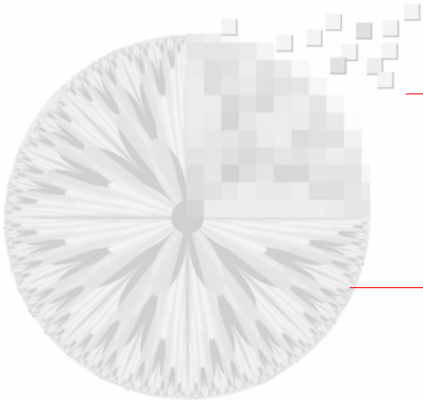


# Digital Image Fundamentals and Image Enhancement in the Spatial Domain

Mohamed N. Ahmed, Ph.D.



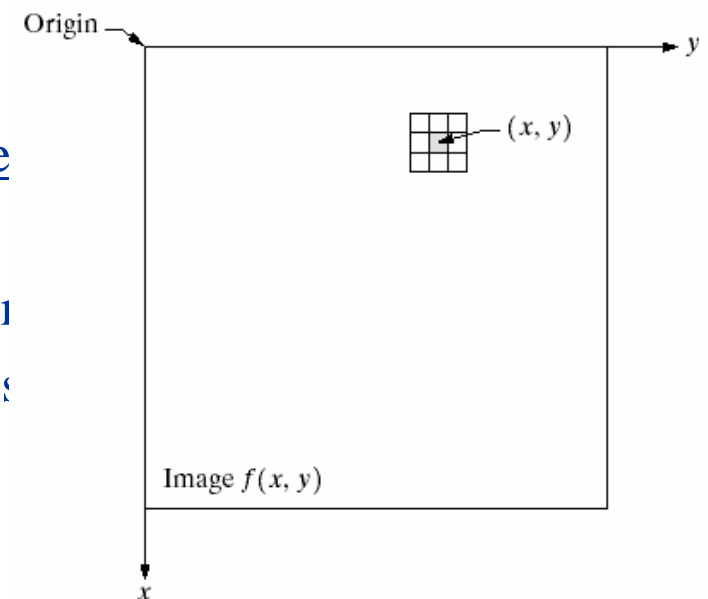
# Introduction

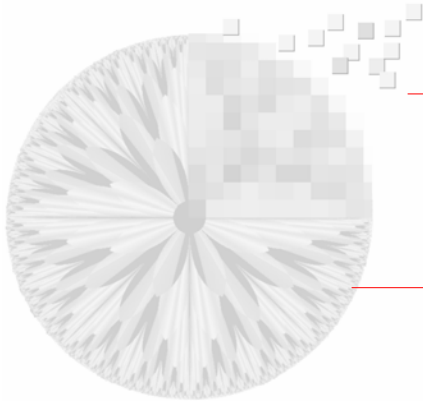
- An image may be defined as 2D function  $f(x,y)$ , where  $x$  and  $y$  are spatial coordinates.
- The amplitude of  $f$  at any pair  $(x,y)$  is called the intensity at that point.



When  $x$ ,  $y$ , and  $f$  are all finite, discrete quantities, we call the image a digital image

So, a digital image is composed of finite number of elements called picture elements or pixels





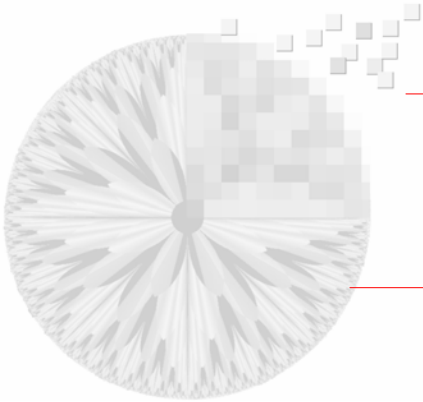
# Introduction

- The field of image processing is related to two other fields:  
*image analysis* and *computer vision*

Image Processing

Computer Vision





# Introduction

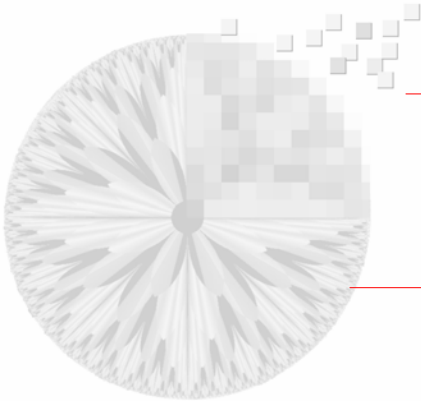
- There are three of processes in the continuum

- **Low Level Processes**

- » Preprocessing, filtering, enhancement

- » sharpening





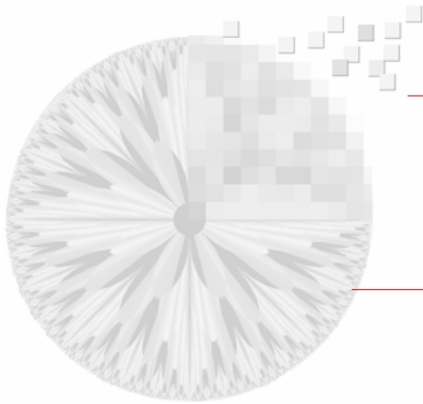
# Introduction

- There are three of processes in the continuum
  - Low Level Processes
    - » Preprocessing, filtering, enhancement
    - » sharpening



- **Mid Level Processes**
  - » segmentation





# Introduction

- There are three of processes in the continuum

- Low Level Processes

- » Preprocessing, filtering, enhancement

- » sharpening



- Mid Level Processes

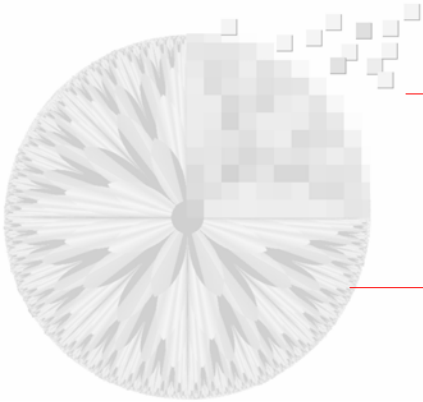
- » segmentation



- **High Level Processes**

- » Recognition





# Origins of DIP

- Newspaper Industry: pictures were sent by Bartlane cable picture between London and New York in early 1920.

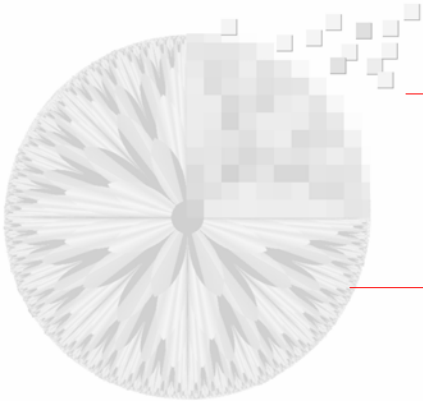
The introduction of the Bartlane Cable reduced the transmission time from a week to three hours

Specialized printing equipment coded pictures for transmission and then reconstructed them at the receiving end.

Visual Quality problems



1921

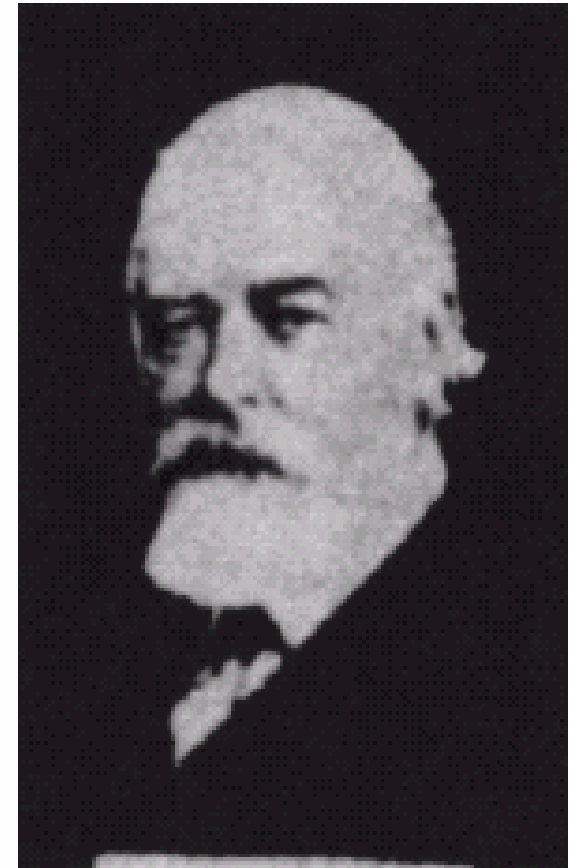


# Origins of DIP

In 1922, a technique based on photographic reproduction made from tapes perforated at the telegraph receiving terminal was used.

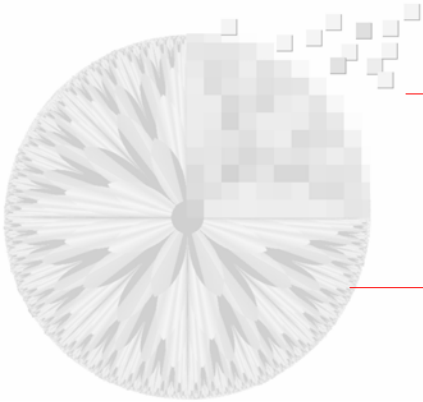
This method had better tonal quality and Resolution

Had only five gray levels



1922

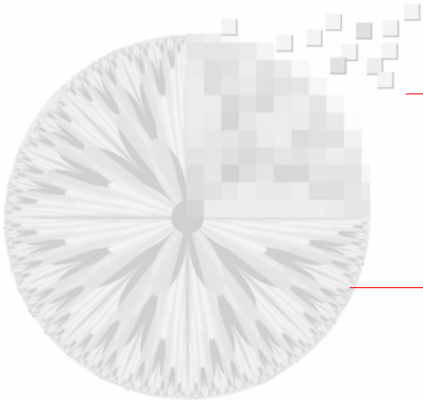




# Origins of DIP

Unretouched cable picture of Generals  
Pershing and Foch transmitted  
Between London and New York in 1929  
Using 15-tone equipment



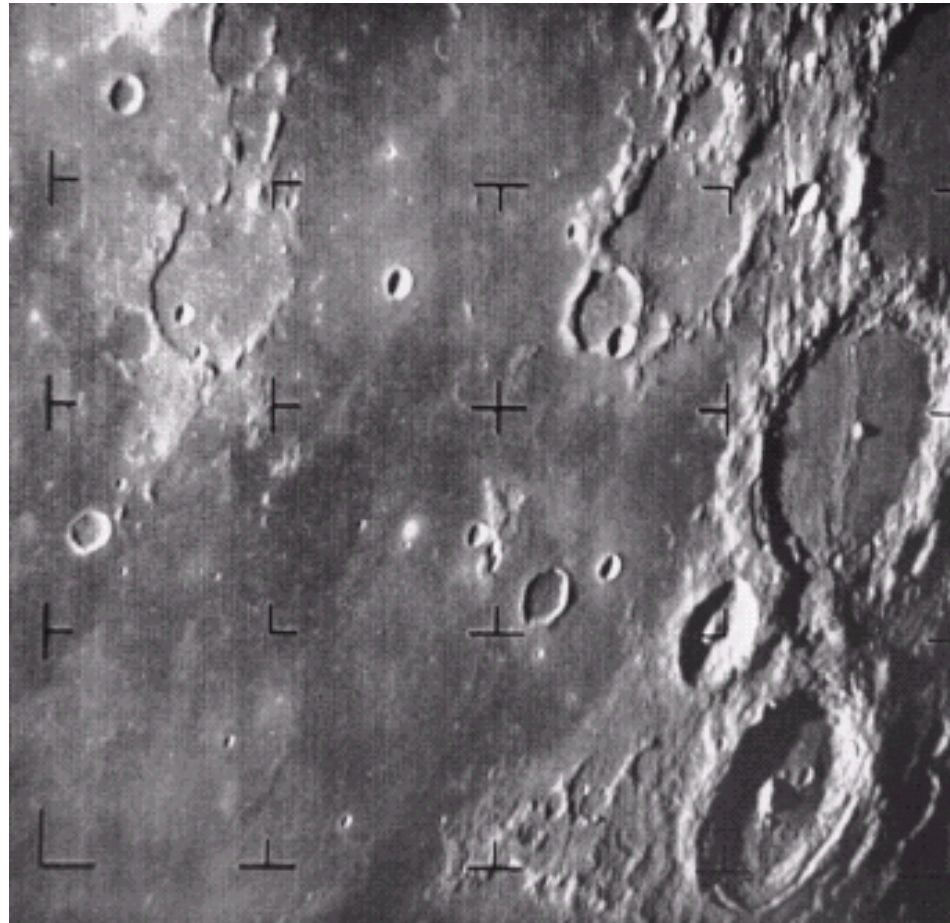


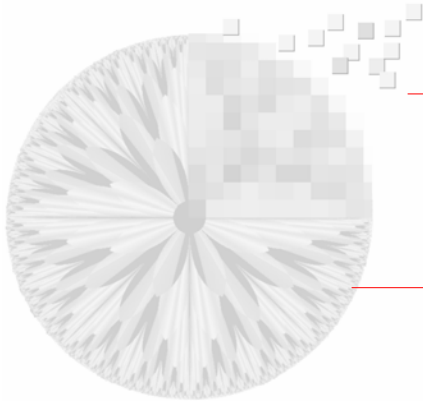
# Origins of DIP

The first picture of the moon by a US Spacecraft.

Ranger 7 took this image  
On July 31st in 1964.

This saw the first use of a digital  
computer to correct for various types  
of image distortions inherent in the  
on-board television camera





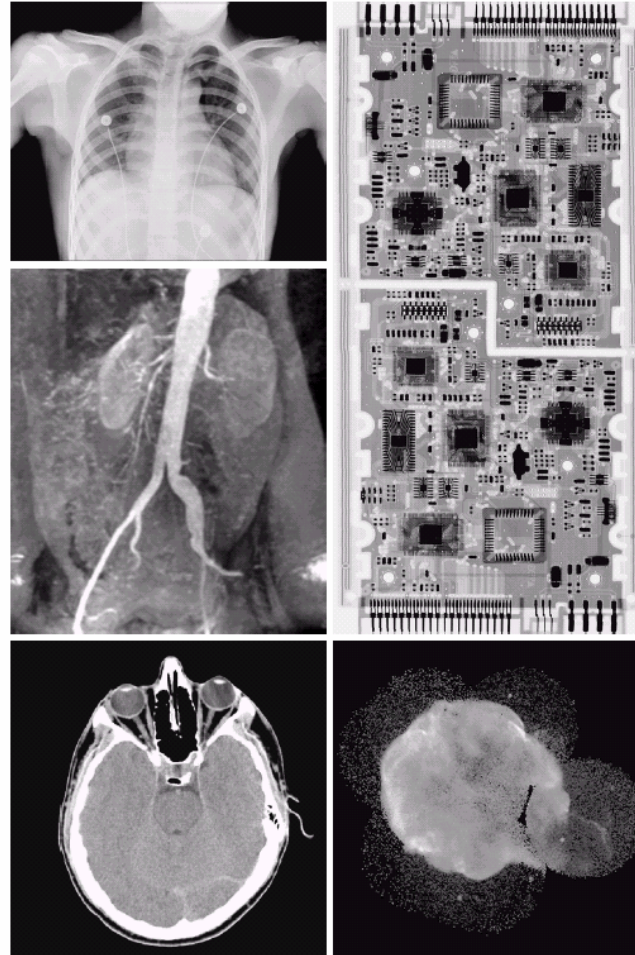
# Applications

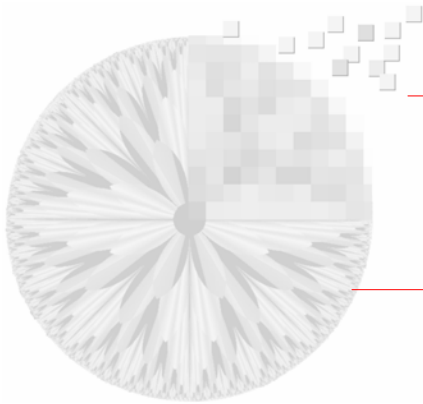
- X-ray Imaging

X-rays are among the oldest sources of EM radiation used for imaging

Main usage is in medical imaging (X-rays, CAT scans, angiography)

The figure shows some of the applications of X-ray imaging

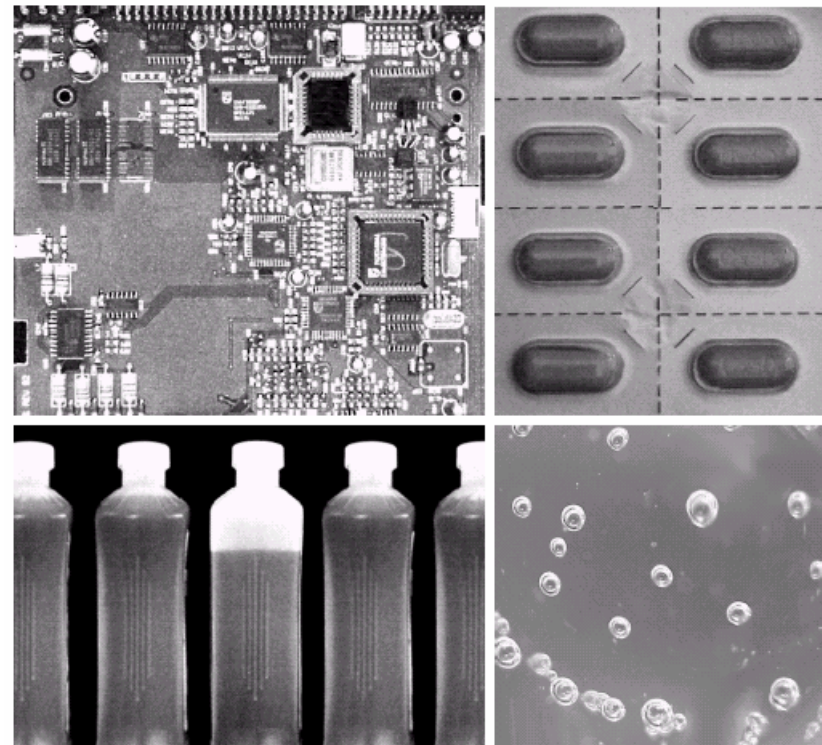




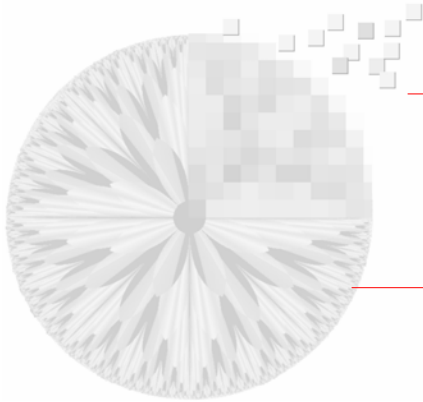
# Applications

- **Inspection Systems**

Some examples of manufactured goods often checked using digital image processing





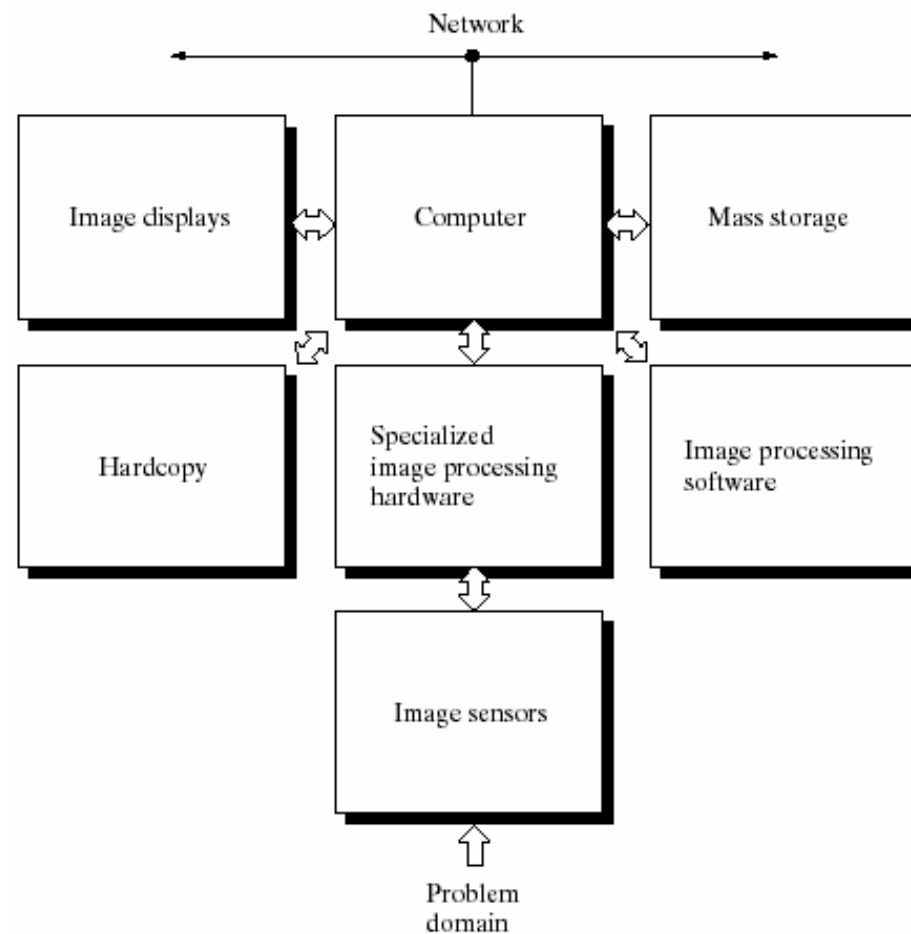


# Applications

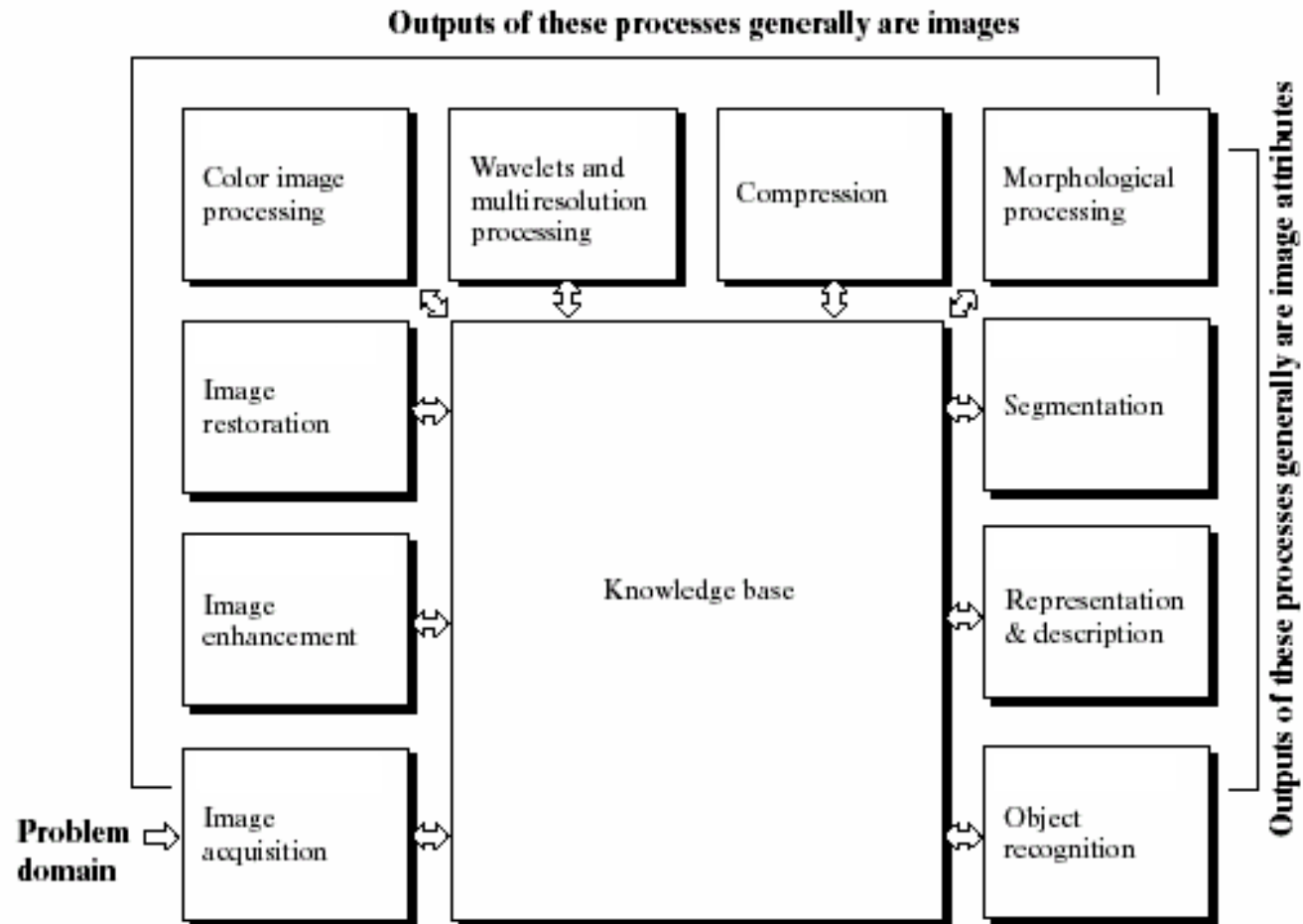
- Finger Prints
- Counterfeiting
- License Plate Reading

