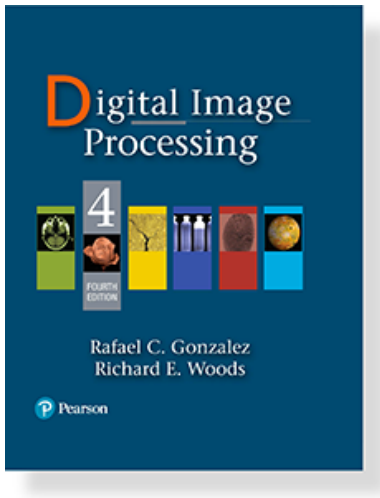
Digital Image Processing

Kuan-Wen Chen
2022/9/20

Digital Image Processing

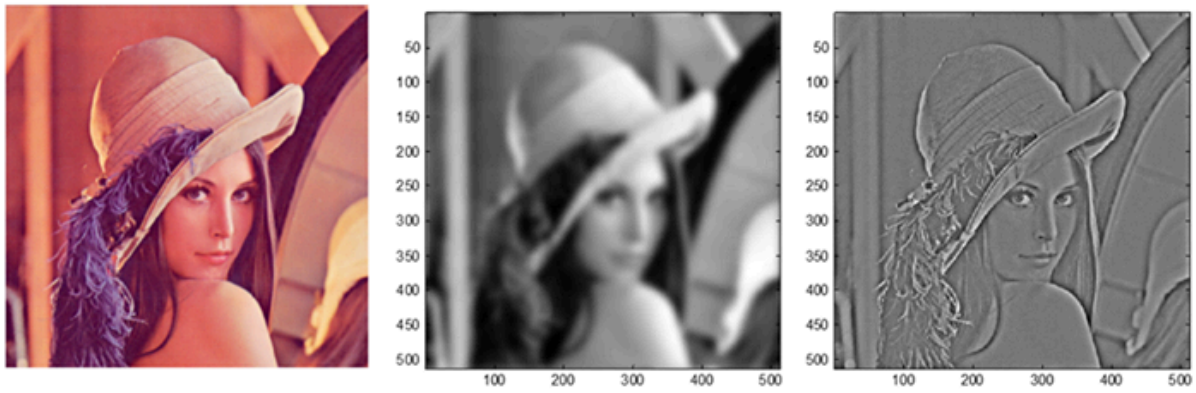
Most popular textbook

- R.C. Gonzalez and R.E. Woods, Digital Image Processing, 4th edition, 2018.



Most famous image

- Lena (512 x 512)



Digital Image Processing

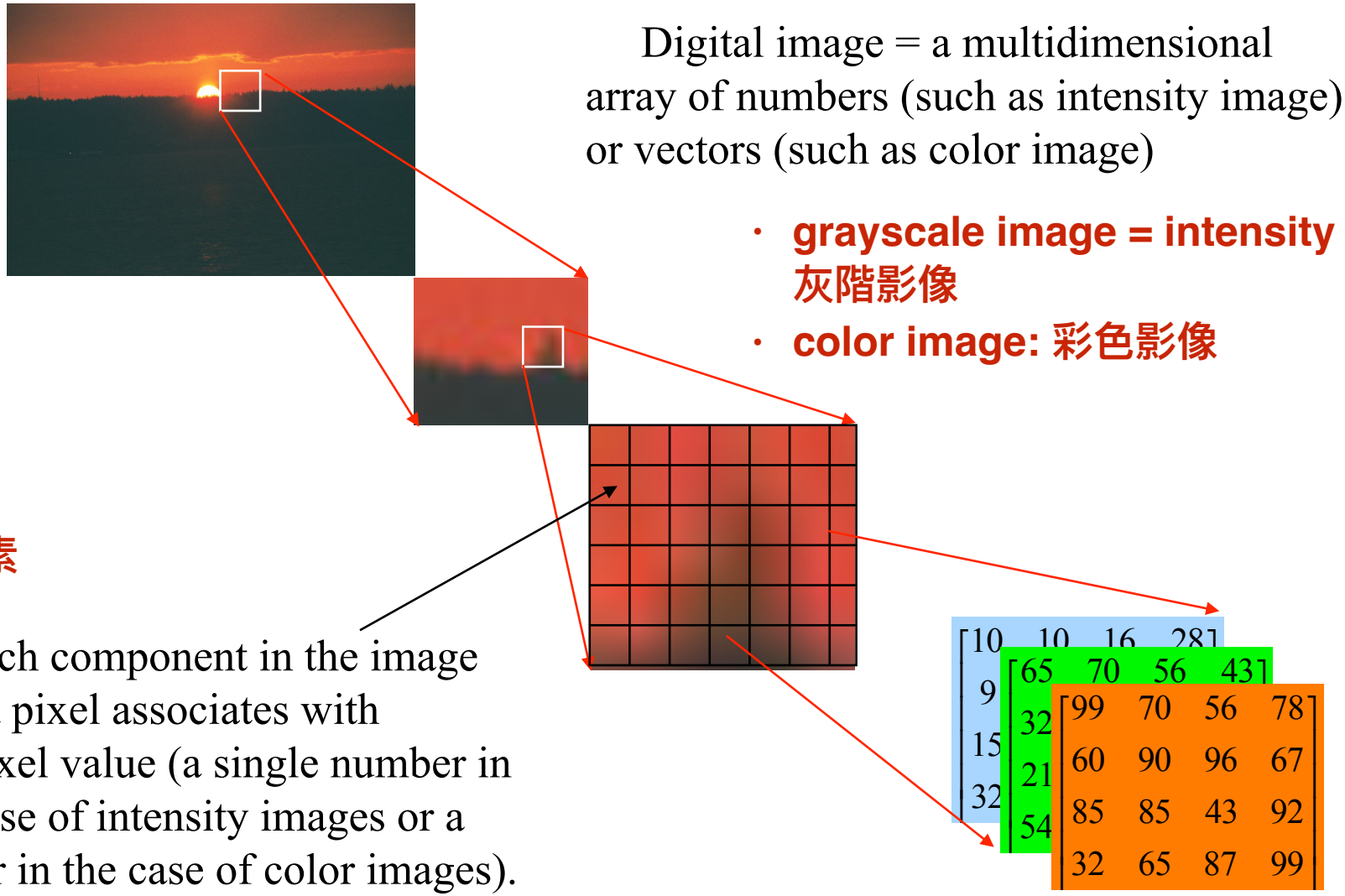
What will I teach?

- Digital image fundamentals
  - Image format (grayscale/color image)
  - Image sampling and quantization

- Spatial domain image processing
  - Point processing
    - image enhancement
    - thresholding
    - histogram
  - Mask processing (spatial filtering)
    - smoothing
    - edge detection and sharpening

Digital Image Fundamentals

Image Format



Digital image = a multidimensional array of numbers (such as intensity image) or vectors (such as color image)

- **grayscale image = intensity image: 灰階影像**
- **color image: 彩色影像**

Pixel: 像素

Each component in the image called pixel associates with the pixel value (a single number in the case of intensity images or a vector in the case of color images).

Image Format

For grayscale image

- **Intensity: 0 ~ 255**
- **1 byte for 1 pixel**

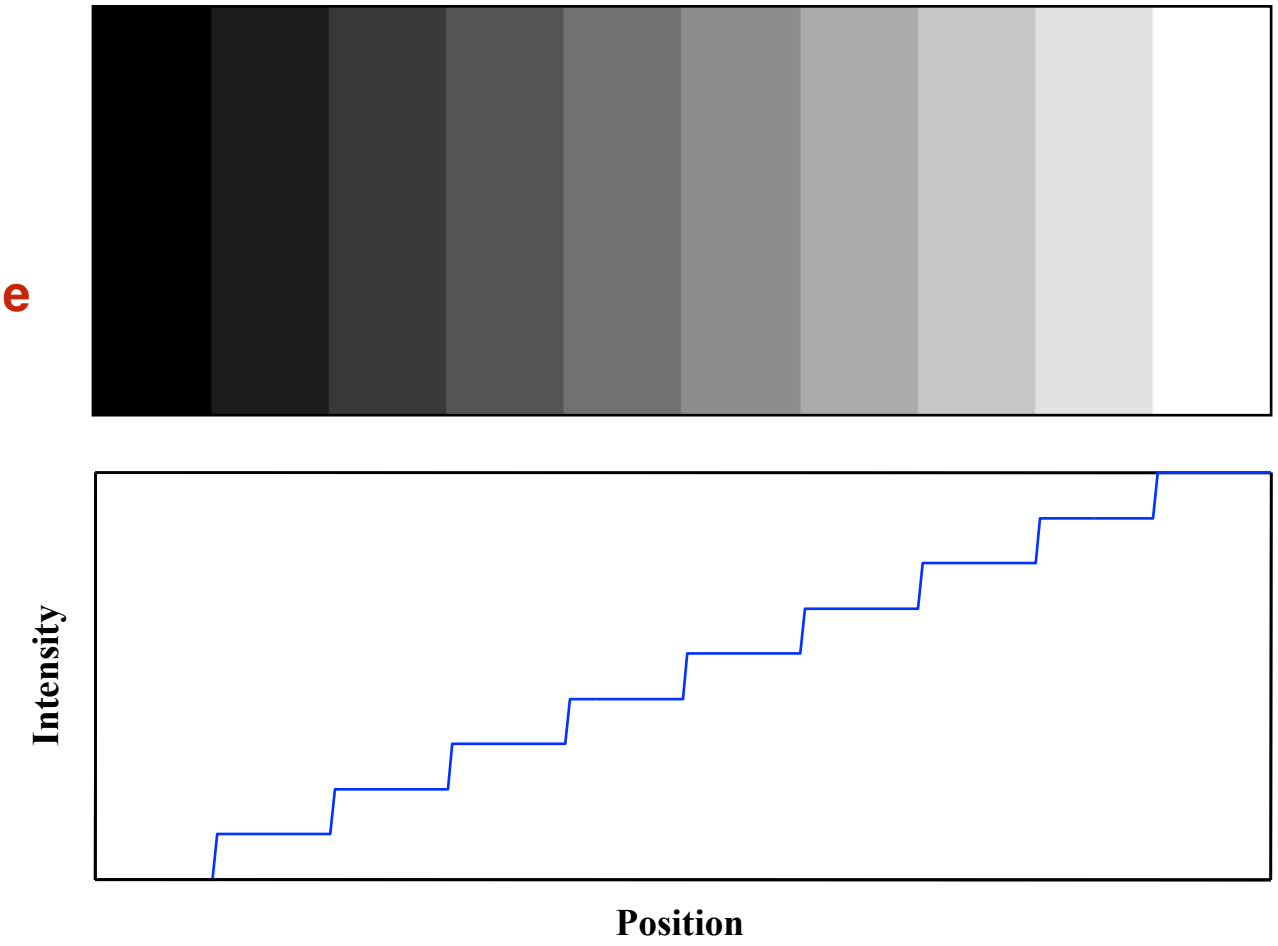
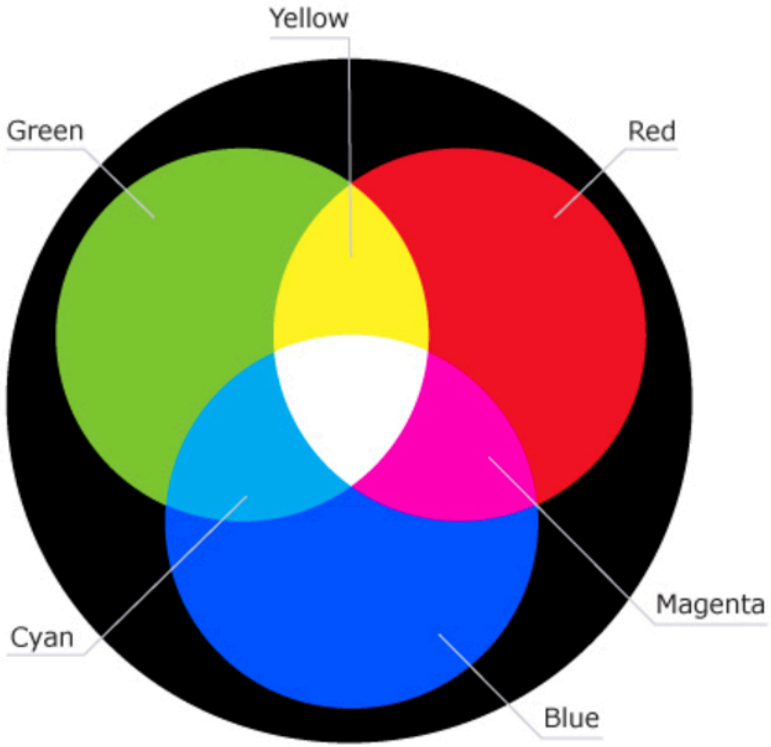
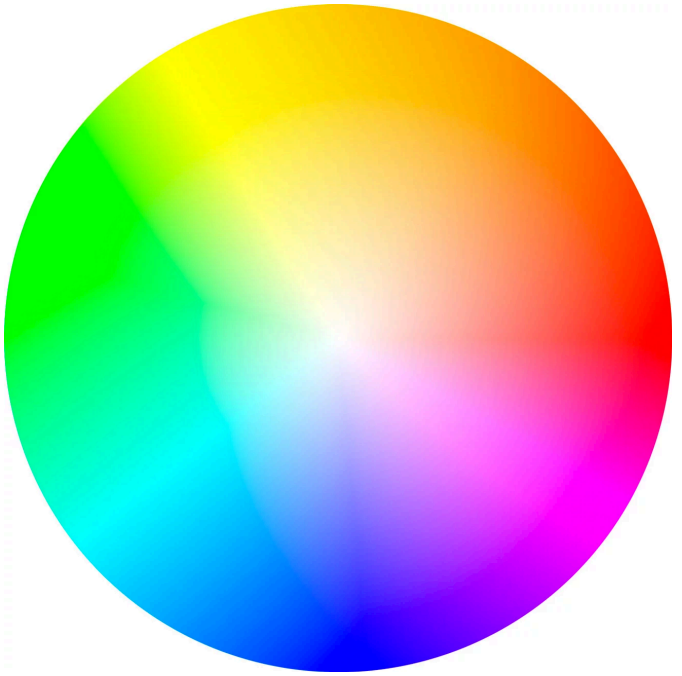


Image Format

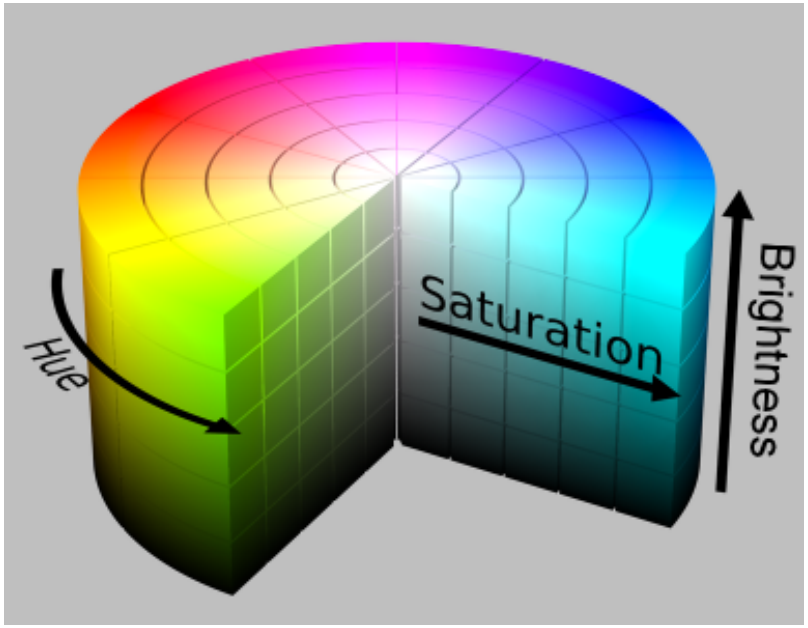




For color image

- **R: 0 ~ 255**
- **G: 0 ~ 255**
- **B: 0 ~ 255**
- **3 byte for 1 pixel, ex: 0x0000FF =**

Image Format



Steps to be followed:

1. Read a RGB image
2. Represent the RGB image in the range [0 1]
3. Find HSI components

$$\theta = \cos^{-1}\left\{\frac{\frac{1}{2}\left[(R-G)+(R-B)\right]}{\left[(R-G)^2+(R-B)(G-B)\right]^{1/2}}\right\}$$

4. H(Hue) = $\begin{cases}\theta & \text{If B<=G}\\ 360-\theta & \text{If B>G}\end{cases}$

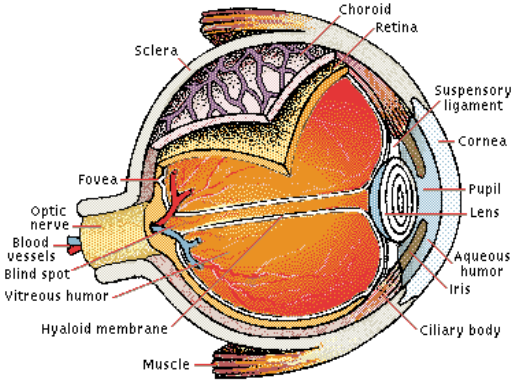
5. S(Saturation) = 1- $\frac{3}{(R+G+B)}\left[\min(R,G,B)\right]$

6. I(Intensity)= $\frac{1}{3}(R+G+B)$

For color image (HSI color model)

