COMP 9517 Computer Vision

Image Processing

Image Analysis

- Manipulation of image data to extract the information necessary for solving an imaging problem
- It is a data reduction process
- Consists of *preprocessing*, *data reduction* and *feature analysis*
  - Preprocessing removes noise, eliminates irrelevant information
  - Data reduction extracts features for the analysis process
  - During feature analysis, the extracted features are examined and evaluated for their use in the application

Image Preprocessing

- Input and output are intensity images
- Aim to improve image, by *suppressing* distortions and *enhancing* image features, so that result is more suitable for a specific application
- Exploit redundancy in image: for example, neighbouring pixels have similar brightness value
- **Spatial** and **frequency** domain techniques

Spatial Domain Techniques

- Operate directly on image pixels:

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

  where
  - $f\,(x,y)$ is the input image
  - $g(x, y)$ is the processed image
  - T is an operator on f, over a neighbour of (x, y)
    - usually square or rectangular neighbour used
    - when T is of size $1x1$, T becomes a gray-level transformation function: $s = T\,(r)$

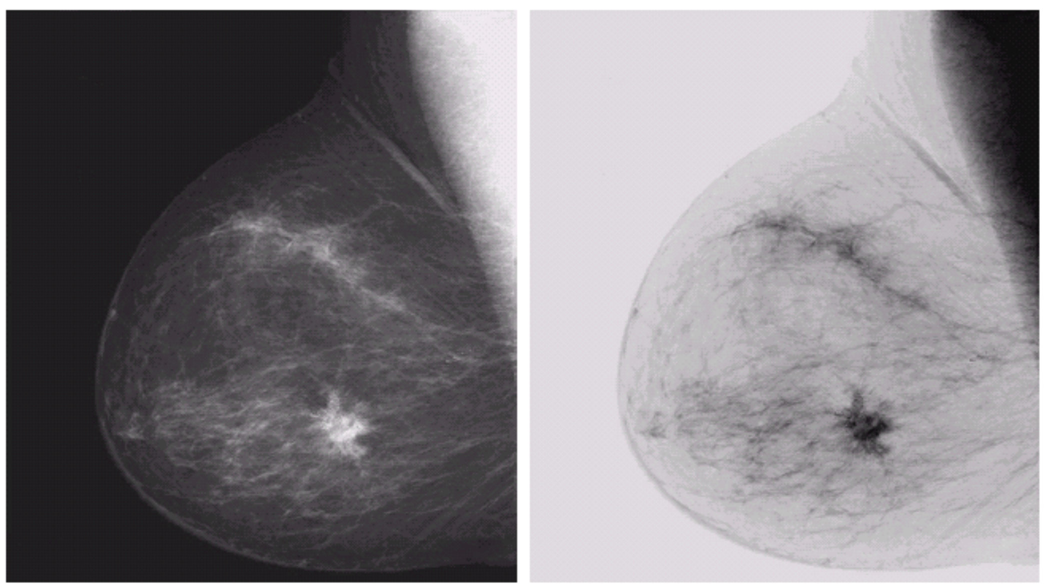
Basic Gray level Transformations

Image Negatives

- For input image with gray levels in range *[0, L-1]* , the negative transformation is

$$s = L - 1 - r, \quad \text{where } r \text{ is the pixel value}$$

- Effects:
    - Produces equivalent of a photo negative
    - Useful for enhancing white or gray detail in dark regions of image, when black areas are dominant



a b

FIGURE 3.4 (a) Original digital mammogram. (b) Negative image obtained using the negative transformation in Eq. (3.2-1). (Courtesy of G.E. Medical Systems.)

Basic Gray level Transformations

Log Transformations

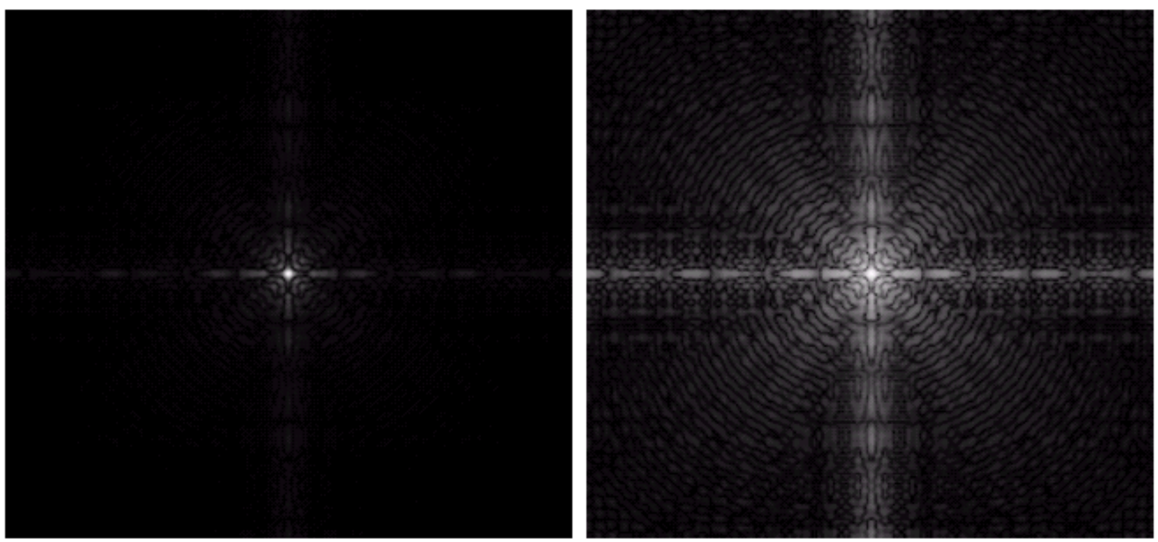
- For input image with gray levels r, the log transformation is

  $s = c \log(1 + r)$, where c is constant, $r >= 0$

- Effects:
  - maps narrow range of low gray-level values into wider range of output values
  - compresses dynamic range of images with large variations in pixel values.

a b

FIGURE 3.5 (a) Fourier spectrum. (b) Result of applying the log transformation given in Eq. (3.2-2) with $c = 1$.

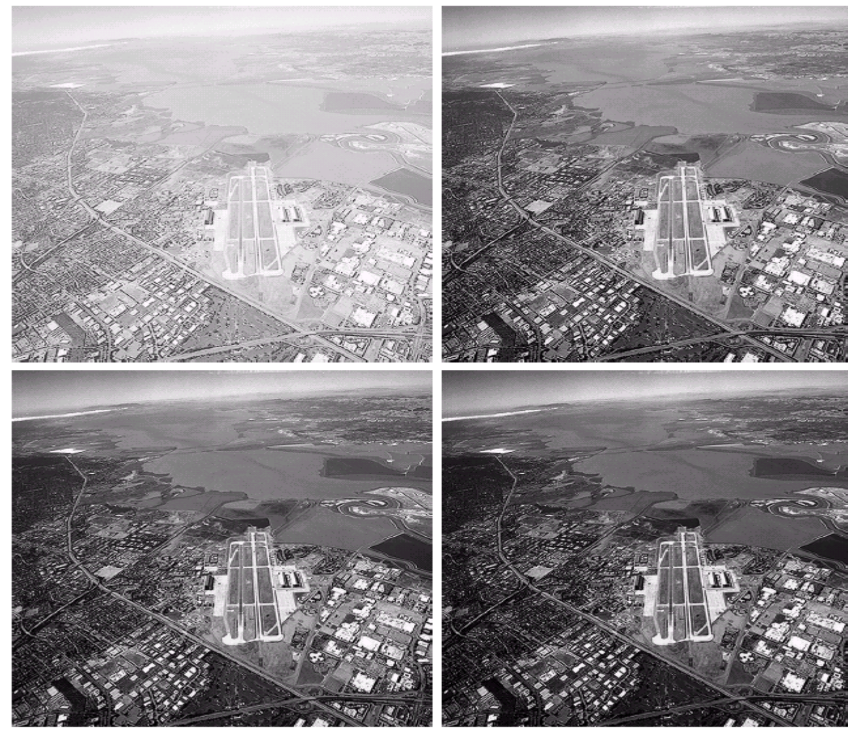


Power-Law Transformations

- Given by $s = c\, r^{\gamma}$ where c, γ are constant
- Similar to log transformation on input-output
- Family of possible transformations by varying γ
- Useful in displaying an image accurately on a computer screen (for example on web sites!) by pre-processing images appropriately before display
- Also useful for general-purpose contrast manipulation