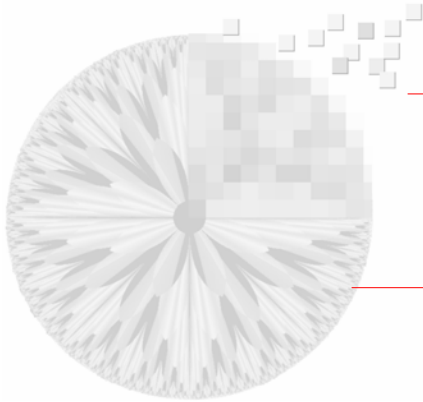


# Components of an Image Processing System



# Steps in Digital Image Processing





## 2. Digital Image Fundamentals



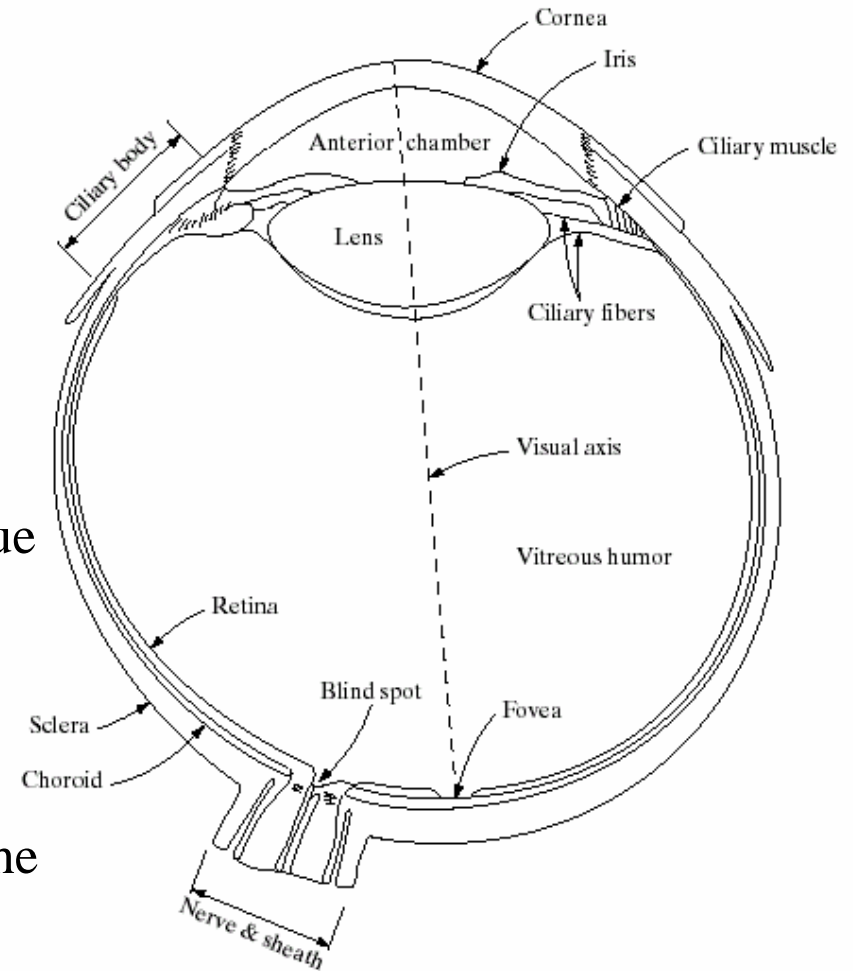
# Structure of the Human Eye

The eye is nearly a sphere with an Average diameter of 20mm

Three membranes enclose the eye:  
Cornea/Sclera, choroid, and retina.

The Cornea is a tough transparent tissue  
Covering the anterior part of the eye  
Sclera is an opaque membrane that  
Covers the rest of the eye

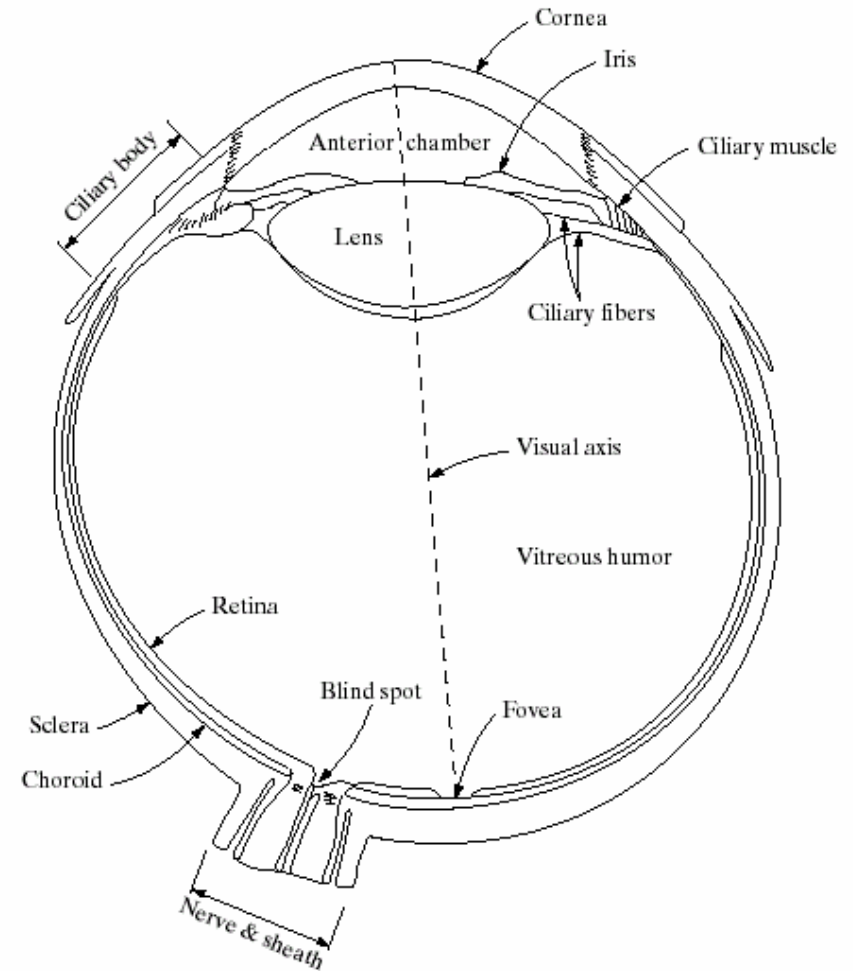
The Choroid has the blood supply to the  
eye



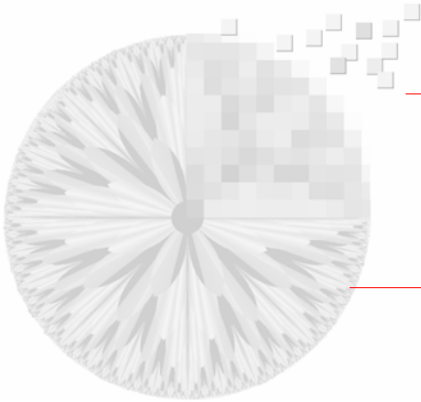
University of Louisville

# Structure of the Human Eye

- Continuous with the choroid is the iris which contracts or expands to control the amount of light entering the eye
- The lens contains 60 to 70 % water, 6% fat, and protein.
- The lens is colored slightly yellow that increases with age
- The Lens absorbs 8% of the visible light. The lens also absorbs high amount of infrared and ultra violet of which excessive amounts can damage the eye

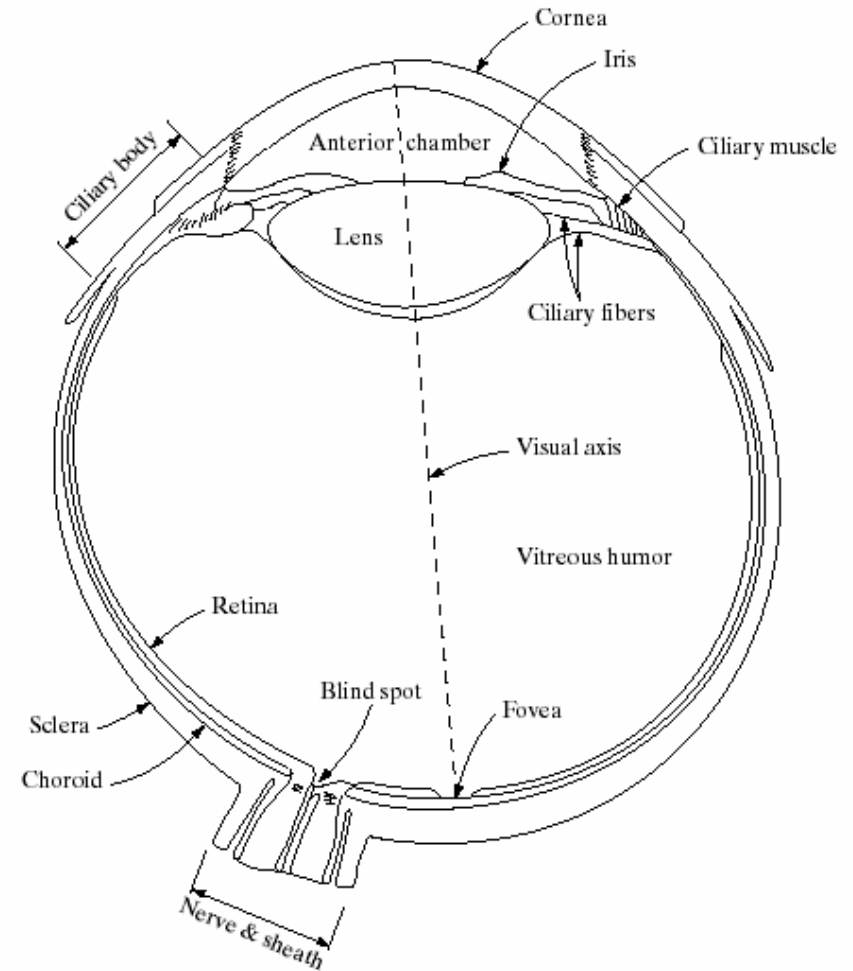


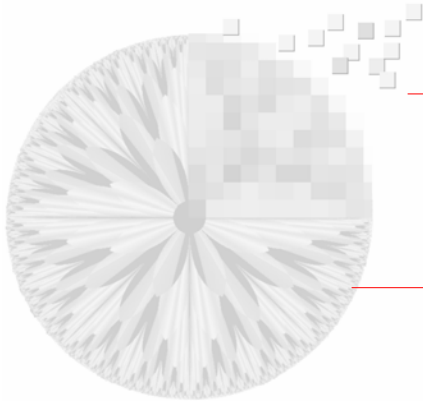
University of Louisville



# The Retina

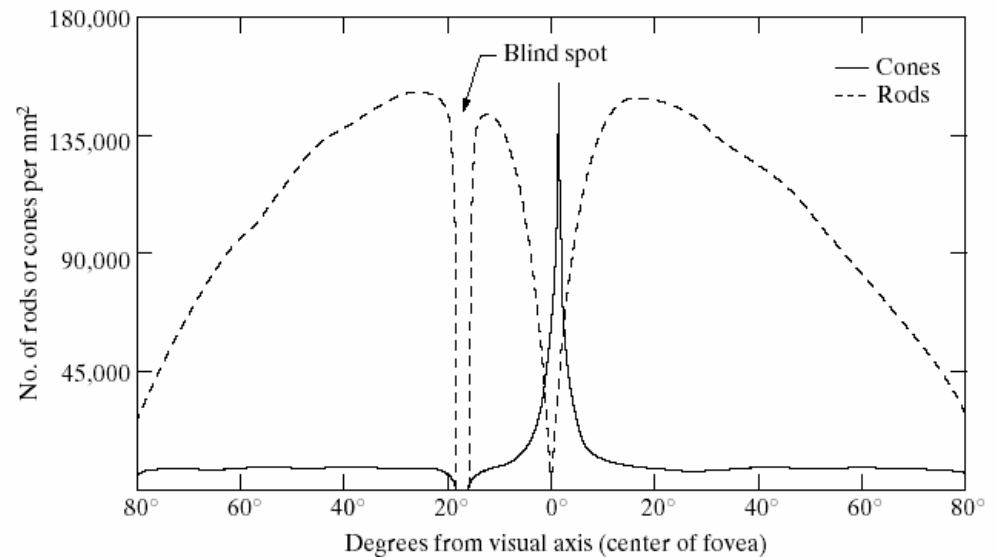
- The innermost membrane is the retina
- When light is properly focused, the image of an outside object is imaged on the retina
- There are discrete light receptors that line the retina: cones and rods



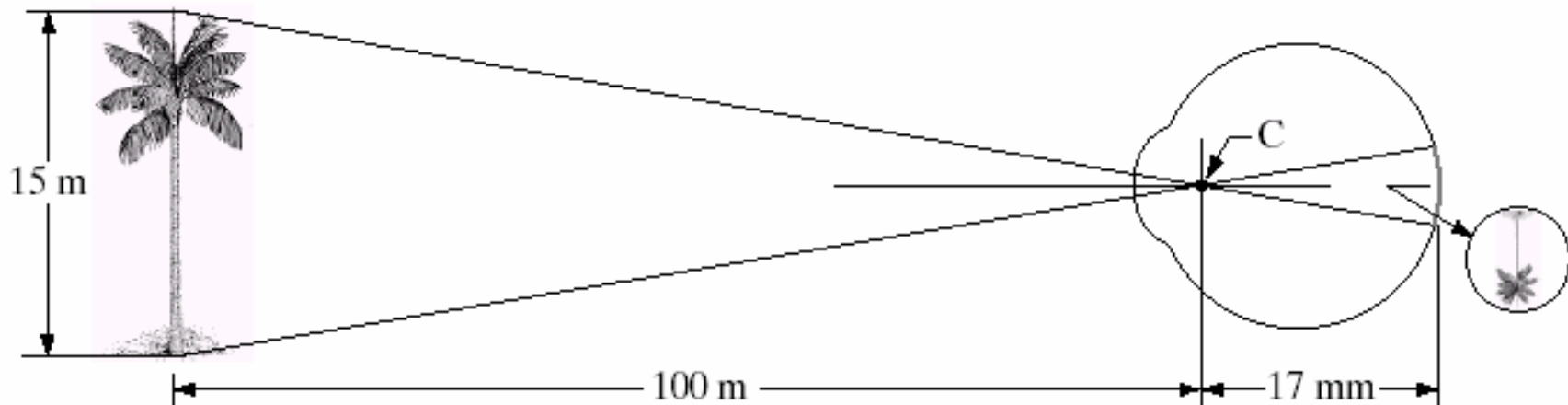


# Rods and Cones

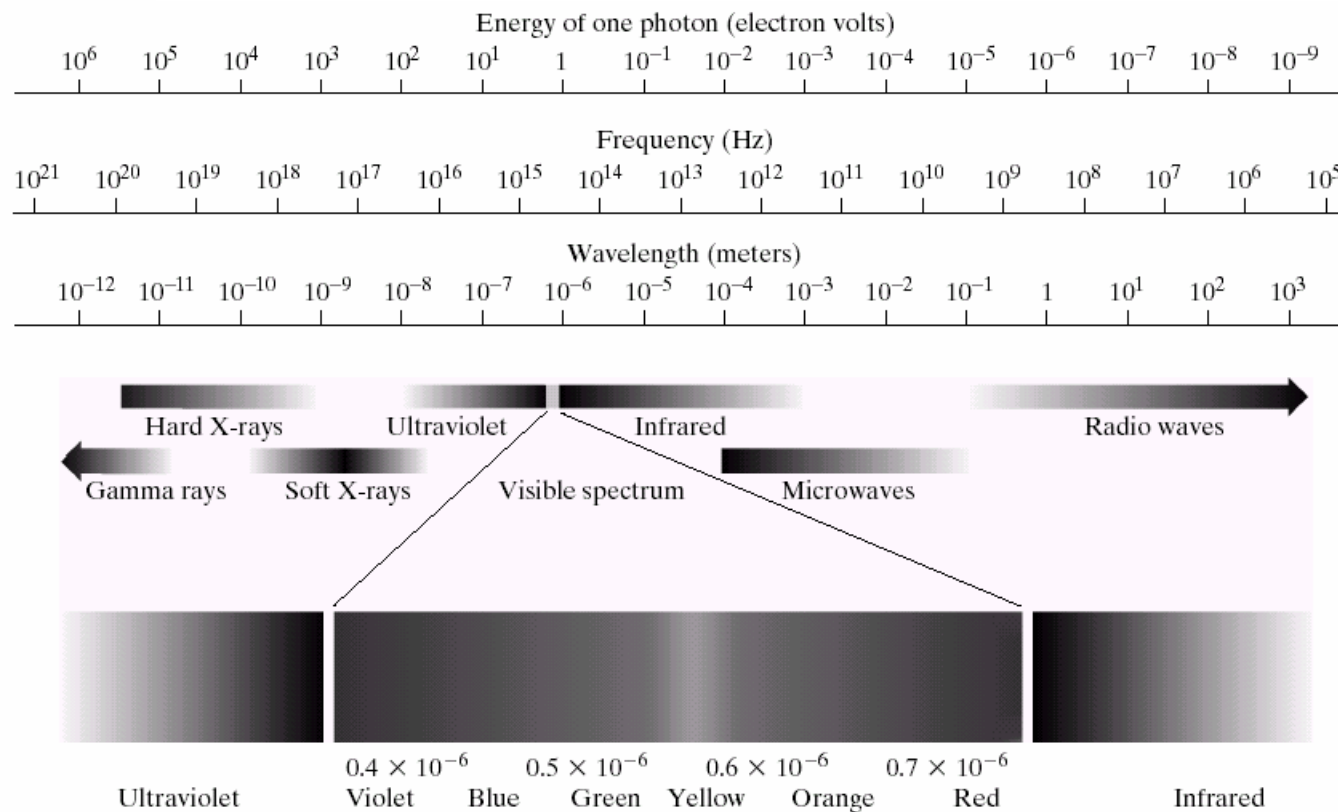
- The cones (7 million) are located in the central portion the retina (fovea). They are highly sensitive to color
- The rods are much larger (75-150 million). They are responsible for giving a general overall picture of the field of view. They are not involved in color vision