- **H: Hue**
- **S: Saturation**
- **I: Intensity**

Brightness Adaptation of Human Eye: Simultaneous Contrast



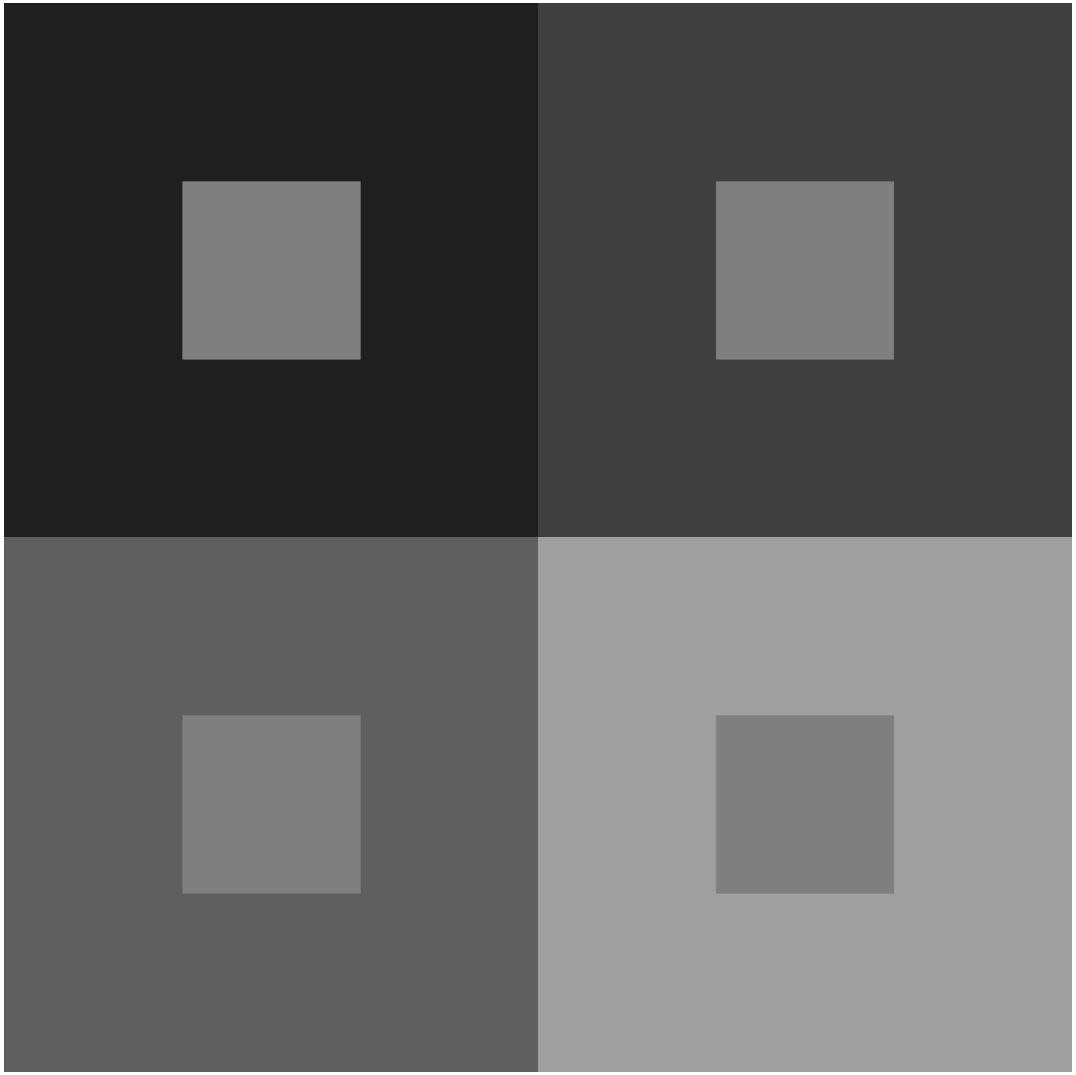
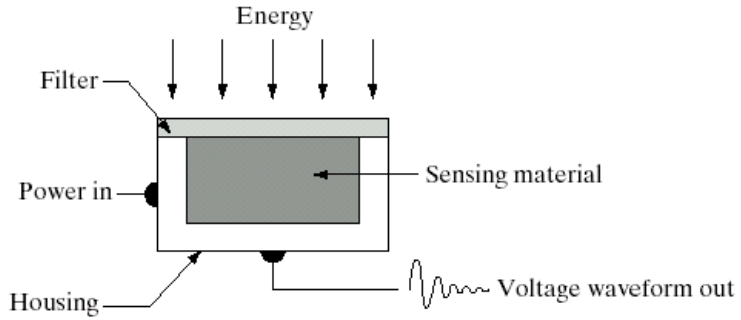
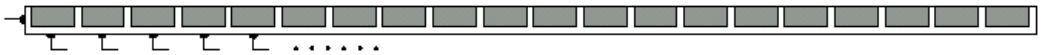


Image Sensors

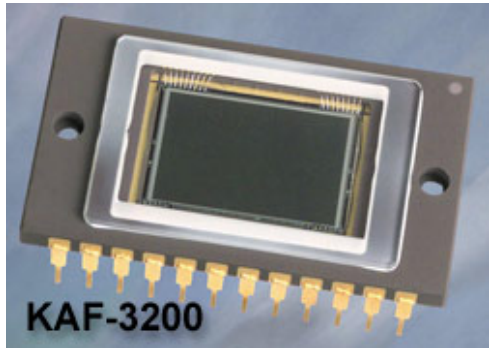


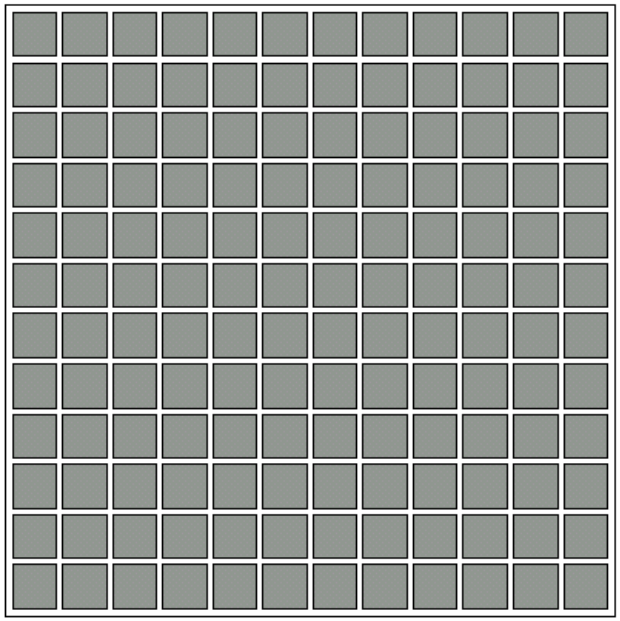
Single sensor



Line sensor

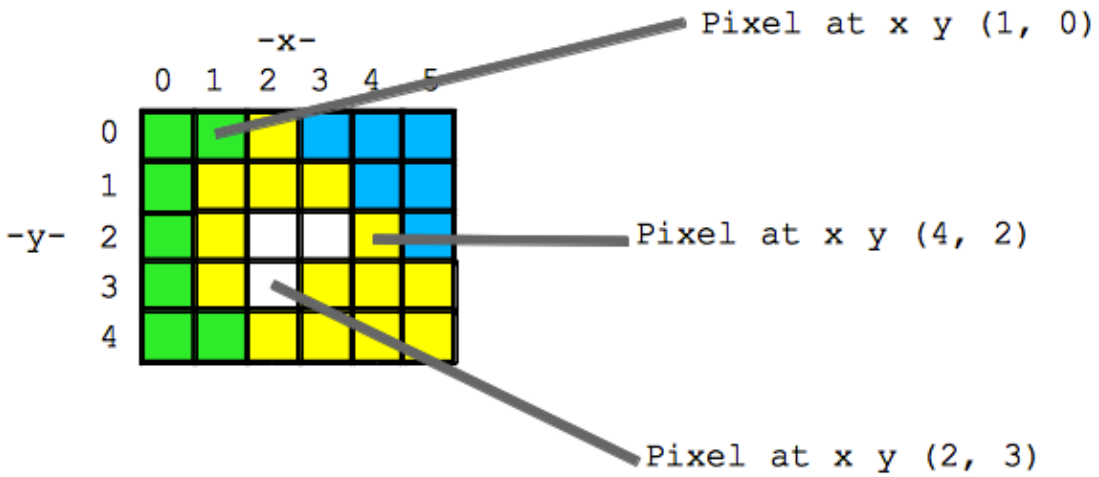
Charge-Coupled Device (CCD)





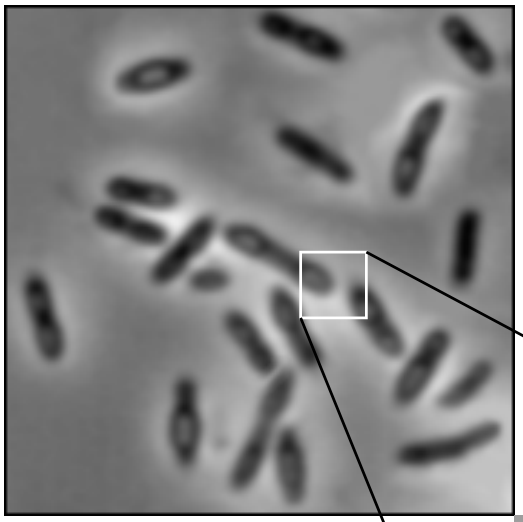
Array sensor

Fundamentals of Digital Images



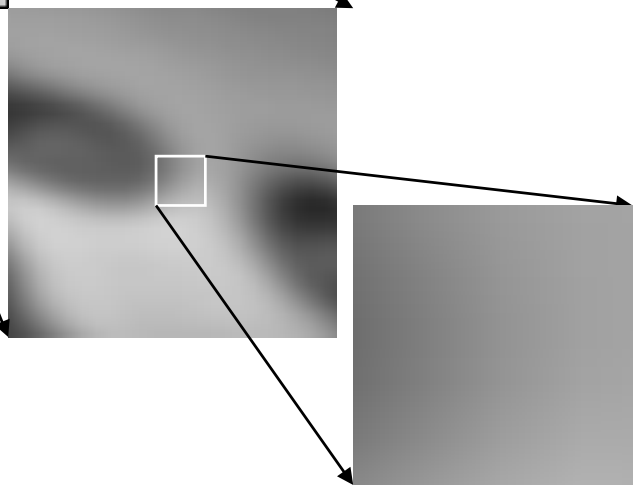
- **An image**: **a multidimensional function of spatial coordinates**.
- **Spatial coordinate**: (x,y) for 2D case such as photograph,
  (x,y,z) for 3D case such as CT scan images
  (x,y,t) for movies
- The function f may represent intensity (for monochrome images) or color (for color images) or other associated values.

Digital Image Types : Intensity Image



Intensity image or monochrome image

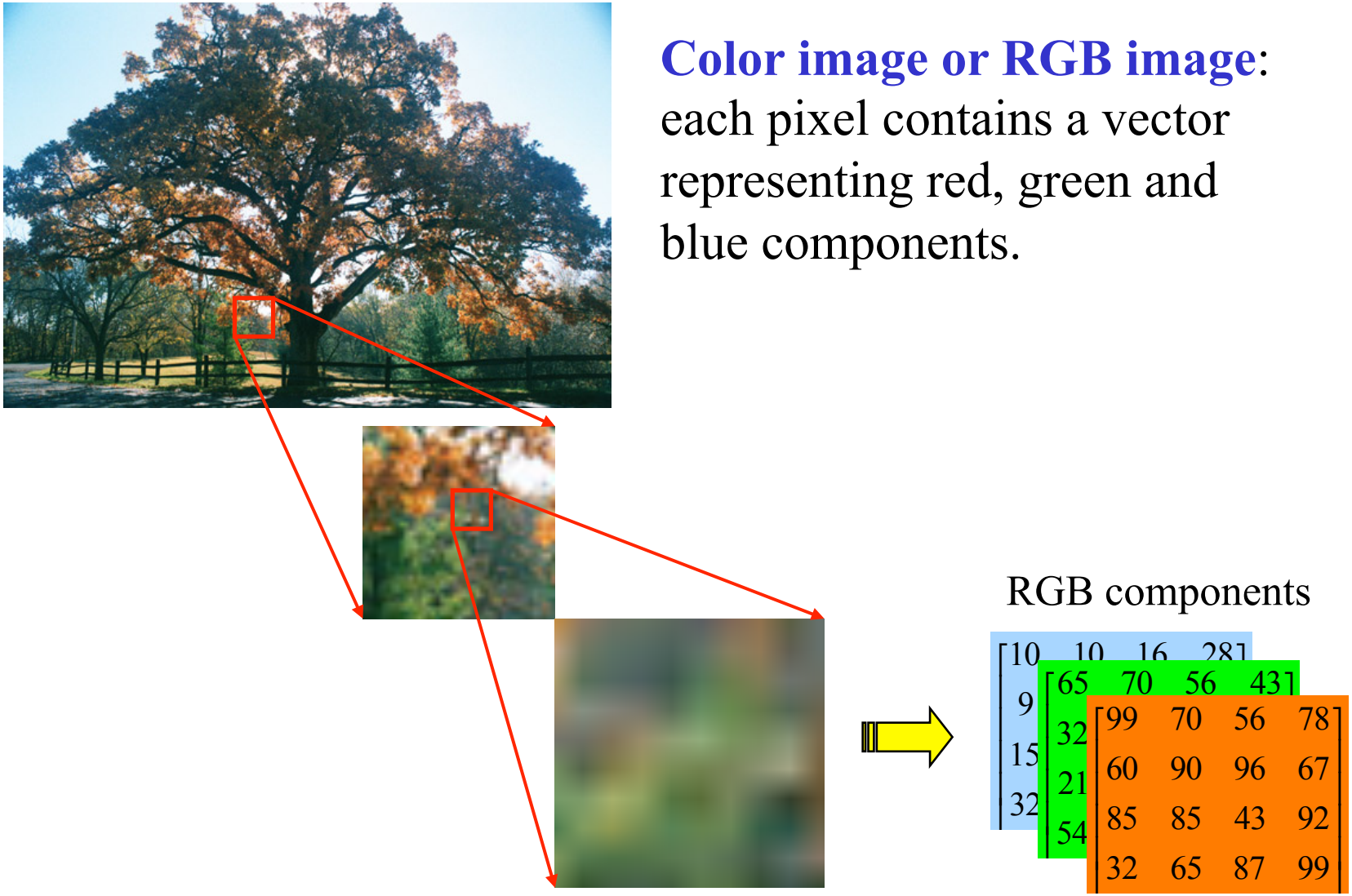
each pixel corresponds to light intensity normally represented in gray scale (gray level).



Gray scale values

$$\begin{bmatrix} 10 & 10 & 16 & 28 \\ 9 & 6 & 26 & 37 \\ 15 & 25 & 13 & 22 \\ 32 & 15 & 87 & 39 \end{bmatrix}$$

Digital Image Types : RGB Image



Color image or RGB image: each pixel contains a vector representing red, green and blue components.

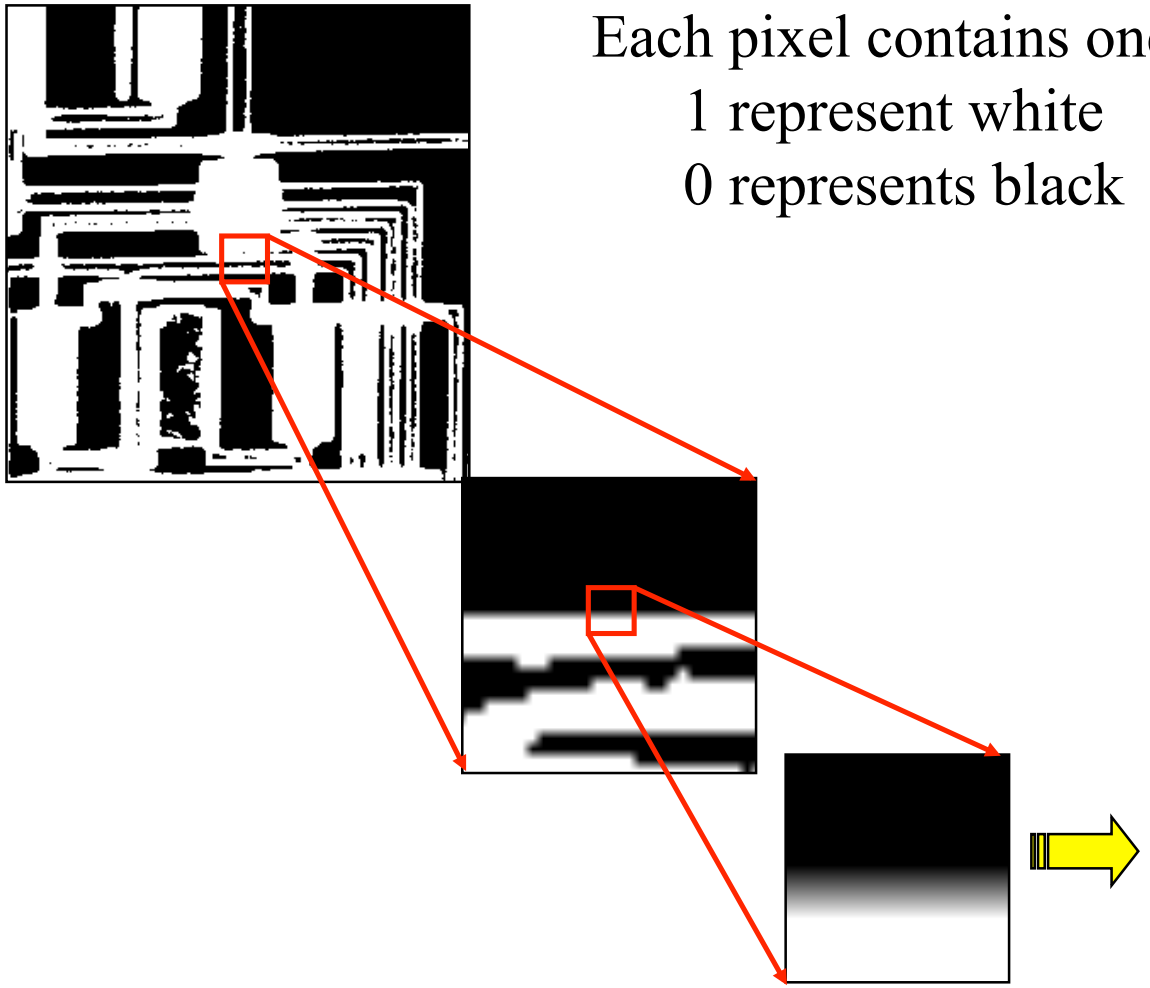
Digital Image Types : Binary Image

Binary image or black and white image

Each pixel contains one bit :