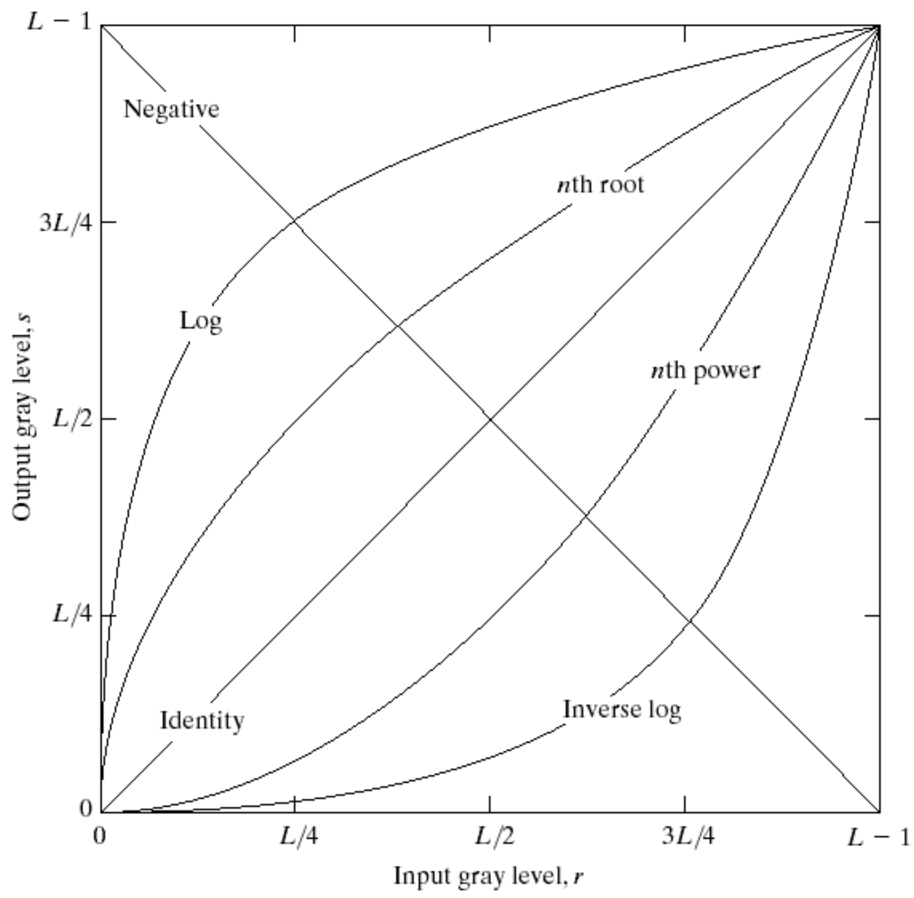
a b
c d

FIGURE 3.9 (a) Aerial image. (b)–(d) Results of applying the transformation in Eq. (3.2-3) with $c = 1$ and $\gamma = 3.0, 4.0,$ and 5.0, respectively. (Original image for this example courtesy of NASA.)



Chapter 3
Image Enhancement in the Spatial Domain

FIGURE 3.3 Some basic gray-level transformation functions used for image enhancement.



Piecewise-linear Transformations

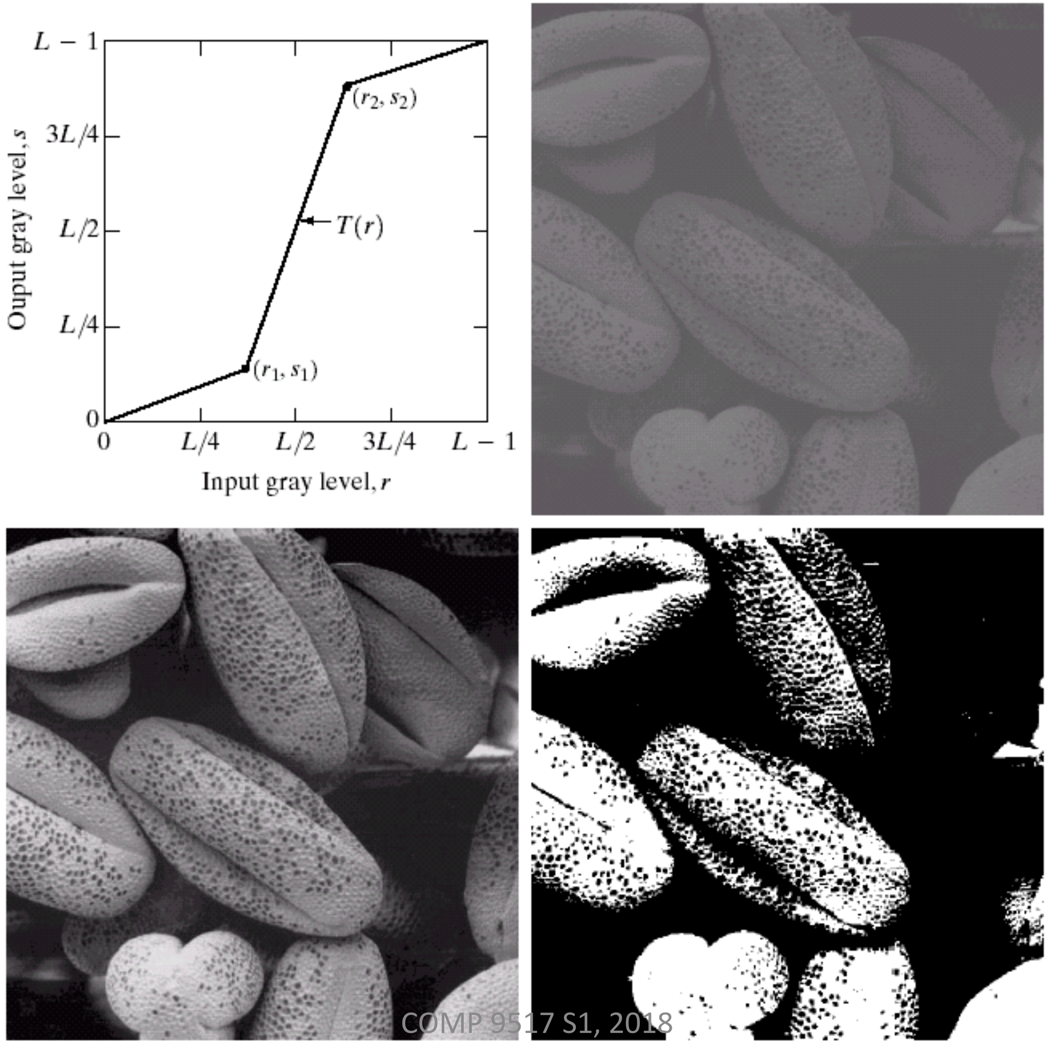
Why?

- Complementary to transformation-based methods
- Forms of piecewise functions can be complex
- However, require more user input

Contrast Stretching

- To increase dynamic range of gray levels in image
  - Interested range being stretched
  - The rest being shrinked

Chapter 3
Image Enhancement in the Spatial Domain



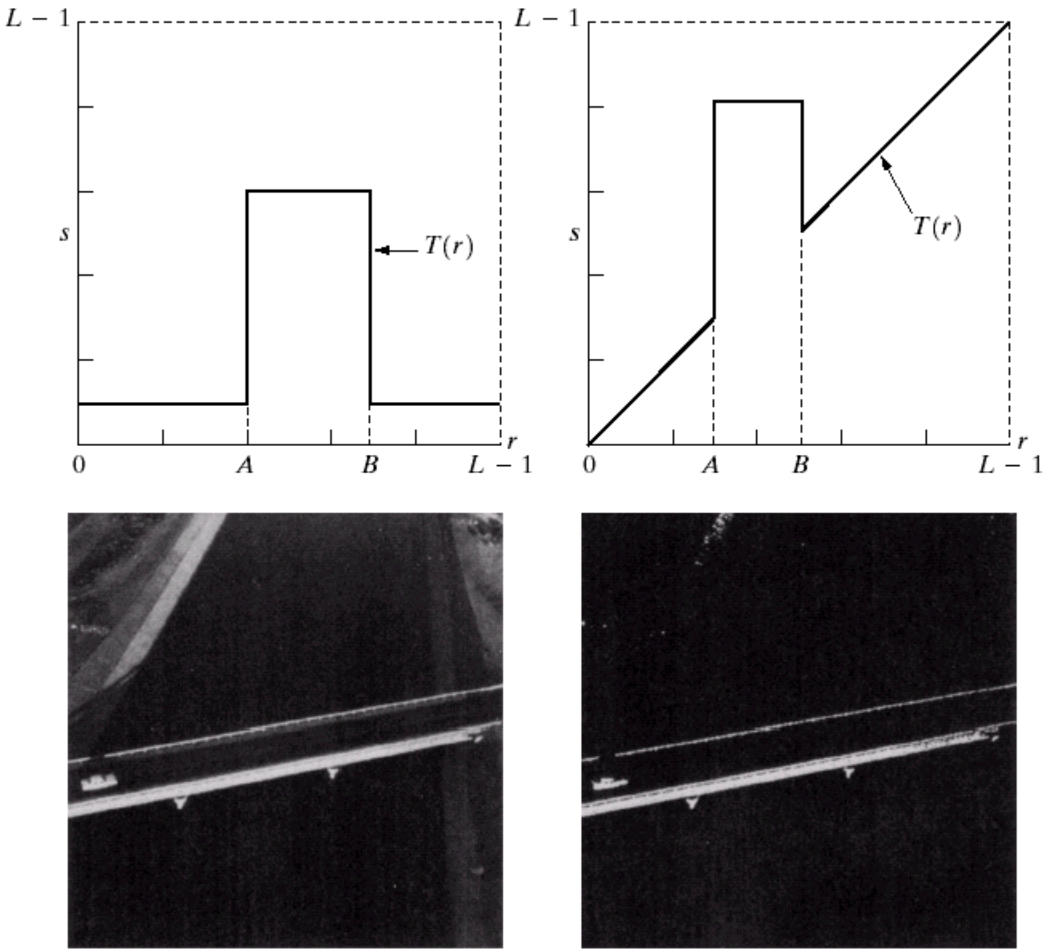
a b
c d

FIGURE 3.10 Contrast stretching. (a) Form of transformation function. (b) A low-contrast image. (c) Result of contrast stretching. (d) Result of thresholding. (Original image courtesy of Dr. Roger Heady, Research School of Biological Sciences, Australian National University, Canberra, Australia.)