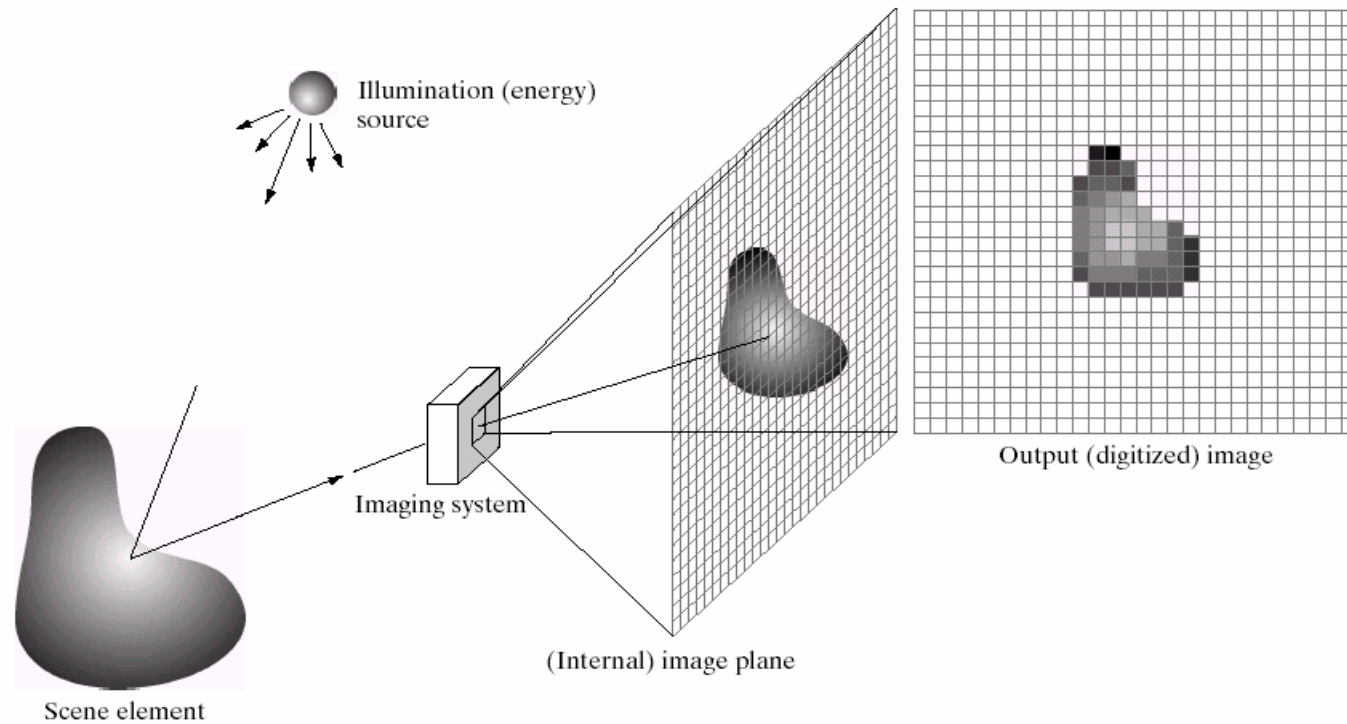
# Image Formation in the Eye

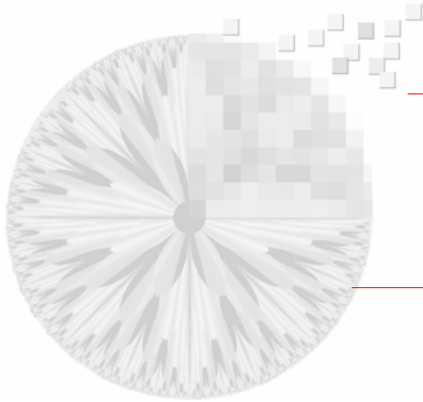


# Electromagnetic Spectrum



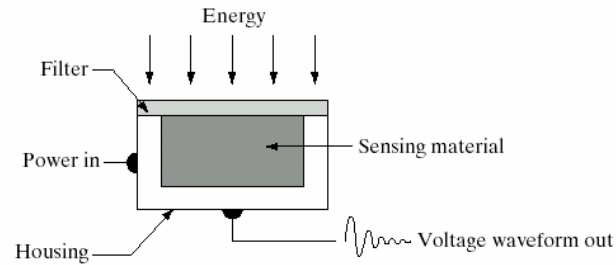
# Image Acquisition



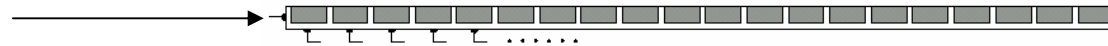


# Image Sensors

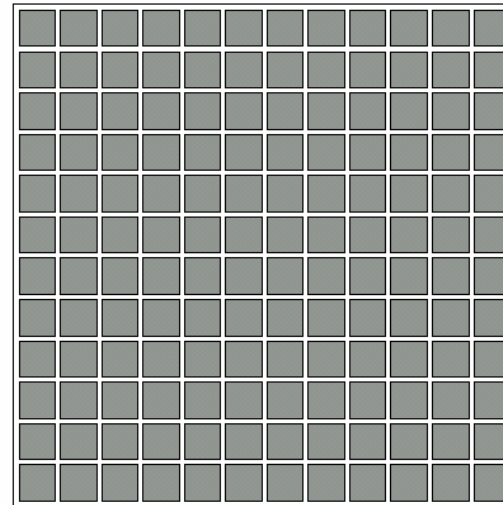
Single Imaging Sensor



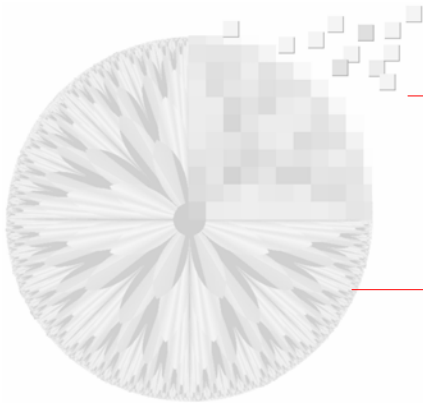
Line sensor



Array of Sensors







# Image Sensors

Single Imaging Sensor

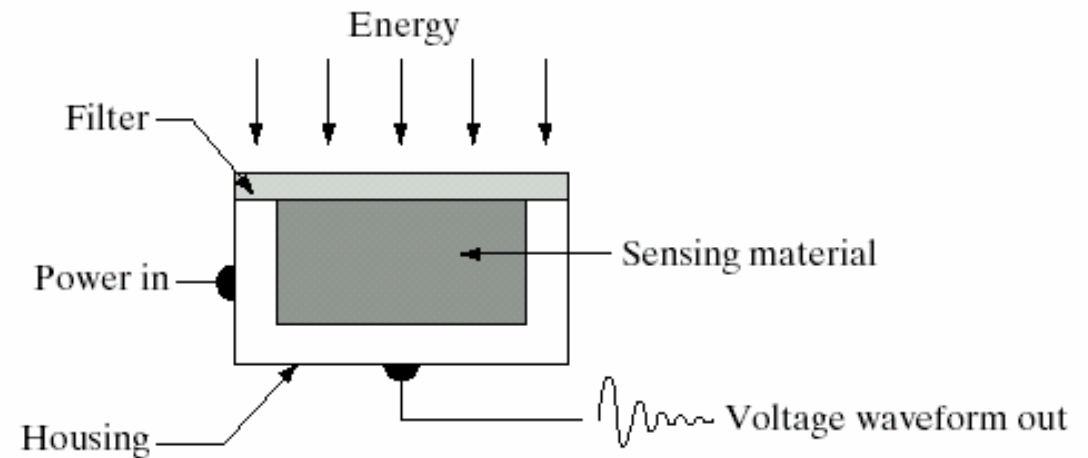
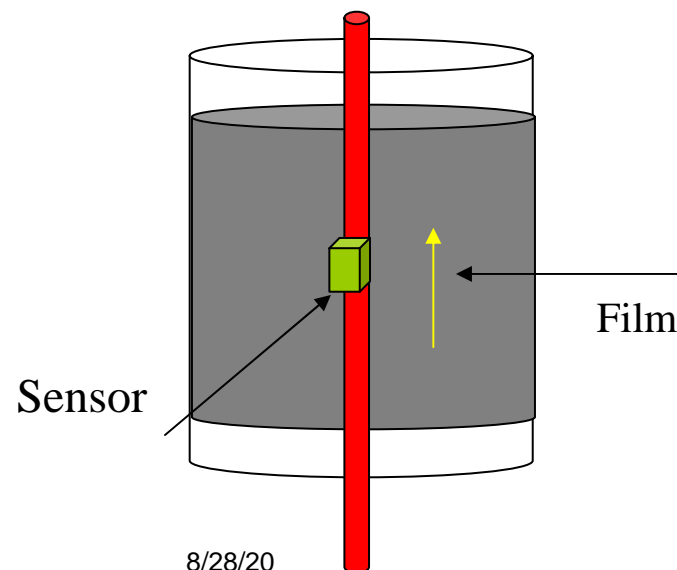
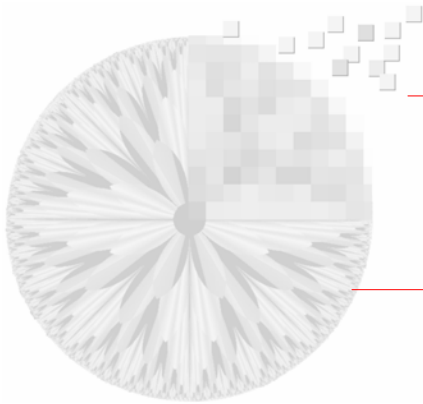


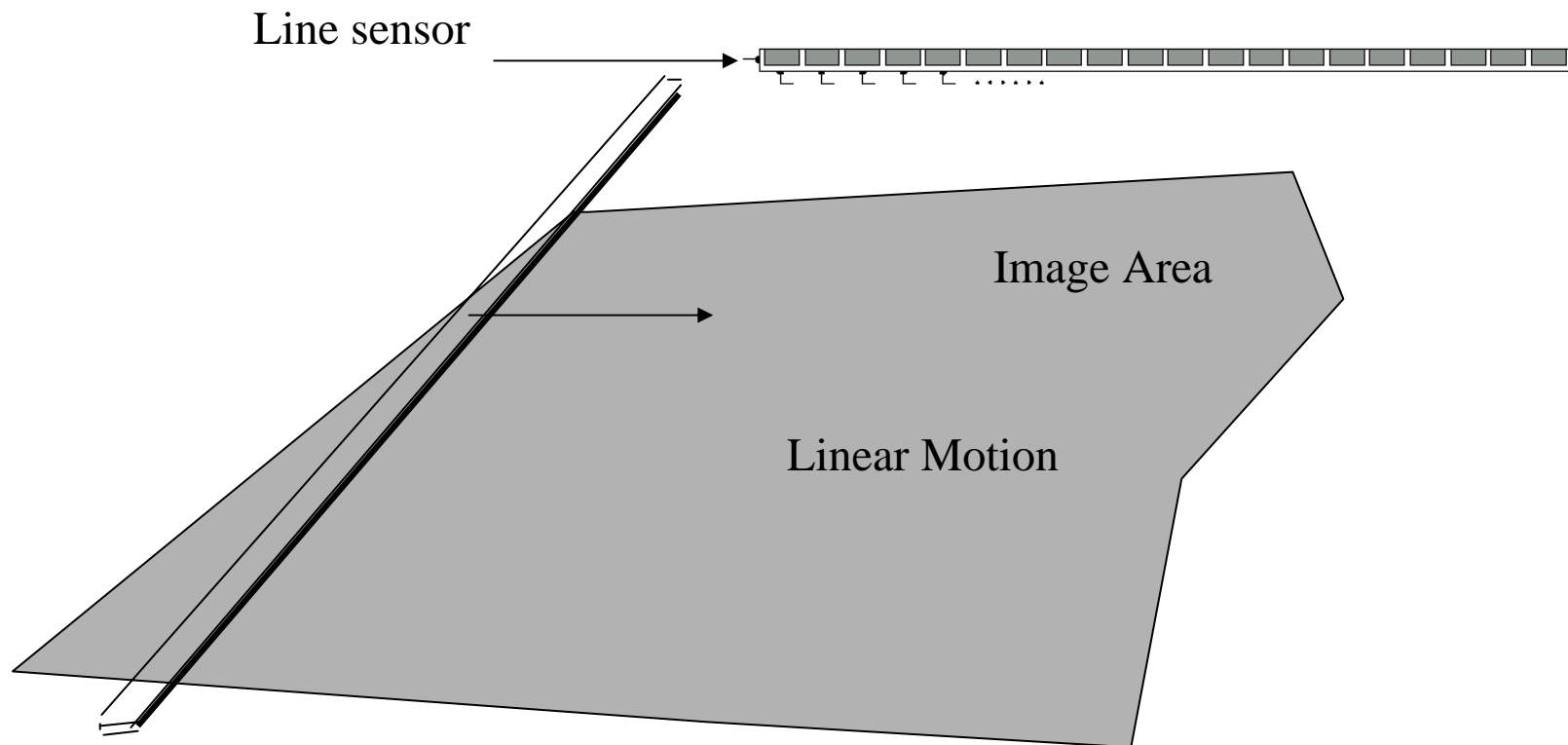
Photo Diode

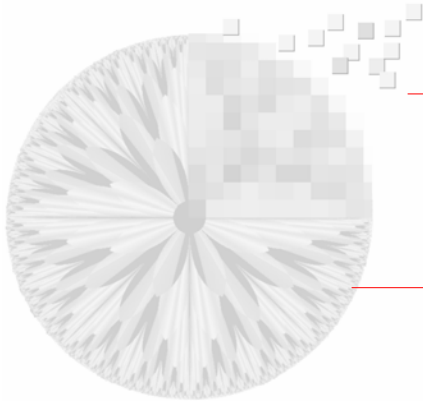


8/28/20  
04

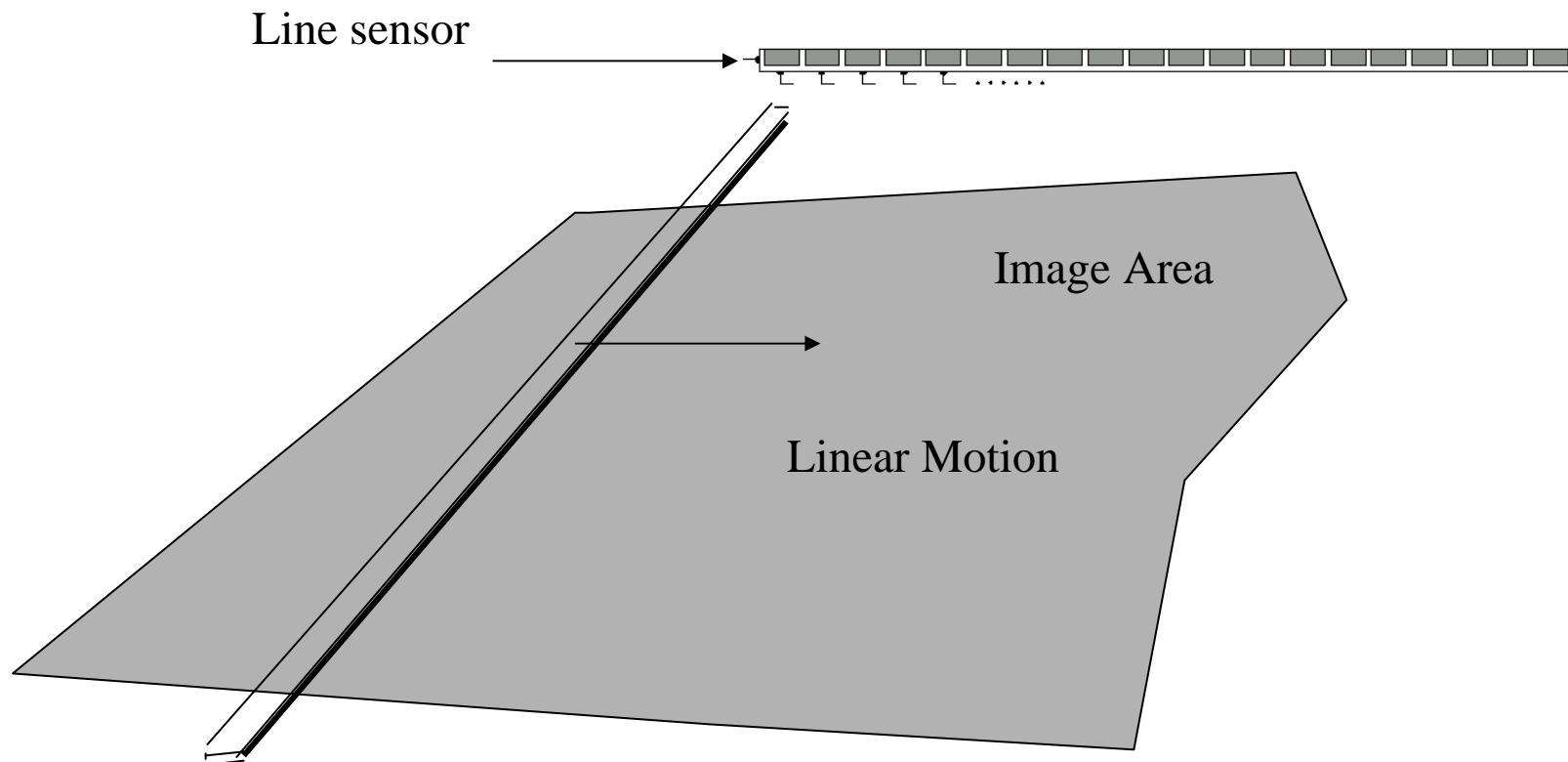


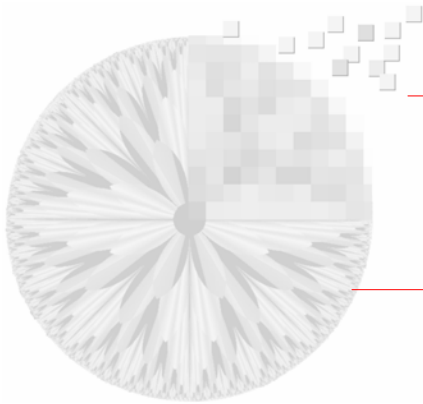
# Image Sensors





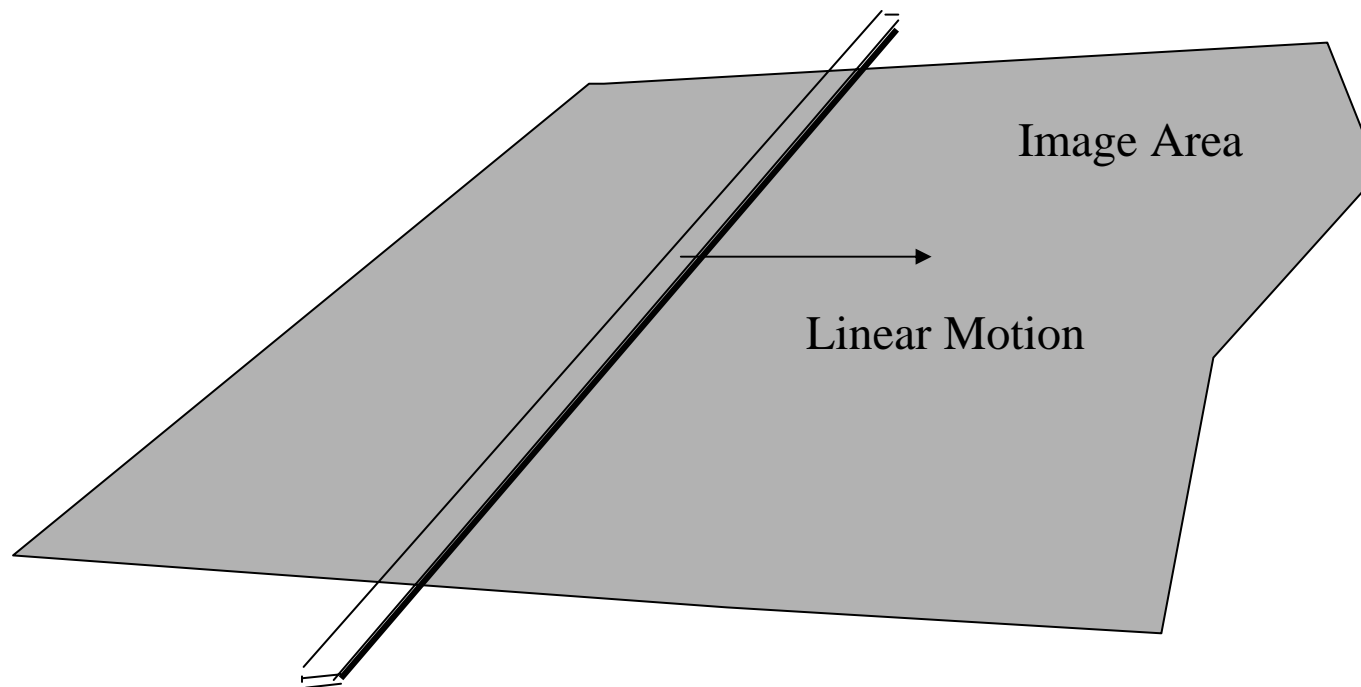
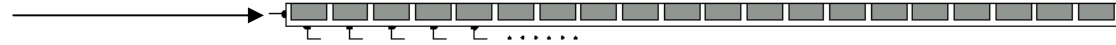
# Image Sensors

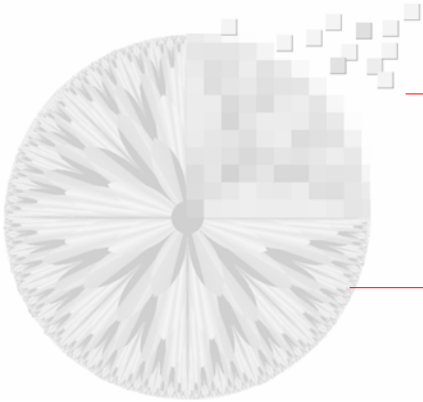




# Image Sensors

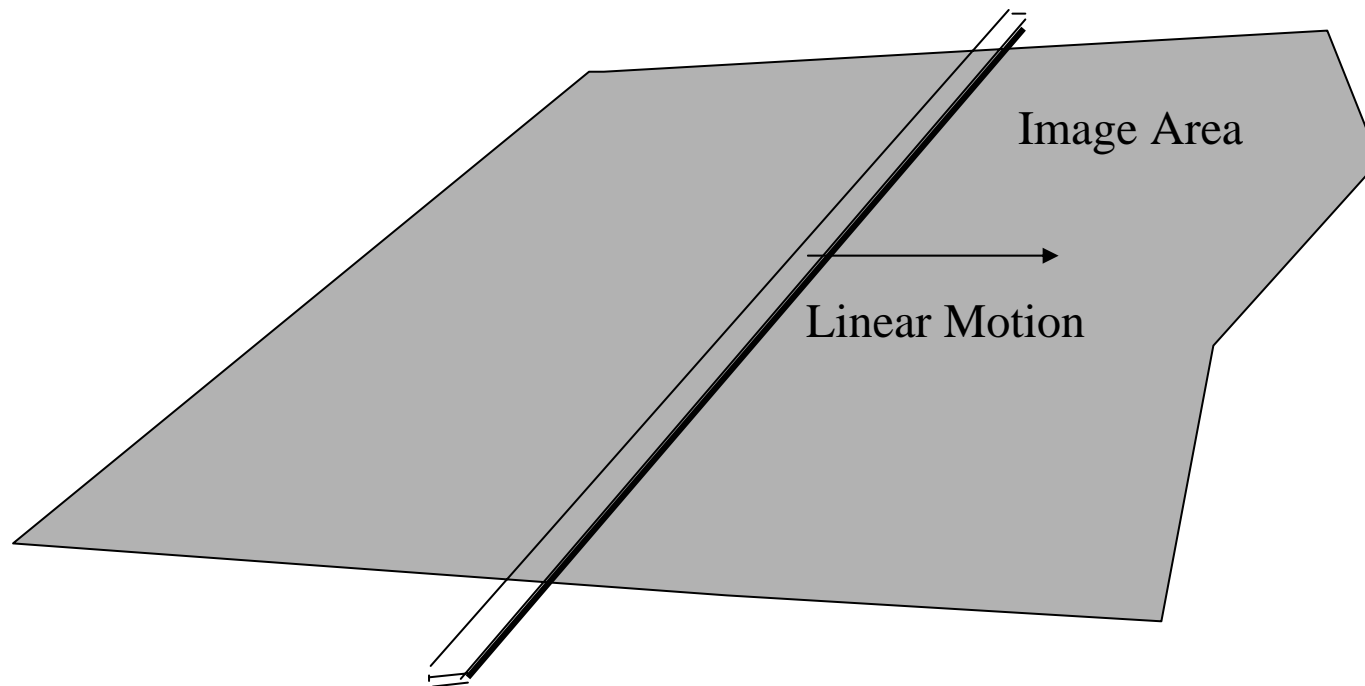
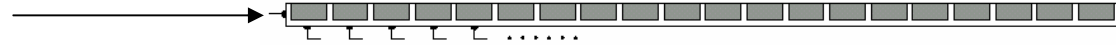
Line sensor

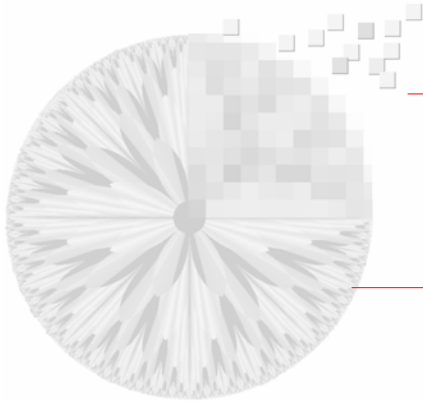




# Image Sensors

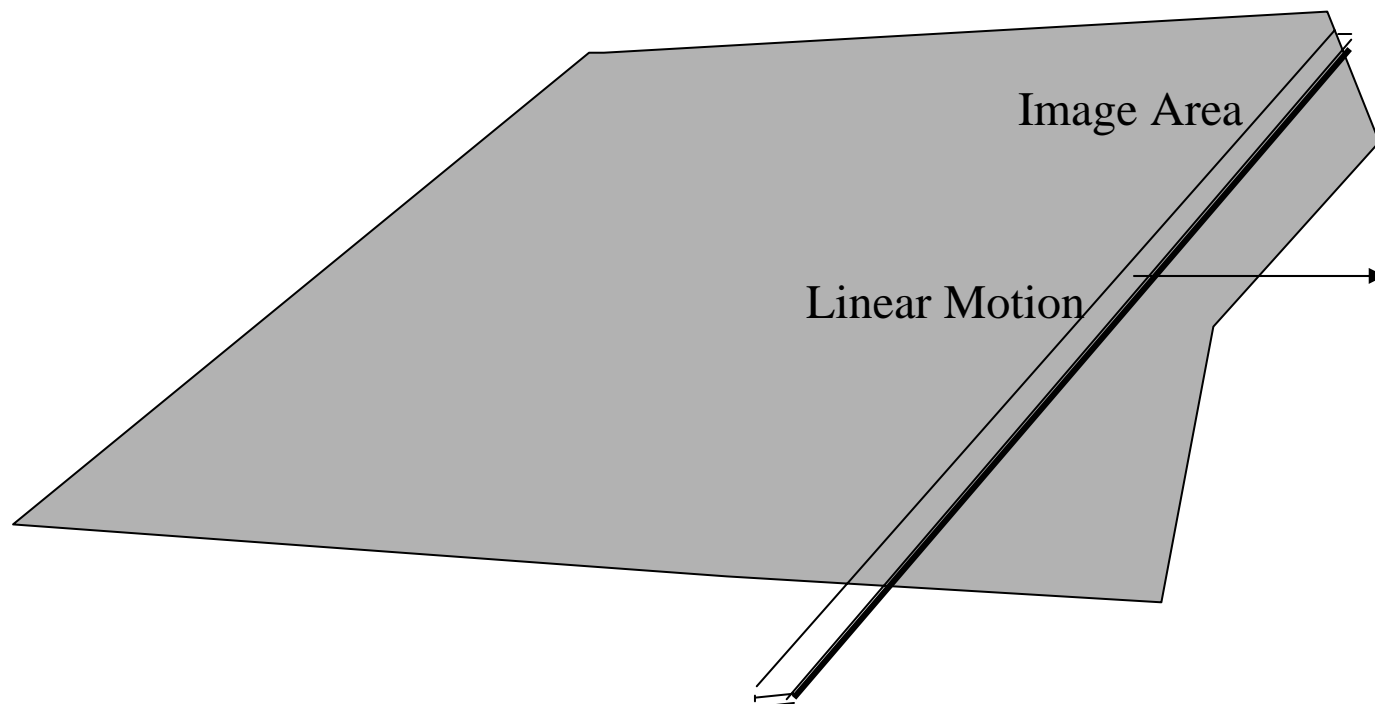
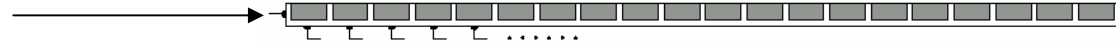
Line sensor