1 represent white

0 represents black



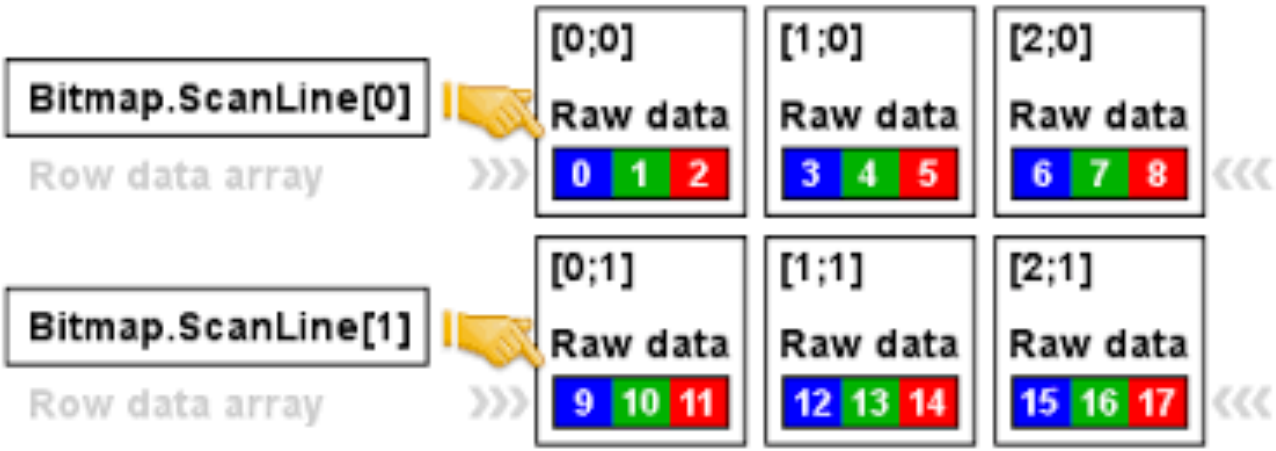
Binary data

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

Digital Image Format: Bitmap (BMP)

A typical BMP file usually contains the following blocks of data:

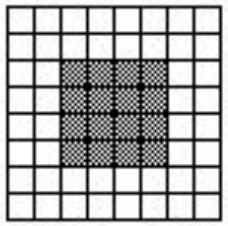
- **BMP File Header:** Stores general information about the BMP file.
- **DIB header:** Stores detailed information about the bitmap image.
- **Color Palette:** Stores the definition of the colors being used for indexed color bitmaps.
  - Image pixels are stored with a color depth of 1, 4, 8, 16, 24, or 32 bits per pixel
- **Bitmap Data:** Stores the actual image, pixel by pixel.



Digital Image Format: Image compression

Image Compression

Image



Pixel Values

0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 1 1 1 1 0 0
0 0 1 1 1 1 0 0
0 0 1 1 1 1 0 0
0 0 1 1 1 1 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0

repeated values
= redundancy,
= opportunity
for compression

Raw pixel data:

0 0 0 0 0 0 0 0, 0 0 0 0 0 0 0 0, 0 0 1 1 1 1 0 0, 0 0 1 1 1 1 0 0, 0 0 1 1 1 1 0 0,
0 0 1 1 1 1 0 0, 0 0 0 0 0 0 0 0, 0 0 0 0 0 0 0 0.

Run-Length Encoded:

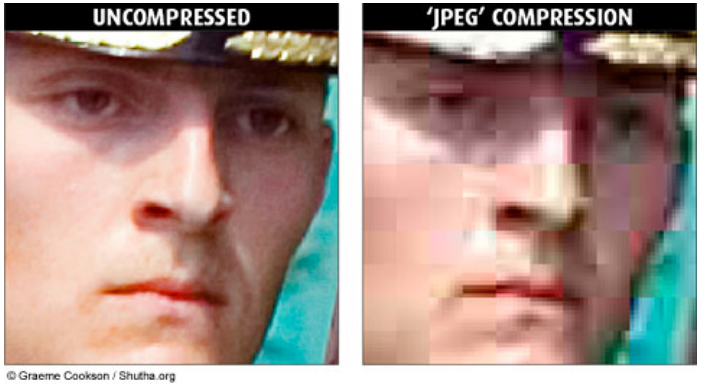
8(0), 8(0), 2(0) 4(1) 2(0), 2(0) 4(1) 2(0), 2(0) 4(1) 2(0), 2(0) 4(1) 2(0) ,8(0), 8(0).

Further Encoded:

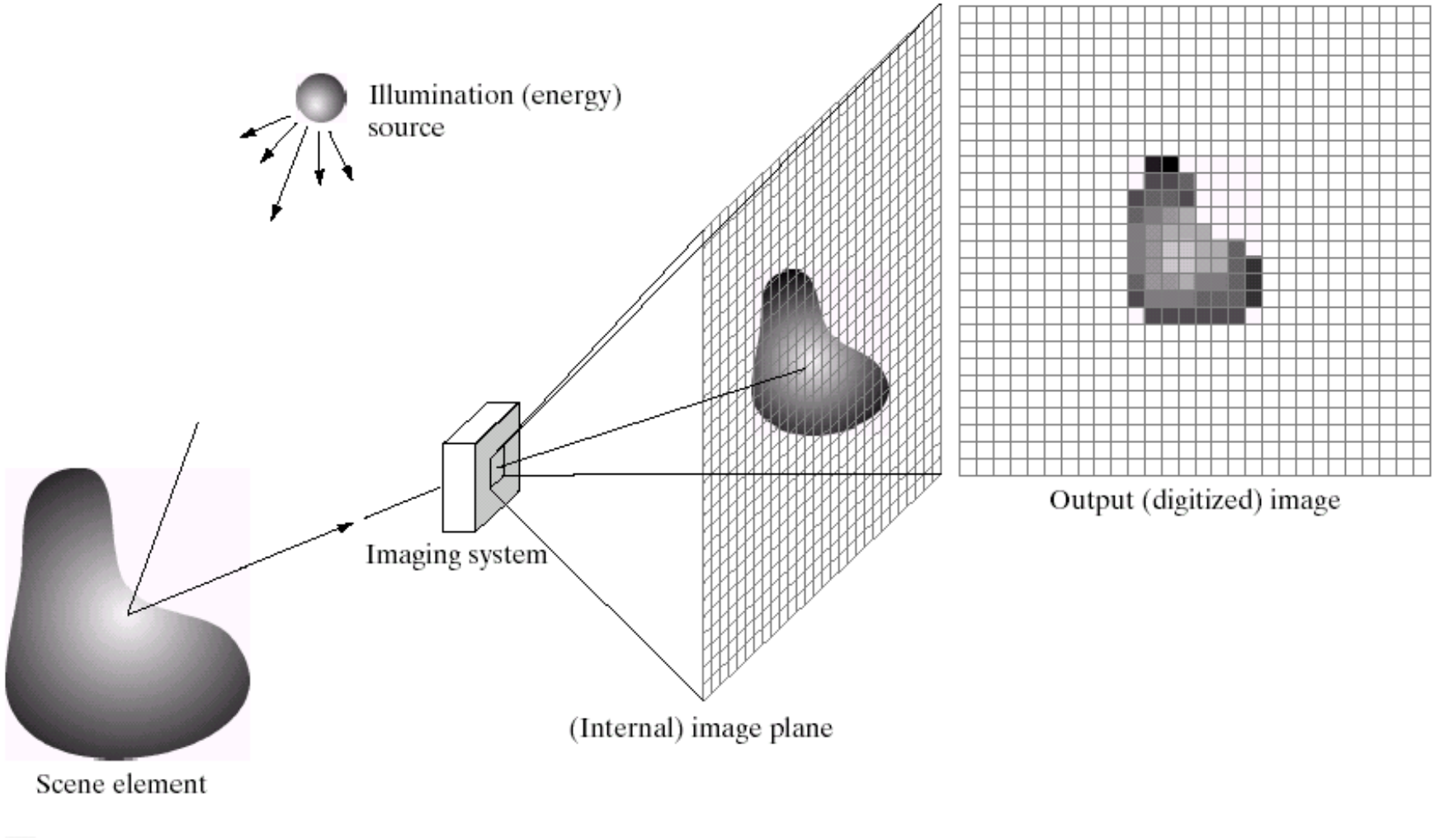
2(8(0),) 4(2(0) 4(1) 2(0),) 2(8(0),).

Symmetry Encoded:

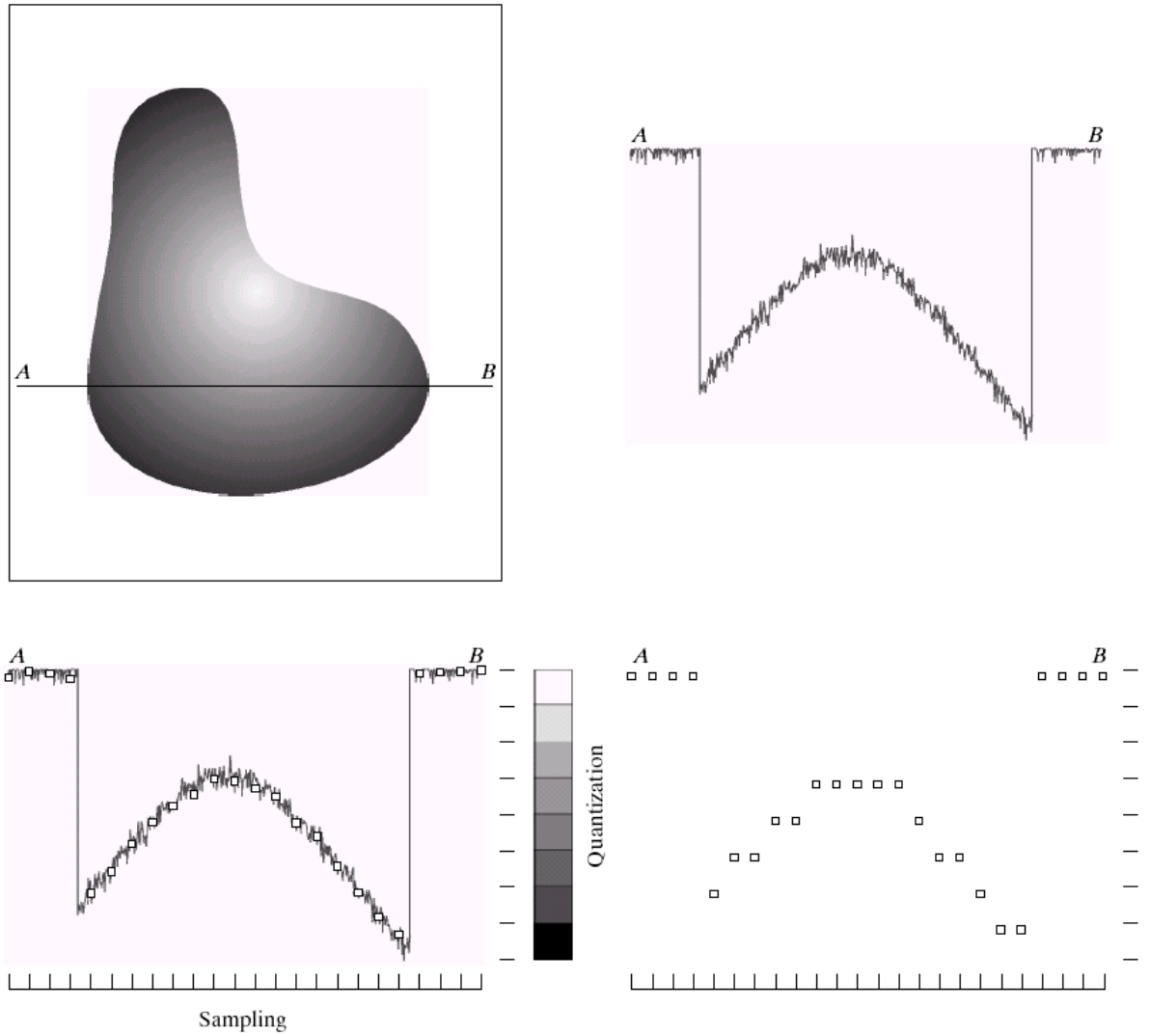
+(2(4(0)), 2(2(0) 2(1)),). "+" = four-fold symmetry



Digital Image Acquisition Process



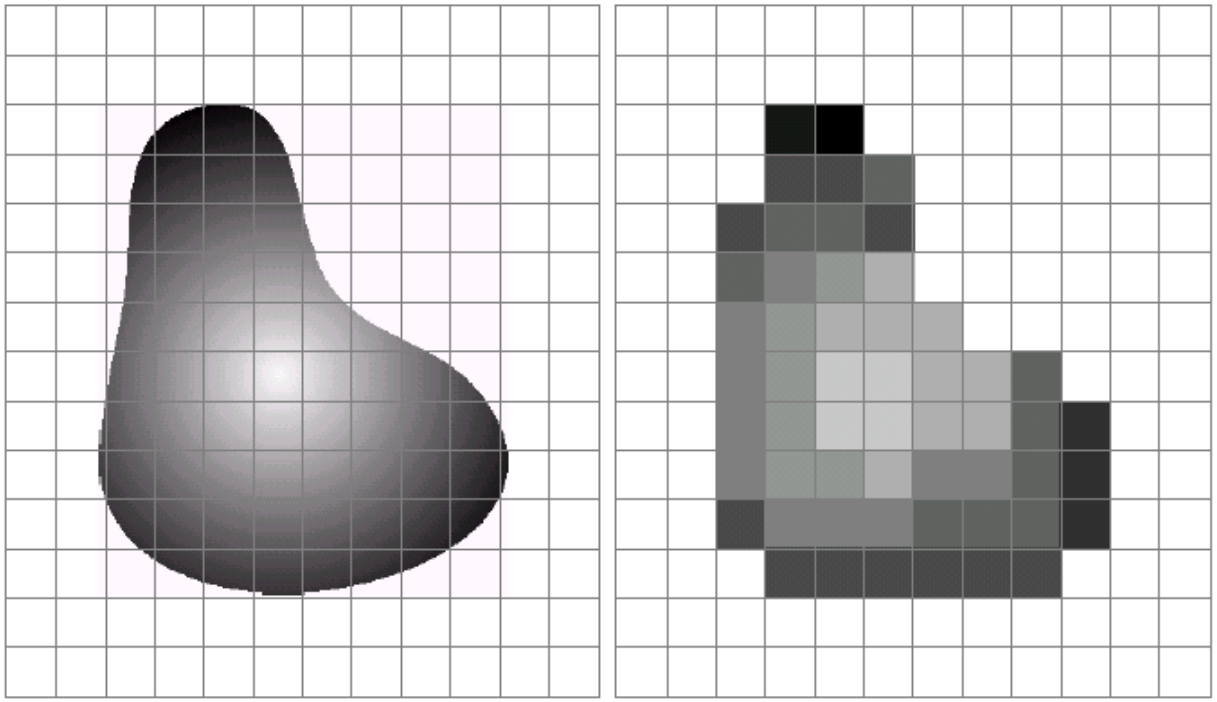
Generating a Digital Image



a b
c d

FIGURE 2.16 Generating a digital image. (a) Continuous image. (b) A scan line from *A* to *B* in the continuous image, used to illustrate the concepts of sampling and quantization. (c) Sampling and quantization. (d) Digital scan line.

Image Sampling and Quantization



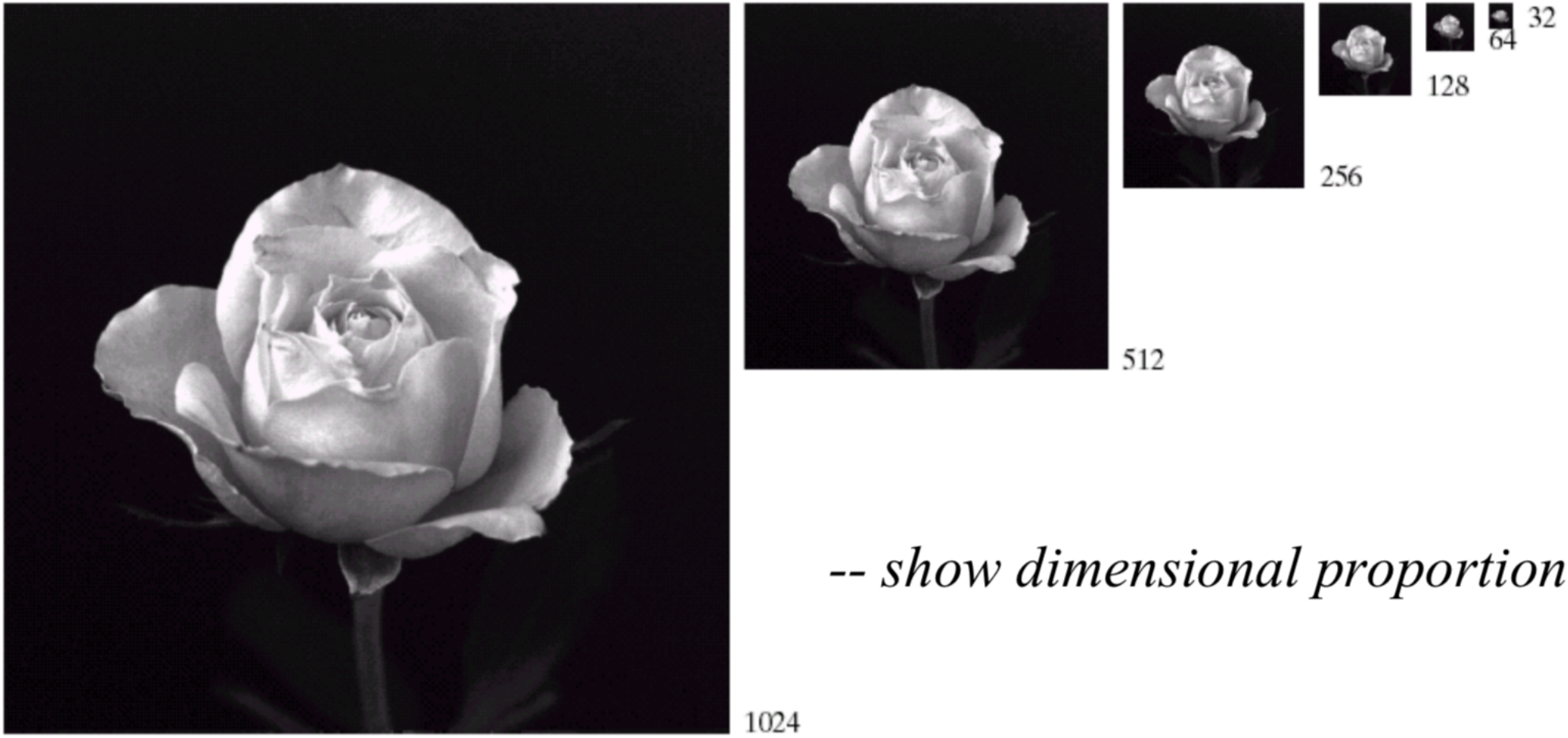
a b

FIGURE 2.17 (a) Continuos image projected onto a sensor array. (b) Result of image sampling and quantization.