Gray-level Slicing

- Highlighting of specific range of gray levels
  - Display high value for all gray levels in range of interest, and low value for all others - produces binary image
  - Brighten the desired range of gray levels, while preserving background and other gray-scale tones of image

Chapter 3
Image Enhancement in the Spatial Domain



a b
c d

FIGURE 3.11 (a) This transformation highlights range [A, B] of gray levels and reduces all others to a constant level. (b) This transformation highlights range [A, B] but preserves all other levels. (c) An image. (d) Result of using the transformation in (a).

Bit-plane Slicing

- Highlights contribution made to total image appearance by specific bits
- Eg, for an 8-bit image, there are 8 1-bit planes
- Useful in compression

Chapter 3
Image Enhancement in the Spatial Domain

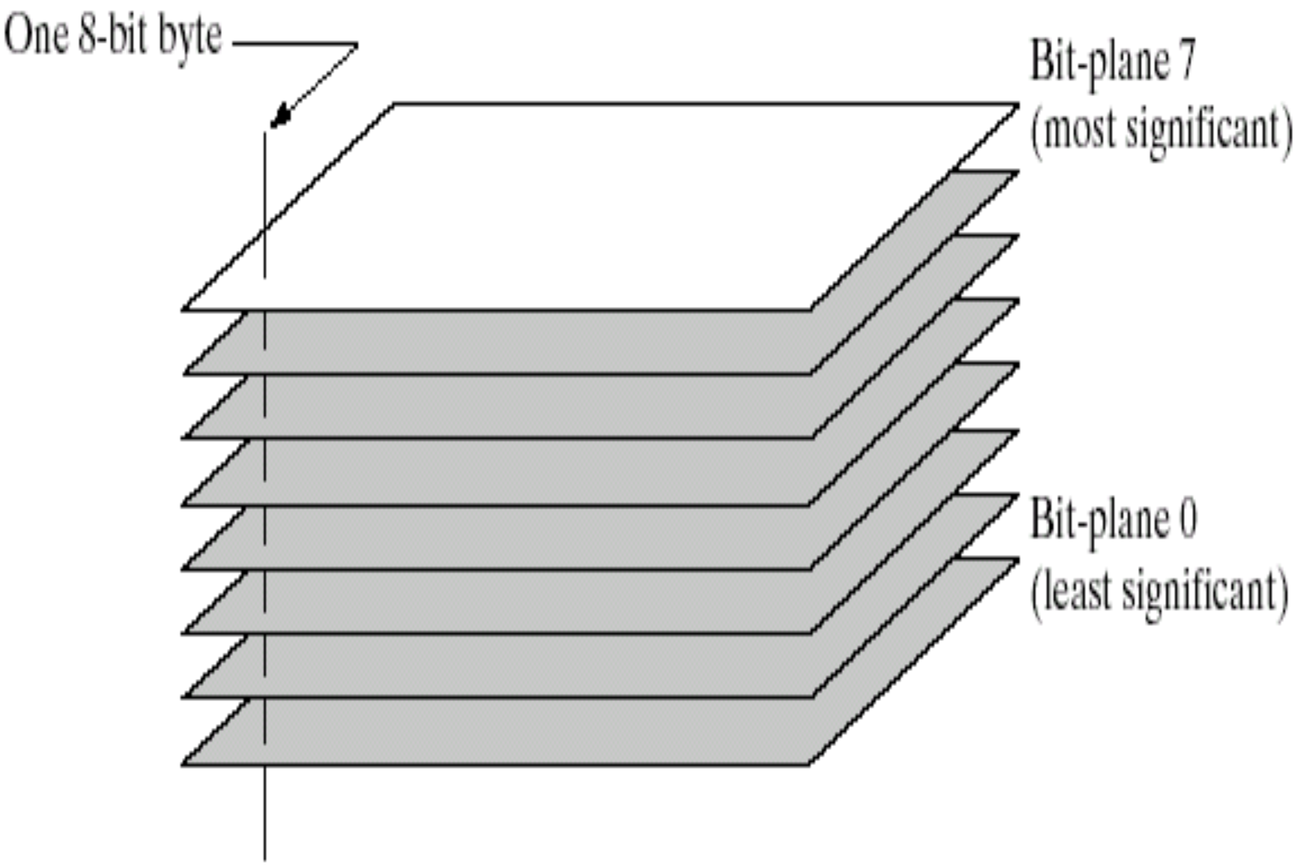


FIGURE 3.12 Bit-plane representation of an 8-bit image.

Chapter 3
Image Enhancement in the Spatial Domain

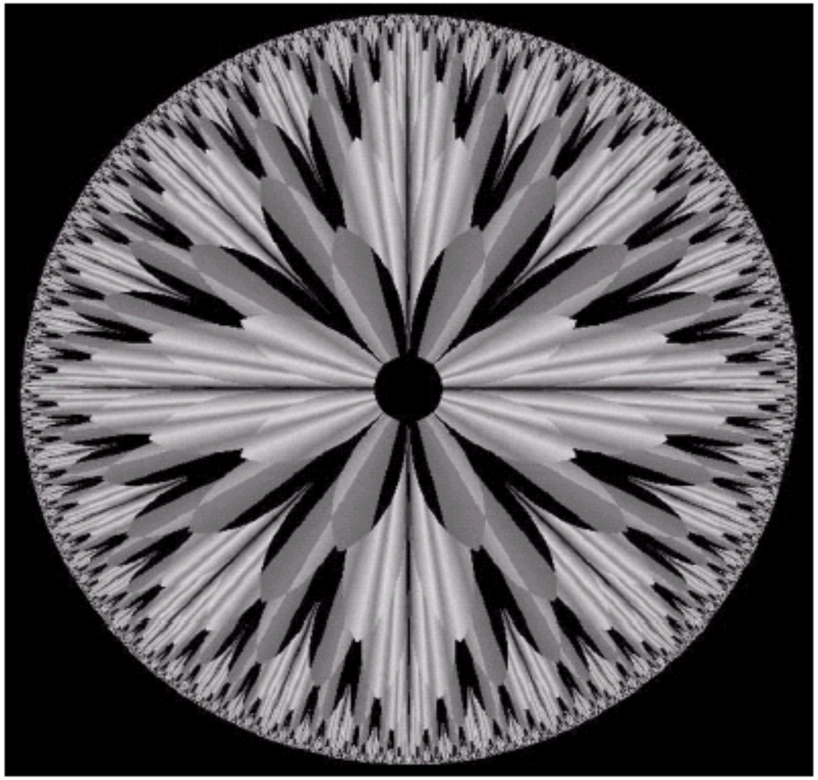


FIGURE 3.13 An 8-bit fractal image. (A fractal is an image generated from mathematical expressions). (Courtesy of Ms. Melissa D. Binde, Swarthmore College, Swarthmore, PA.)