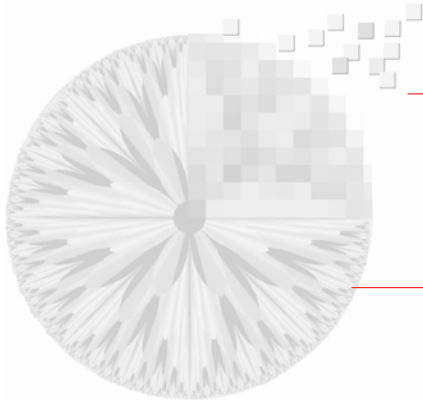




# Image Sensors

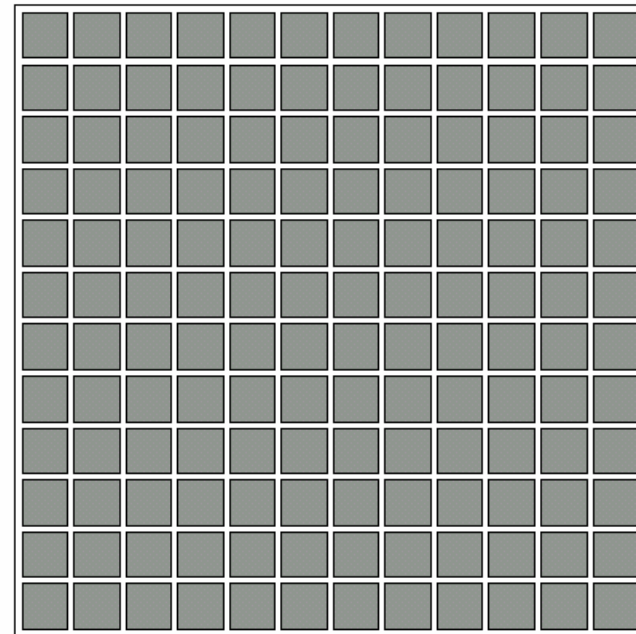
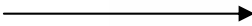
Line sensor





# Image Sensors

Array of Sensors

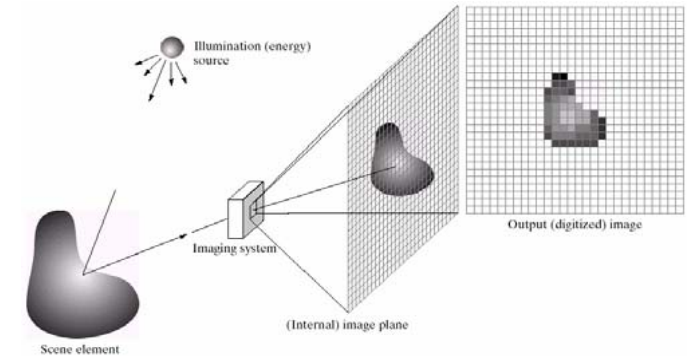


CCD Camera



# Image Formation Model

$$f(x,y)=i(x,y)r(x,y)$$



where

- 1)  $i(x,y)$  the amount of illumination incident to the scene

$$0 < i(x, y) < \infty$$

- 2)  $r(x,y)$  the reflectance from the objects

$$0 < r(x, y) < 1$$





# Image Formation Model

- For Monochrome Images :  $l = f(x, y)$

where

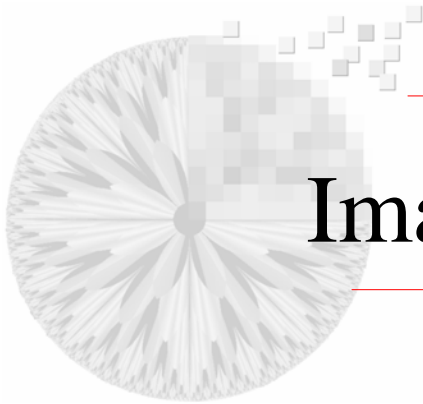
- »  $l_{min} < l < l_{max}$
- »  $l_{min} > 0$
- »  $l_{max}$  should be finite

The Interval  $[l_{min}, l_{max}]$  is called the **gray scale**

In practice, the gray scale is from 0 to L-1, where L is the # of gray levels

0            > Black

L-1        > White

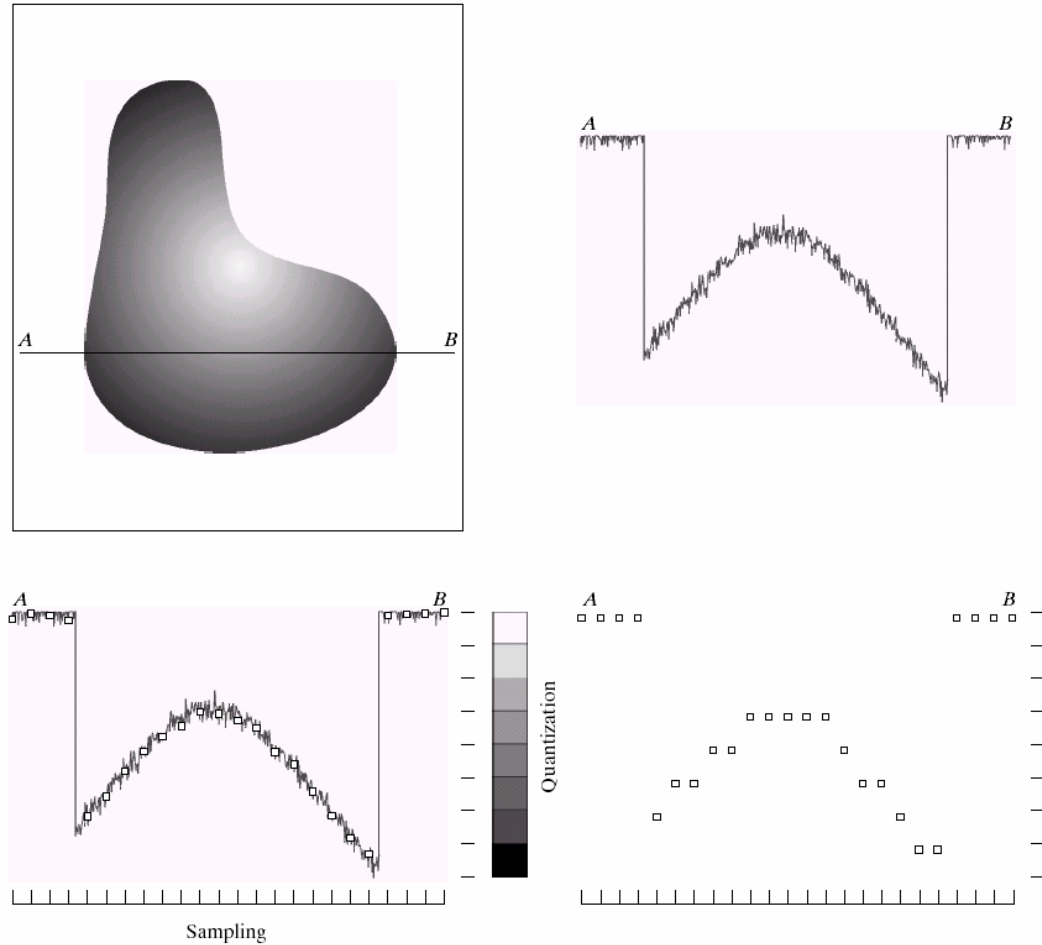


# Image Sampling and Quantization

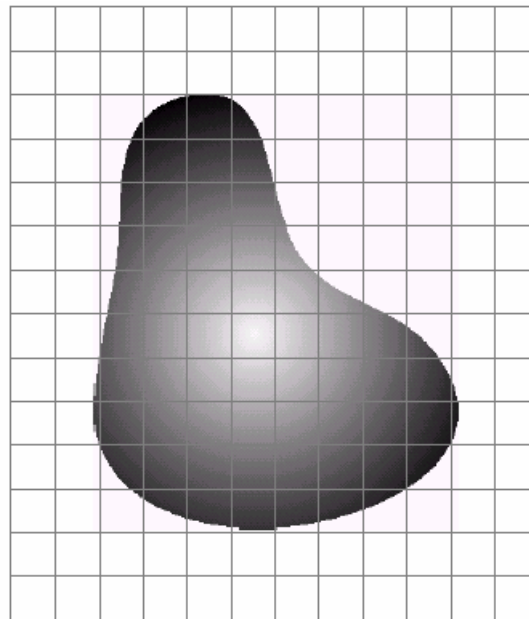


- Sampling is the quantization of coordinates
- Quantization is the quantization of gray levels

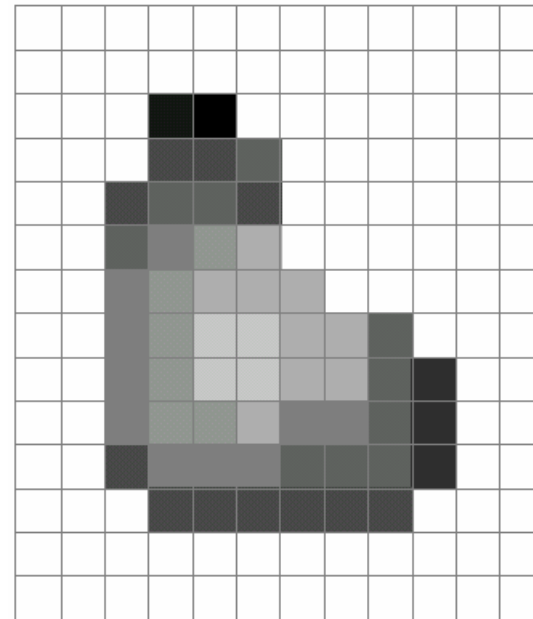
# Image Sampling and Quantization



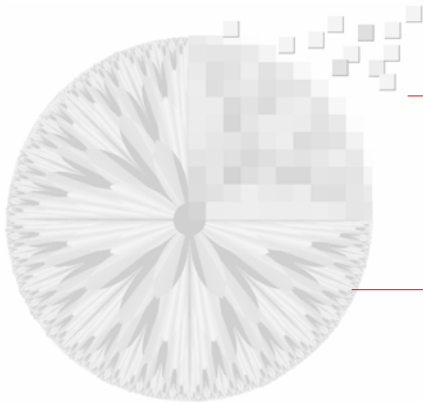
# Sampling and Quantization



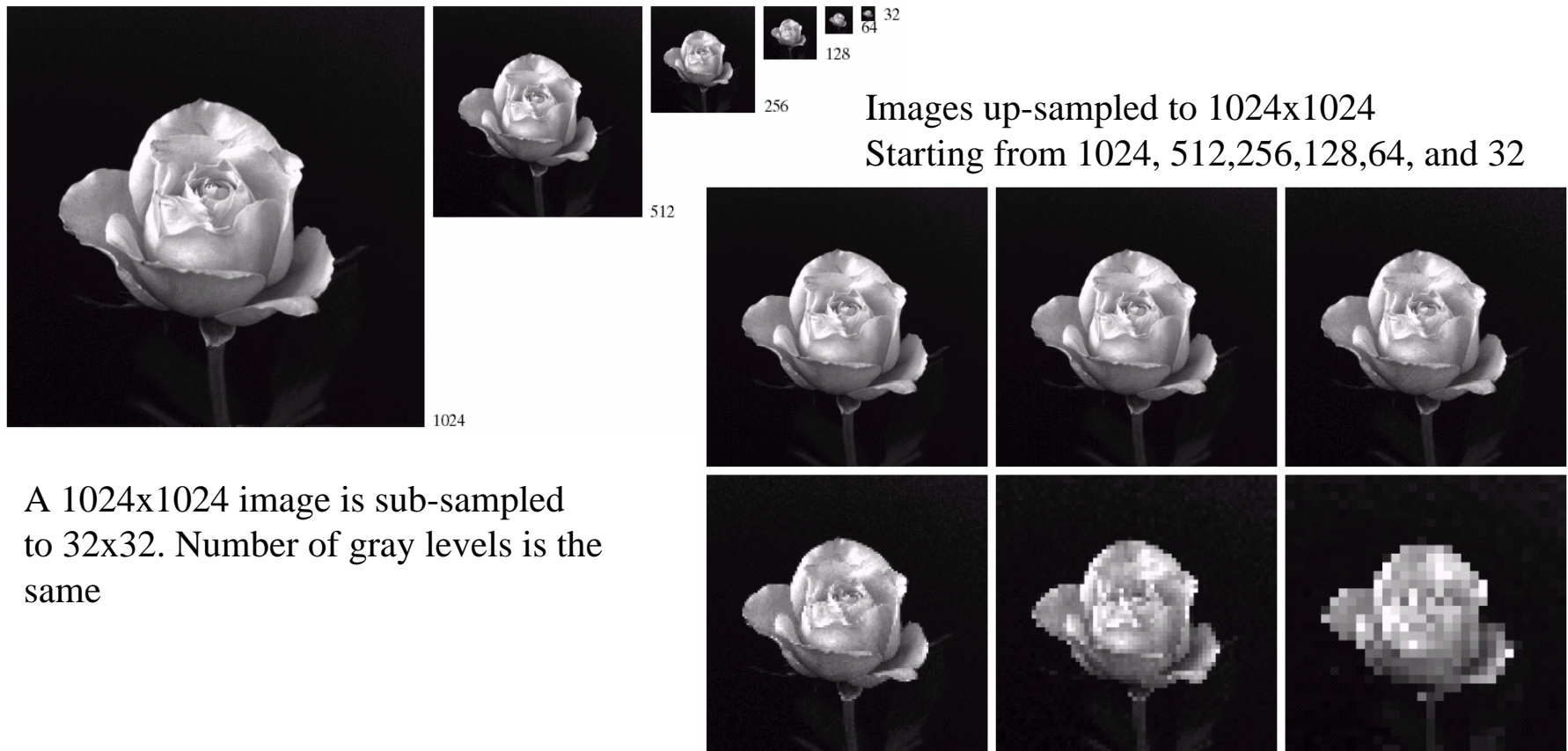
**Continuous Image projected  
onto a sensor array**



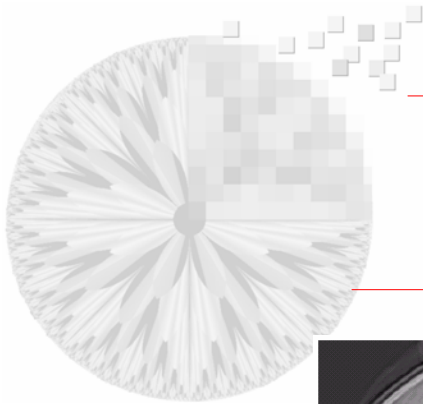
**Results of Sampling and  
Quantization**



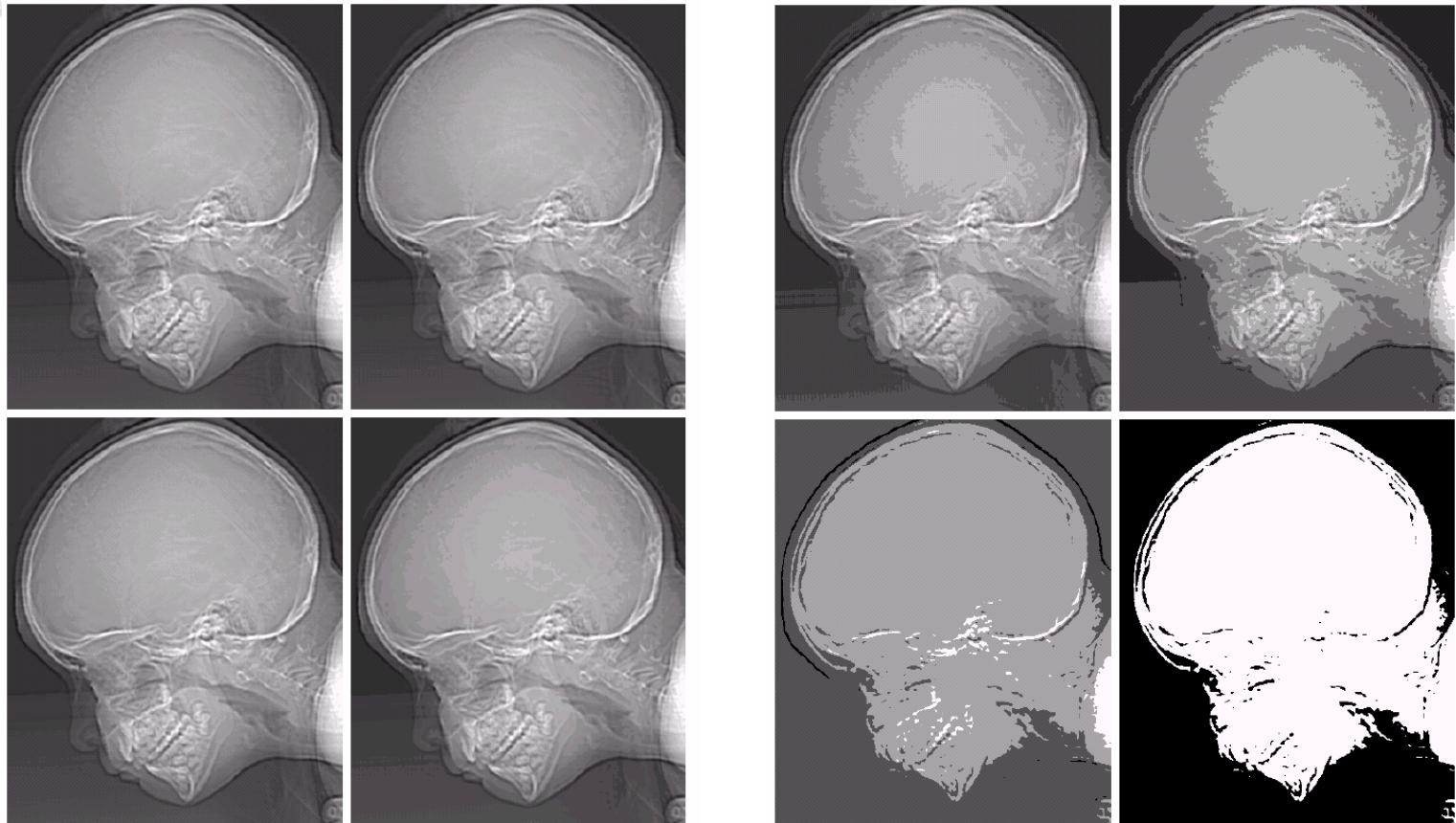
# Effect of Sampling



A 1024x1024 image is sub-sampled to 32x32. Number of gray levels is the same



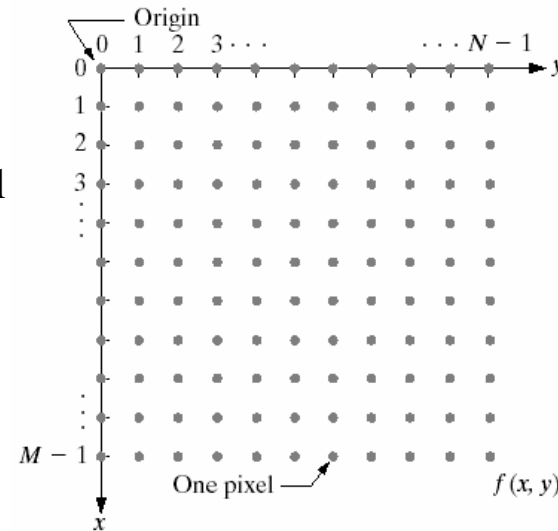
# Effect of Quantization



**An X-ray Image represented by different number of gray levels: 256, 128, 64, 32, 16, 8, 4, and 2.**

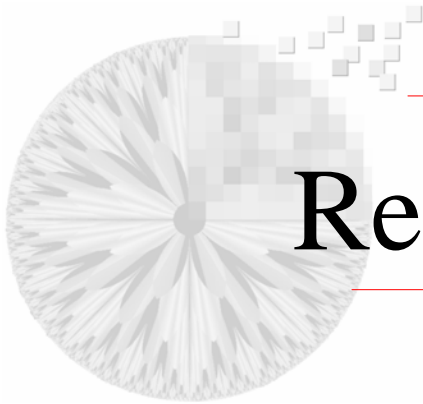
# Representing Digital Images

The result of Sampling and Quantization is a matrix of real Numbers. Here we have an image  $f(x,y)$  that was sampled To produce M rows and N columns.



$$f(x, y) = \begin{bmatrix} f(0,0) & f(0,1) & \dots & f(0, N-1) \\ \dots & f(1,1) & \dots & \dots \\ \dots & \dots & \dots & \dots \\ f(M-1,0) & \dots & \dots & f(M-1, N-1) \end{bmatrix}$$





# Representing Digital Images

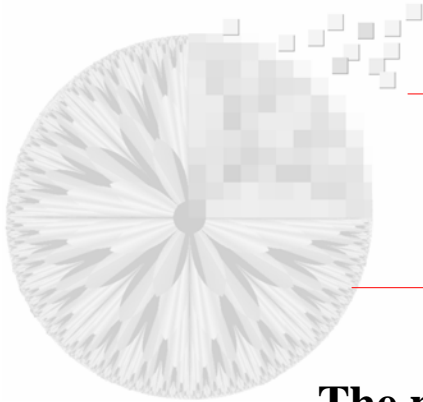
- There is no requirements about  $M$  and  $N$
- Usually  $L = 2^k$
- Dynamic Range :  $[0, L-1]$

**The number of bits required to store an image**

$$b = M \times N \times k \quad \text{where } k \text{ is the number of bits/pixel}$$

**Example** : The size of a 1024 x 1024 8bits/pixel image is  $2^{20}$  bytes = 1 MBytes





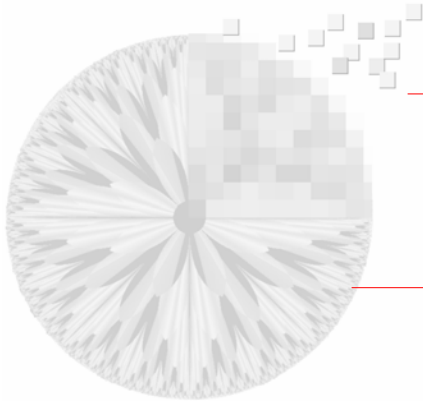
# Image Storage

**The number of bits required to store an image**

$$b = M \times N \times k \quad \text{where } k \text{ is the number of bits/pixel}$$

$N/k$	1 ( $L = 2$ )	2 ( $L = 4$ )	3 ( $L = 8$ )	4 ( $L = 16$ )	5 ( $L = 32$ )	6 ( $L = 64$ )	7 ( $L = 128$ )	8 ( $L = 256$ )
32	1,024	2,048	3,072	4,096	5,120	6,144	7,168	8,192
64	4,096	8,192	12,288	16,384	20,480	24,576	28,672	32,768
128	16,384	32,768	49,152	65,536	81,920	98,304	114,688	131,072
256	65,536	131,072	196,608	262,144	327,680	393,216	458,752	524,288
512	262,144	524,288	786,432	1,048,576	1,310,720	1,572,864	1,835,008	2,097,152
1024	1,048,576	2,097,152	3,145,728	4,194,304	5,242,880	6,291,456	7,340,032	8,388,608
2048	4,194,304	8,388,608	12,582,912	16,777,216	20,971,520	25,165,824	29,369,128	33,554,432
4096	16,777,216	33,554,432	50,331,648	67,108,864	83,886,080	100,663,296	117,440,512	134,217,728
8192	67,108,864	134,217,728	201,326,592	268,435,456	335,544,320	402,653,184	469,762,048	536,870,912

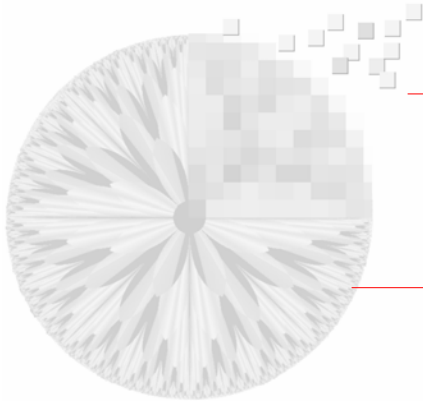
The number of storage bits depending on width and height ( $N \times N$ ), and the number Of bits/pixel  $k$ .



