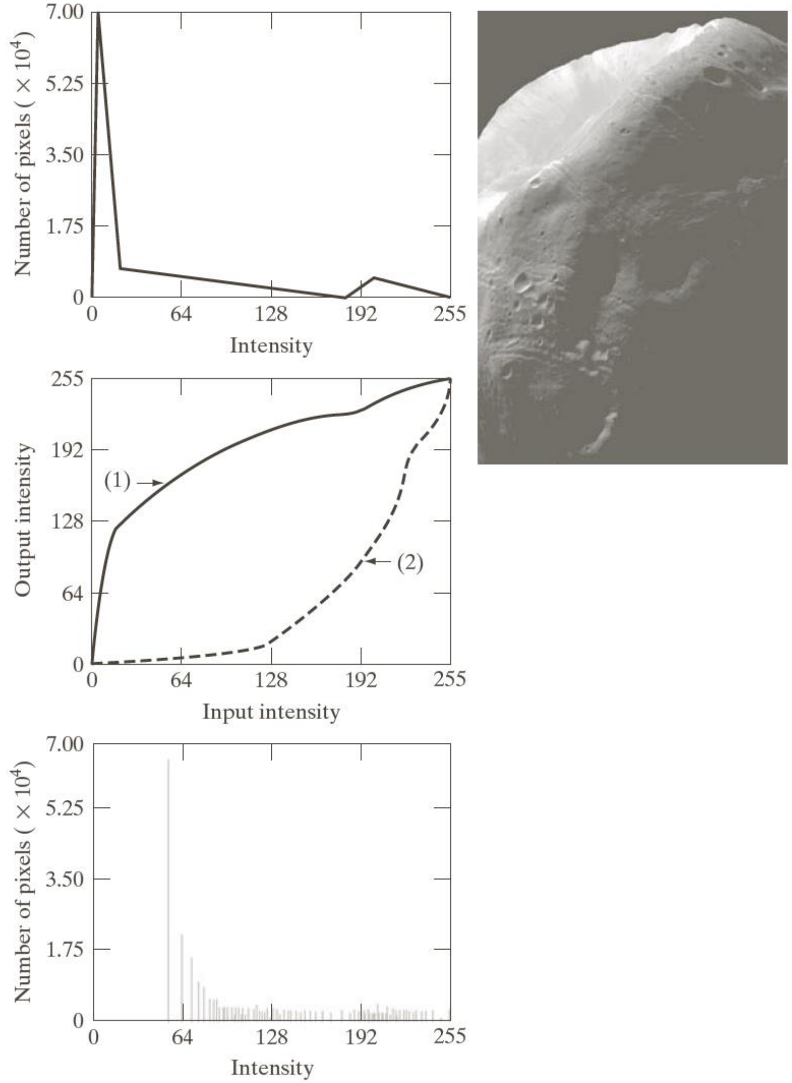






Histogram Processing

- Example: image; specified histogram; transformation functions; final histogram (curve (2))



a c
b
d

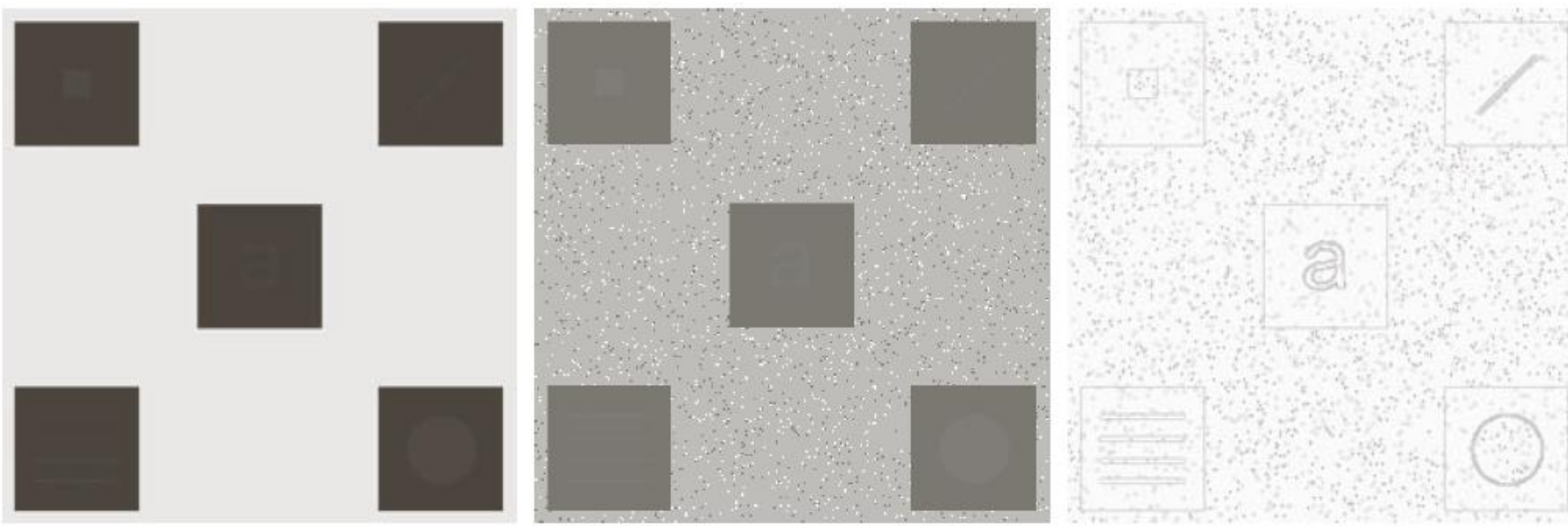
FIGURE 3.25
(a) Specified histogram.
(b) Transformations.
(c) Enhanced image using mappings from curve (2).
(d) Histogram of (c).