Chapter 3
Image Enhancement in the Spatial Domain

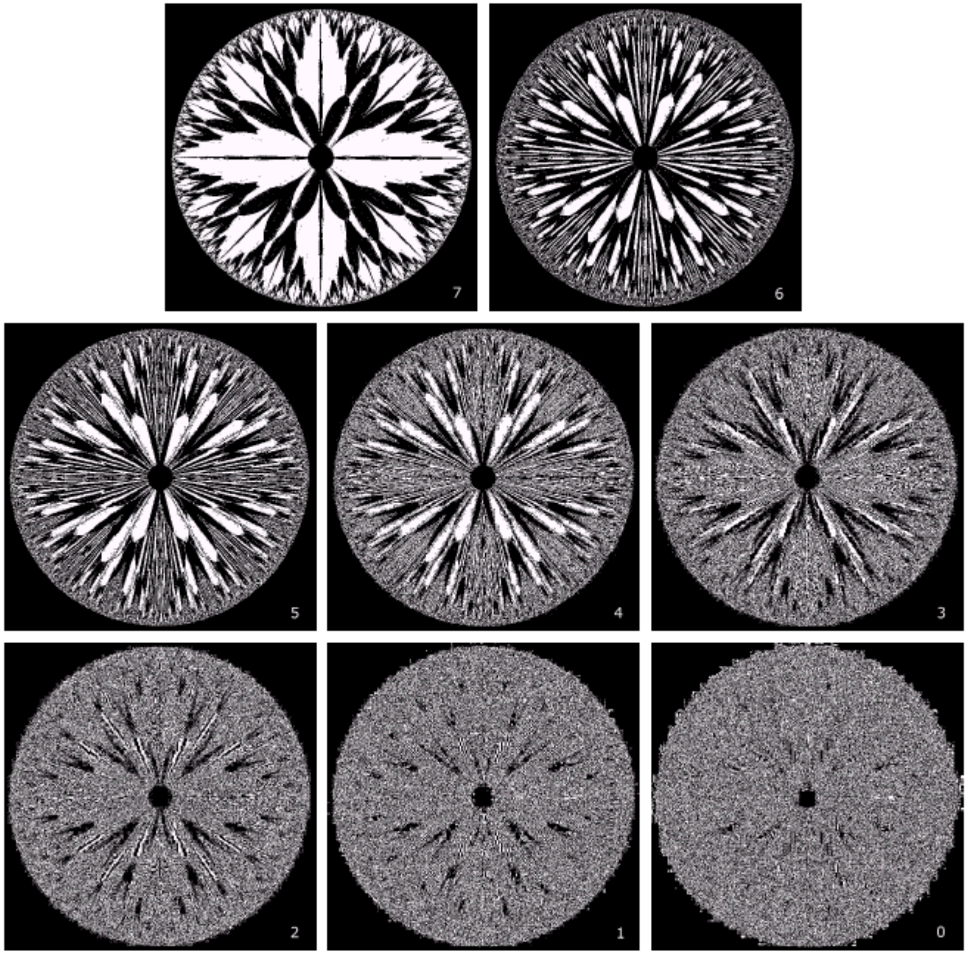
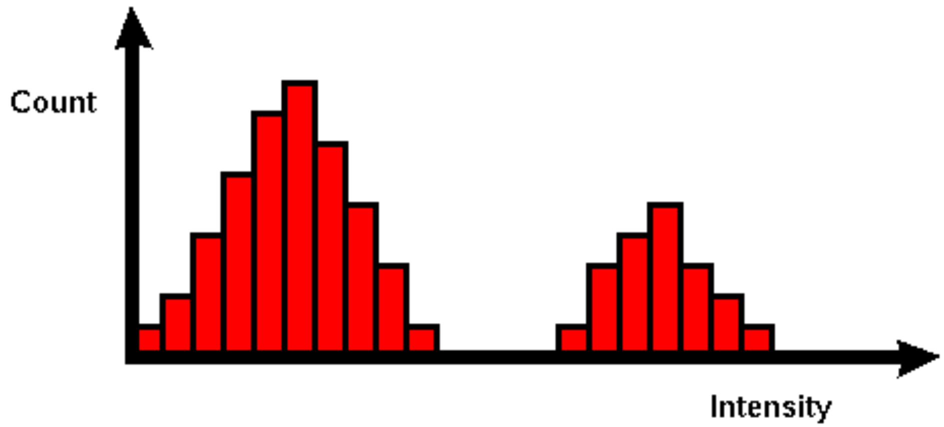


FIGURE 3.14 The eight bit planes of the image in Fig. 3.13. The number at the bottom, right of each image identifies the bit plane.

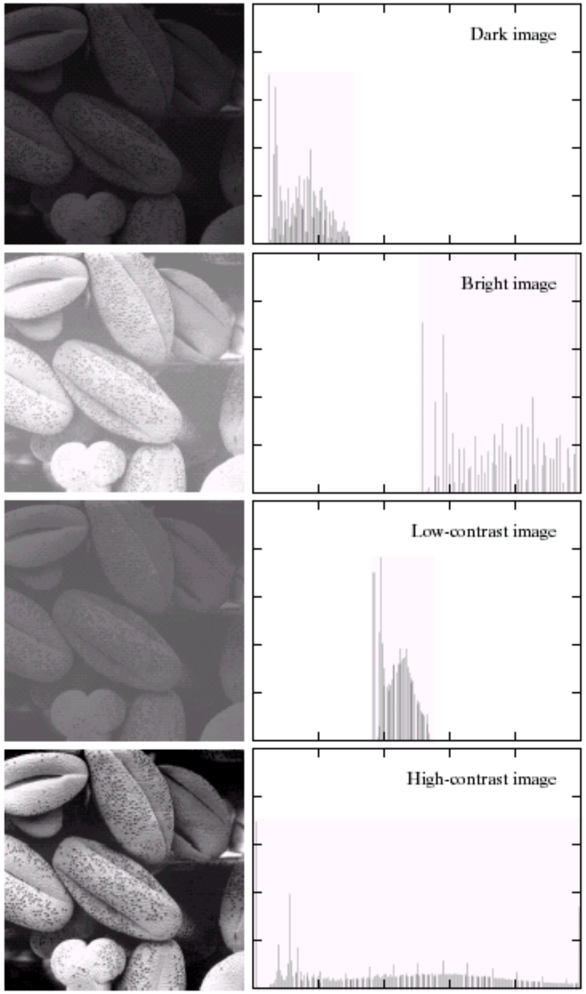
Histogram Processing

Histogram

- plots the number of pixels for each level value



https://homepages.inf.ed.ac.uk/rbf/HIPR2/histb.gif



Histogram Processing

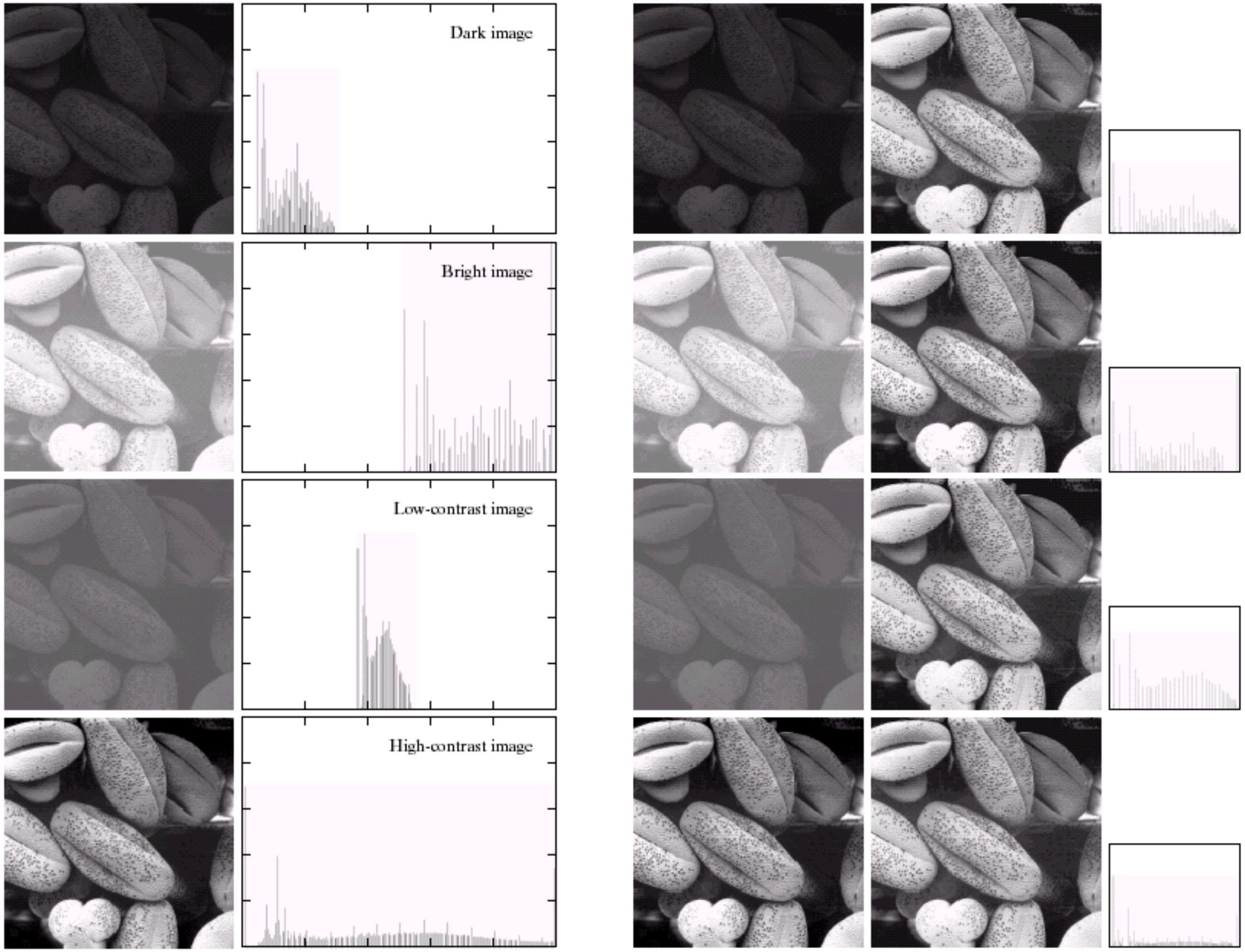
Histogram Equalization

- To get an image with equally distributed brightness levels over the whole brightness scale