# File Formats

- PGM/PPM
- RAW
- JPEG
- GIF
- TIFF
- PDF
- EPS





# File Formats

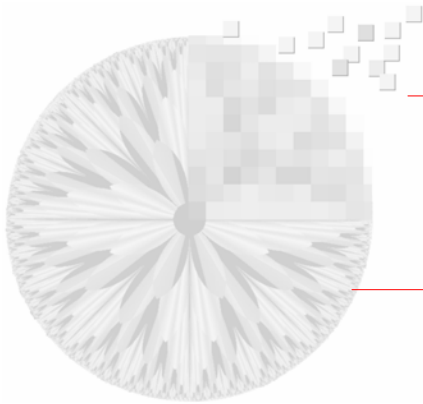
- **The TIFF File**

TIFF -- or Tag Image File Format -- was developed by Aldus Corporation in 1986, specifically for saving images from scanners, frame grabbers, and paint/photo-retouching programs.

Today, it is probably the most versatile, reliable, and widely supported bit-mapped format. It is capable of describing bi-level, grayscale, palette-color, and full-color image data in several color spaces.

It includes a number of compression schemes and is not tied to specific scanners, printers, or computer display hardware.

The TIFF format does have several variations, however, which means that occasionally an application may have trouble opening a TIFF file created by another application or on a different platform

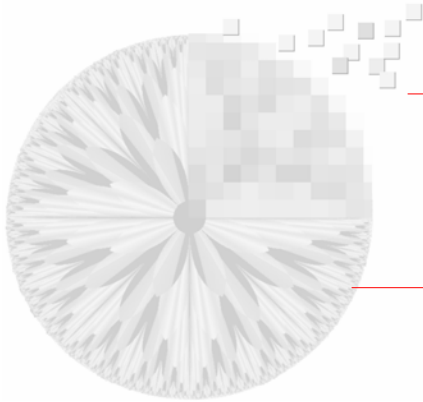


# File Formats

- **The GIF File**

GIF -- or Graphics Interchange Format -- files define a protocol intended for the on-line transmission and interchange of raster graphic data in a way that is independent of the hardware used in their creation or display.

- The GIF format was developed in 1987 by CompuServe for compressing eight-bit images that could be telecommunicated through their service and exchanged among users.
- The GIF file is defined in terms of blocks and sub-blocks which contain relevant parameters and data used in the reproduction of a graphic. A GIF data stream is a sequence of protocol blocks and sub-blocks representing a collection of graphics

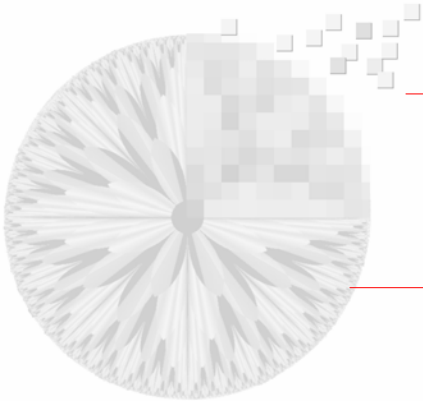


# File Formats

- **The JPEG File**

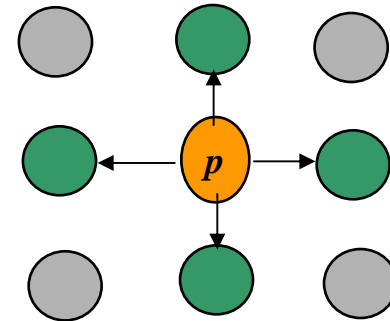
JPEG is a standardized image compression mechanism. The name derives from the Joint Photographic Experts Group, the original name of the committee that wrote the standard. In reality, JPEG is not a file format, but rather a method of data encoding used to reduce the size of a data file. It is most commonly used within file formats such as JFIF and TIFF.

- JPEG File Interchange Format (JFIF) is a minimal file format which enables JPEG bitstreams to be exchanged between a wide variety of platforms and applications. This minimal format does not include any of the advanced features found in the TIFF JPEG specification or any application specific file format.
- JPEG is designed for compressing either full-color or grayscale images of natural, real-world scenes. It works well on photographs, naturalistic artwork, and similar material, but not so well on lettering or simple line art. It is also commonly used for on-line display/transmission; such as on web sites.
- A 24-bit image saved in JPEG format can be reduced to about one-twentieth of its original size.



# Neighbors of a Pixel

- A pixel  $p$  at coordinates  $(x,y)$  has 4 neighbors:  $(x-1,y)$ ,  $(x+1,y)$ ,  $(x,y-1)$ ,  $(x,y+1)$ .
- These pixels are called  $N_4(p)$



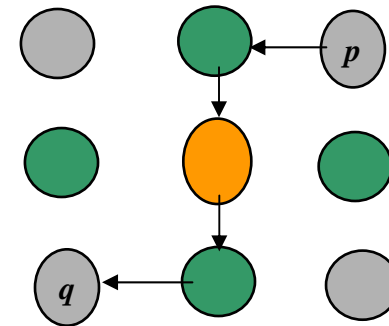
- $N_8(p)$  are the eight immediate neighbors of  $p$

# Adjacency and Connectivity

- Two pixels are connected if:
  - They are neighbors
  - Their gray levels satisfy certain conditions (e.g. :  $g_1 = g_2$ )
- \*Two pixels  $p, q$  are 4 adjacent if  $q \in N_4(p)$
- \*Two pixels  $p, q$  are 8 adjacent if  $q \in N_8(p)$

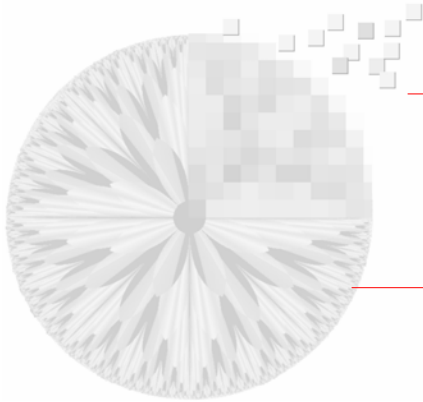
# Adjacency and Connectivity

- Path :
  - A digital path from  $p$  to  $q$  is the set of pixel coordinates linking  $p$  and  $q$ .



- Region:
  - A region is a connected set of pixels





# Distance Measures

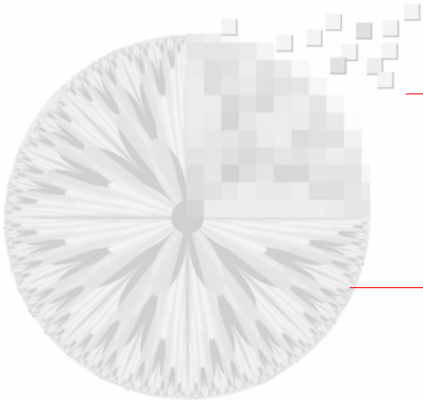
Assume we have 3 pixels:  $p:(x,y)$ ,  $q:(s,t)$  and  $z:(v,w)$

A distance function  $D$  is a metric that satisfies the following conditions:

- a)  $D(p, q) \geq 0$ ,  $D(p, q) = 0$  iff  $p = q$
- b)  $D(p, q) = D(q, p)$
- c)  $D(p, z) \leq D(p, q) + D(q, z)$

Example: Euclidean Distance :

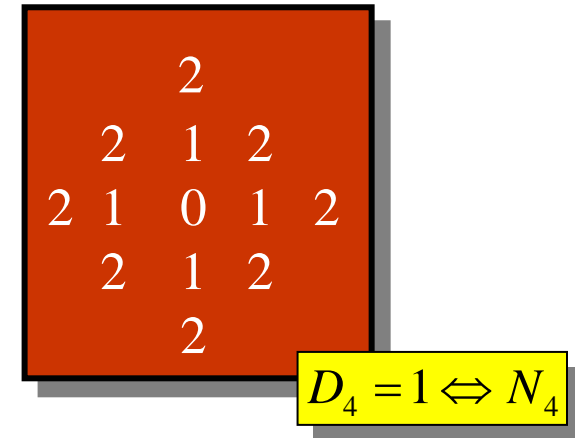
$$D(p, q) = \sqrt{(x - s)^2 + (y - t)^2}$$



# Distance Measures

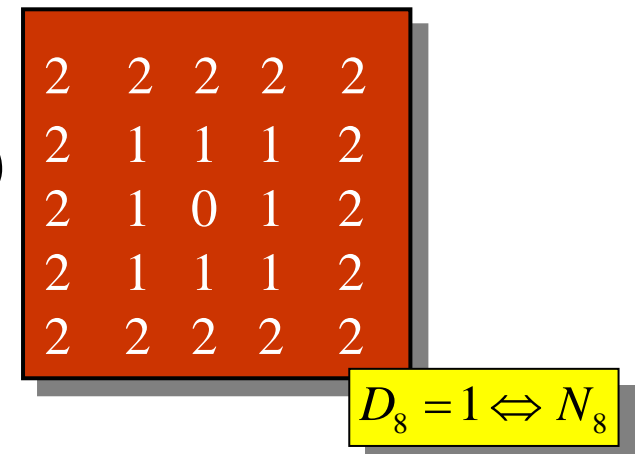
- City Block Distance :

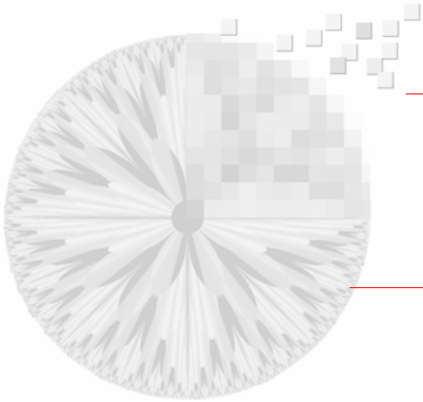
$$D_4(p, q) = |x - s| + |y - t|$$



- Chess Board Distance

$$D_8(p, q) = \max(|x - s|, |y - t|)$$



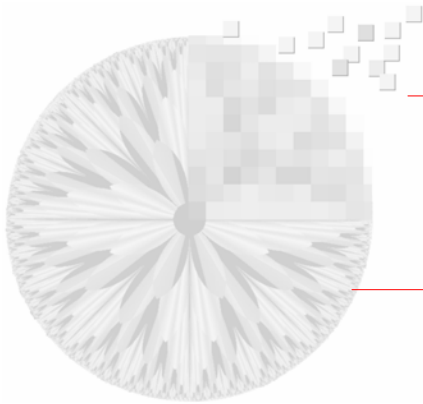


# Image Scaling

- Pixel Replication
- Bilinear Interpolation
- Bicubic Interpolation

A	B
C	D

A	e	B	
f	g	h	
C	i	D	

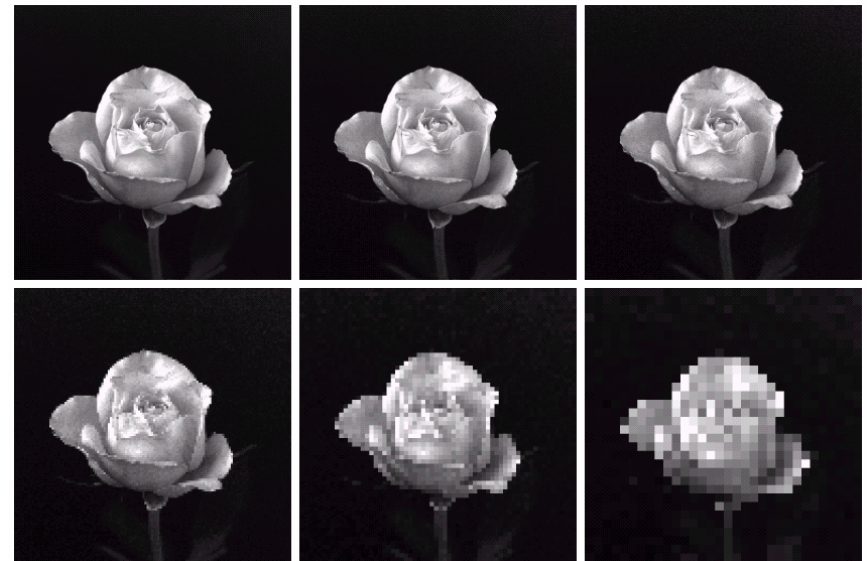


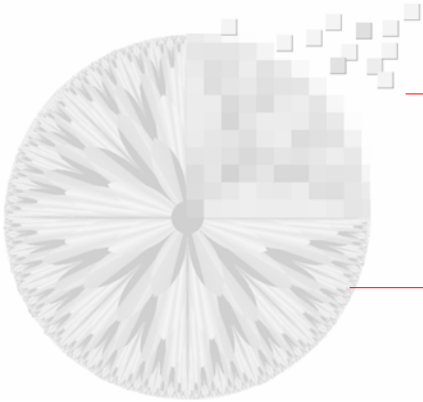
# Image Interpolation

- Pixel Replication:

Use the nearest neighbor to construct the zoomed image

Useful in doubling the image size

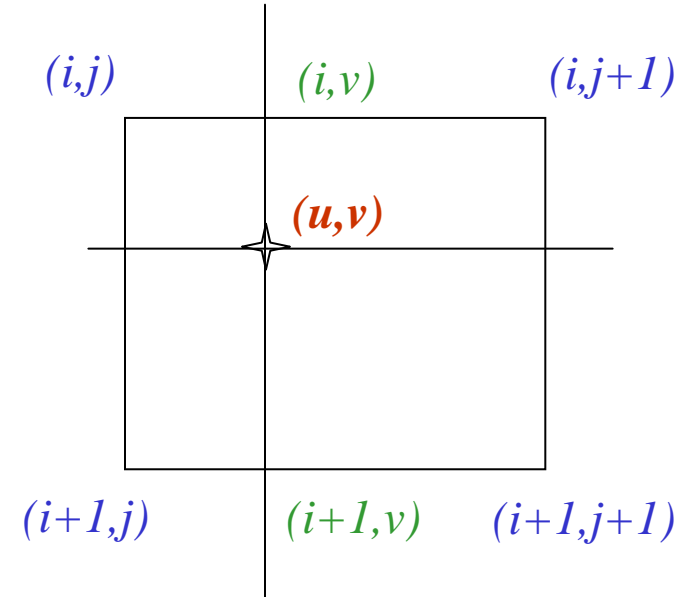




# Image Interpolation

- Bilinear Interpolation

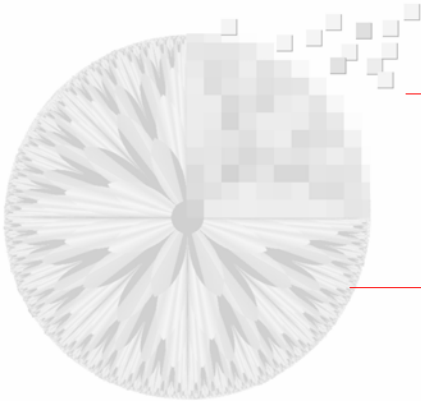
Use 4 nearest neighbors to calculate the image value.



$$f(u, v) = f(i, v)(1 - (u - i)) + f(i + 1, v)(u - i)$$

$$f(i, v) = f(i, j)(1 - (v - j)) + f(i + 1, j)(v - j)$$

$$f(i + 1, v) = f(i, j + 1)(1 - (v - j)) + f(i + 1, j + 1)(v - j)$$



# Image Interpolation

- Cubic Interpolation

Use 16 nearest neighbors

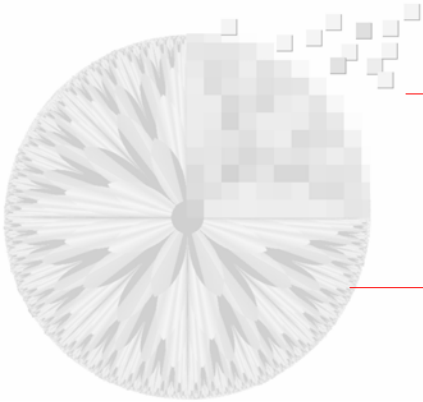
The contribution of each pixel depends on its distance from the output pixel

Usually we use spline curve to give smoother output.

$$f(u, v) = P^T(v) F P(u)$$

where

$$P(u) = \begin{bmatrix} p_1(u) \\ p_2(u) \\ p_3(u) \\ p_4(u) \end{bmatrix} \quad 0 < u, v < 1$$



# Image Interpolation

- Cubic Interpolation

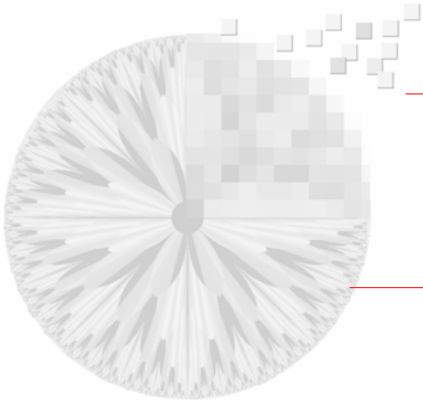
$$F = \begin{bmatrix} f_{i-1,j-1} & f_{i-1,j} & & \\ & f_{i,j} & & \\ & & & \\ f_{i+2,j} & & & f_{i+1,j+2} \end{bmatrix}$$

$$p_1(t) = (-t^3 + 2t^2 - t) / 2$$

$$p_2(t) = (3t^3 - 5t^2 + 2) / 2$$

$$p_3(t) = (-3t^3 + 4t^2 + t) / 2$$

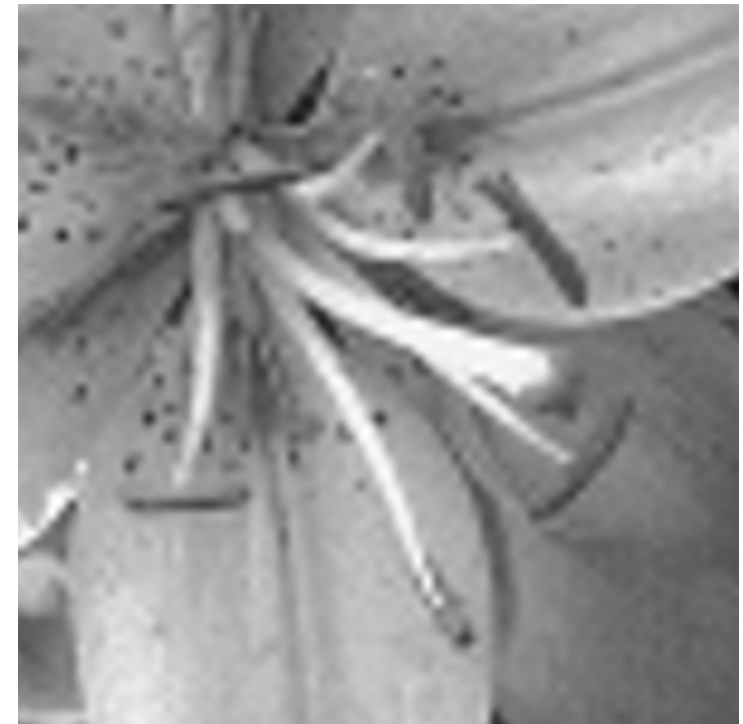
$$p_4(t) = (t^3 - t^2) / 2$$



# Image Interpolation

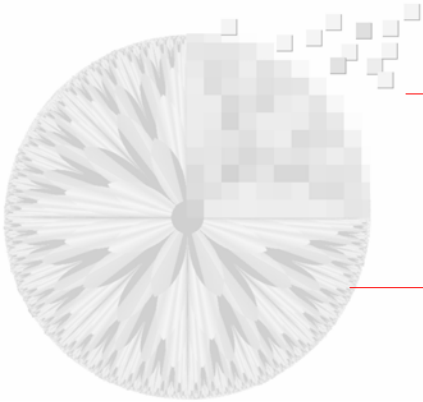


4x Bilinear Interpolation



4x Bicubic Interpolation





# Image Interpolation



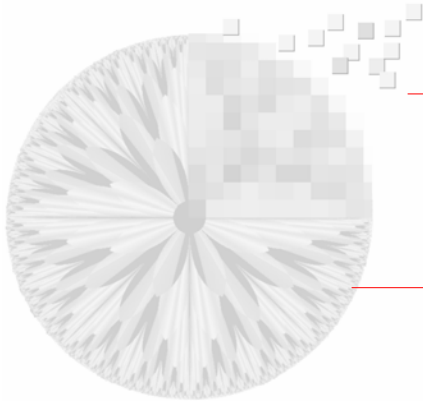
4x BiCubic Interpolation



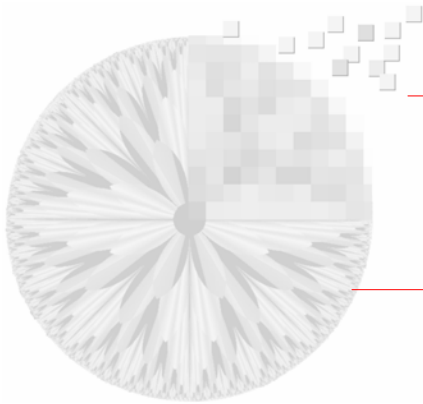
4x Edge Directed Interpolation

# Image Interpolation





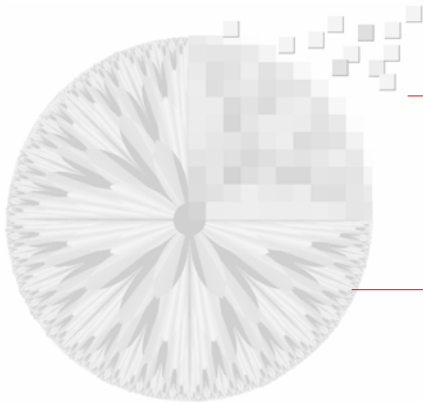
### 3. Image Enhancement in the Spatial Domain



# Image Enhancement

The objective of Image Enhancement is to process image data so that the result is more suitable than the original image



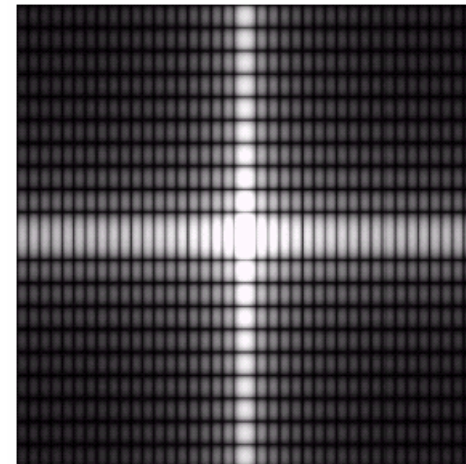
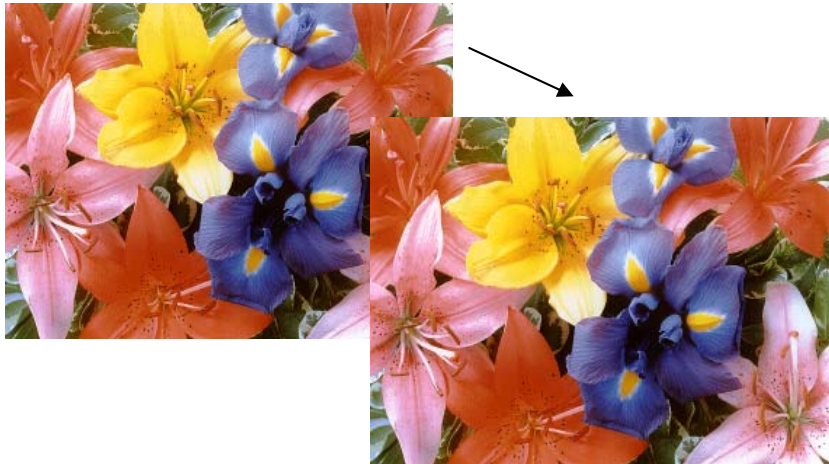