Histogram Processing

- Local histograms are computed from the neighbourhood of the pixel.
- Local histograms may be computed
  - Based on the local histogram computed earlier. (Why?)
  - Non-overlapping regions may end up to blocking
- The mean and the variance of the neighbourhood are used to perform local changes in the image
  - Many zero components in the histogram
    - 3x3 neigbourhood, 9 out of 256 possible values
  - Contrast is corresponding to the local variance

Histogram Processing

- Histogram equalization: global histogram, local histogram
- Original image has imperceptible noise, smooth regions

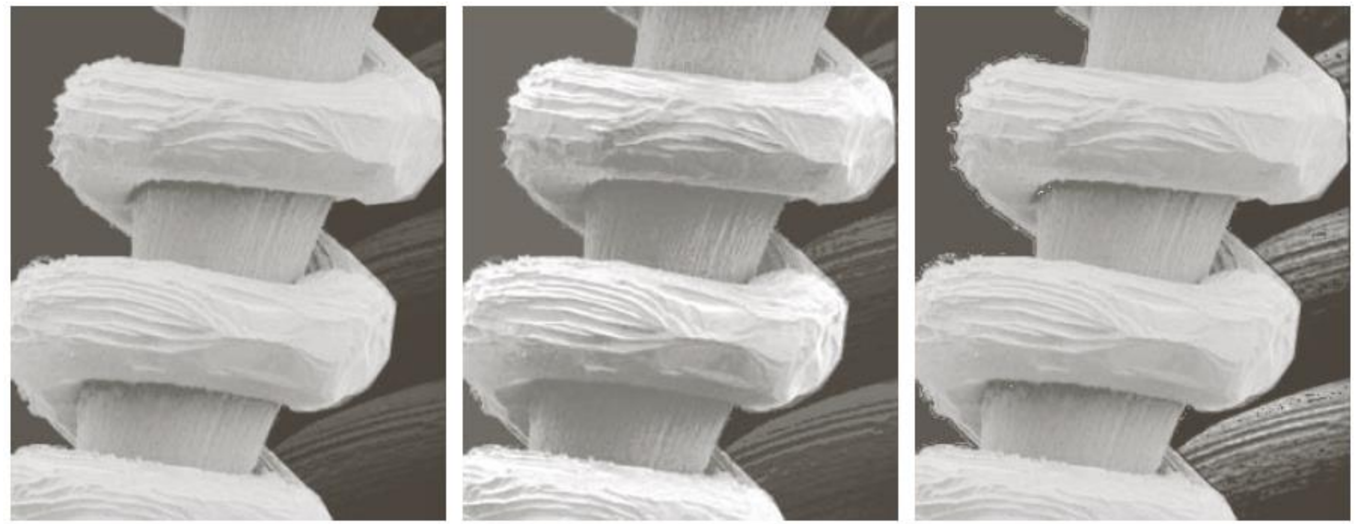


a b c

FIGURE 3.26 (a) Original image. (b) Result of global histogram equalization. (c) Result of local histogram equalization applied to (a), using a neighborhood of size 3×3.

Histogram Processing

- Example on global and local histogram equalization



a b c

FIGURE 3.27 (a) SEM image of a tungsten filament magnified approximately 130×. (b) Result of global histogram equalization. (c) Image enhanced using local histogram statistics. (Original image courtesy of Mr. Michael Shaffer, Department of Geological Sciences, University of Oregon, Eugene.)

Spatial Filtering

- Spatial filtering is a basic tool in image processing
  - Filter, lowpass/highpass filter
  - Filter may be a mask, kernel, template, window
- Linear spatial filtering with filter w

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

- Correlation vs. convolution
  - Moving a mask and computing values at each location
  - In convolution the filter is rotated 180 degree