$$s = T(r) = floor(T-1)\sum_{n=0}^{f_{i,j}} p_n$$

, where p_n = number of pixels with intensity n / total number of pixels

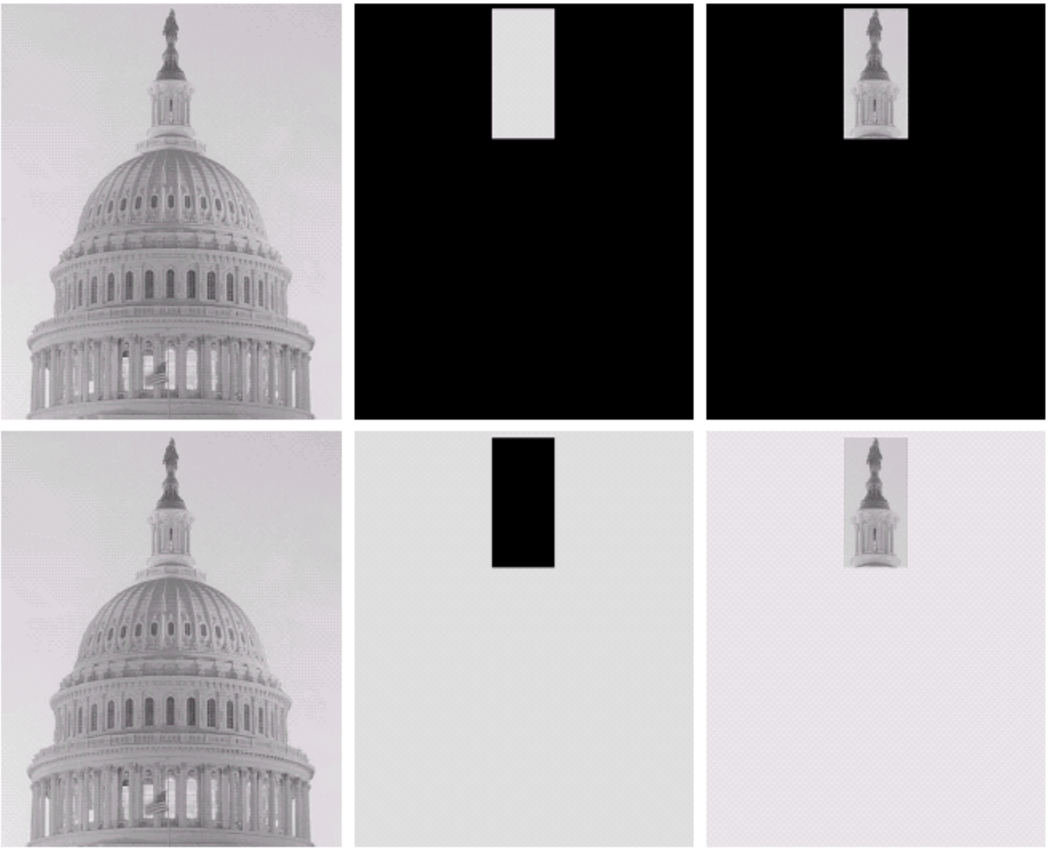
Chapter 3
Image Enhancement in the Spatial Domain



Arithmetic/Logic Operations

- On pixel-by-pixel basis between 2 or more images
- AND and OR operations are used for *masking*-selecting subimages as RoI
- Subtraction and addition are the most useful arithmetic operations

Chapter 3
Image Enhancement in the Spatial Domain



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).

Image Averaging

- Noisy image $g\ (x, y)$ formed by adding noise $n\ (x, y)$ to uncorrupted image $f\ (x, y)$:

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

- Assume that at each (x, y), the noise is uncorrelated and has zero average value.

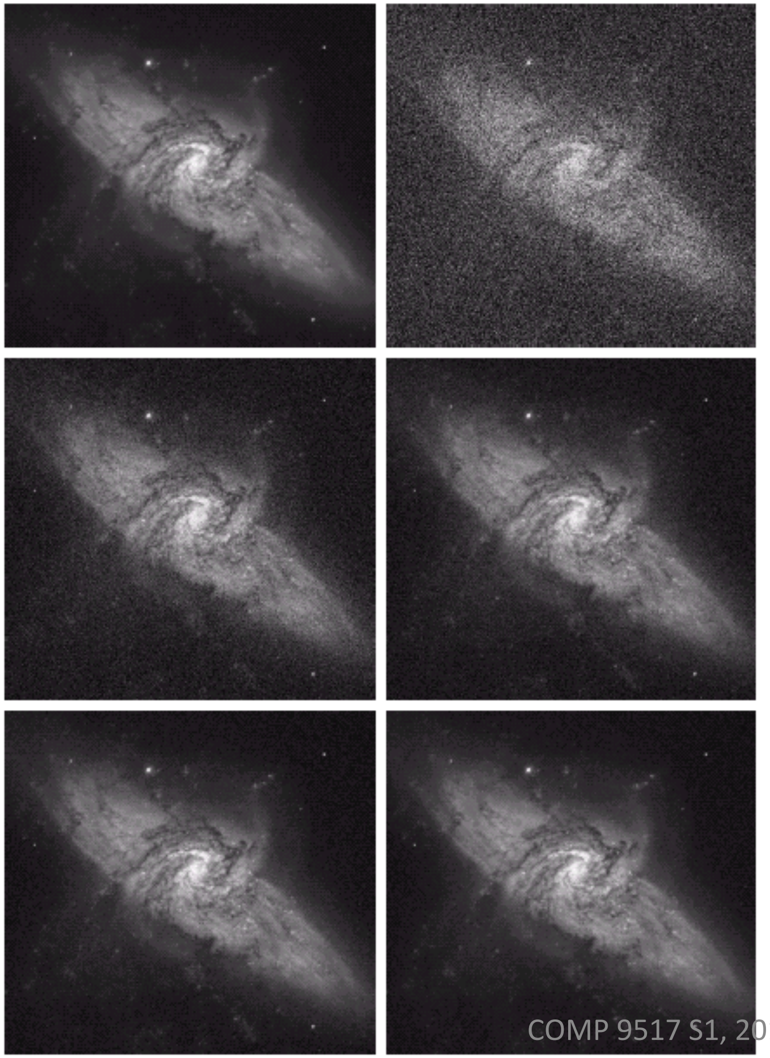
- **Aim:** To obtain smoothed result by adding a set of noisy images $g_i\ (x, y)$, $i = 1, 2, ..., K$

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

- As K increases, the variability of the pixel values decreases

- Assumes that images are spatially registered

Chapter 3
Image Enhancement in the Spatial Domain



a b
c d
e f

FIGURE 3.30 (a) Image of Galaxy Pair NGC 3314. (b) Image corrupted by additive Gaussian noise with zero mean and a standard deviation of 64 gray levels. (c)–(f) Results of averaging $K = 8, 16, 64$, and 128 noisy images. (Original image courtesy of NASA.)

Spatial Filtering

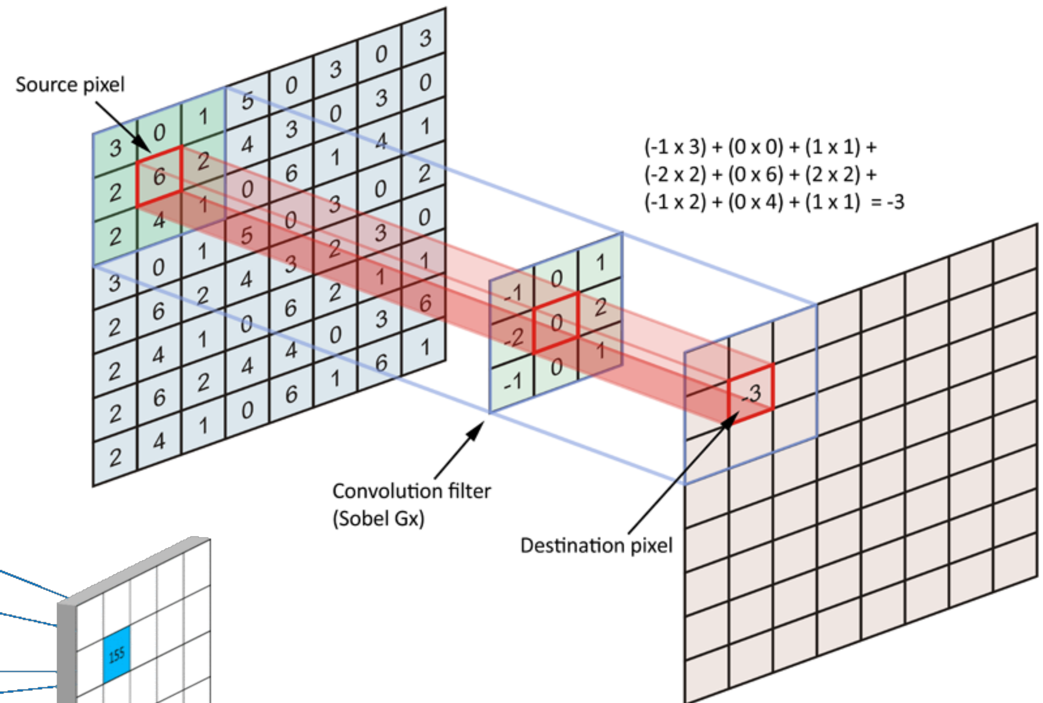
- These methods use a small *neighbourhood* of a pixel in the input image to produce a new brightness value for that pixel
- Also called *filtering* techniques
- Neighbourhood of *(x, y)* is usually a square or rectangular subimage centred at *(x, y)*- called *filter/mask/kernel/template/window*
- A *linear transformation* calculates a value in the output image *g (i, j)* as a linear combination of brightnesses in a local neighbourhood of the pixel in the input image *f (i, j)*, weighted by coefficients *h*:

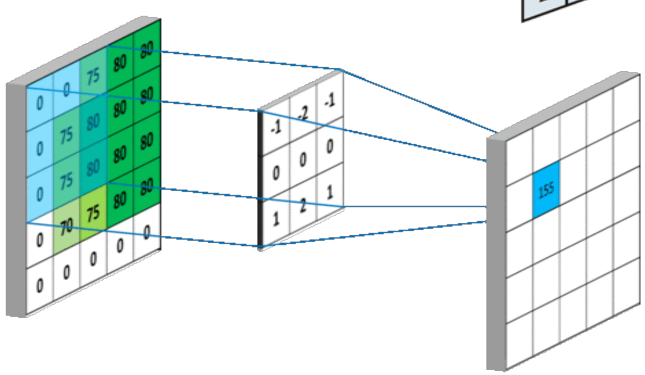
$$g(i, j) = \sum \sum h(i-m, j-n) f(m,n)$$

- This is a *discrete convolution* with a convolution mask h

Spatial Filtering

Convolution





Smoothing Spatial Filters

- **Aim:** To suppress noise, other small fluctuations in image - may be result of sampling, quantisation, transmission, environment disturbances during acquisition
- Uses redundancy in the image data
- May blur sharp edges, so care is needed

Neighbourhood Averaging

$$g(x, y) = \frac{1}{P} \sum_{(n,m) \in S} f(n,m)$$

- Replace intensity at pixel (x, y) with the **average** of the intensities in a neighbourhood of (x, y).
- We can also use a **weighted average**, giving more importance to some pixels over others in the neighbourhood- reduces blurring
- Neighbourhood averaging (also called **mean filter**) blurs edges.

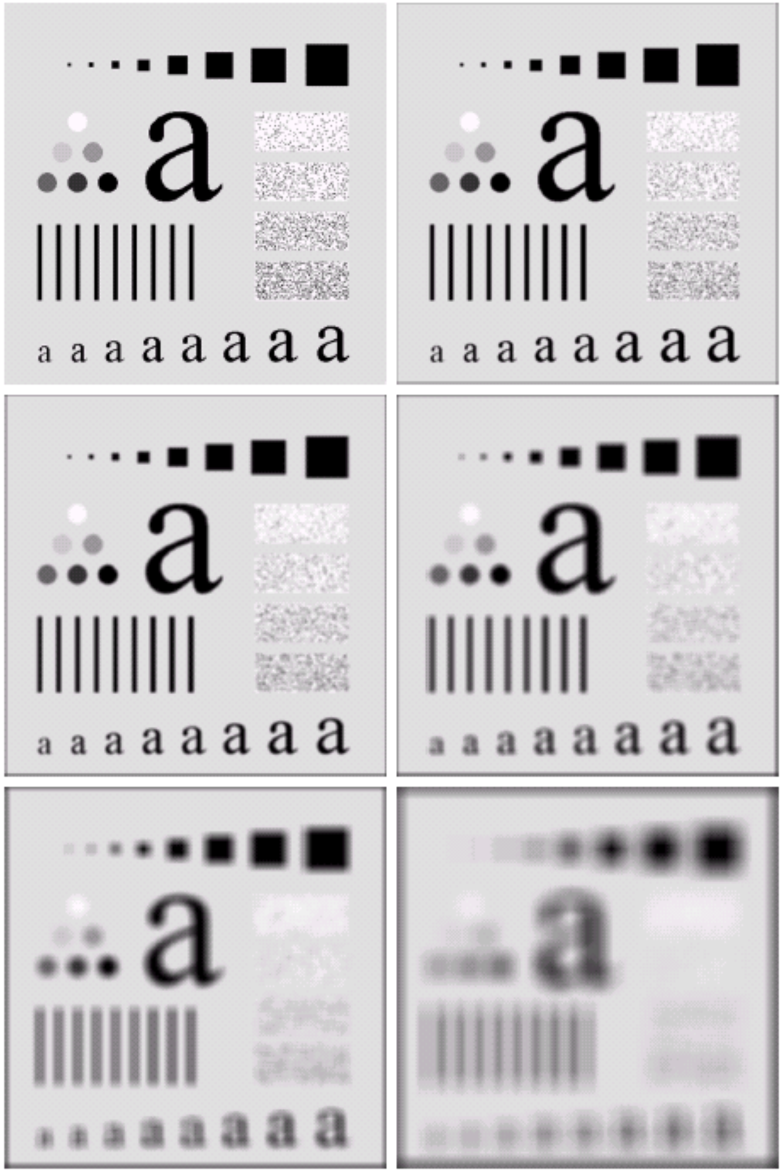
Chapter 3
Image Enhancement in the Spatial Domain

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

a b

FIGURE 3.34 Two 3×3 smoothing (averaging) filter masks. The constant multiplier in front of each mask is equal to the sum of the values of its coefficients, as is required to compute an average.

Chapter 3
Image Enhancement in the Spatial Domain



a b
c d
e f

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.

Order-Statistics Filters

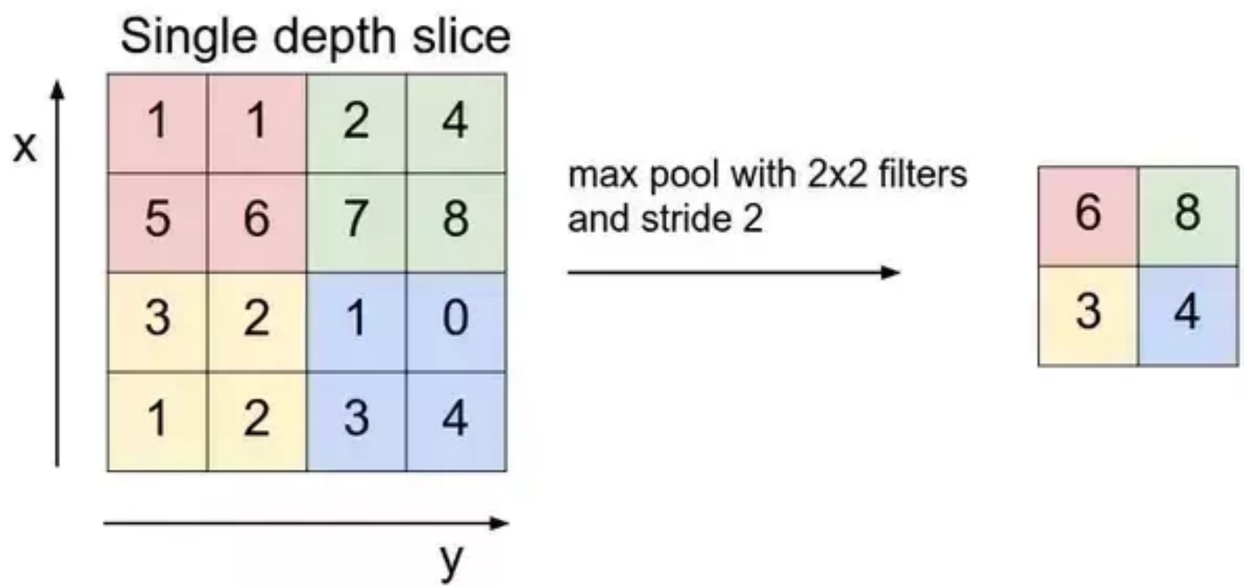
- **Nonlinear spatial filters-** response based on ordering the pixels in the neighbourhood, and replacing centre value with the ranking results.