Image sampling: discretize an image in the spatial domain

Spatial resolution / image resolution: pixel size or number of pixels

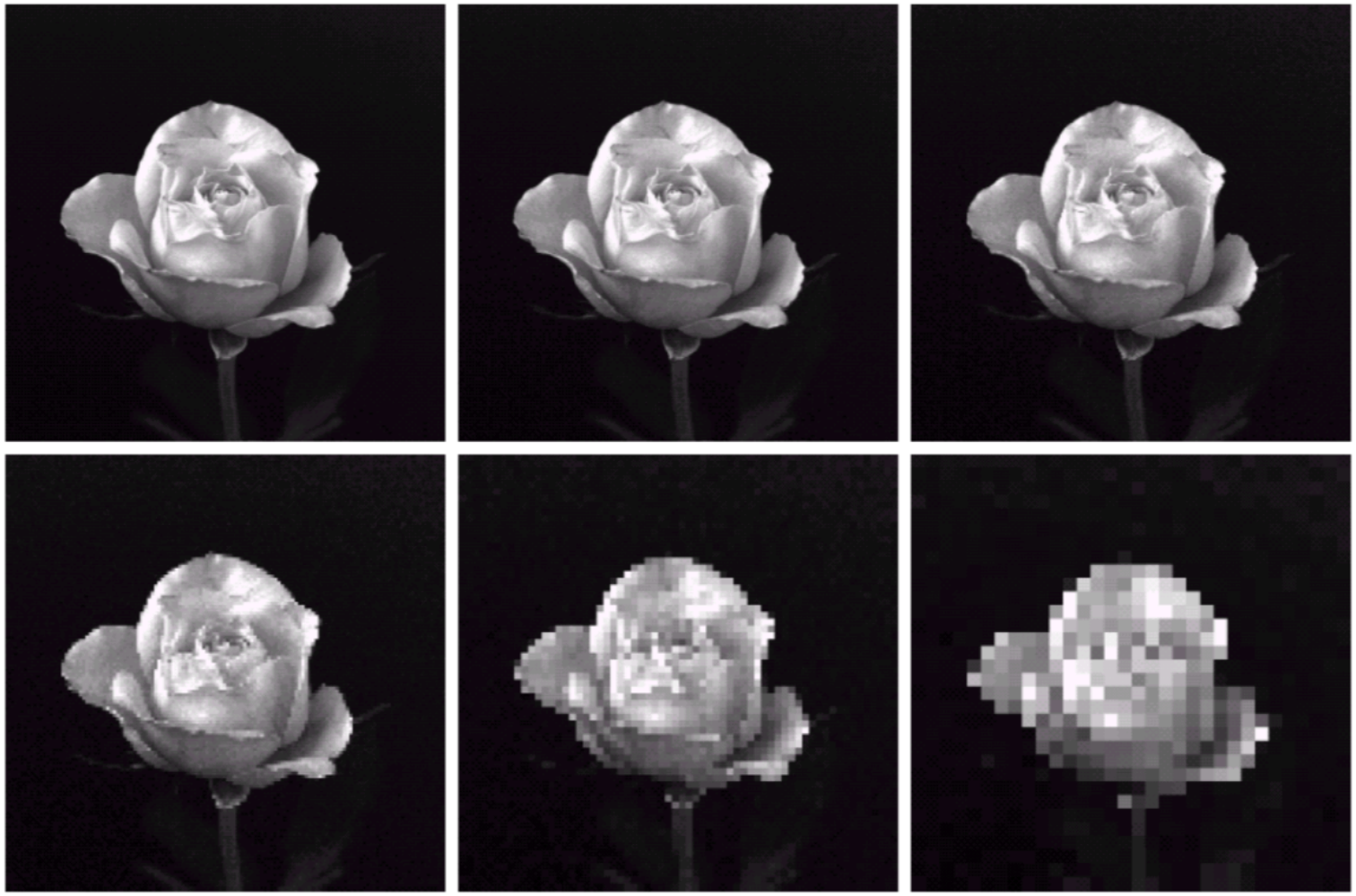
Spatial Resolution



-- show dimensional proportion

FIGURE 2.19 A 1024 × 1024, 8-bit image subsampled down to size 32 × 32 pixels. The number of allowable gray levels was kept at 256.

-- zoom-in to show the effects of subsampling

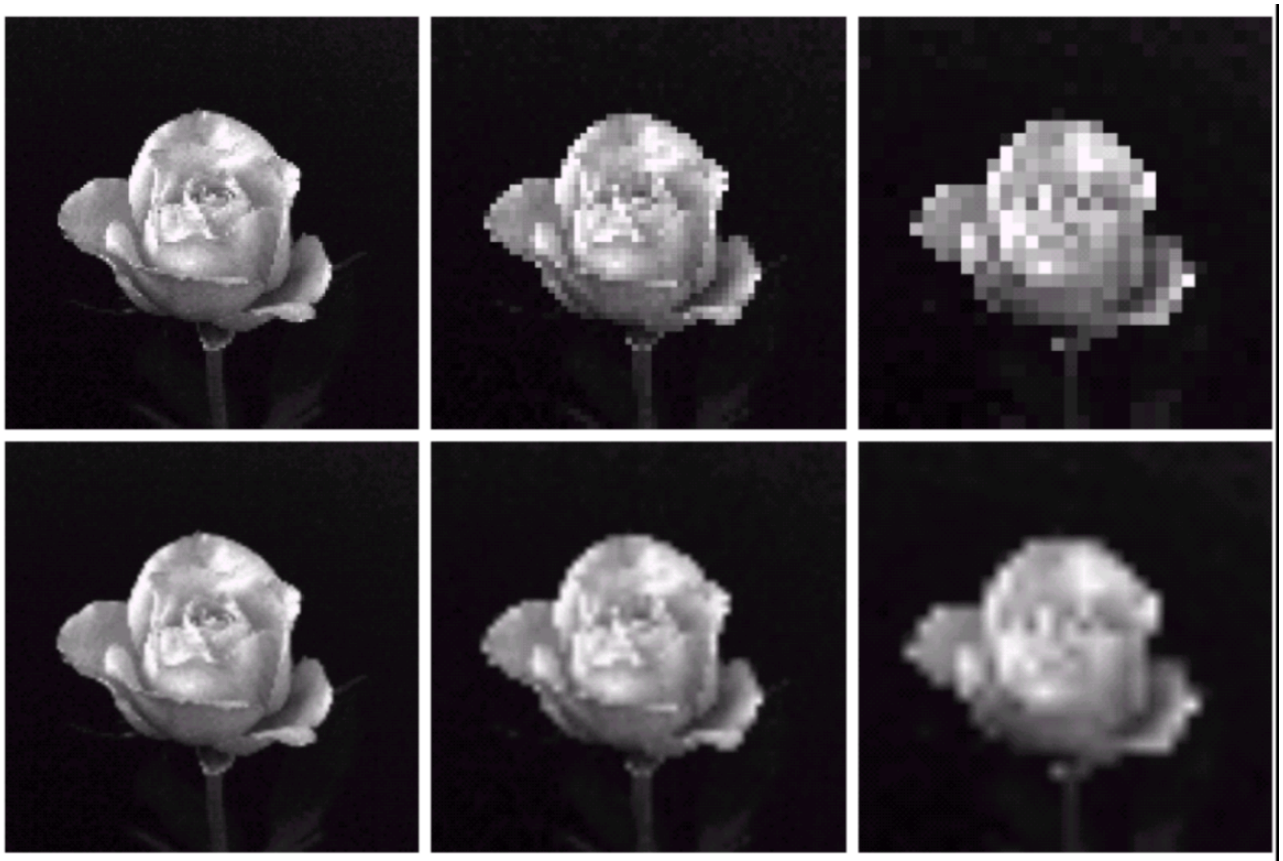


a b c
d e f

FIGURE 2.20 (a) 1024 × 1024, 8-bit image. (b) 512 × 512 image resampled into 1024 × 1024 pixels by row and column duplication. (c) through (f) 256 × 256, 128 × 128, 64 × 64, and 32 × 32 images resampled into 1024 × 1024 pixels.

Image Interpolation

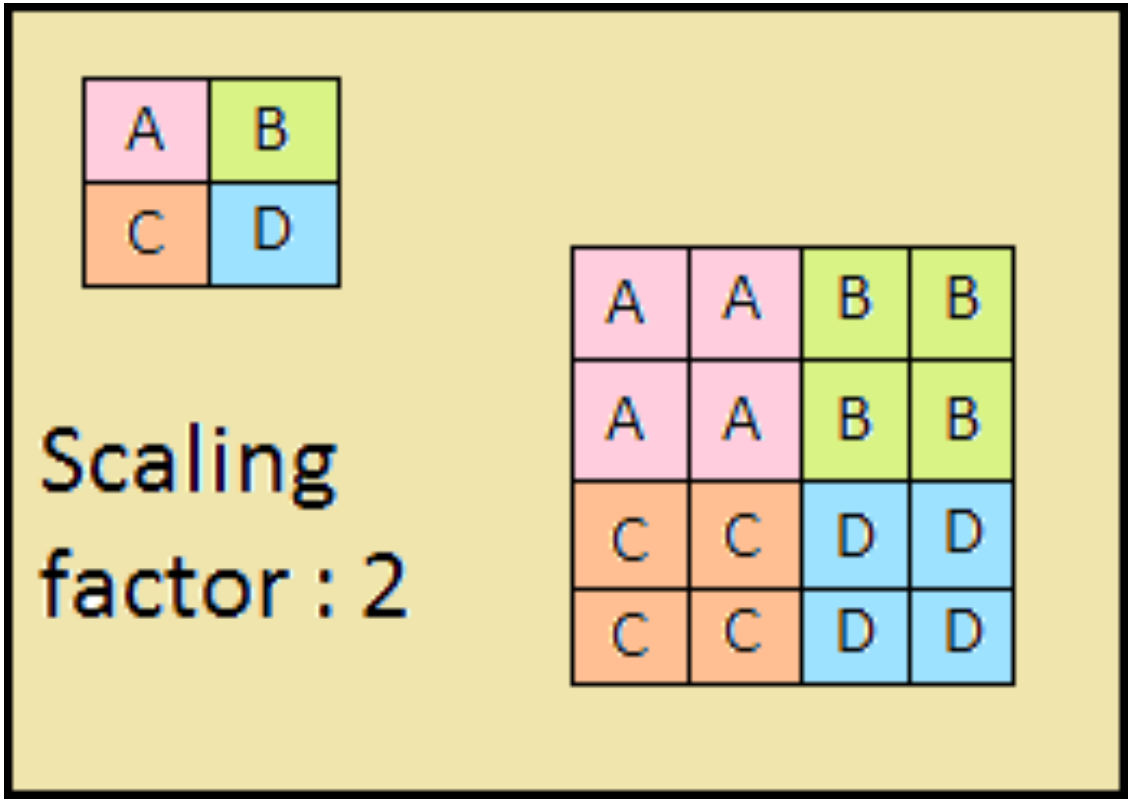
- Nearest-Neighbor Interpolation
- Bilinear Interpolation



a b c
d e f

FIGURE 2.25 Top row: images zoomed from 128 × 128, 64 × 64, and 32 × 32 pixels to 1024 × 1024 pixels, using nearest neighbor gray-level interpolation. Bottom row: same sequence, but using bilinear interpolation.

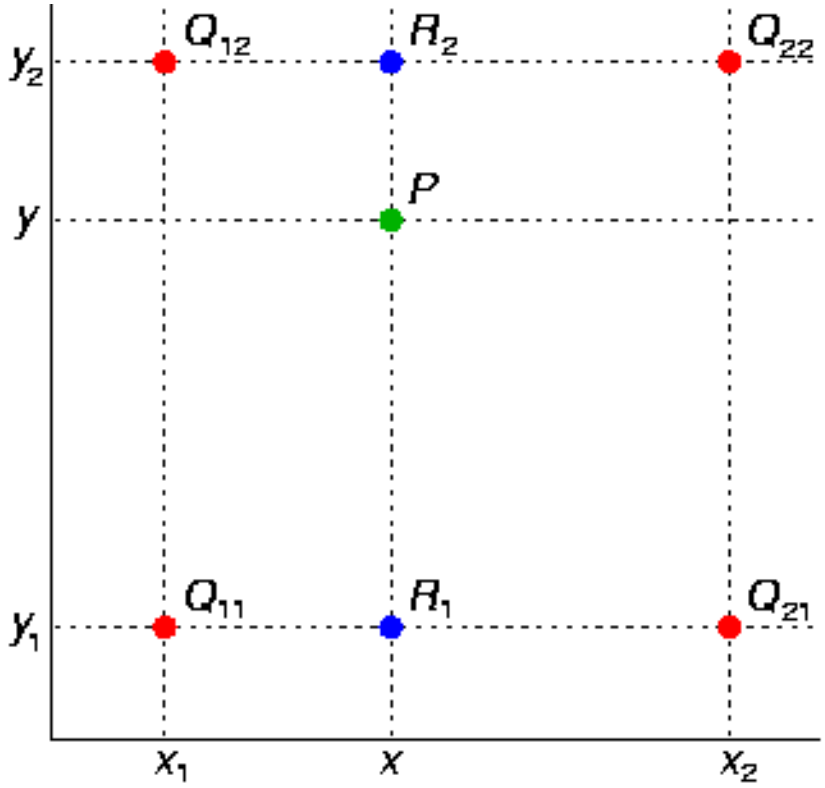
Nearest-Neighbor Interpolation



Bilinear Interpolation

$$f(x, y_1) \approx \frac{x_2 - x}{x_2 - x_1} f(Q_{11}) + \frac{x - x_1}{x_2 - x_1} f(Q_{21}),$$
$$f(x, y_2) \approx \frac{x_2 - x}{x_2 - x_1} f(Q_{12}) + \frac{x - x_1}{x_2 - x_1} f(Q_{22}).$$

$$f(x, y) \approx \frac{y_2 - y}{y_2 - y_1} f(x, y_1) + \frac{y - y_1}{y_2 - y_1} f(x, y_2)$$



Spatial Domain Image Processing

Image Enhancement

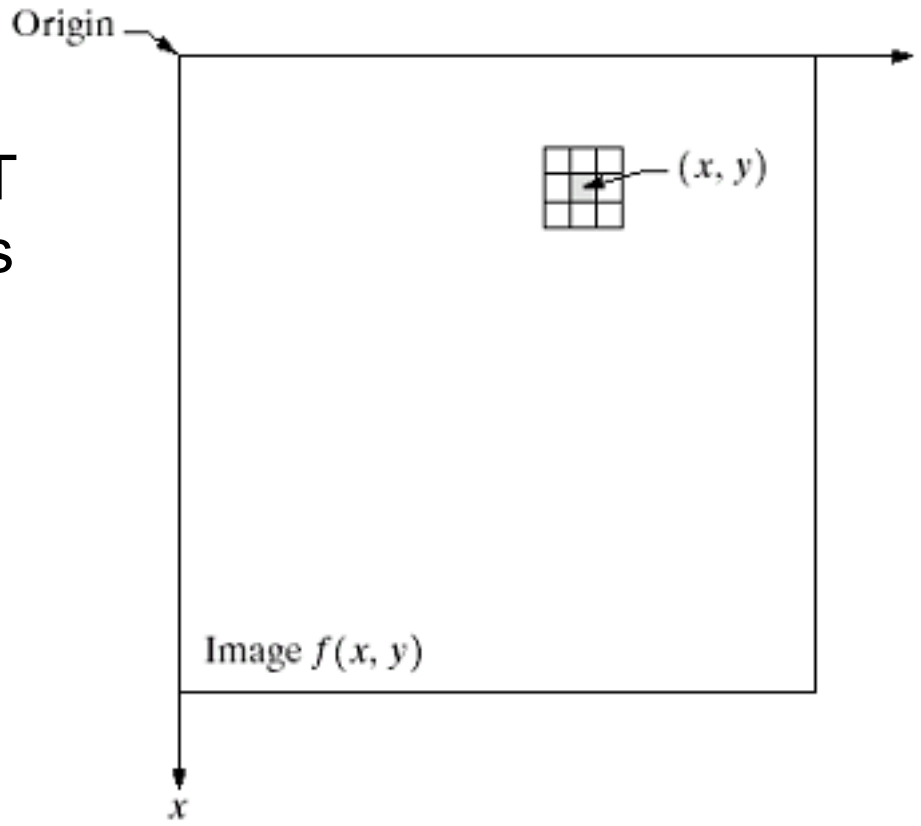
Image Enhancement means improvement of images to be suitable for specific applications.