Spatial Filtering

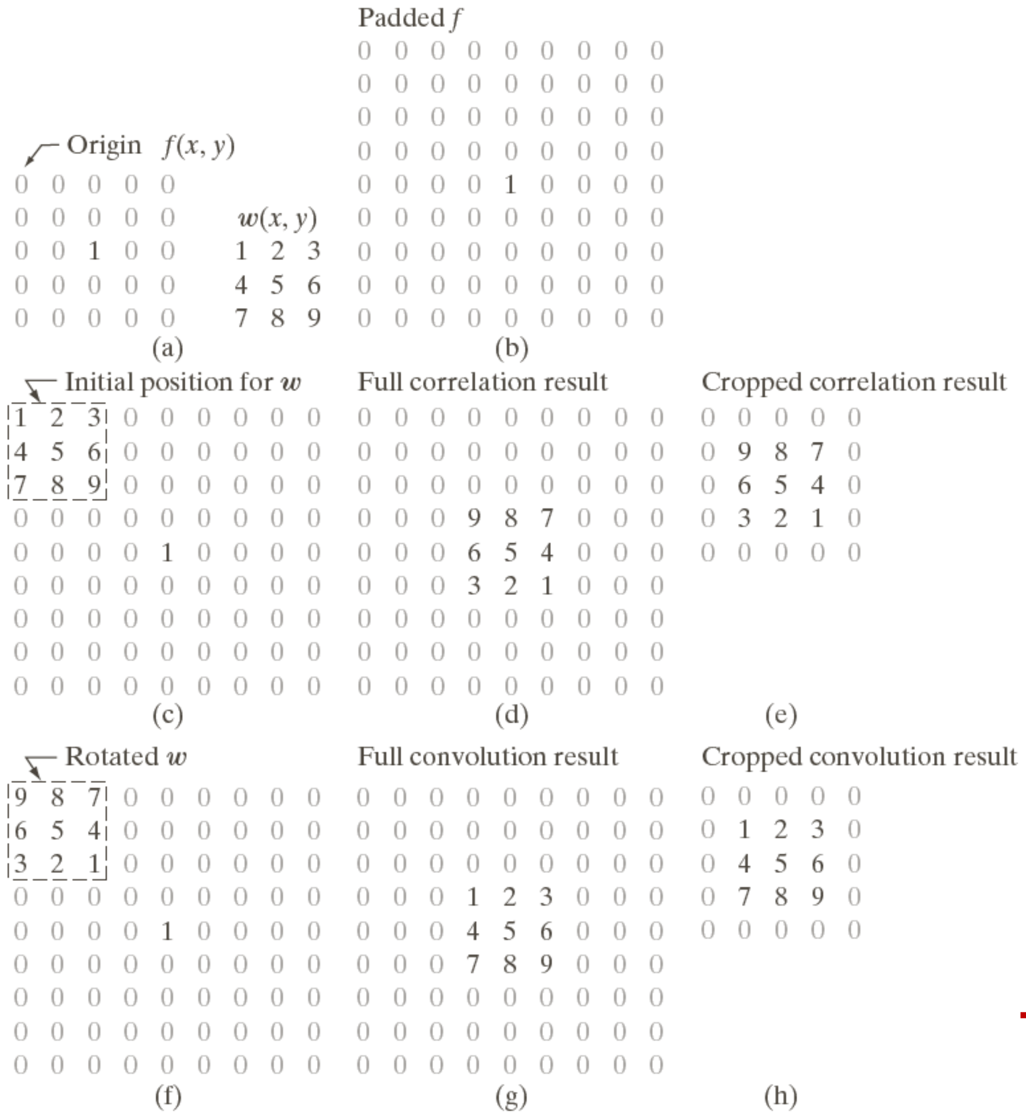
- Convolution

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

- Using correlation and convolution is a matter of preference.
  - Filter for the other one is found by rotation.
  - One important difference: convolution copies the function with a unit impulse.

Spatial Filtering

- Correlation and convolution
- Padding is required in practice for both operations
  - Zeros
  - Linear
  - Reflection
  - Also higher continuity



Spatial Filtering, smoothing

- Smoothing filters for
  - Blurring, e.g removing small details for better feature detection
  - Noise removal
- Lowpass filters compute local averages over the image,
  - Both positive effects and negative effects
    - removing sharp internsity variations
- The mask size may vary
  - Interesting objects become blobby-like for easier detection