# Image Enhancement

Image Enhancement

Spatial Domain

Frequency Domain



# Spatial Domain Enhancement

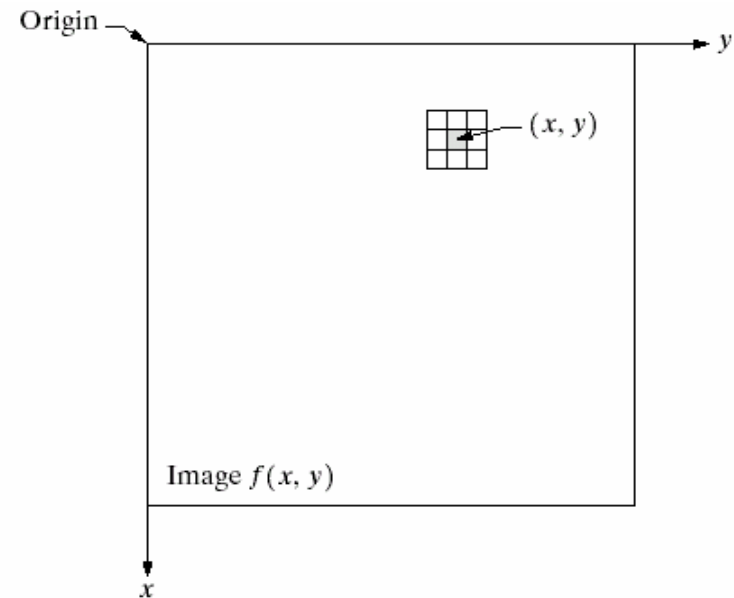
- Let  $f(x,y)$  be the original image and  $g(x,y)$  be the processed image

Then

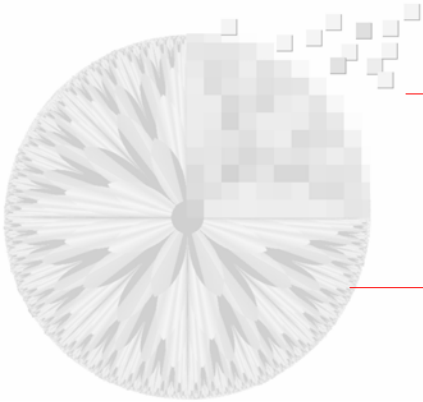
$$g(x, y) = T(f(x, y))$$

where  $T$  is an operator over a certain neighborhood of the image centered at  $(x,y)$

Usually, we operate on a small rectangular region around  $(x,y)$



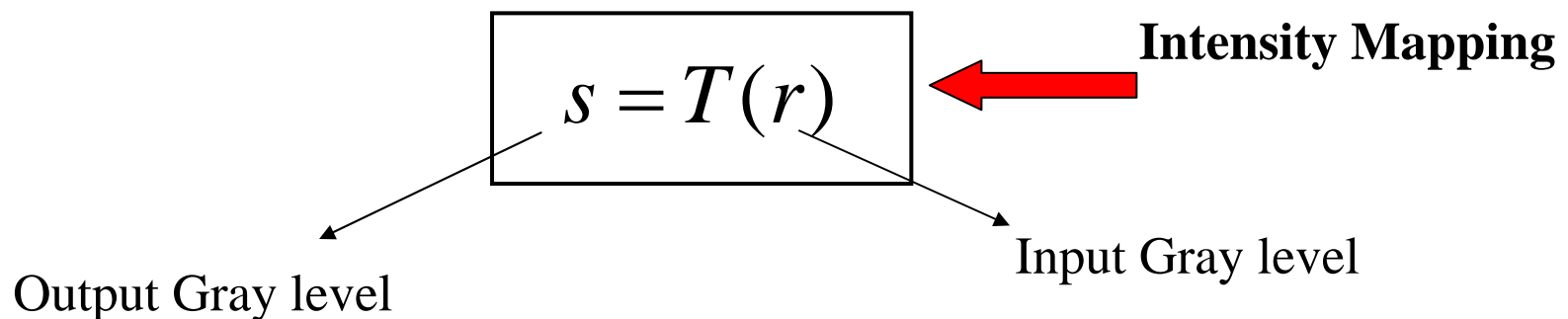


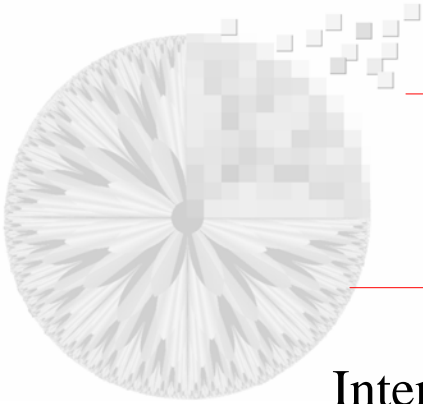


# Intensity Mapping

- The simplest form of  $T$  is when the neighborhood is  $1 \times 1$  pixel (single pixel)
- In this case,  $g$  depends only on the gray level at  $(x,y)$

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

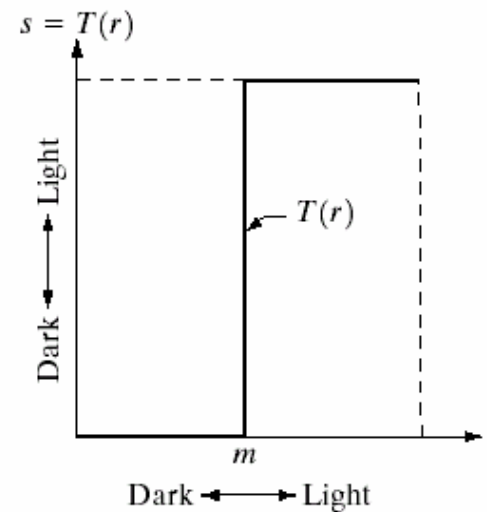
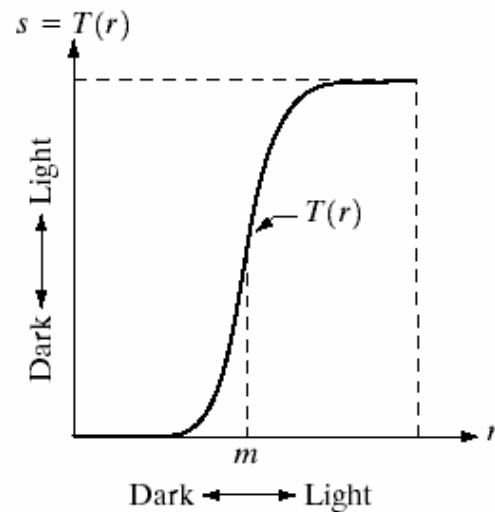




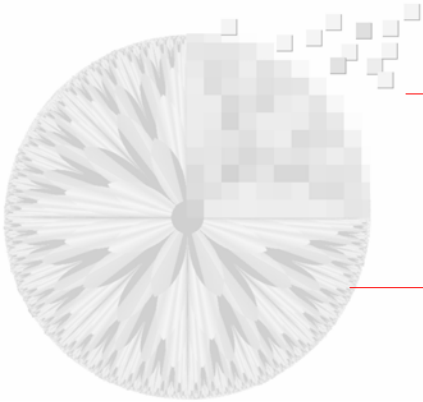
# Intensity Mapping

Intensity mapping is used to :

- a) Increase Contrast
- b) Vary range of gray Levels







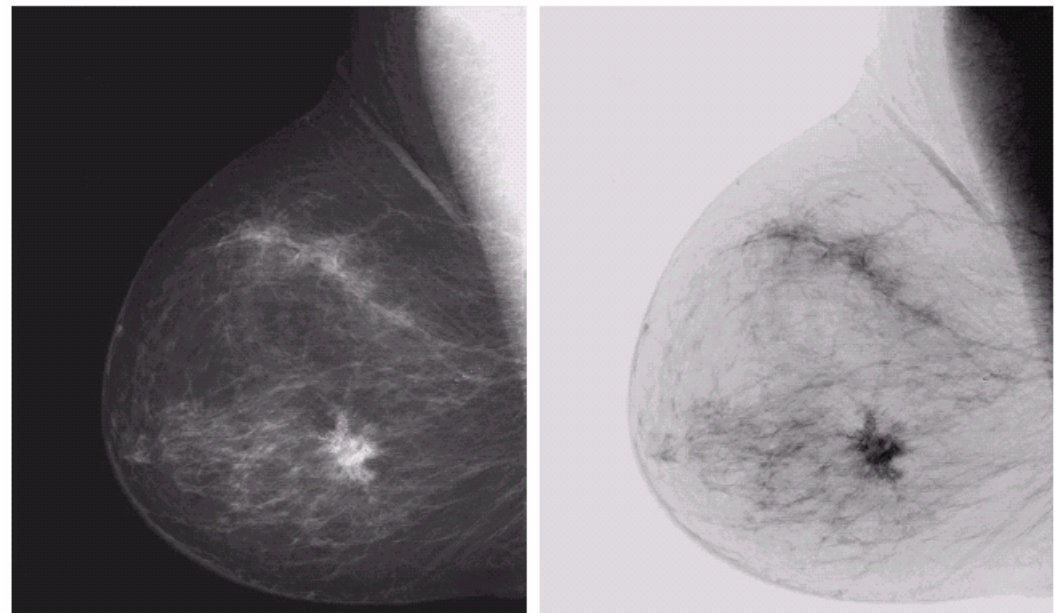
# Image Mapping

- **A) Image Negative**

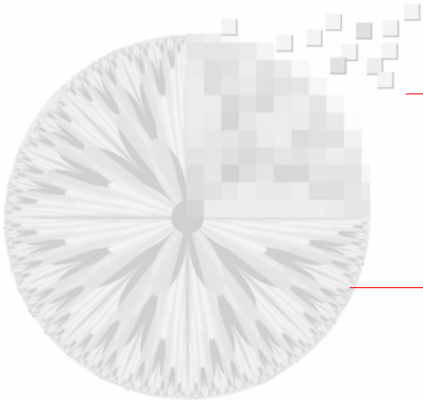
$$s = (L - 1) - r$$

Example:  $L=256$

$$s = 255 - r$$



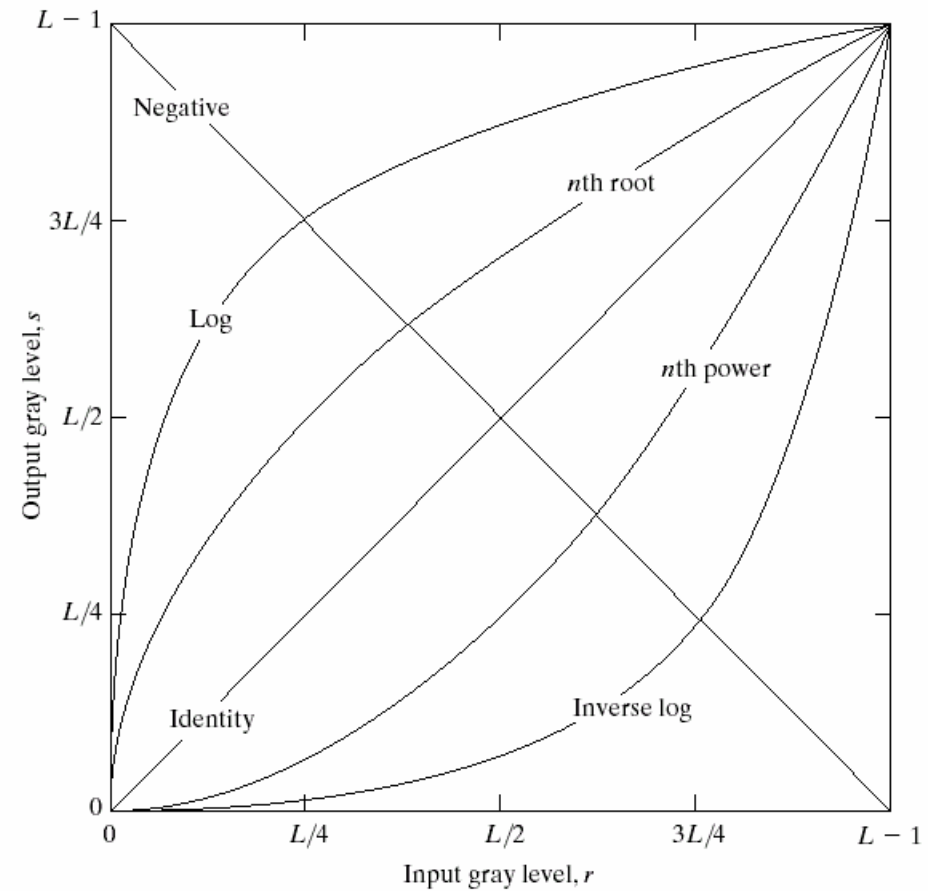
This operation enhances details in dark regions

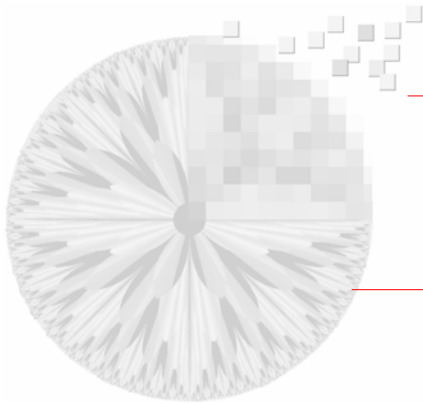


# Image Mapping

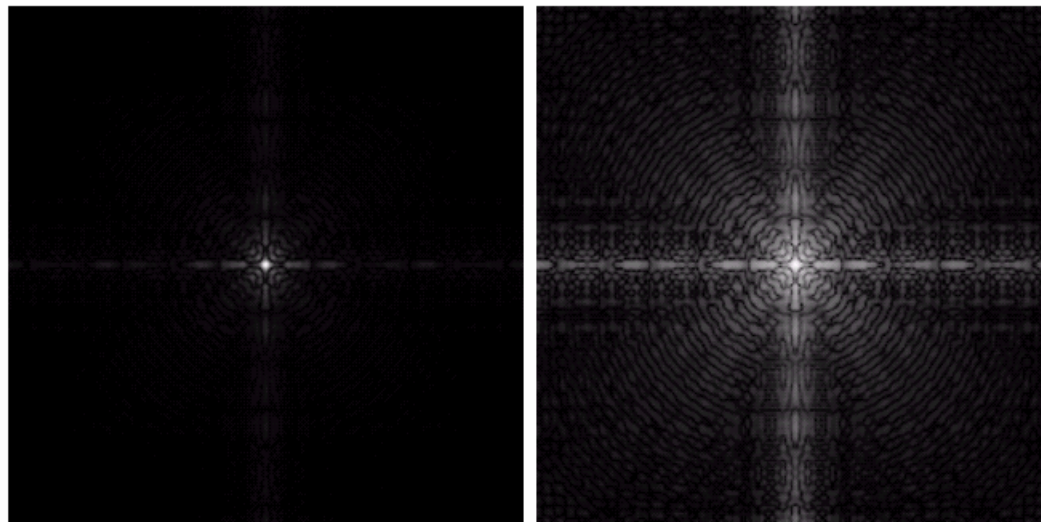
- B) Log Transformations

$$s = c \log(1 + r)$$

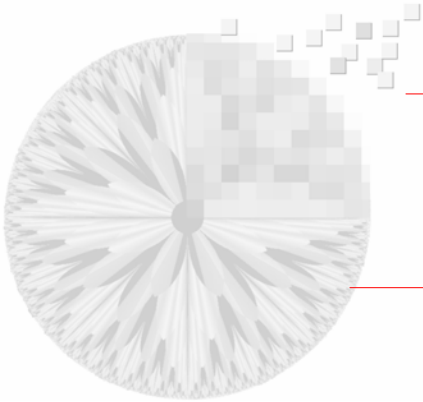




# Image Mapping



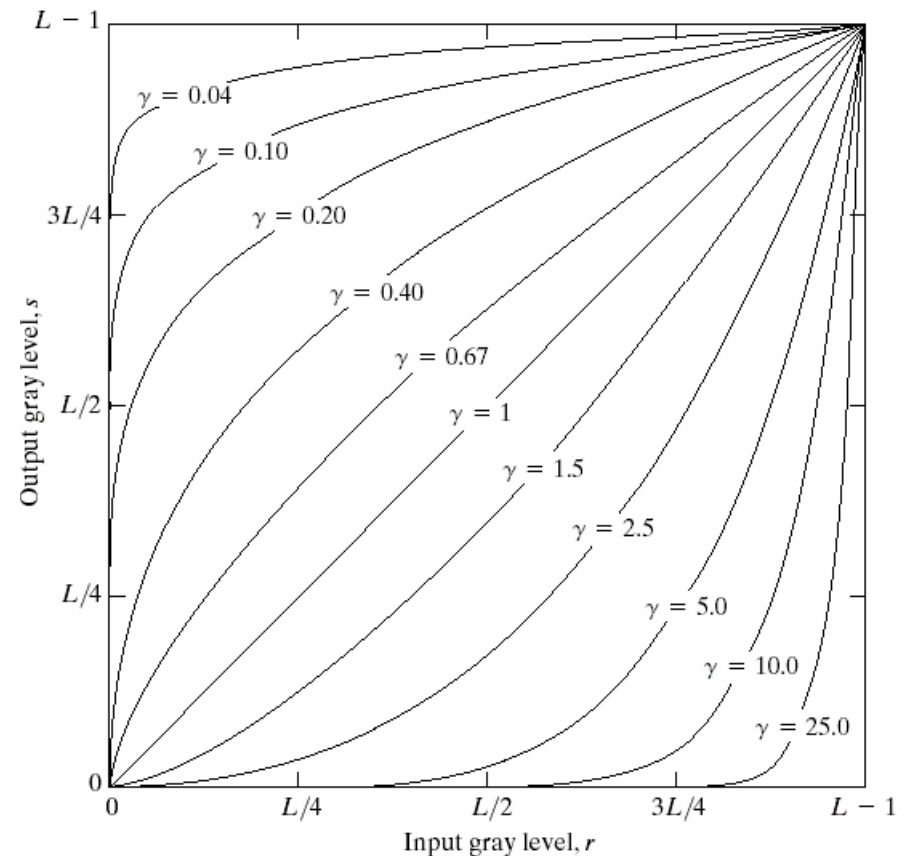
Fourier Spectrum and Result of applying log transformation  $c=1$

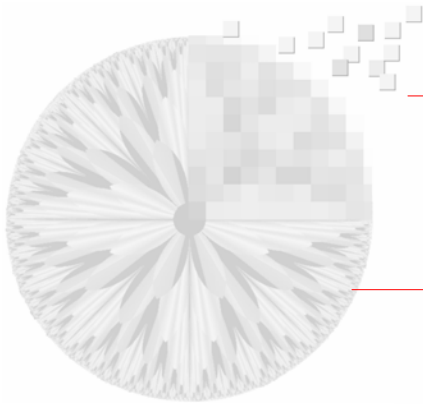


# Image Mapping

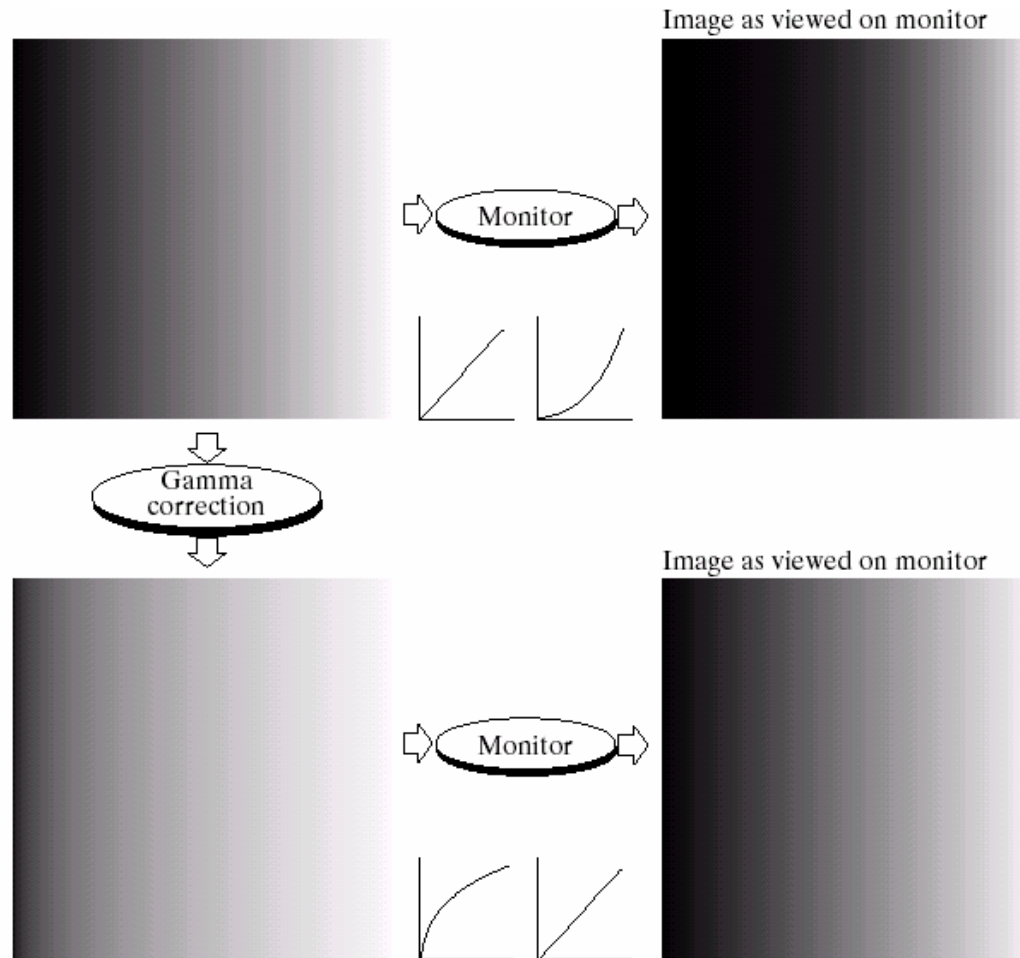
- C) Power Transformation

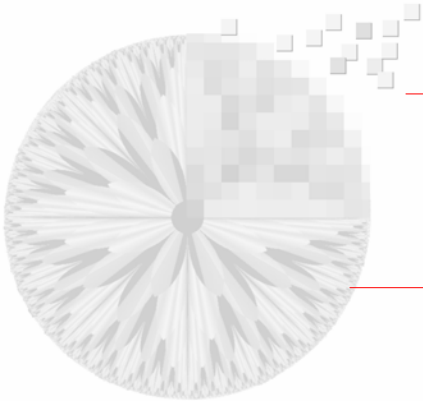
$$s = cr^\gamma$$



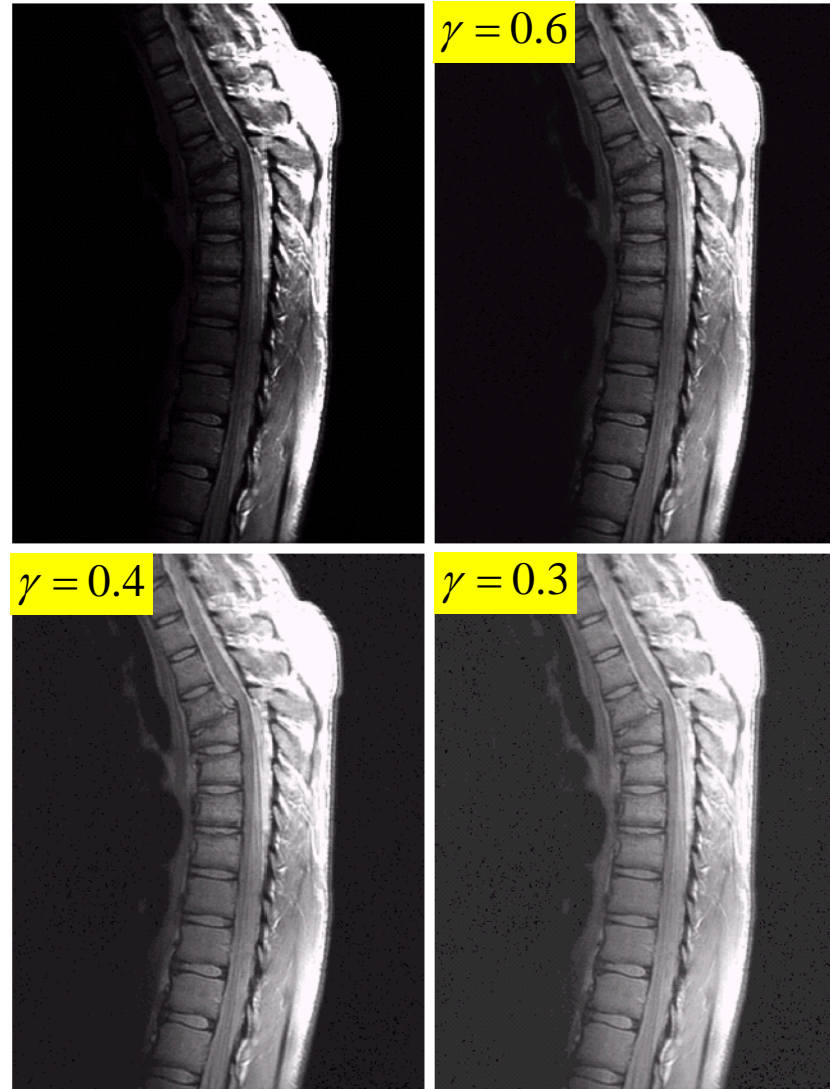


# Gamma Correction

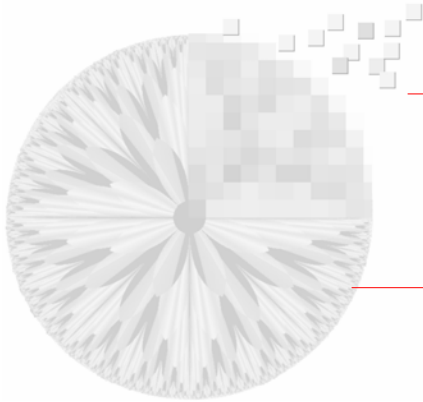




# Gamma Correction



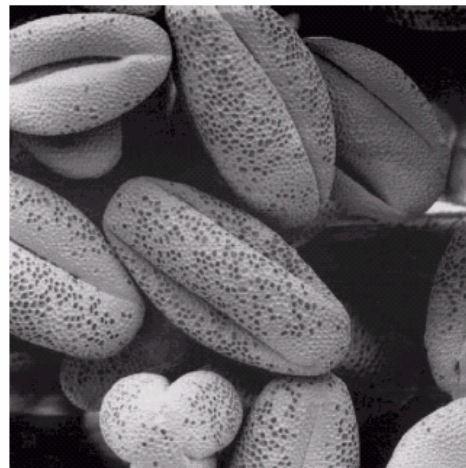
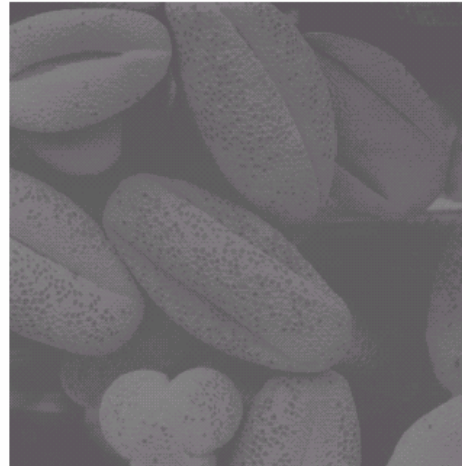
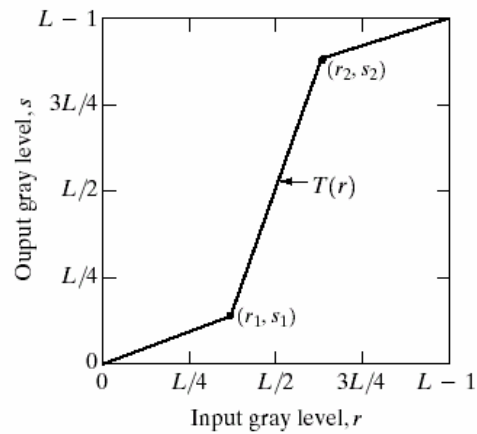