Example:

Note: each image enhancement technique that is suitable for one application may not be suitable for other applications.

Neighbourhood

- For example, an operator T utilizes only the pixels in the area of the image spanned by the neighborhood, e.g., a 3x3 neighborhood.

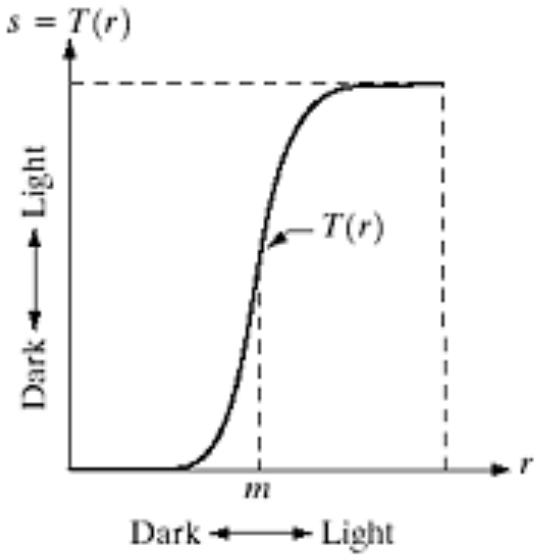


Point processing - 1x1 neighborhood

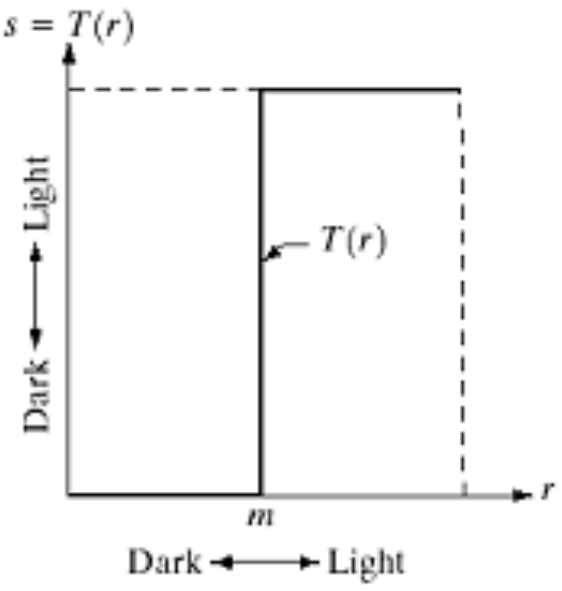
- Gray-level Transformation Function

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

Contrast Stretching



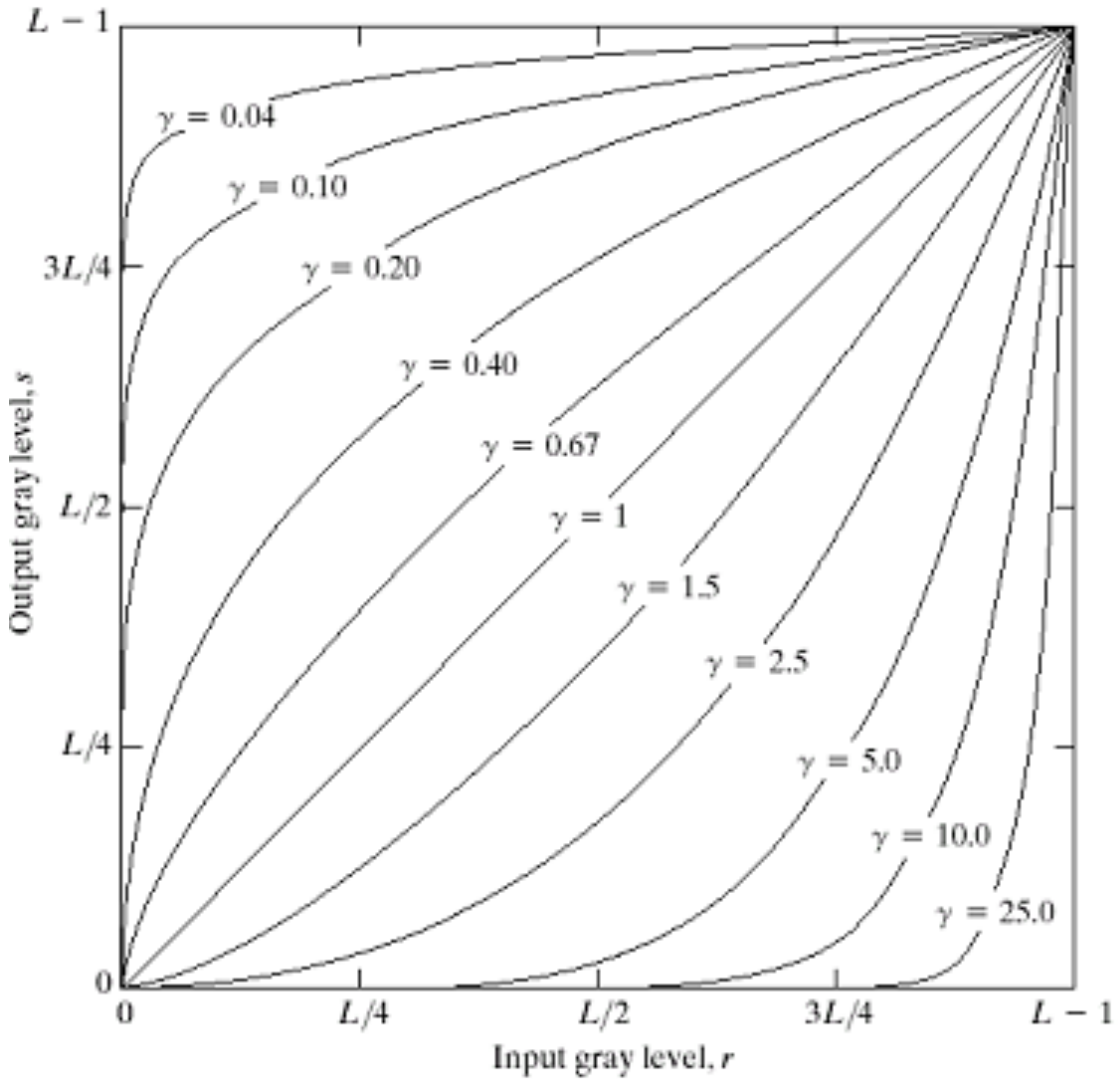
Thresholding



Gamma Correction

gamma

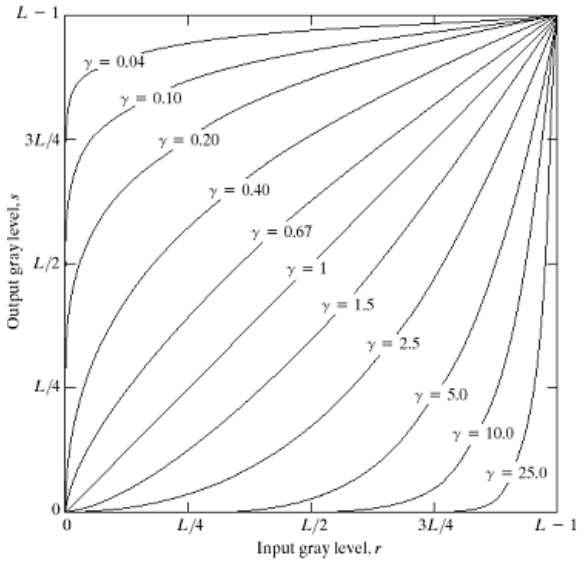
$$s = c\, r^{\gamma}$$

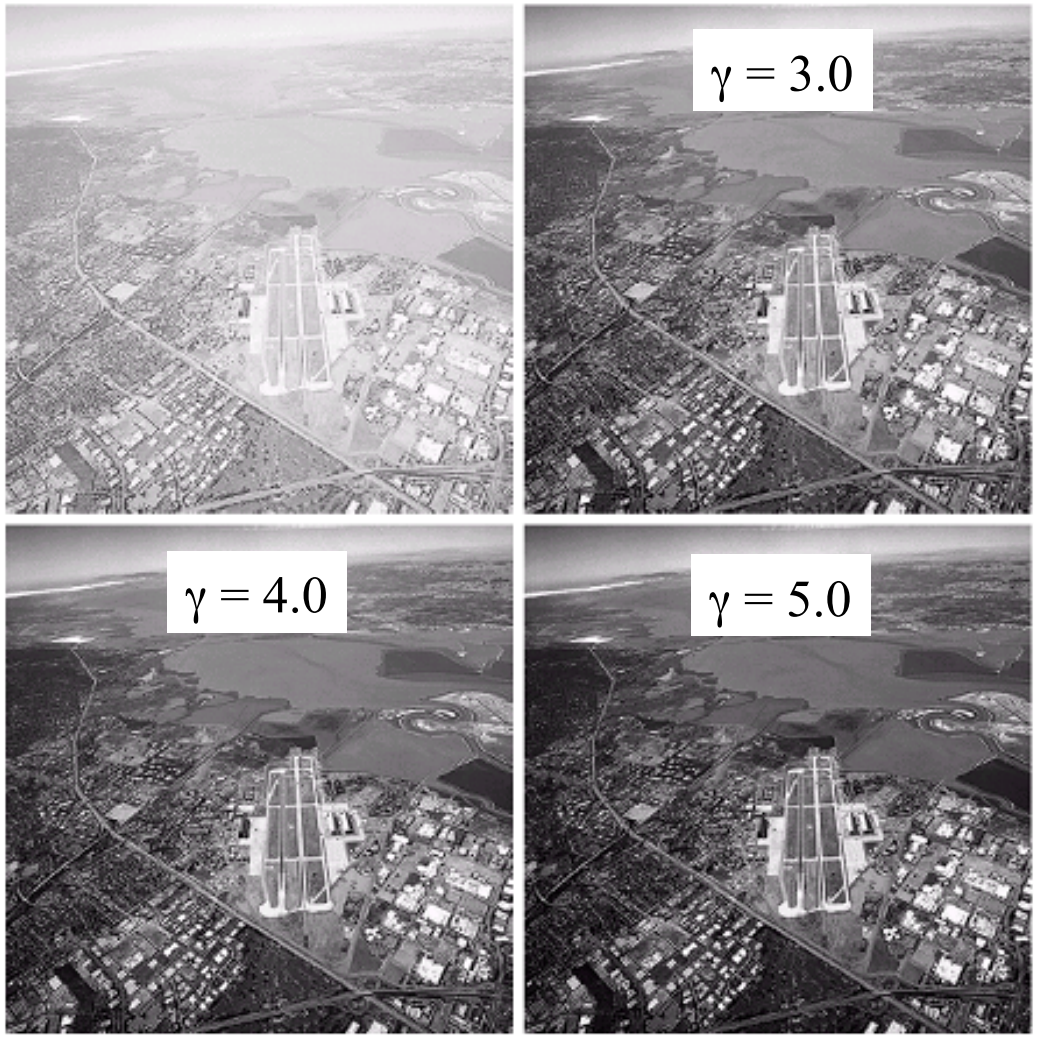


Gamma Correction

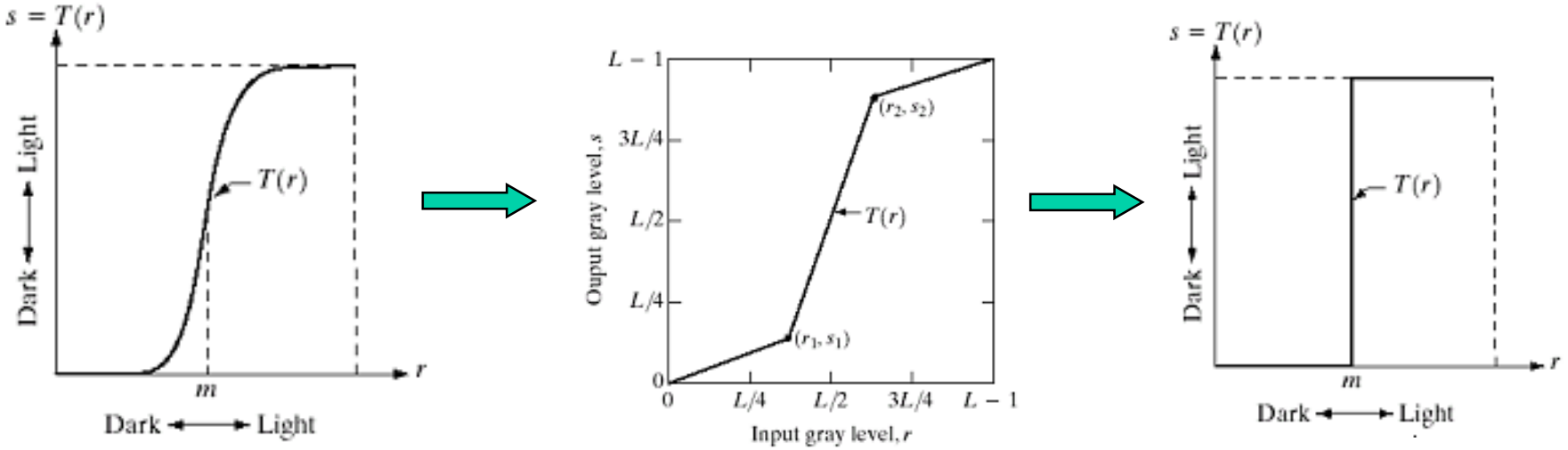
gamma

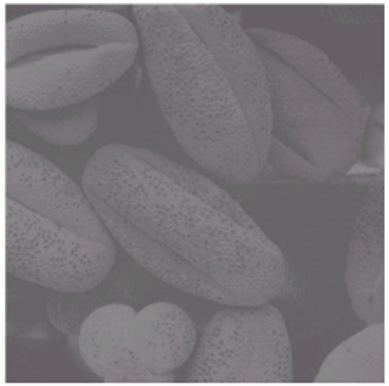
$$s = c\, r^{\gamma}$$

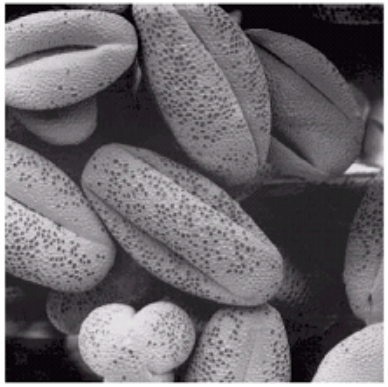




Contrast Stretching







Full-Scale Histogram Stretch

$(r_1, s_1) = (r_{\min}, 0)$

$(r_2, s_2) = (r_{\max}, L\text{-}1)$

Thresholding

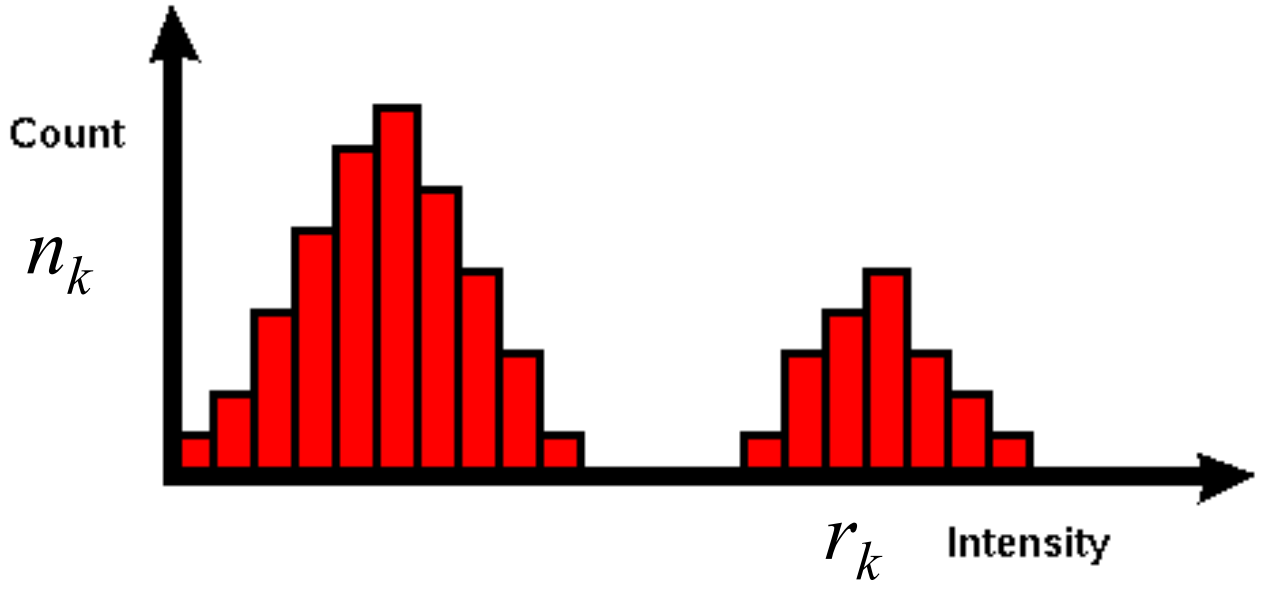
Histogram

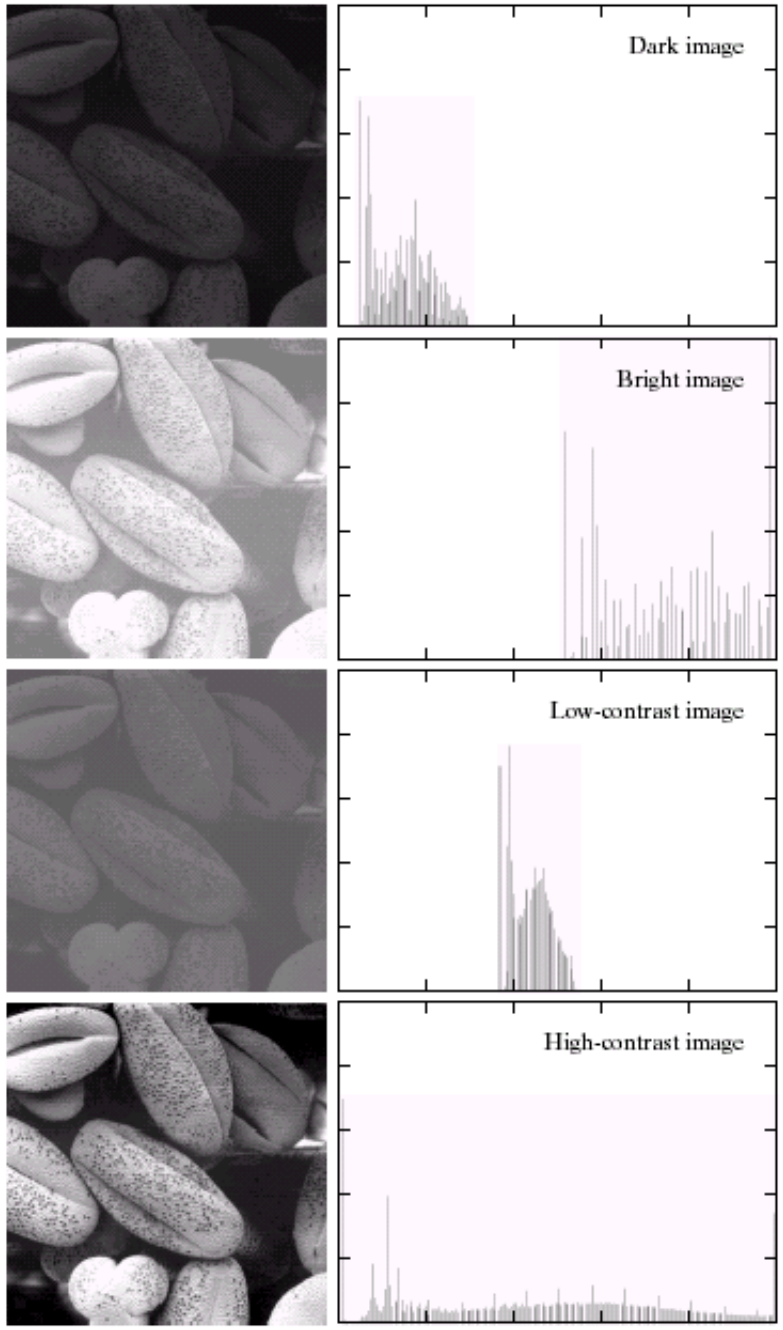
- Histogram of a digital image is a distribution function

$$h(r_k) = n_k$$

  - where r_k is the kth gray level

    and n_k is the number of pixels having gray level r_k





Different types of histograms:

- Normalized histogram

$$p(r_k) = n_k / n \ , \quad k = 1, \ ..., \ L$$

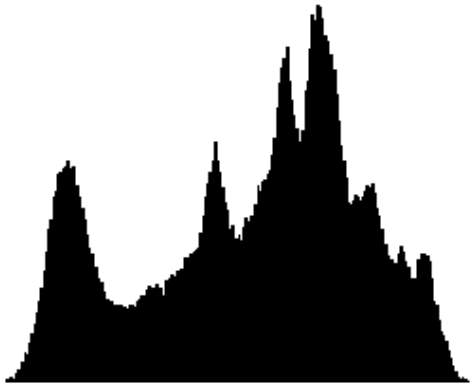
- Histogram is useful for
  - -- image enhancement
  - -- image compression
  - -- image segmentation
  - -- etc.

→ Want to have a more flat histogram !

=> Histogram Equalization

Histogram Equalization

Image
Enhancement



Histogram
Equalization

To make histogram distributed uniformly

Aogirithm of Histogram Equalization

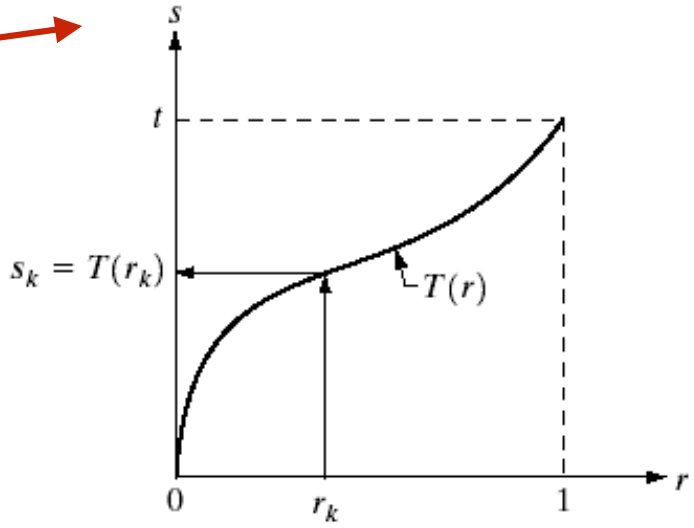
1. *Compute the histogram of the input image:*