Spatial Filtering, smoothing

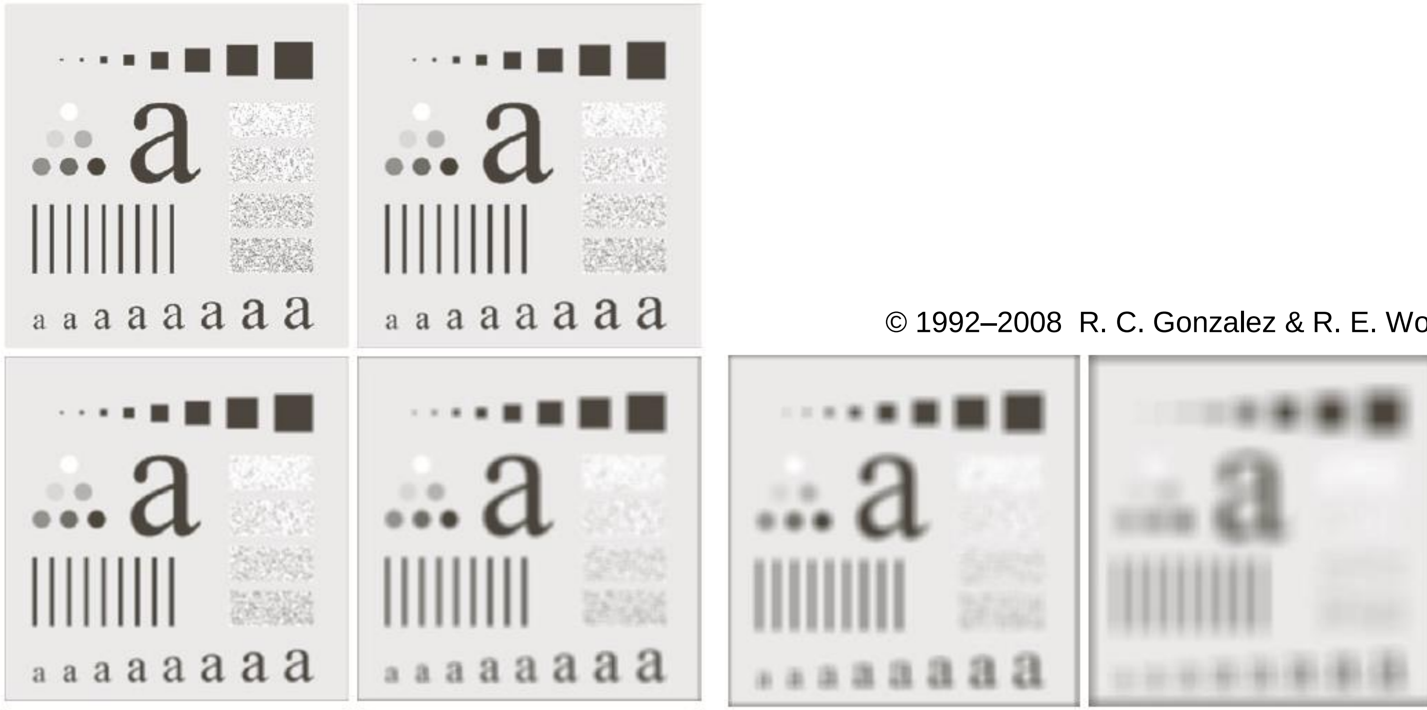


FIGURE 3.33 (a) Original image, of size 500 × 500 pixels. (b)–(f) Results of smoothing with square averaging filter masks of sizes $m = 3, 5, 9, 15$, and 35, respectively. The black squares at the top are of sizes 3, 5, 9, 15, 25, 35, 45, and 55 pixels, respectively; their borders are 25 pixels apart. 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 intensity 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.

Spatial Filtering, smoothing

- Nonlinear filters offer many design options
- Order-statistics filters
  - median, max, min filters,

$$f(x,y) = \underset{(s,t)\in S_{xy}}{median}\{g(s,t)\}$$

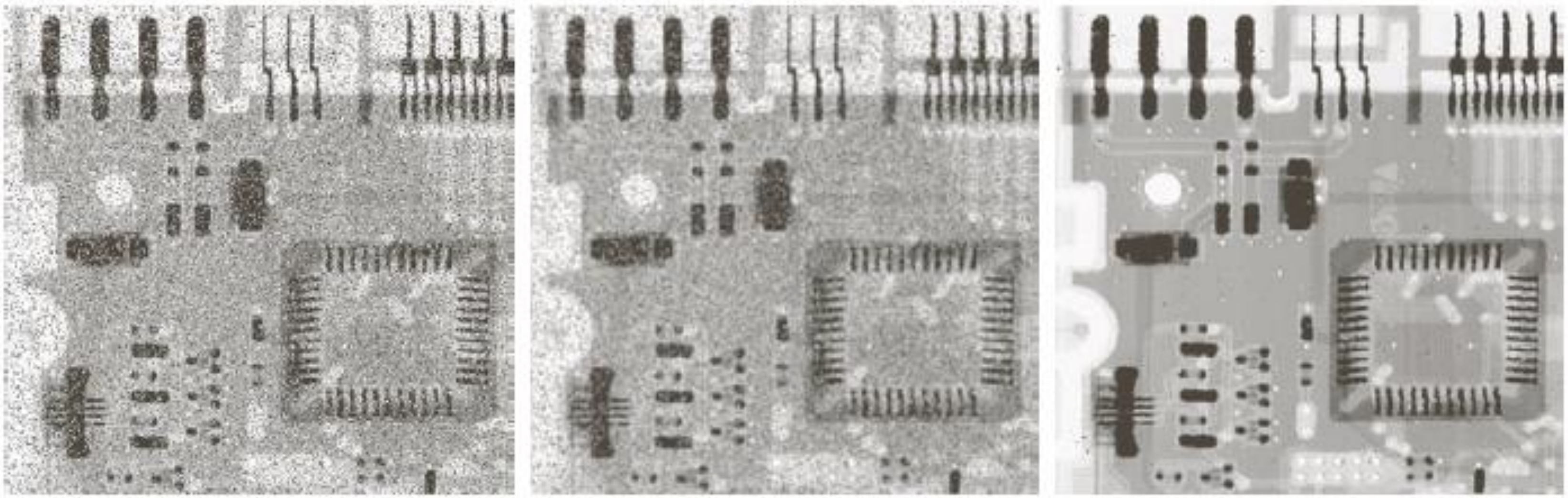
  - Midpoint filter: mean between the min and max
  - Alpha-trimmed mean filter: varying between mean and median

$$f(x,y) = \frac{1}{mn-d} \sum_{(s,t)\in S_{xy}} g_r(s,t)$$

  - d is between 0 and $mn-1$ (mean, median)