# Gamma Correction



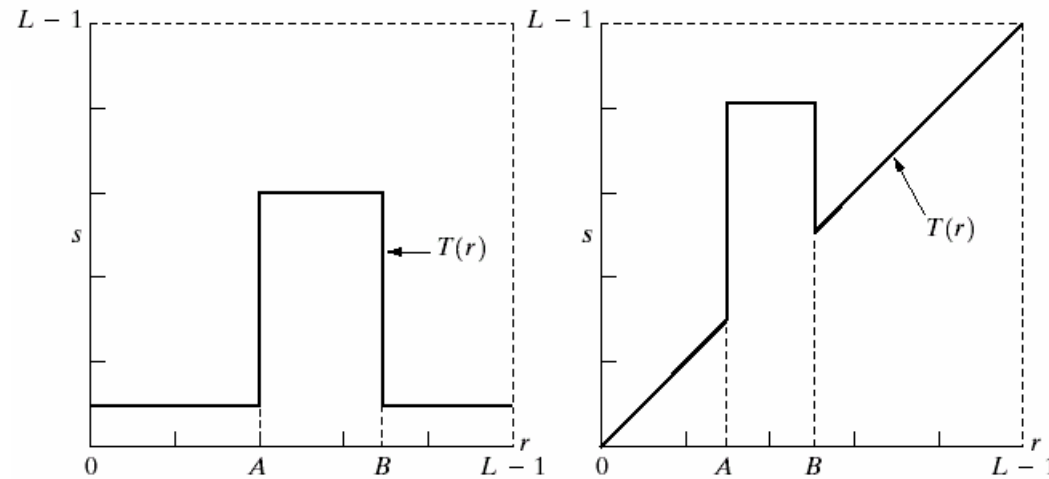
# Contrast Stretching



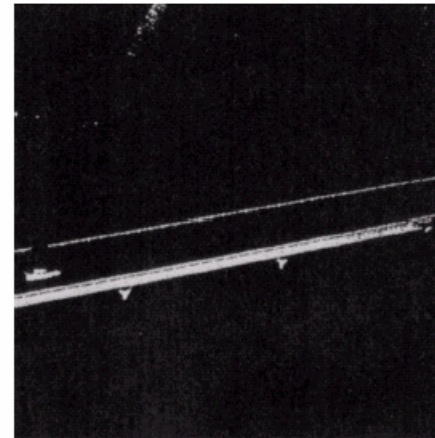
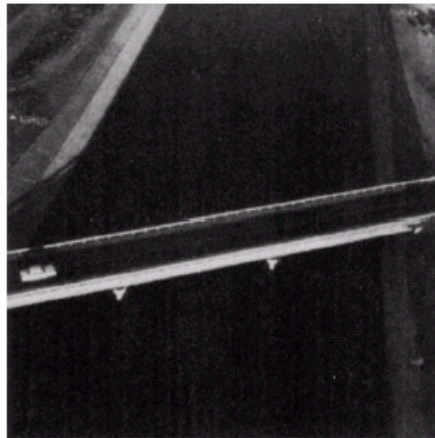
Contrast stretching.  
 (a) Form of transformation function. (b) A low-contrast image. (c) Result of contrast stretching. (d) Result of thresholding. (Original image courtesy of Dr. Roger Heady, Research School of Biological Sciences, Australian National University, Canberra, Australia.)

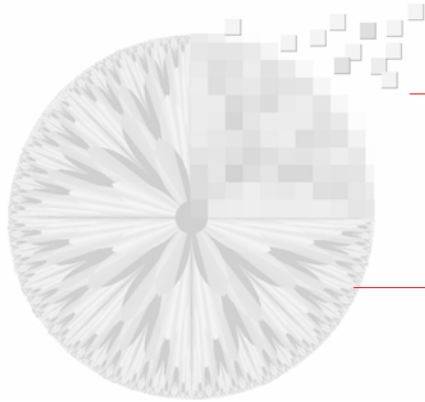


# Contrast Stretching



- (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).

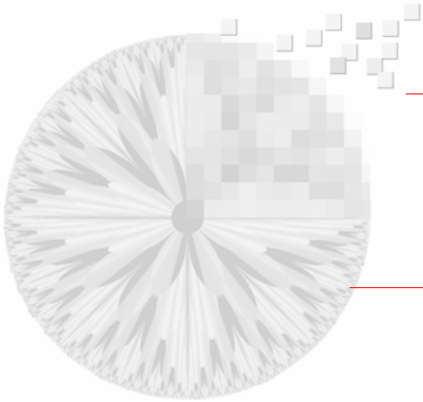




# Workshop

## Using Photoshop

1. Image -> Adjustments -> perform:
  - a) Image negative,
  - b) Approx  $\gamma=0.3$ ,  $\gamma=2.4$ ,
  - c) Clipping at 200
2. Use the Brightness and Contrast curves to increase the level of brightness of the image
4. Threshold Image: Image -> Adjustments -> Threshold



# Histogram

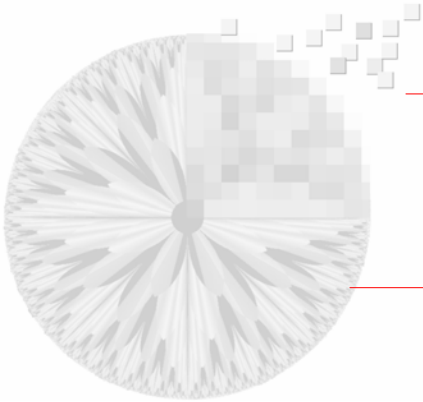
- The Histogram of a digital image is a function :

$$h(r_k) = n_k$$

where

$r_k$  is the  $k^{th}$  gray level

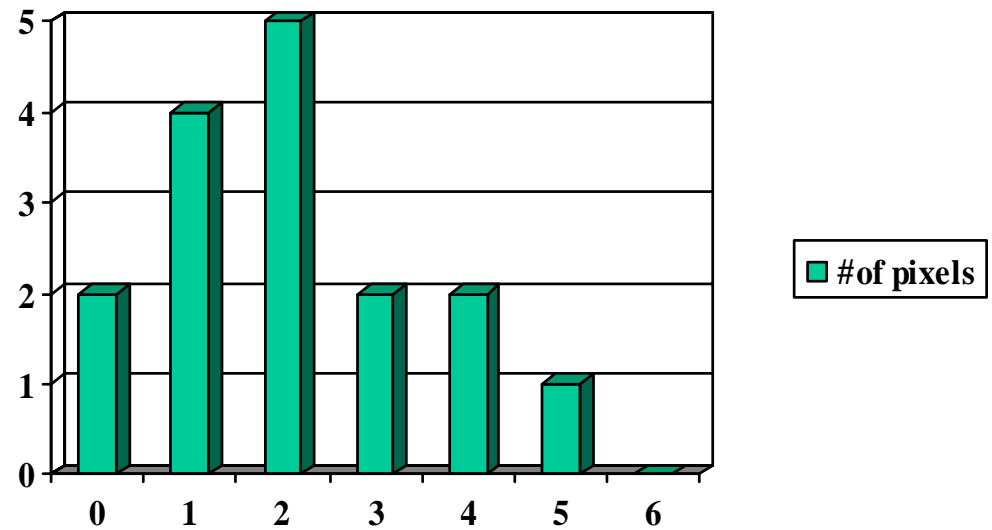
$n_k$  is the number of pixels having gray level  $r_k$

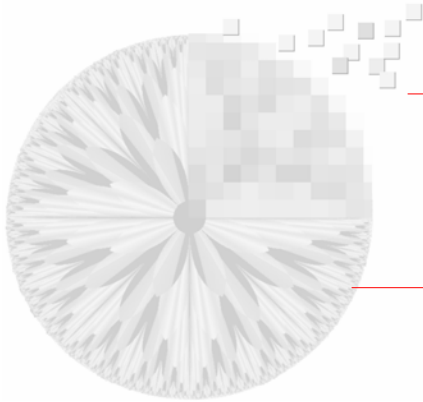


# Histogram

- Example:

0	0	2	2
1	1	2	5
1	1	3	4
2	2	3	4





# Normalized Histogram

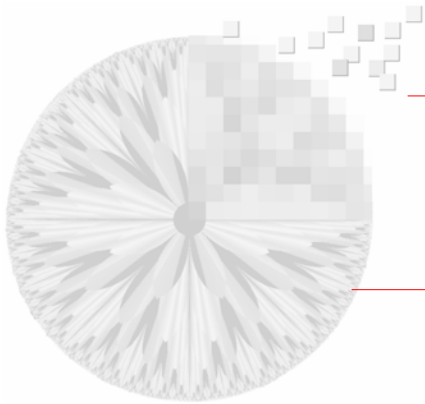
- Normally, we normalize  $h(r_k)$  by

$$p(r_k) = \frac{h(r_k)}{n} = \frac{n_k}{n}$$

- So, we have

$$\sum_{k=0}^{L-1} p(r_k) = \sum_{k=0}^{L-1} \frac{n_k}{n} = 1$$

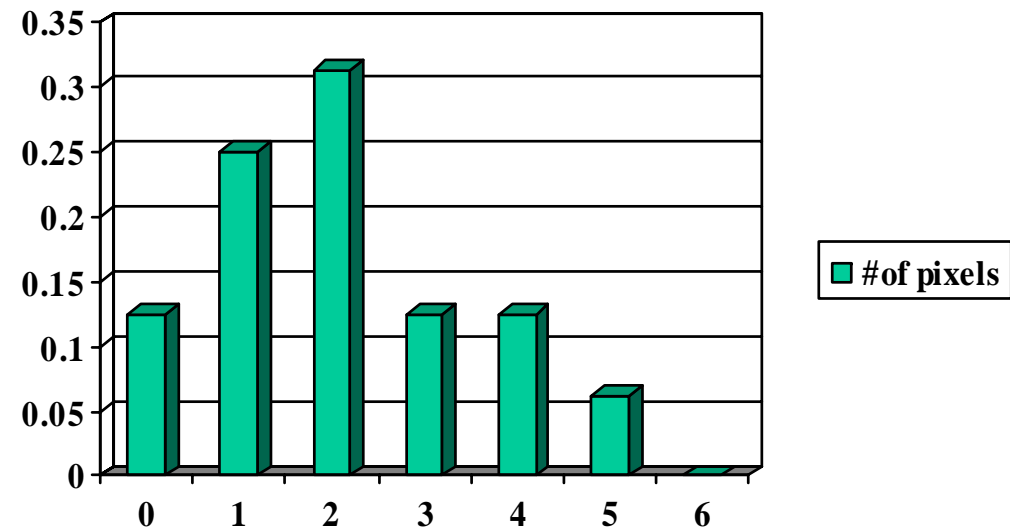
- $p(r_k)$  can be sought of as the probability of a pixel to have a certain value  $r_k$



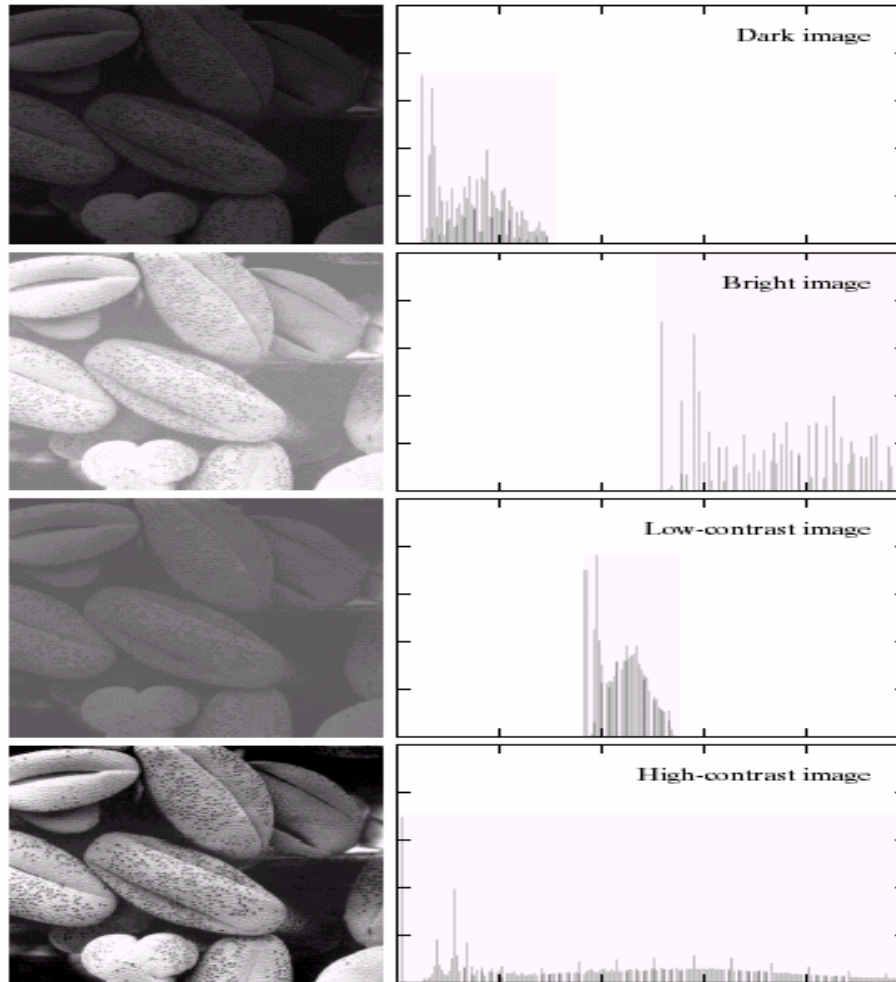
# Normalized Histogram

- Example:  $n=16$

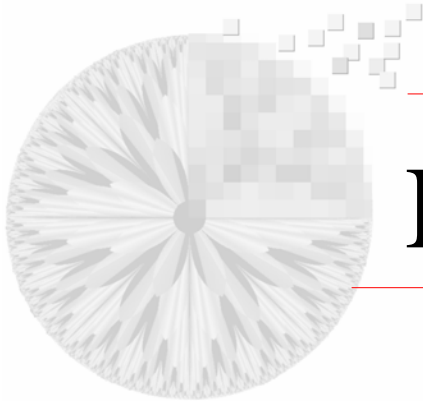
0	0	2	2
1	1	2	5
1	1	3	4
2	2	3	4



# Histogram



**Note:** Images with uniformly Distributed histograms have higher Contrast and high dynamic range

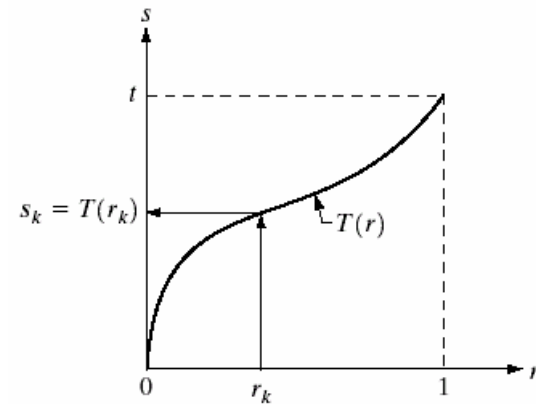


# Histogram Equalization

- Define a transformation  $s = T(r)$

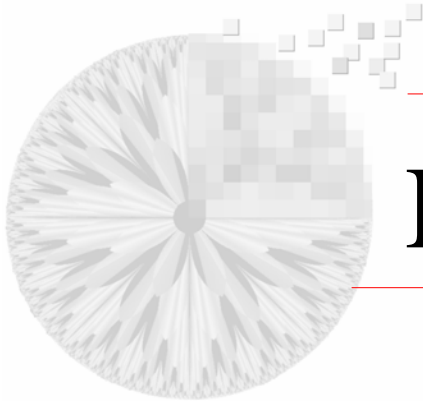
with

$$T(r) = \int_0^r p_r(w)dw$$



where  $p_r(r)$  is the probability histogram of image  $r$





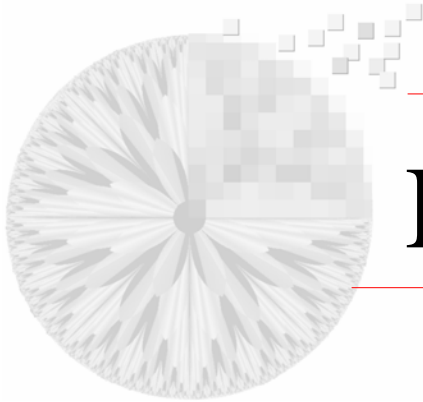
# Histogram Equalization

- Now let's calculate  $p_s(s)$

$$p_s(s) = p_r(r) \frac{dr}{ds}$$

$$\frac{ds}{dr} = \frac{d}{dr} T(r) = \frac{d}{dr} \int_0^r p_r(w) dw$$

$$\frac{ds}{dr} = p_r(r)$$



# Histogram Equalization

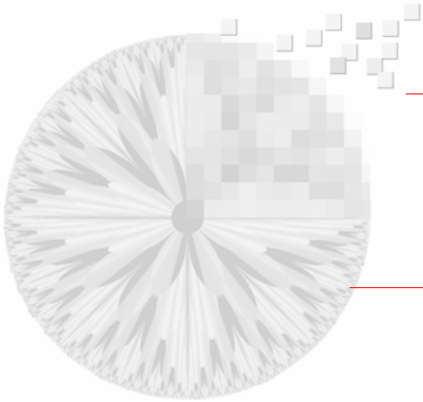
- So,

$$\frac{dr}{ds} = \frac{1}{p_r(r)}$$

Then

$$p_s(s) = p_r(r) \cdot \frac{1}{p_r(r)} = 1$$

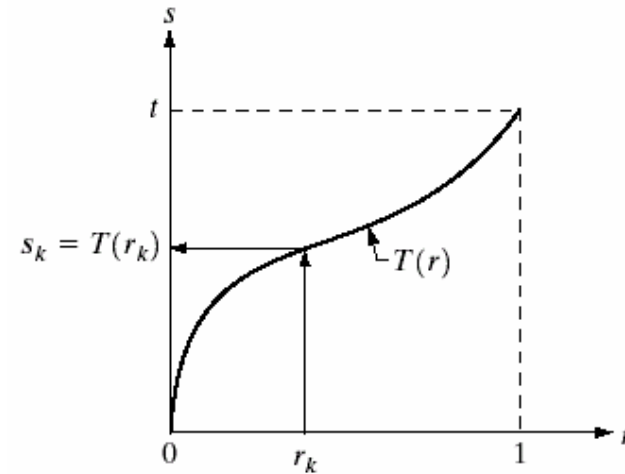
Which means that using the transformation  $T(r) = \int_0^r p_r(w)dw$  the resulting probability is uniform independent of the original image



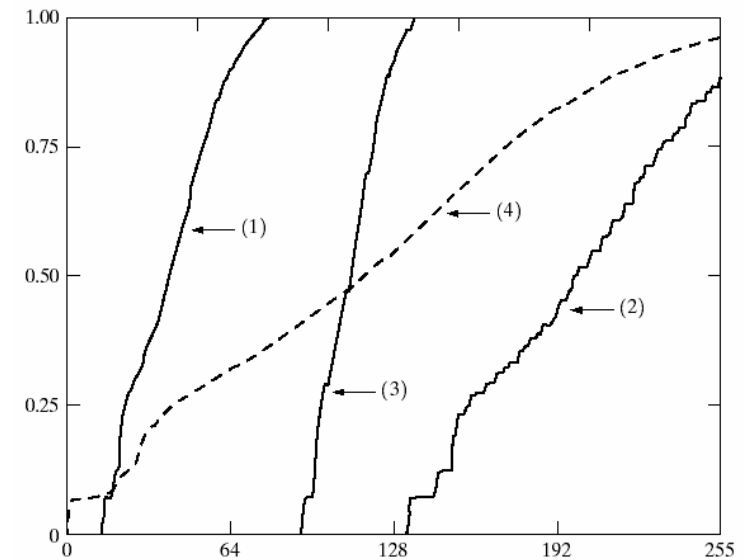
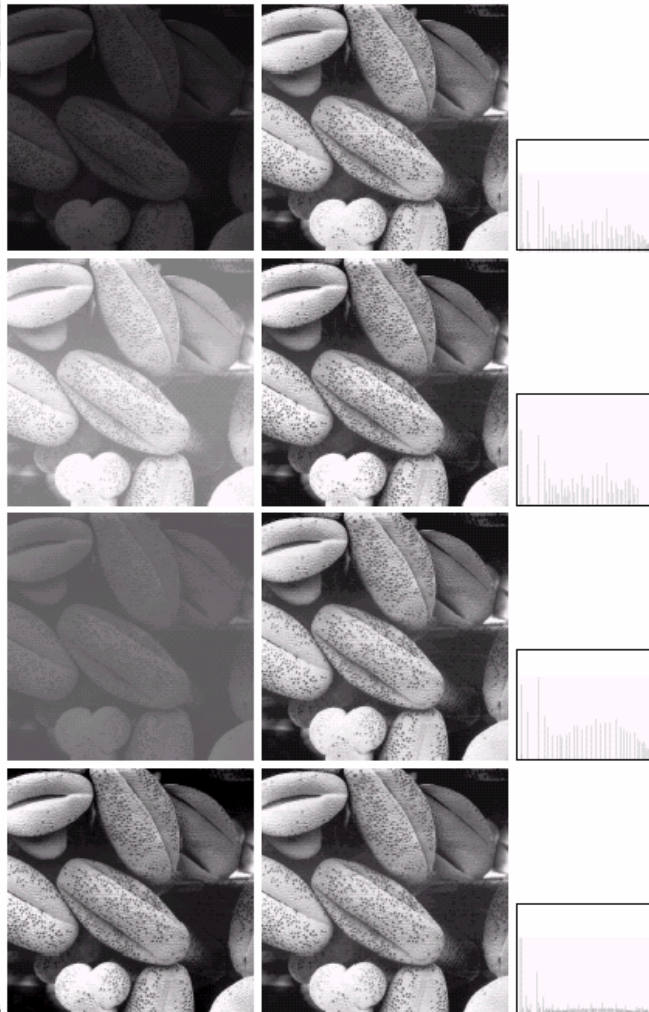
# Histogram Equalization