$$h(k) = \#\{(x,y)|f(x,y)=k\}, \text{ where } k = 0 \text{ to } 255.$$

2. *Compute the transformation function:*

$$T(k) = 255 * \sum_{j=0}^{k} \frac{h(j)}{n}$$

Cumulative normalized histogram



3. *Transform the value of each pixel by*

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

Histogram Equalization Example

Intensity	# pixels	Accumulative Sum of P_r
0	20	20/100 = 0.2
1	5	(20+5)/100 = 0.25
2	25	(20+5+25)/100 = 0.5
3	10	(20+5+25+10)/100 = 0.6
4	15	(20+5+25+10+15)/100 = 0.75
5	5	(20+5+25+10+15+5)/100 = 0.8
6	10	(20+5+25+10+15+5+10)/100 = 0.9
7	10	(20+5+25+10+15+5+10+10)/100 = 1.0
Total	100	1.0

Histogram Equalization Example

Intensity (r)	No. of Pixels (n_j)	Acc Sum of P_r	Output value	Quantized Output (s)
0	20	0.2	0.2x7 = 1.4	1
1	5	0.25	0.25*7 = 1.75	1
2	25	0.5	0.5*7 = 3.5	3
3	10	0.6	0.6*7 = 4.2	4
4	15	0.75	0.75*7 = 5.25	5
5	5	0.8	0.8*7 = 5.6	5
6	10	0.9	0.9*7 = 6.3	6
7	10	1.0	1.0x7 = 7	7
Total	100			

Mask Processing

Mask processing - Spatial Filtering

- Filter, Mask, Kernel, Template, Window
- Coefficients
- Linear Filtering *vs* Nonlinear Filtering (e.g., median filtering)

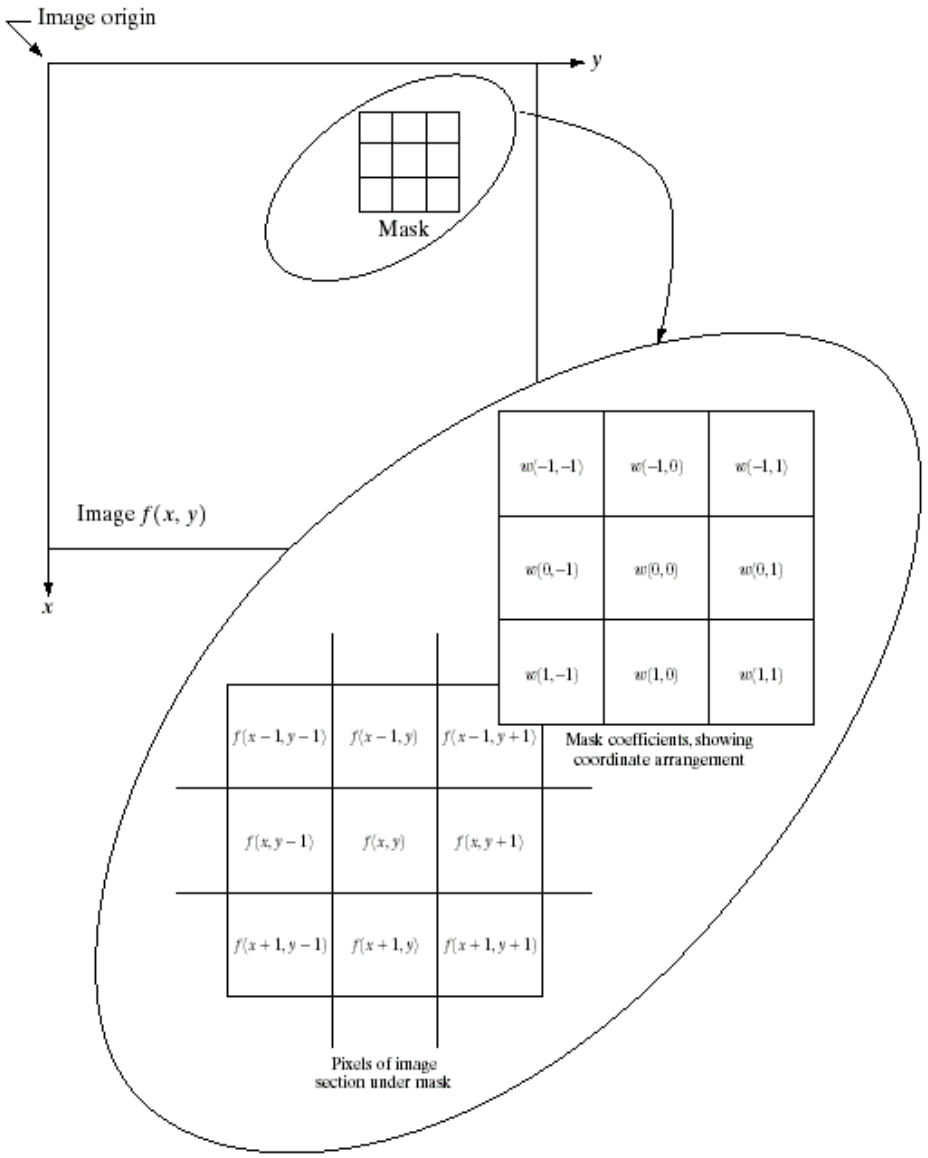


FIGURE 3.32 The mechanics of spatial filtering. The magnified drawing shows a 3×3 mask and the image section directly under it; the image section is shown displaced out from under the mask for ease of readability.

Linear Smoothing Filters – averaging filters

$\frac{1}{9} \times$

1	1	1
1	1	1
1	1	1

Box Filter

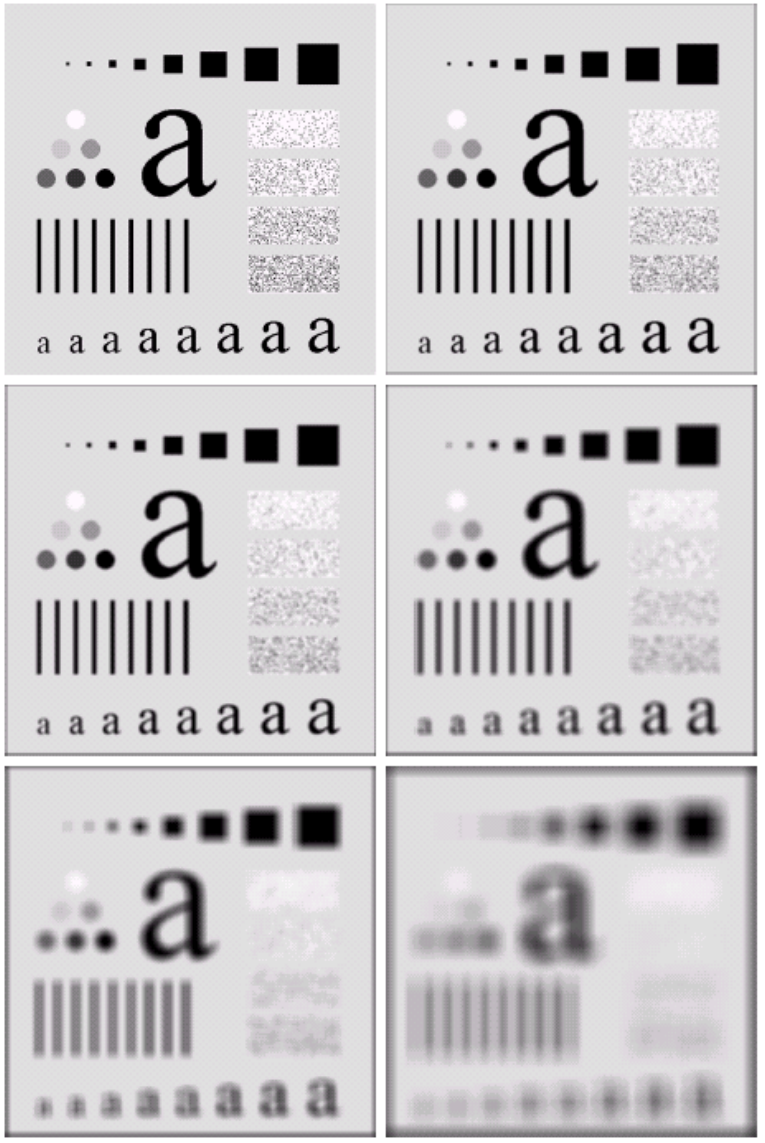
$\frac{1}{16} \times$

1	2	1
2	4	2
1	2	1

Weighted Average

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

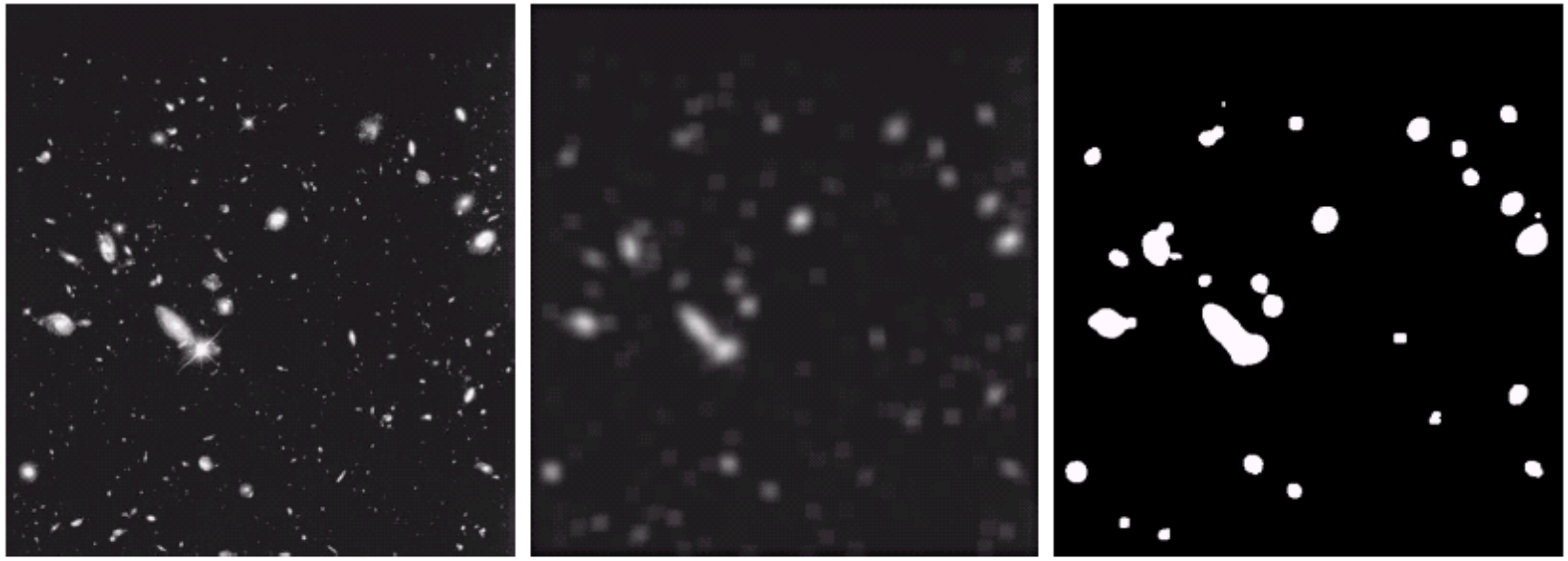
Linear Smoothing Filters – averaging filters



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.

Linear Smoothing Filters – averaging filters



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.)