Spatial Filtering, smoothing

- Original image with salt-and-pepper noise
- 3x3 mean filter, median filter



a b c

FIGURE 3.35 (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, sharpening

- Sharpening filters
  - The goal is to highlight intensity variations
- Smoothing/averaging is analogous to integration, then sharpening is corresponding to spatial differentiation
- First and second order derivatives for digital images
  - Special interest in smooth areas and intensity discontinuities (ramps, points, lines, edges)

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

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

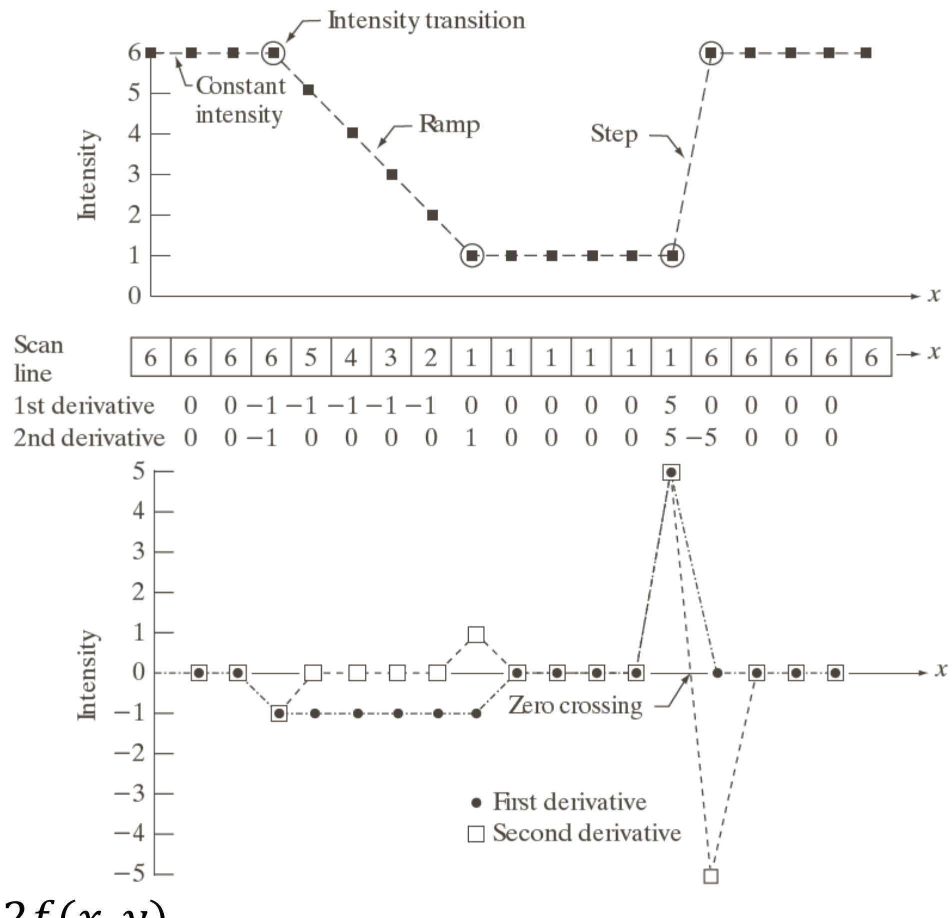
Spatial Filtering, sharpening

- Image line and the corresponding first and second derivatives

- In two dimensions, the Laplacian is

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

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



Spatial Filtering, sharpening

- In two dimensions, the Laplacian is

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

- Filtering outputs dark images with intensity changes highlighted
- A sharpened results after summing