Median Filter

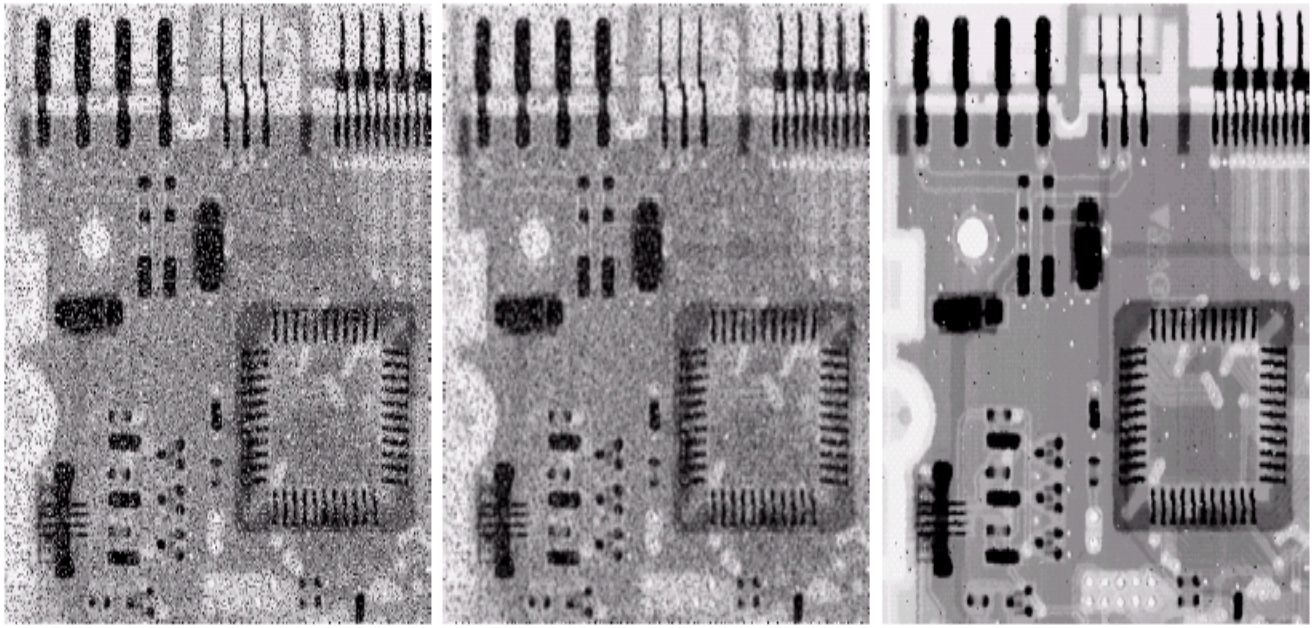
- In median filtering, intensity of each pixel is replaced by the *median* of the intensities in a neighbourhood of the pixel
- Median M of a set of values is the middle value such that half the values in the set are less than M and the other half greater than M
- Median filtering forces points with distinct intensities to be more like their neighbours, thus eliminating isolated intensity spikes
- Also, isolated pixel clusters (light or dark), whose area is <= n^2/2, are eliminated by nxn median filter
- Good for impulse noise (salt-and-pepper noise)
- Other examples are max and min filters

Pooling

- Max/average/median pooling
  - Provides translation invariance
  - Reduces computations
  - Popular in deep convolutional neural networks (deep learning)



Chapter 3
Image Enhancement in the Spatial Domain



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 Spatial Filters

Edge Detection

- Goal is to highlight fine detail, or enhance detail that has been blurred
- Spatial differentiation is the tool - strength of response of derivative operator is proportional to degree of discontinuity of the image at the point where operator is applied
- Image differentiation enhances edges, and deemphasizes slowly varying gray-level values

Derivative definitions

For 1-D function f(x), the first order derivative is approximated as:

$df/dx = f(x+1) - f(x)$

The second-order derivative is approximated as:

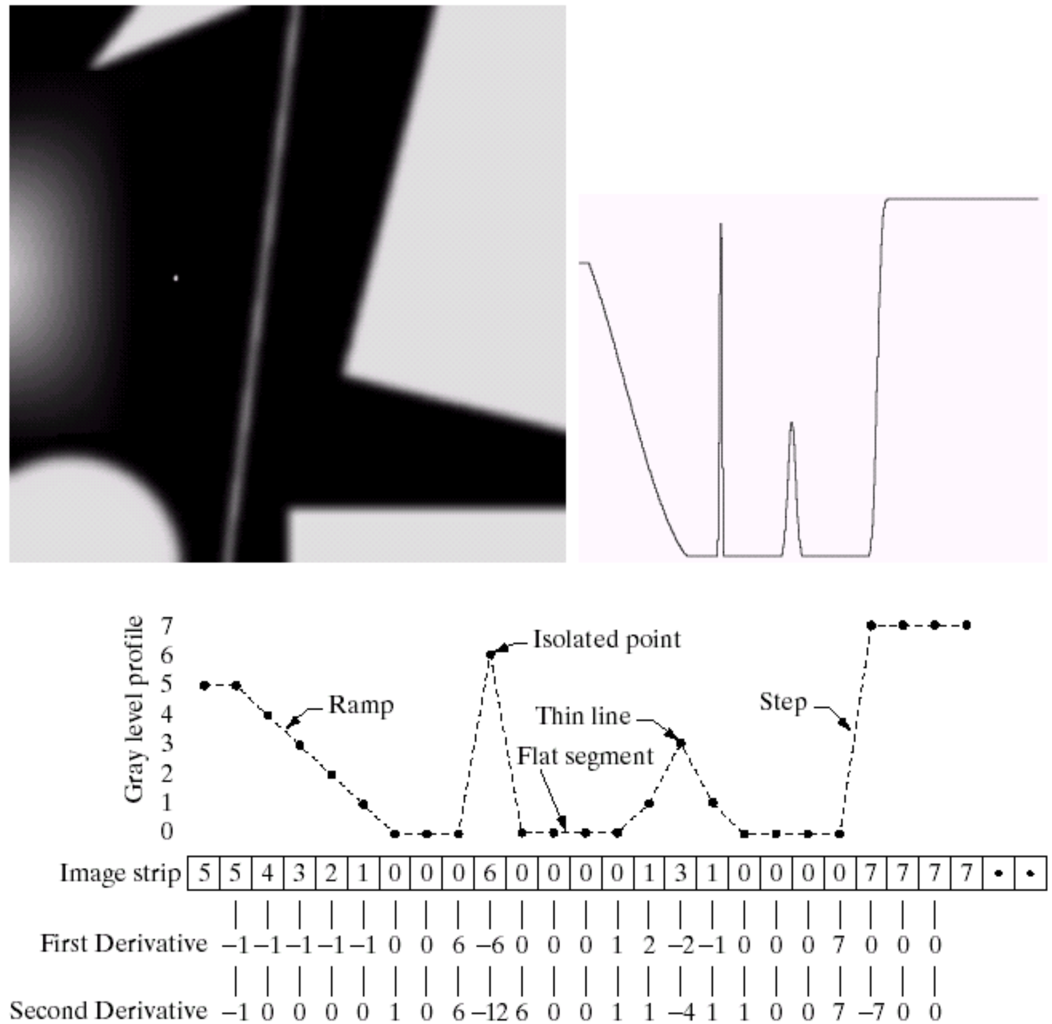
$d^2f/dx^2 = f(x+1) + f(x-1) - 2f(x)$

These are partial derivatives, so extension to 2-D is easy

Chapter 3
Image Enhancement in the Spatial Domain

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).



Basic idea

- Horizontal scan of the image
- Edge modelled as a ramp- to represent blurring due to sampling
- First derivative is
  - Non-zero along ramp
  - zero in regions of constant intensity
  - constant during an intensity transition
- Second derivative is
  - Nonzero at onset and end of ramp
  - Stronger response at isolated noise point
  - zero everywhere except at onset and termination of intensity transition
- Thus, magnitude of first derivative can be used to detect the presence of an edge, and sign of second derivative to determine whether a pixel lies on dark or light side of an edge.

Derivative Summary

- First-order derivatives produce thicker edges
- Second-order derivatives produce stronger response to fine detail (thin lines, isolated points)
- First-order derivatives have stronger response to gray-level step
- Second-order derivatives produce double response at step changes in gray level

Gradient Operator

First-order derivatives implemented using gradient operator

G_x and G_y may be obtained by using masks

The magnitude of the gradient vector is

$$G[f(x, y)] = \sqrt{G_x^2 + G_y^2}$$

This is commonly approximated by:

$$G[f(x, y)] = |G_x| + |G_y|$$

Now use numerical techniques to compute these give rise to different masks, e.g.

Roberts' 2x2 cross-gradient operators, Sobel's 3x3 masks

Chapter 3
Image Enhancement in the Spatial Domain

a
b c
d e

FIGURE 3.44 A 3 × 3 region of an image (the z's are gray-level values) and masks used to compute the gradient at point labeled z_5. All masks coefficients sum to zero, as expected of a derivative operator.

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

−1	−2	−1
0	0	0
1	2	1

−1	0	1
−2	0	2
−1	0	1

The Laplacian

Second order derivatives based on the Laplacian. For a function *f (x, y),* the Laplacian is defined by

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

This is a linear operator, as all derivative operators are.

In discrete form:

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

and similarly in *y* direction.

Summing them gives us

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

Laplacian ctd

- There are other forms of the Laplacian- can include diagonal directions, for example.
- Laplacian highlights grey-level discontinuities and produces dark featureless backgrounds.
- The background can be recovered by adding or subtracting the Laplacian image to the original image.