Application: Averaging before Thresholding

Nonlinear Smoothing Filters – Order-Statistics Filters

Median Filter

-- the 50th percentile of a ranked set of numbers

-- effective for reducing *impulse noise,*
or *salt-and-pepper noise*

Max Filter

-- the 100th percentile filter

Min Filter

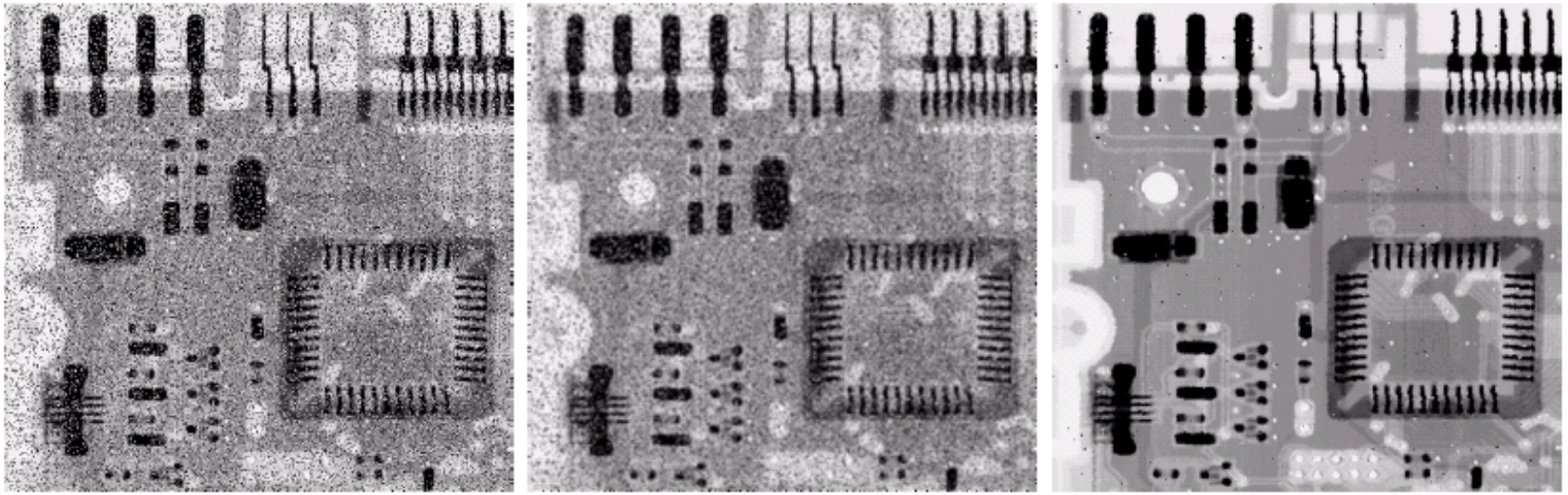
-- the 0th percentile filter

Nonlinear Smoothing Filters – Order-Statistics Filters

Salt-and-pepper noise image

3x3 averaging filter

3x3 median filter



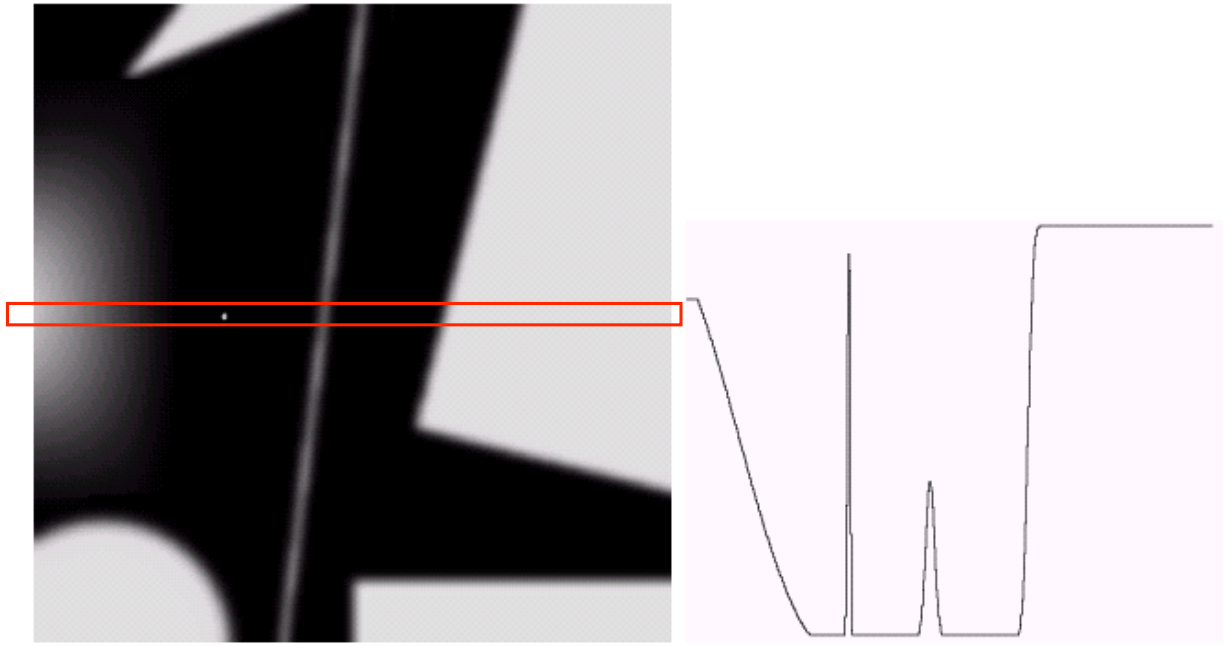
a b c

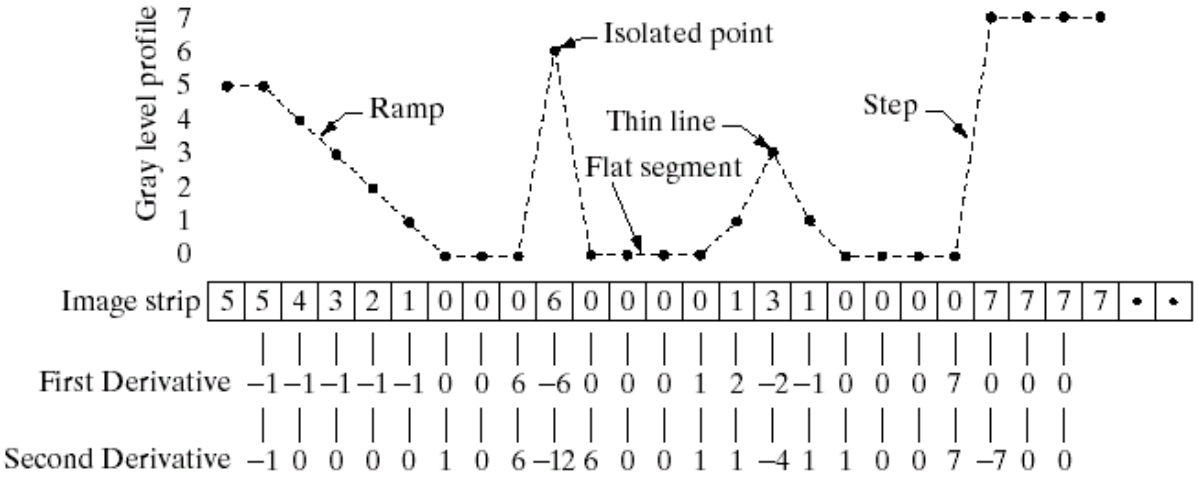
FIGURE 3.37 (a) X-ray image of circuit board corrupted by salt-and-pepper noise. (b) Noise reduction with a 3×3 averaging mask. (c) Noise reduction with a 3×3 median filter. (Original image courtesy of Mr. Joseph E. Pascente, Lixi, Inc.)

Edge detection and sharpening

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).





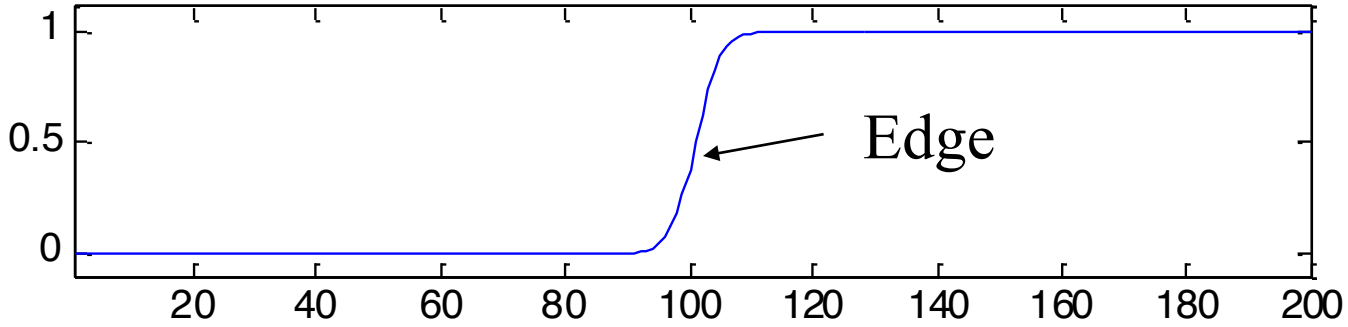
for edge detection

for sharpening

Edge detection

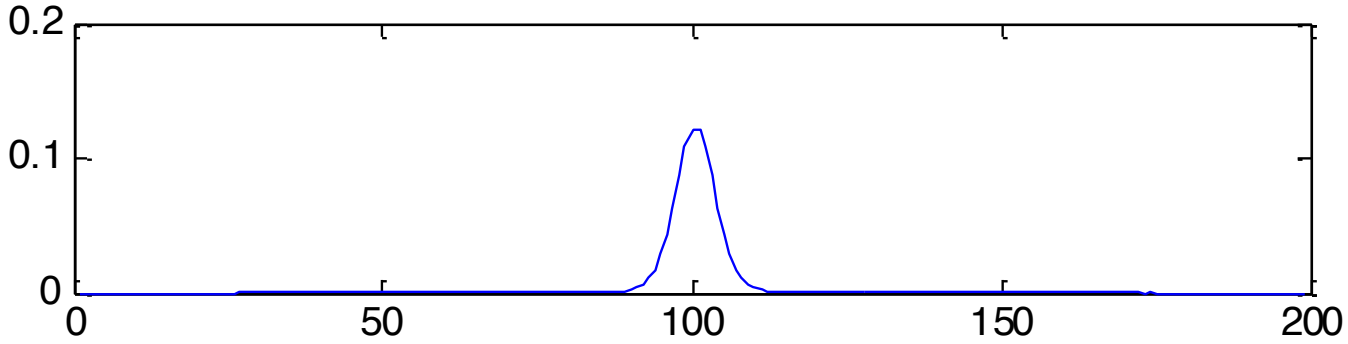
Intensity profile

$p(x)$



1st derivative

$\frac{dp}{dx}$



Edge detection

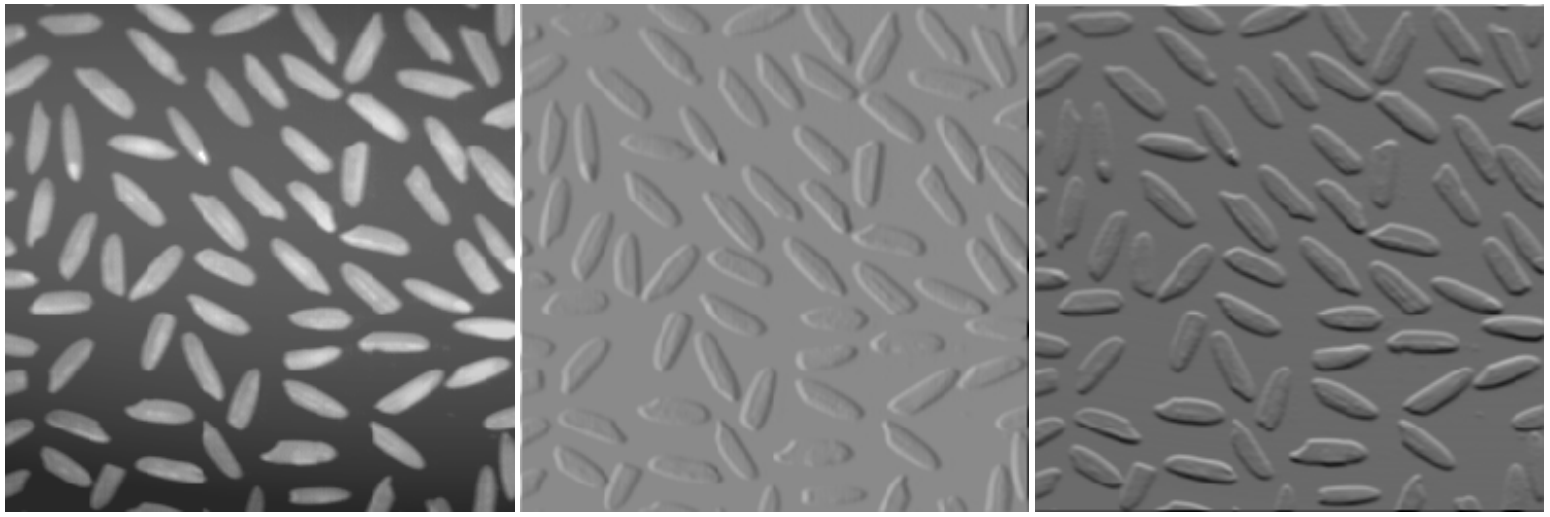
Sobel operators

-1	0	1
-2	0	2
-1	0	1

to compute $\frac{\partial P}{\partial x}$

-1	-2	-1
0	0	0
1	2	1

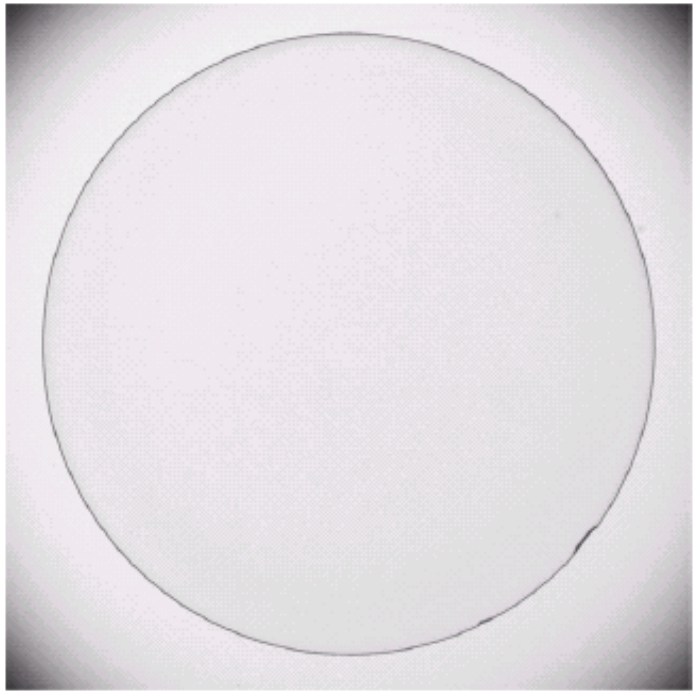
to compute $\frac{\partial P}{\partial y}$

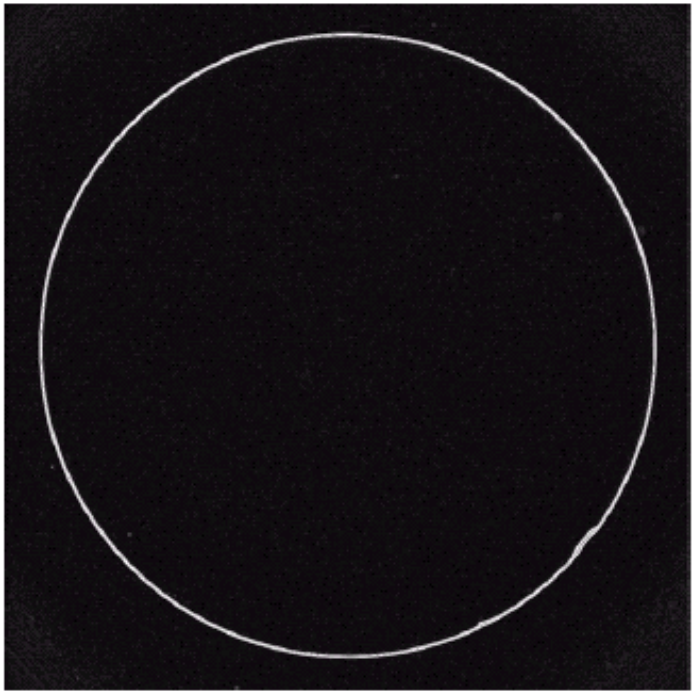


P $\frac{\partial P}{\partial x}$ $\frac{\partial P}{\partial y}$

Gradient magnitude

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

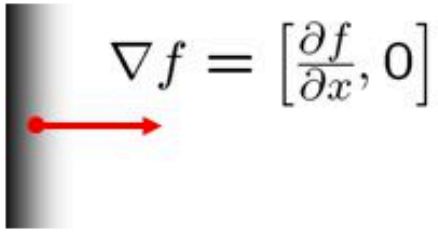




a b

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.)