$$g(x,y) = f(x,y) + c[\nabla^2 f(x,y)]$$

0	1	0
1	−4	1
0	1	0

1	1	1
1	−8	1
1	1	1

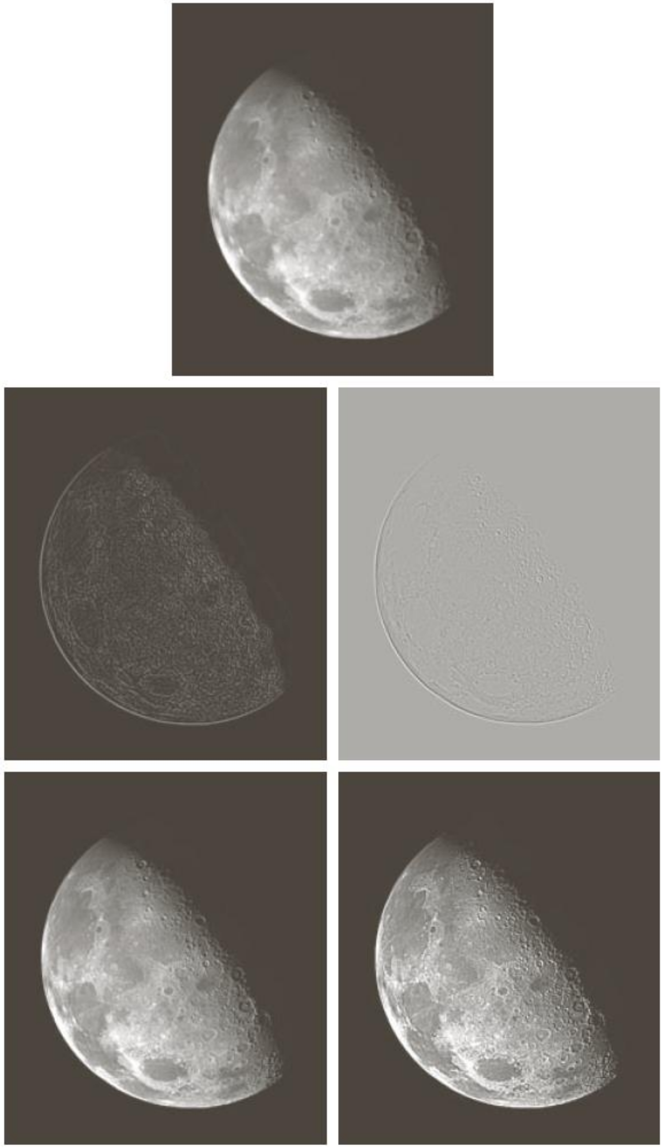
a, b

0	−1	0
−1	4	−1
0	−1	0

−1	−1	−1
−1	8	−1
−1	−1	−1

Spatial Filtering, sharpening

- Original image
- Image after Laplacian filter (both negative and positive values, negative values clipped to 0),
- Image after scaling the results from the Laplacian filter (now from 0 to L-1)
- Sharpened images with filters a) and b)

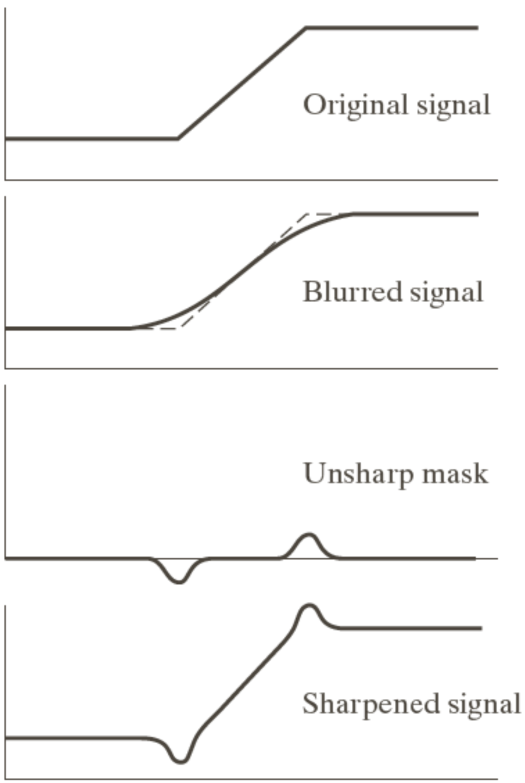


Spatial Filtering, sharpening

- Unsharp masking for ramps is similar to second-order derivative: a blurred image is added to the original

$$g_{mask}(x,y) = f(x,y) - \bar{f}(x,y)$$
$$g(x,y) = f(x,y) + k\, g_{mask}(x,y)$$

- $k \geq 0, k = 1$ normally
- $k < 1,$ de-emphasizing unsharp mask
- $k > 1$, for highboost filtering



Spatial Filtering, sharpening

- Original image, blurred, unsharp mask, $k = 1, k > 1$

Spatial Filtering, Gradient

- Gradient provides information on relative changes in the image intensity
  - Useful in visual inspection for finding defects

$$\nabla f = grad(f) = \begin{bmatrix} g_x \\ g_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix}$$

$$M(x, y) = mag(\nabla f) = \sqrt{g_x^2 + g_y^2}$$

  - $M(x, y)$ is also called the *gradient image* corresponding to the magnitude of the change in the gradient direction (rotation invariant)