Chapter 3
Image Enhancement in the Spatial Domain

0	1	0
1	−4	1
0	1	0

1	1	1
1	−8	1
1	1	1

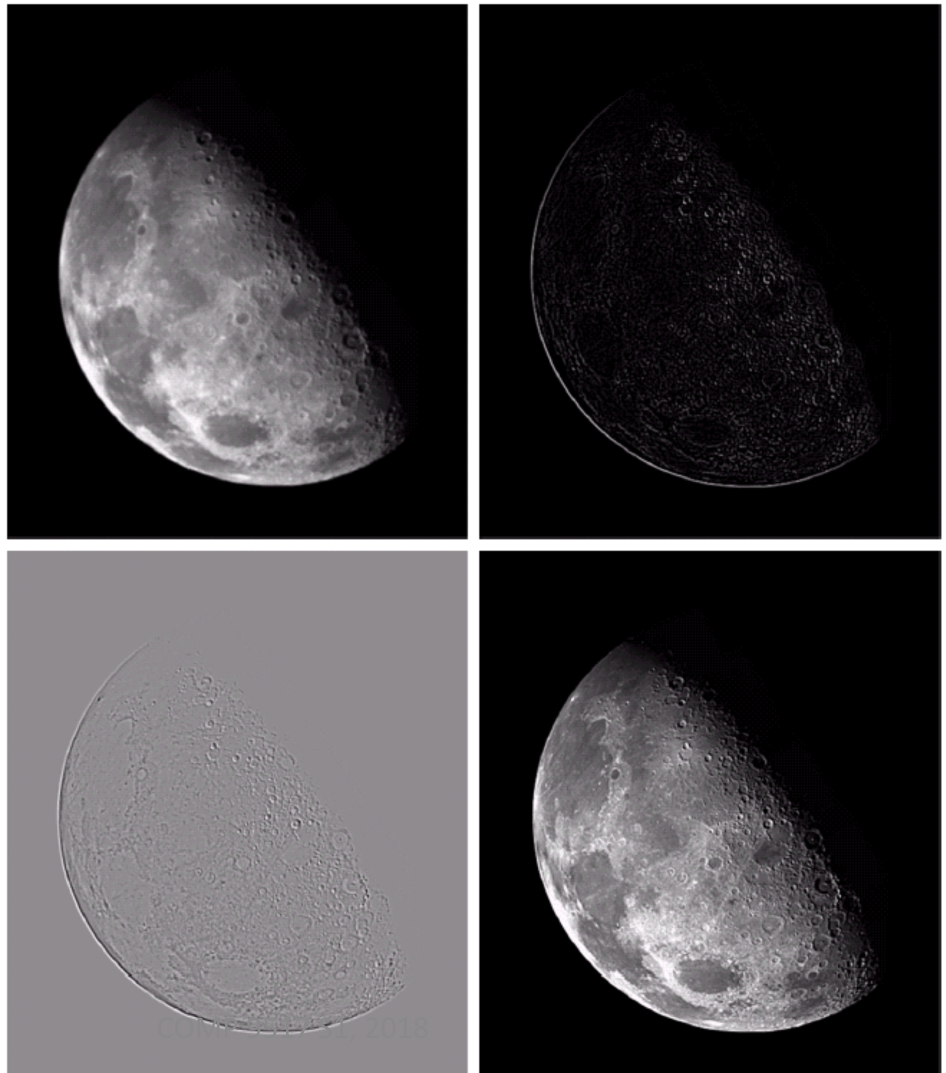
0	−1	0
−1	4	−1
0	−1	0

−1	−1	−1
−1	8	−1
−1	−1	−1

a b
c d

FIGURE 3.39 (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.

Chapter 3
Image Enhancement in the Spatial Domain



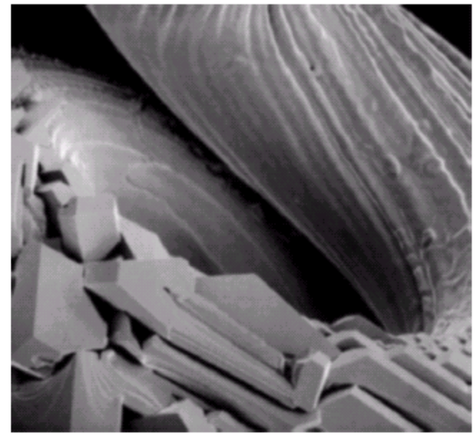
a b
c d

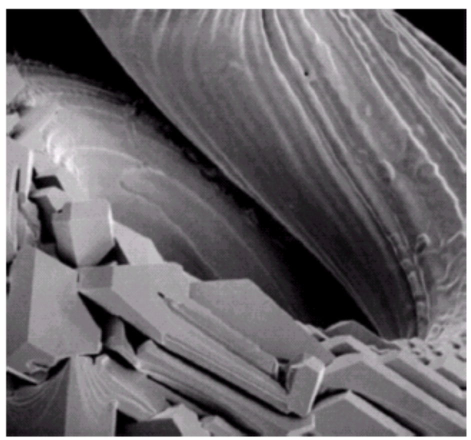
FIGURE 3.40
(a) Image of the North Pole of the moon.
(b) Laplacian-filtered image.
(c) Laplacian image scaled for display purposes.
(d) Image enhanced by using Eq. (3.7-5).
(Original image courtesy of NASA.)

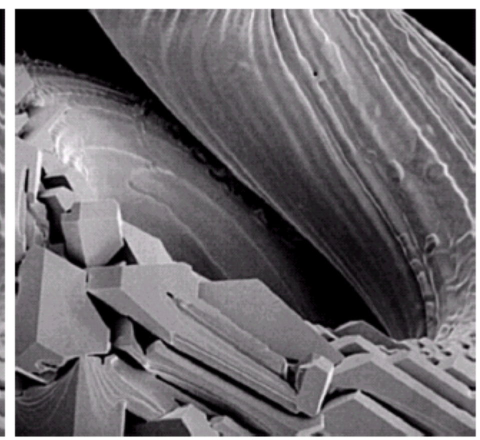
Chapter 3
Image Enhancement in the Spatial Domain

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.)

Chapter 3 Image Enhancement in the Spatial Domain

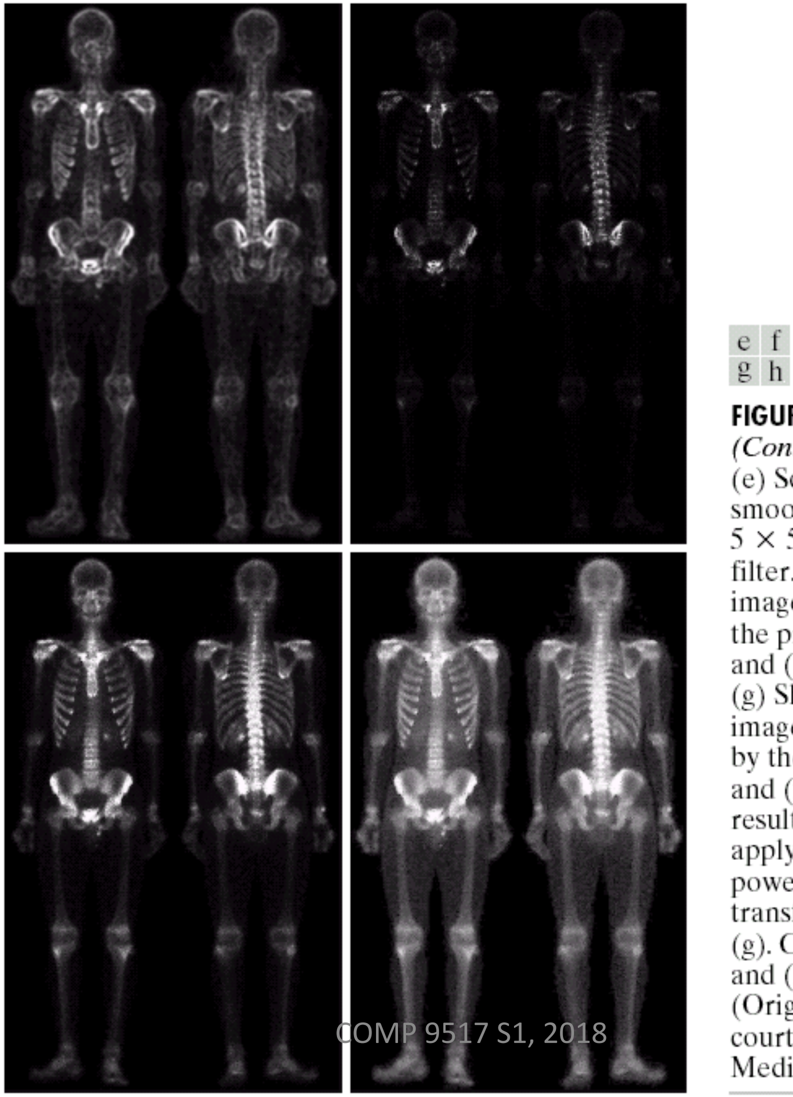


FIGURE 3.46 *(Continued)* (e) Sobel image smoothed with a 5 × 5 averaging filter. (f) Mask image formed by the product of (c) and (e). (g) Sharpened image obtained by the sum of (a) and (f). (h) Final result obtained by applying a power-law transformation to (g). Compare (g) and (h) with (a). (Original image courtesy of G.E. Medical Systems.)

References and Acknowledgements

- Chapter 3, Gonzalez and Woods 2002
- Szeliski 3.1-3.3
- Some content are extracted from the above resources