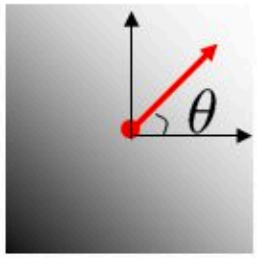
Gradient direction



$$\nabla f = \left[\frac{\partial f}{\partial x}, 0\right]$$

$$\nabla f = \left[0, \frac{\partial f}{\partial y}\right]$$

$$\nabla f = \left[\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}\right]$$

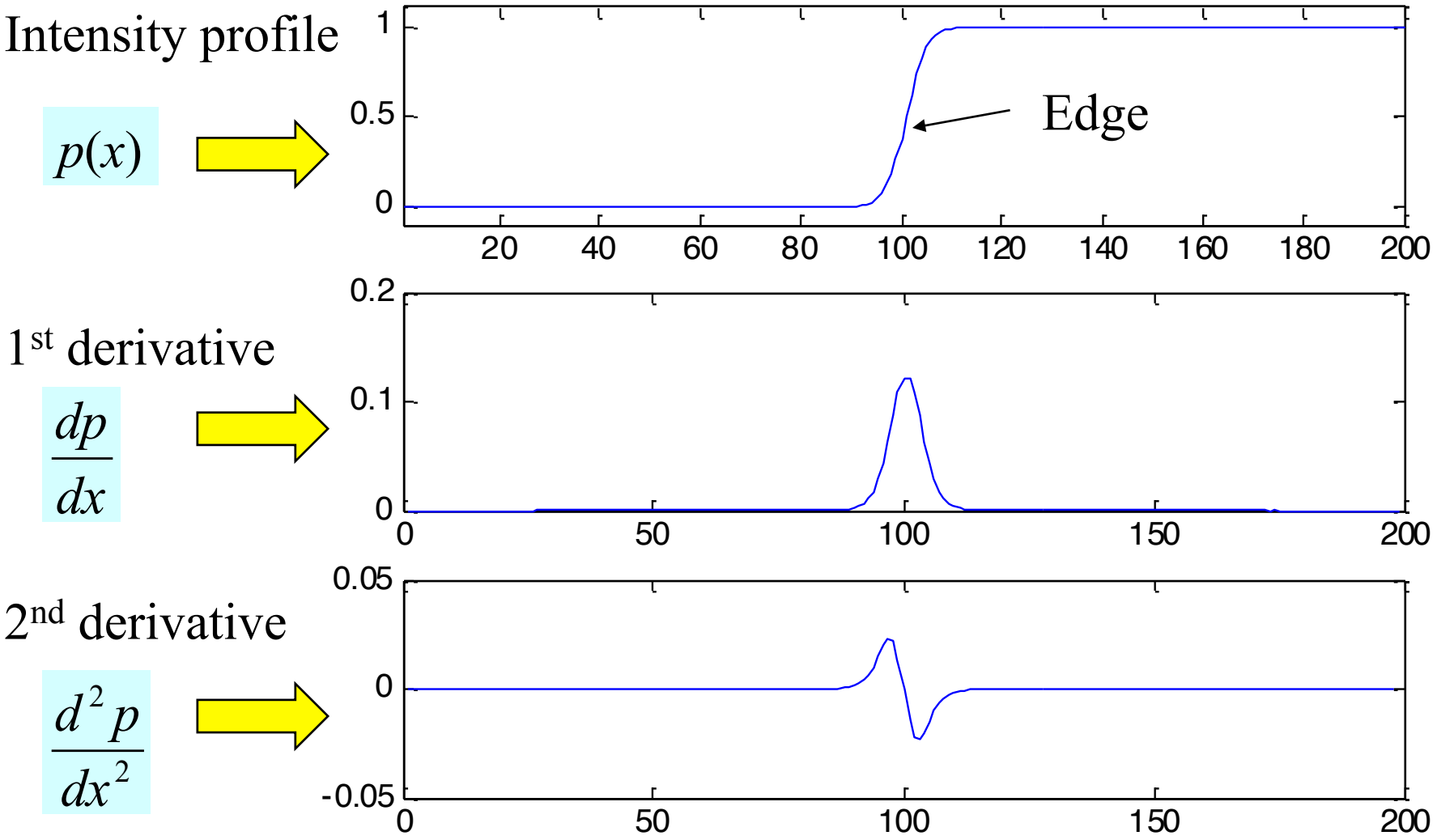


$$\nabla f = \left[\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}\right]$$

The gradient direction is given by:

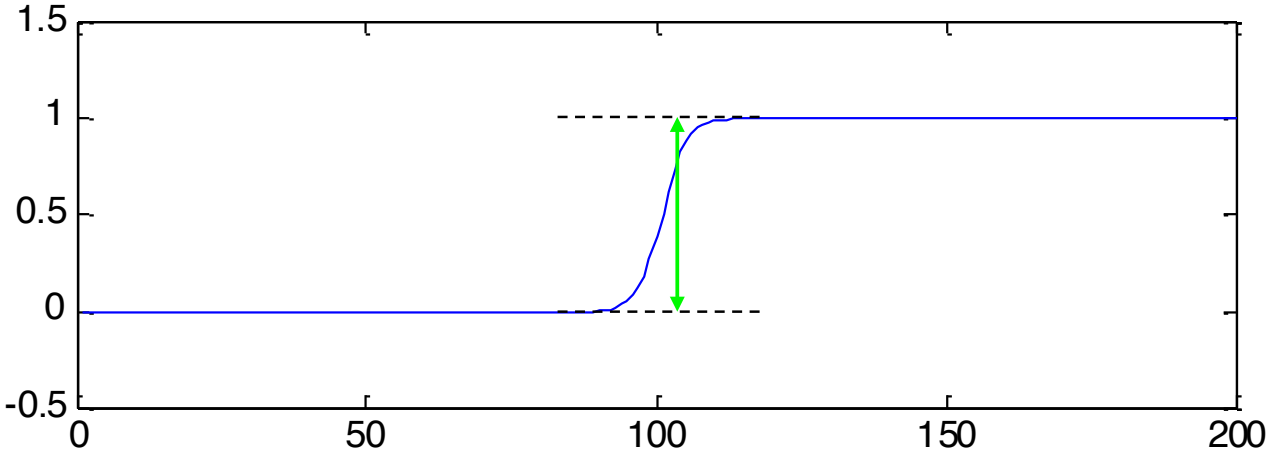
$$\theta = \tan^{-1}\left(\frac{\partial f}{\partial y} / \frac{\partial f}{\partial x}\right)$$

Laplacian sharpening

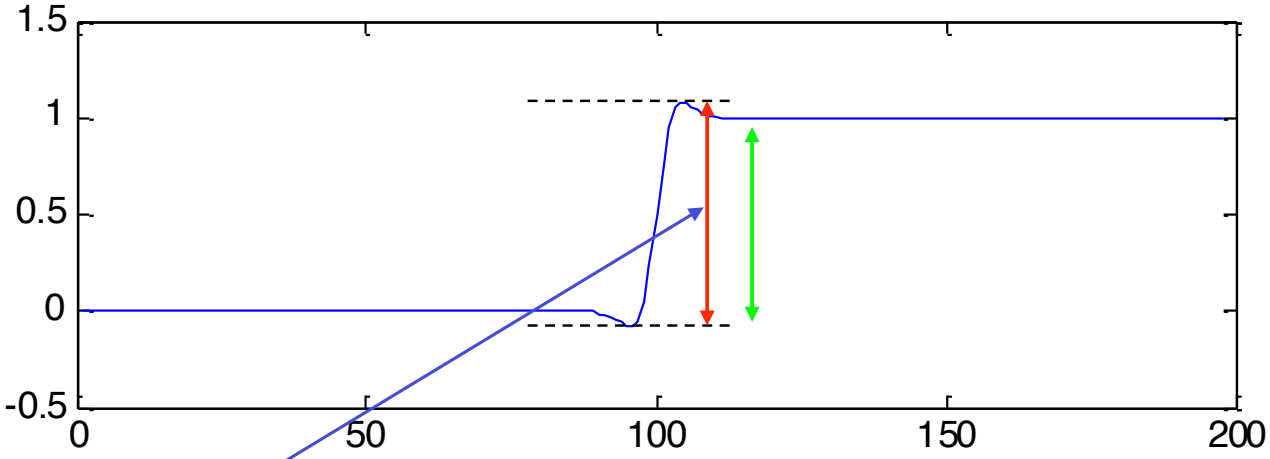


Laplacian sharpening

$p(x)$

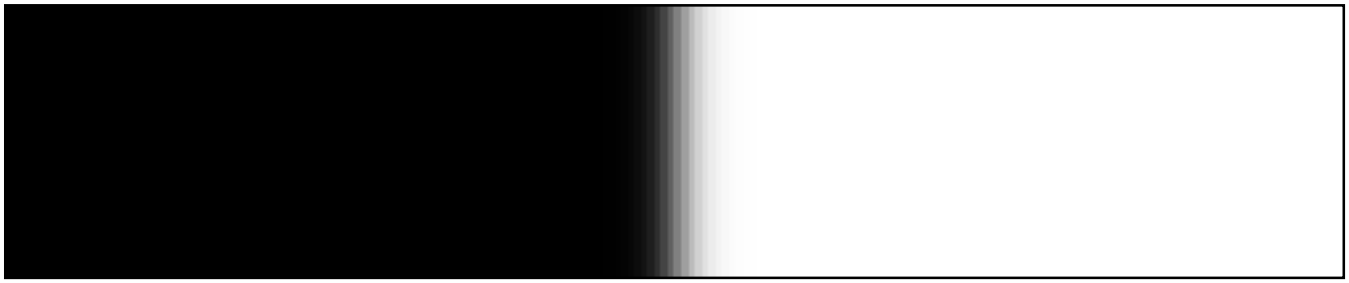


$$p(x) - 10\frac{d^2 p}{dx^2}$$



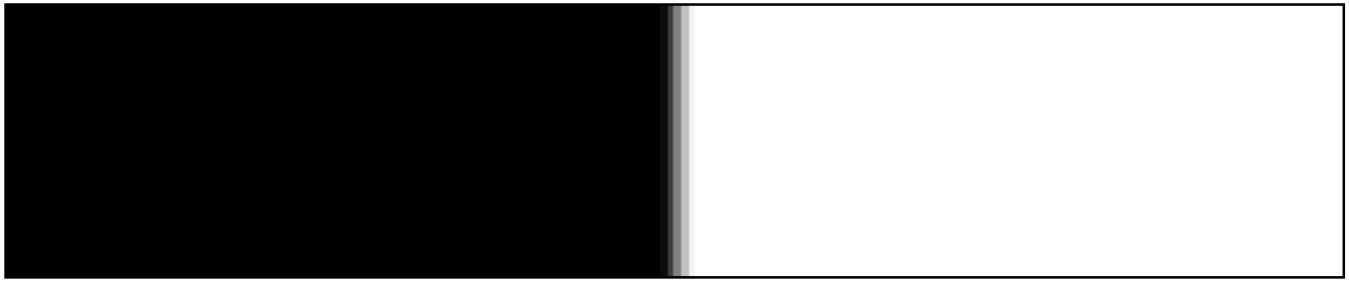
Laplacian sharpening results in larger intensity discontinuity near the edge.

Laplacian sharpening



Before sharpening

$$p(x)$$



After sharpening

$$p(x) - 10\frac{d^2 p}{dx^2}$$

Laplacian sharpening

Used for estimating image Laplacian $\nabla^2 P = \frac{\partial^2 P}{\partial x^2} + \frac{\partial^2 P}{\partial y^2}$

-1	-1	-1
-1	8	-1
-1	-1	-1

0	-1	0
-1	4	-1
0	-1	0

→ The center of the mask is positive

or

1	1	1
1	-8	1
1	1	1

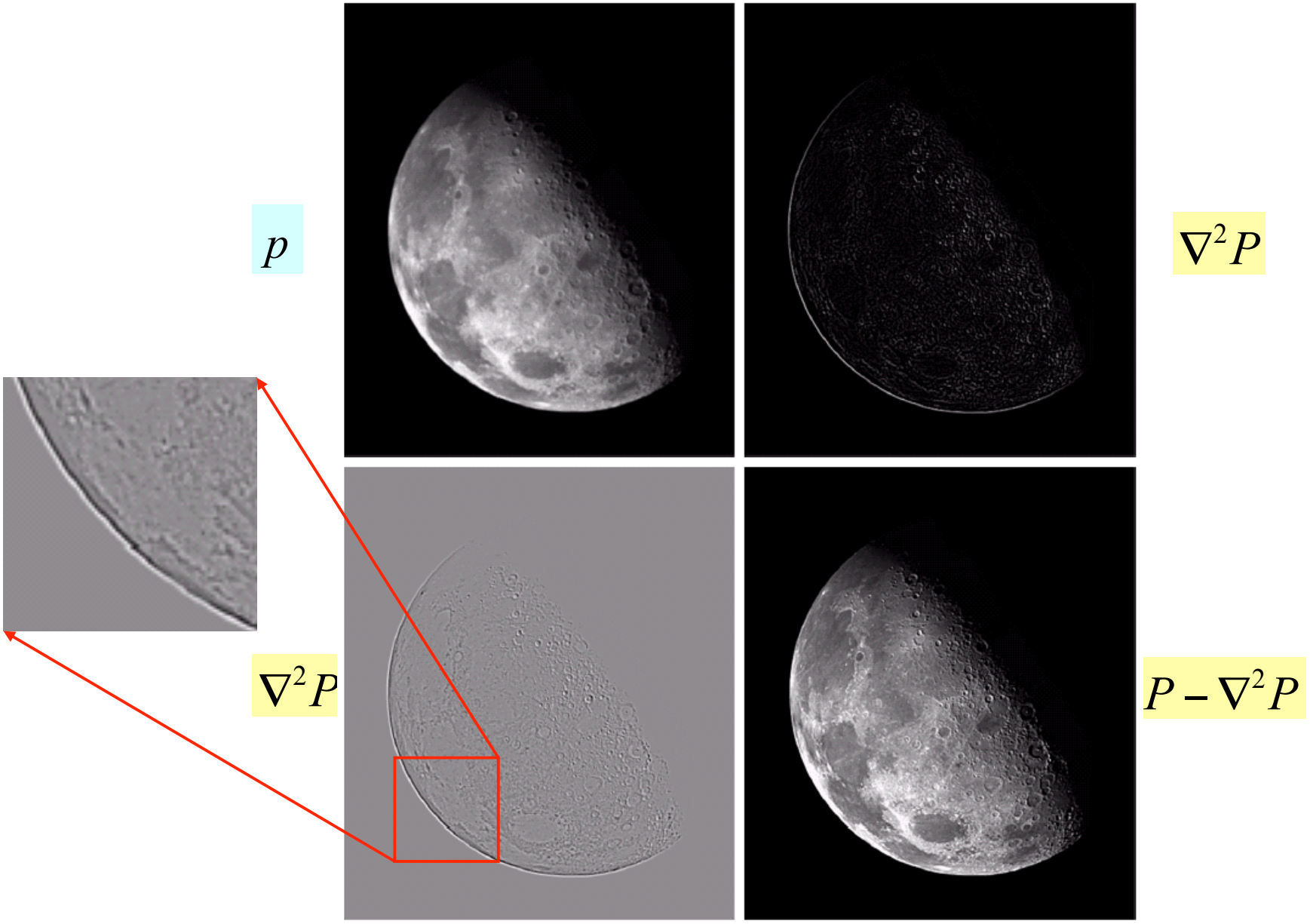
0	1	0
1	-4	1
0	1	0

→ The center of the mask is negative

Application: Enhance edge, line, point

Disadvantage: Enhance noise

Laplacian sharpening



Laplacian sharpening

Mask for $\nabla^2 P$

1	1	1
1	-8	1
1	1	1

or

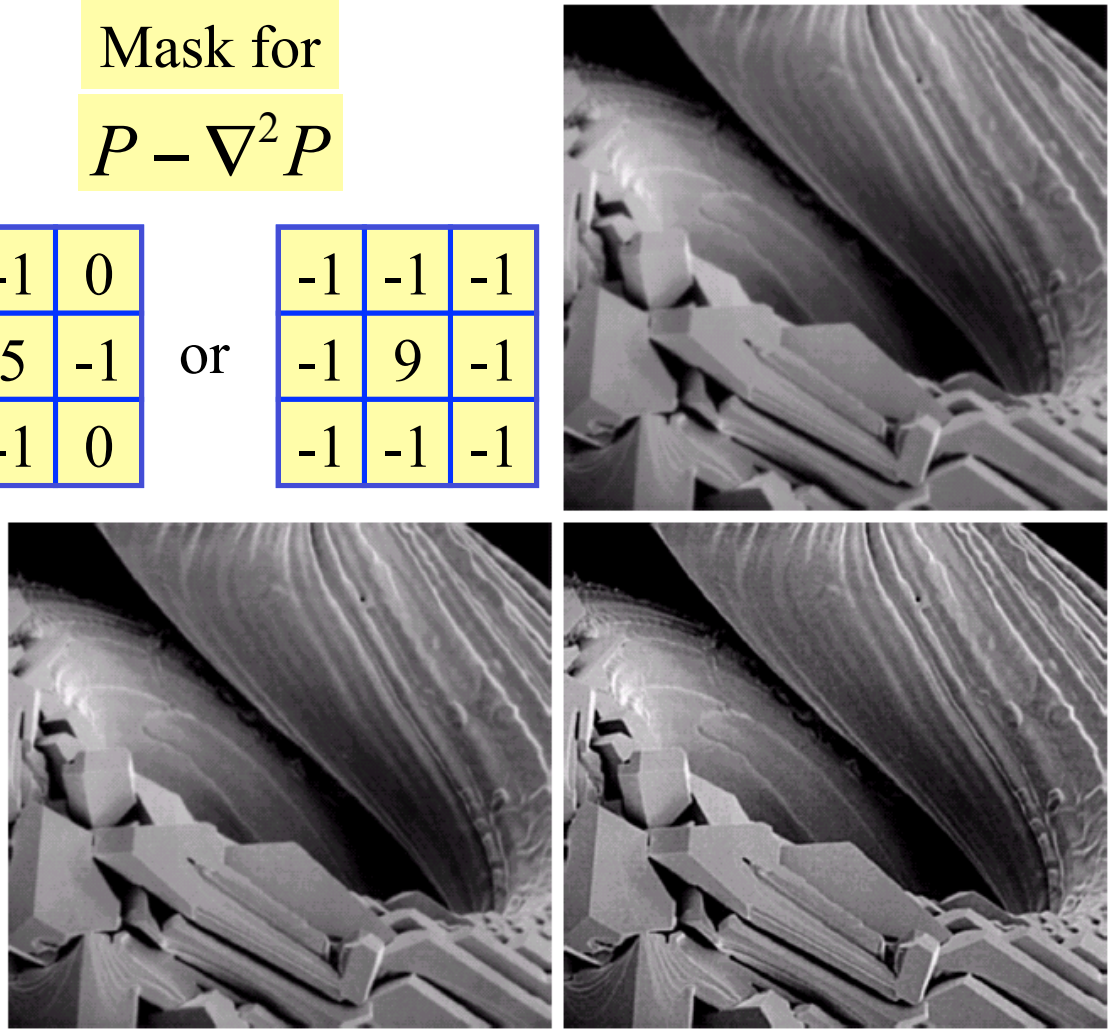
0	1	0
1	-4	1
0	1	0

Mask for $P - \nabla^2 P$

0	-1	0
-1	5	-1
0	-1	0

or

-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.)

(Images from Rafael C. Gonzalez and Richard E. Wood, Digital Image Processing, 2nd Edition.

Digital Image Processing

Kuan-Wen Chen
2022/9/20