Spatial Filtering, Gradient

- In practice $M(x,y) = |g_x| + |g_y|$ is also used
  - Preserves description of intensity changes
  - Isotropic property is lost (in general)
- Roberts operators for pixel z_5

$$g_x = (z_8 - z_5),\ g_y = (z_6 - z_5)$$

$$g_x = (z_9 - z_5),\ g_y = (z_8 - z_6)$$

$$M(x,y) = \sqrt{(z_9 - z_5)^2 + (z_8 - z_6)^2}$$

$$M(x,y) \approx |z_9 - z_5| + |z_8 - z_6|$$

- Mask is of even size -> not symmetric

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

−1	0
0	1

0	−1
1	0

Spatial Filtering, Gradient

- For a 3x3 neigbourhood the approximation for the gradient becomes

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

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

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

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

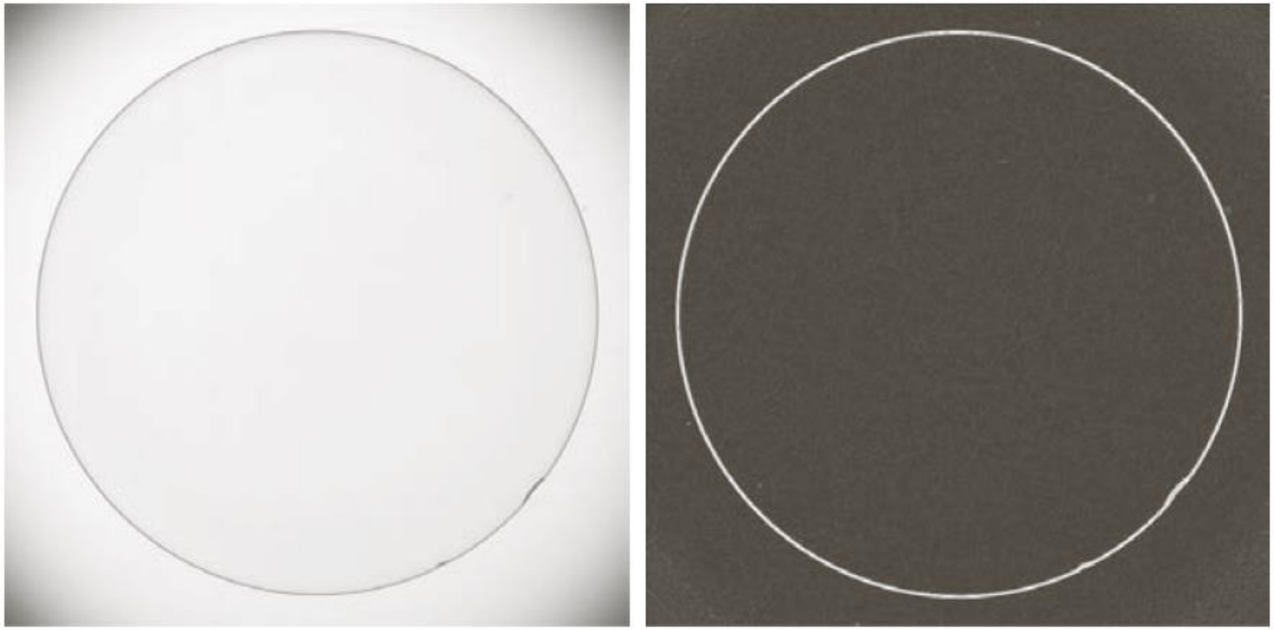
- Sobel operators (with linear components g_x and g_y)
  - Higher weight for the center point
  - For a constant image the response is zero

−1	−2	−1
0	0	0
1	2	1

−1	0	1
−2	0	2
−1	0	1

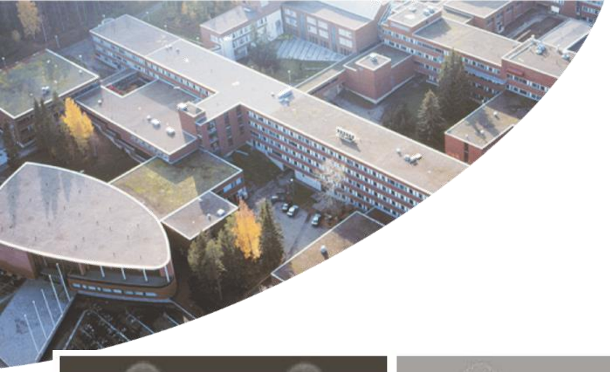
Spatial Filtering, Gradient

- Detecting defects in a lens
  - Preprocessing for automatic inspection
  - Partial removal of gradual background changes

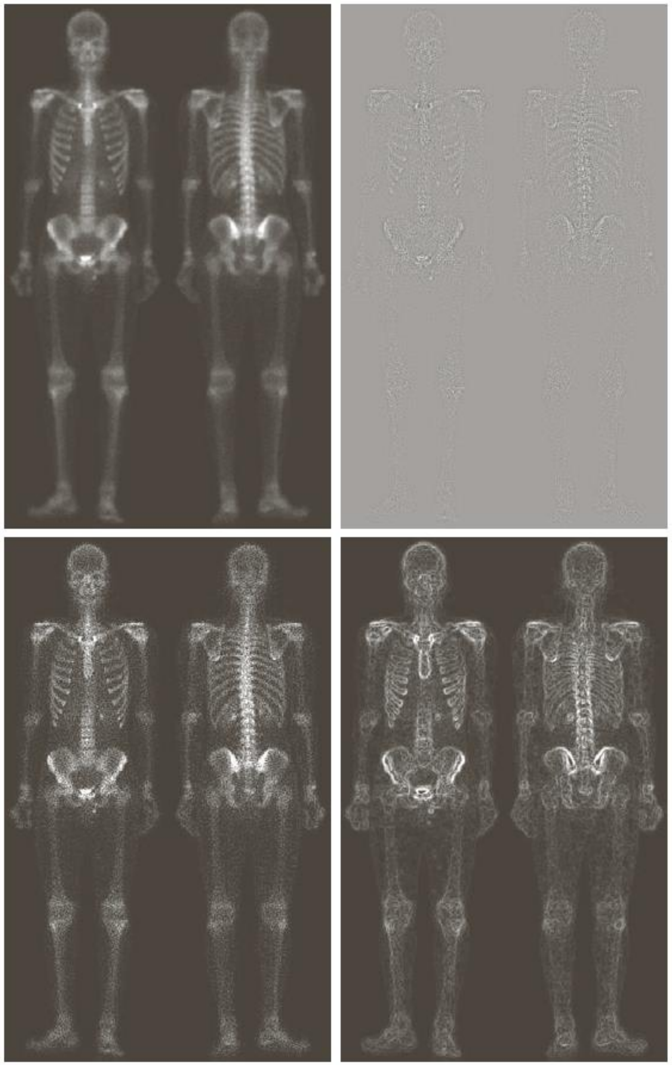


a b

FIGURE 3.42 (a) Optical image of contact lens (note defects on the boundary at 4 and 5 o'clock). (b) Sobel gradient. (Original image courtesy of Pete Sites, Perceptics Corporation.)



Spatial Filtering

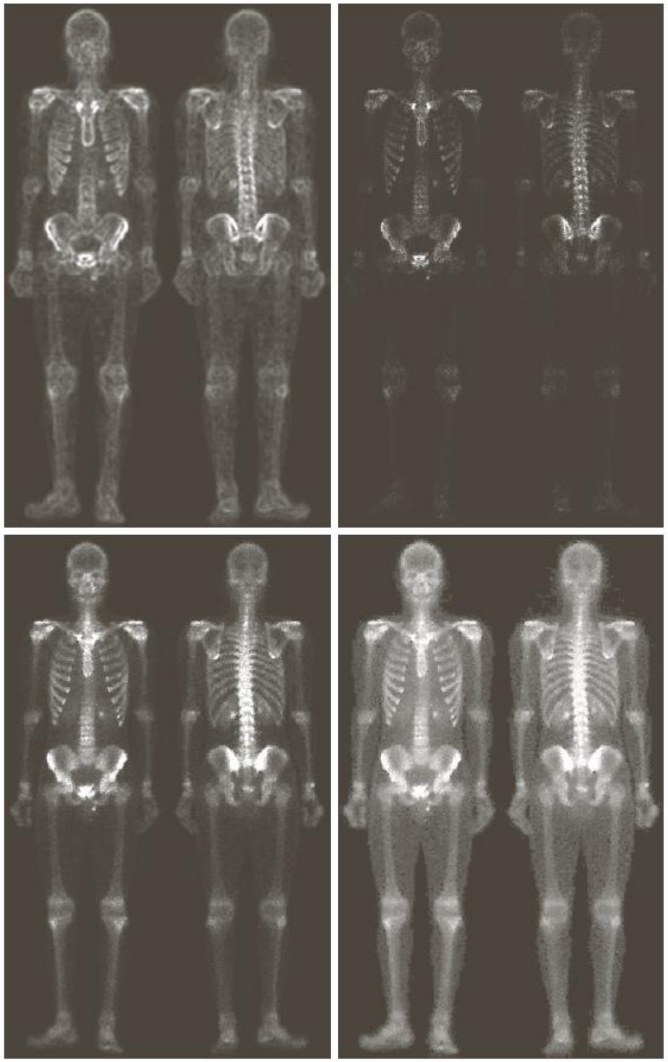


a b
c d

FIGURE 3.43
(a) Image of whole body bone scan. (b) Laplacian of (a). (c) Sharpened image obtained by adding (a) and (b). (d) Sobel gradient of (a).

e f
g h

FIGURE 3.43
(Continued)
(e) Sobel image smoothed with a 5×5 averaging filter. (f) Mask image formed by the product of (c) and (e). (g) Sharpened image obtained by the sum of (a) and (f). (h) Final result obtained by applying a power-law transformation to (g). Compare (g) and (h) with (a). (Original image courtesy of G.E. Medical Systems.)



Summary

- Intensity variations contain important information
- Histograms carry information on the distribution of intensities
- Applications in
  - Image enhancement
  - Compensating nonlinearities in display technologies
  - Support for later analysis