In discrete form :

$$s_k = \sum_{j=0}^k p_r(r_j) = \sum_{j=0}^k \frac{n_j}{n} \quad 0 \leq k \leq L-1$$

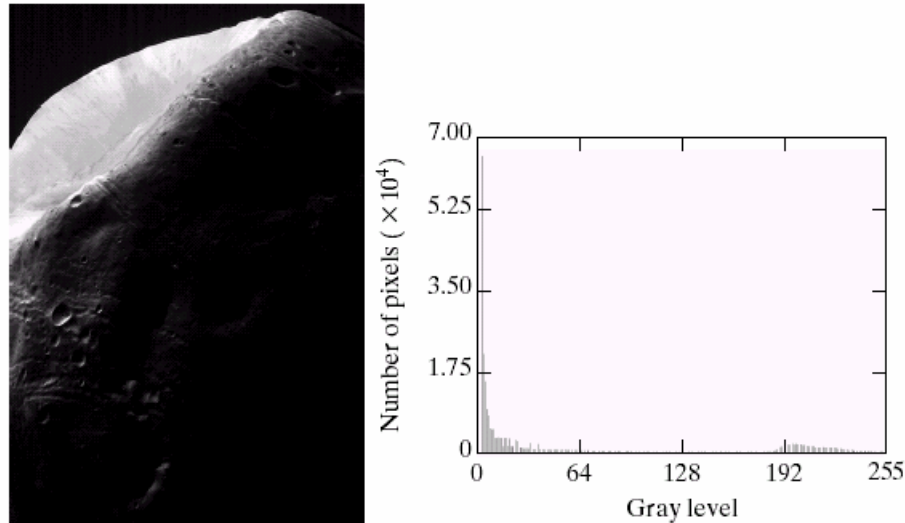


# Histogram Equalization



Transformation Functions

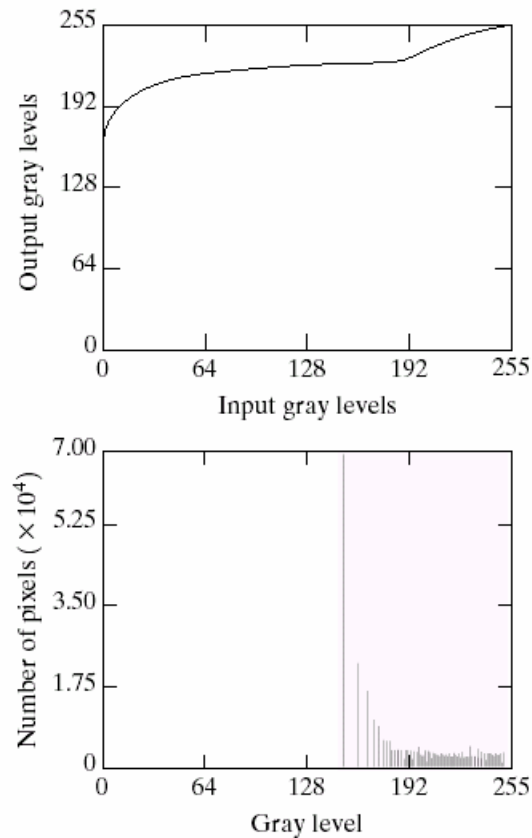
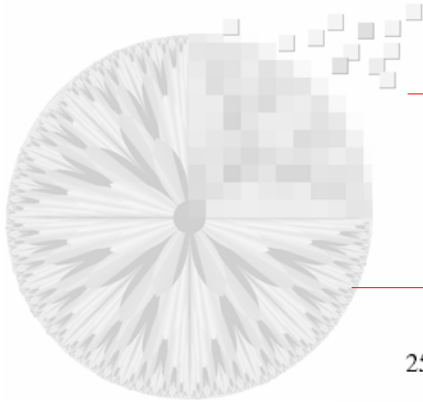
# Histogram Equalization



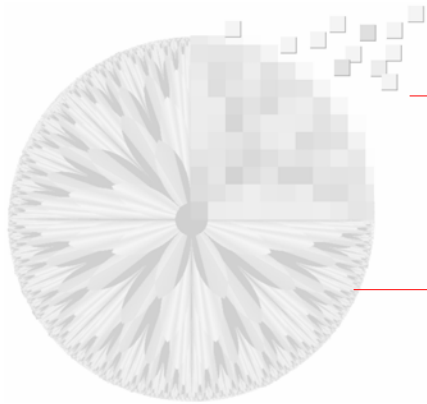
a b

**FIGURE 3.20** (a) Image of the Mars moon Phobos taken by NASA's *Mars Global Surveyor*. (b) Histogram. (Original image courtesy of NASA.)

# Histogram Equalization



**FIGURE 3.21**  
 (a) Transformation function for histogram equalization.  
 (b) Histogram-equalized image (note the washed-out appearance).  
 (c) Histogram of (b).



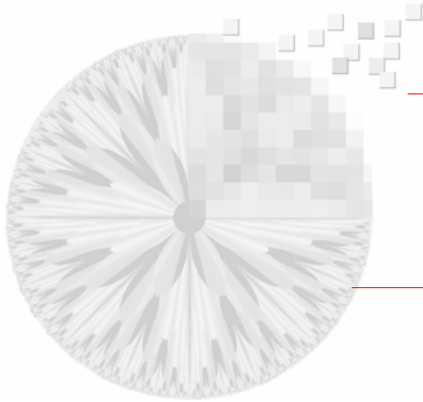
# Workshop

1. Obtain the histogram equalization curve for the following example

0	0	2	2
1	1	2	5
1	1	3	4
2	2	3	4

Using PhotoShop

2. Calculate the Histogram: Image->Histogram
3. Perform Histogram Equalization

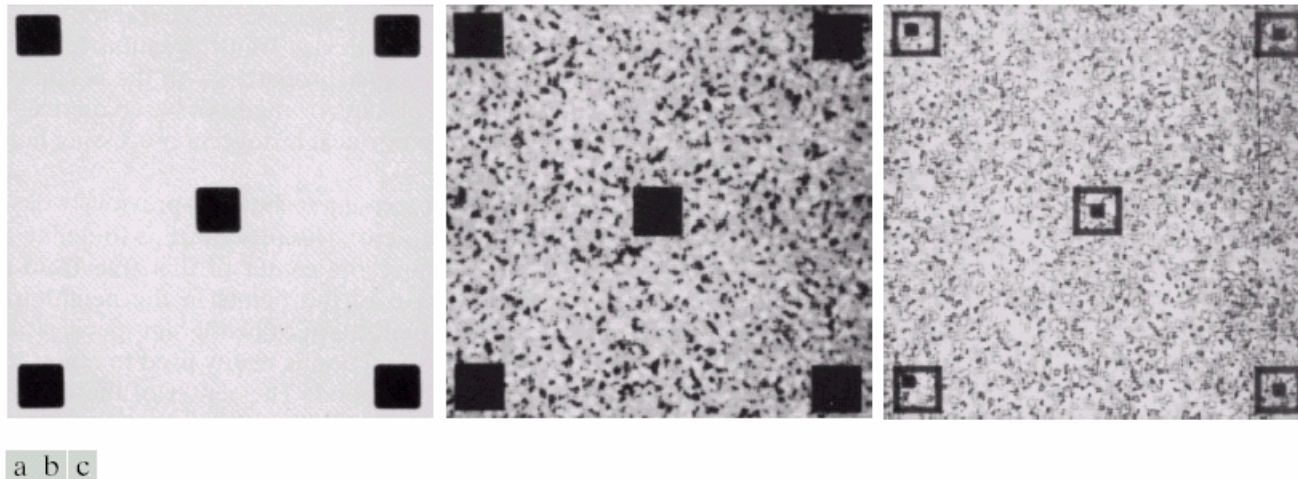


# Local Enhancement

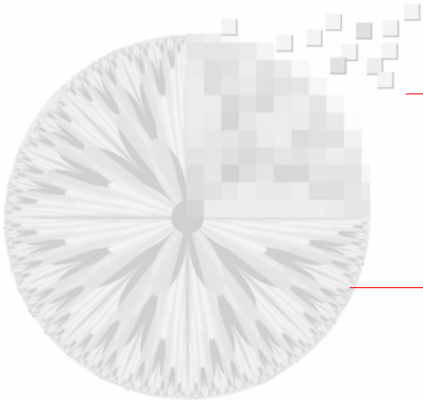
- Instead of calculating the histogram for the whole image and then do histogram equalization,
  - First divide the image into blocks
  - Perform histogram equalization on each block



# Local Histogram Equalization



**FIGURE 3.23** (a) Original image. (b) Result of global histogram equalization. (c) Result of local histogram equalization using a  $7 \times 7$  neighborhood about each pixel.



# Local Statistics

- From the local histogram, we can compute the  $n^{th}$  moment

$$\mu_n(r) = \sum_{i=0}^{L-1} (r_i - m)^n p(r_i)$$

where

$$m = \sum_{i=0}^{L-1} r_i p(r_i)$$

$$\mu_0 = 1$$

$$\mu_1 = 0$$

$$\mu_2 = \sum_{i=0}^{L-1} (r_i - m)^2 p(r_i) = \sigma^2$$

← Variance

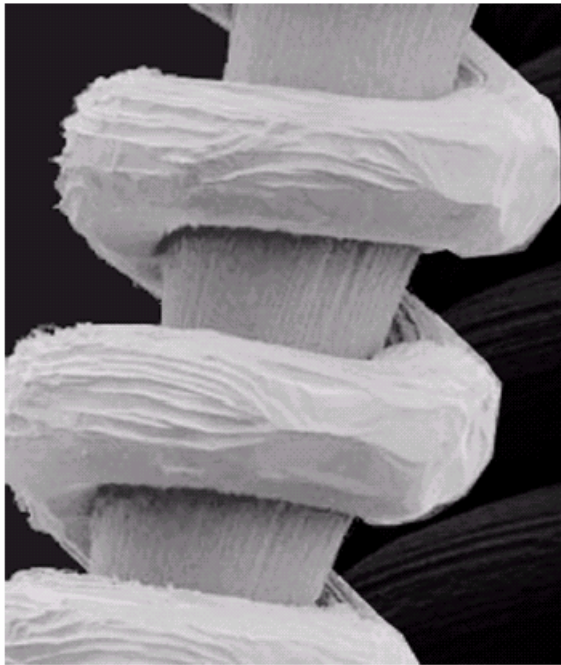


# Enhancement By Local Statistics

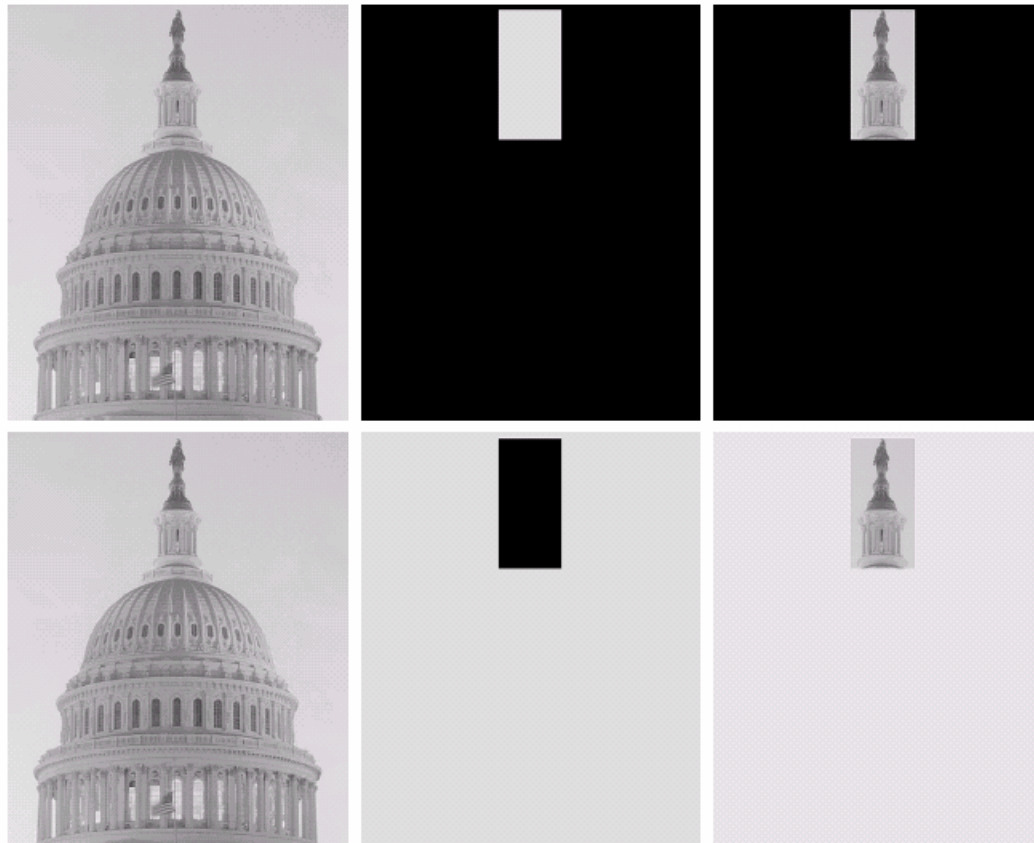
- Assume we want to change only dark areas in the image and leave light areas unchanged

$$g(x, y) = \begin{cases} \alpha f(x, y) & \text{if } m_{xy} \leq m \quad \sigma_{xy} \leq Th \\ f(x, y) & \text{otherwise} \end{cases}$$

# Enhancement By Local Statistics



# Enhancement By Arithmetic Operations



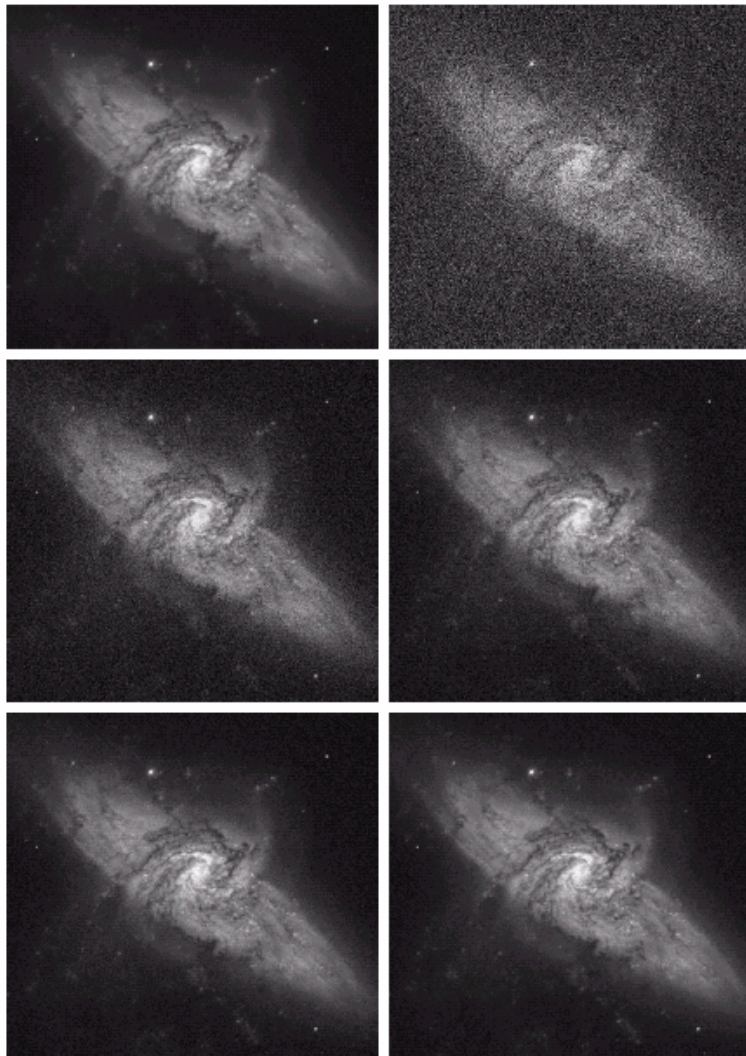
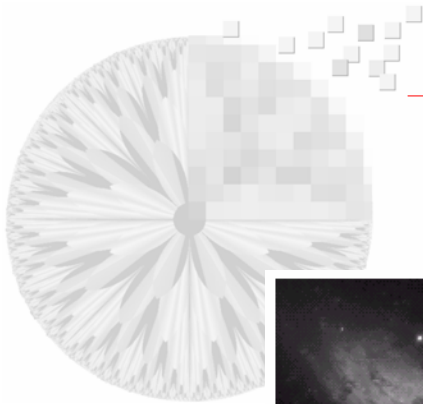
a	b	c
d	e	f

**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).



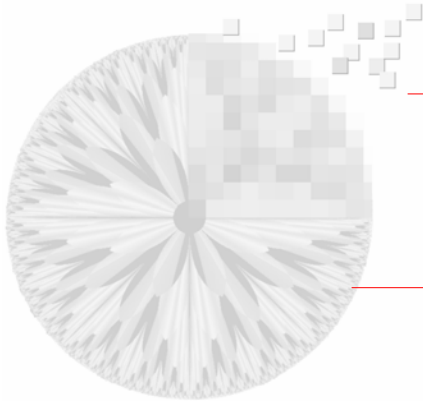
# Image Averaging



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

a	b
c	d
e	f

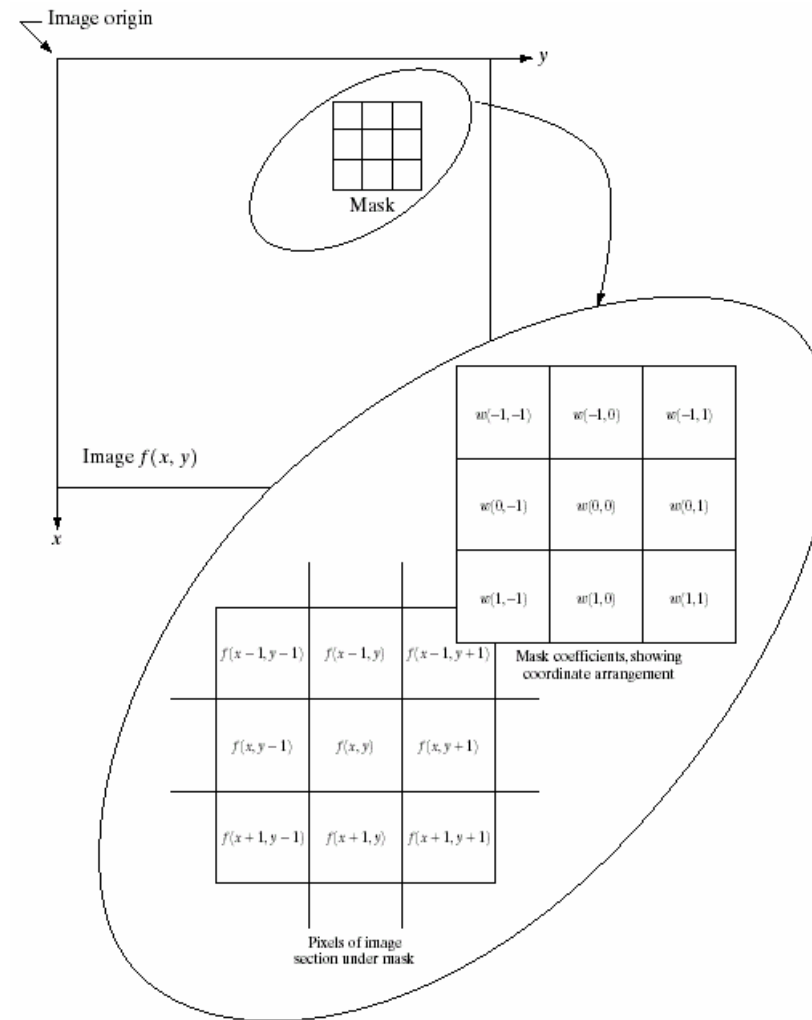
**FIGURE 3.30** (a) Image of Galaxy Pair NGC 3314. (b) Image corrupted by additive Gaussian noise with zero mean and a standard deviation of 64 gray levels. (c)–(f) Results of averaging  $K = 8, 16, 64$ , and  $128$  noisy images. (Original image courtesy of NASA.)



# Spatial Filtering

- Spatial filtering is performed by convolving the image with a mask or a kernel
- Spatial filters include sharpening, smoothing, edge detection, noise removal, etc.

# Basics of Spatial Filtering



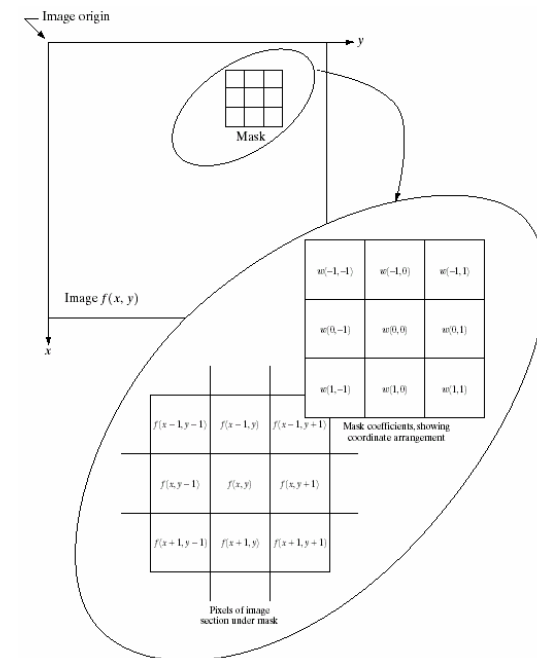


# Basics of Spatial Filtering

- In general, linear filtering of an image  $f$  of size  $M \times N$  with filter size  $m \times n$  is given by the expression

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

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





# Smoothing Spatial Filters

- The output of a smoothing spatial filter is simply the average of the pixels contained in the neighborhood of the filter mask.
- These filters are sometimes called *averaging filters* and also *lowpass filters*
- By replacing the value of the pixel with the average of a window around it, the result is an image with reduced sharp transitions



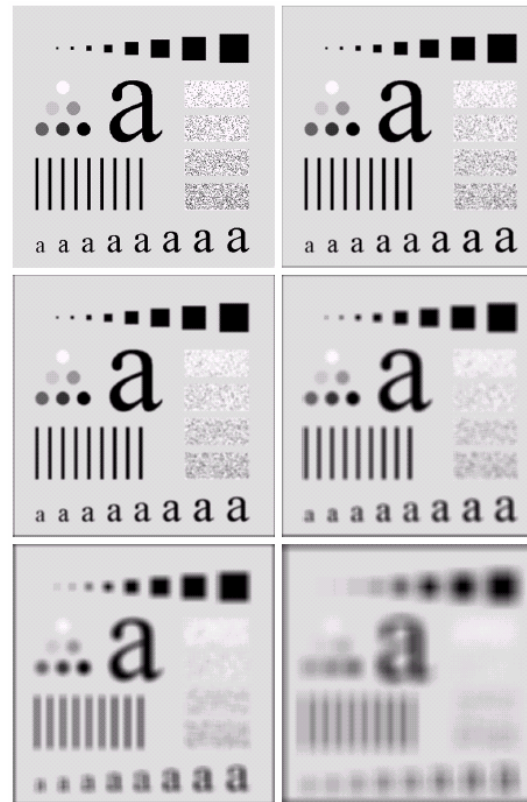
# Smoothing Spatial Filters

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

In general

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

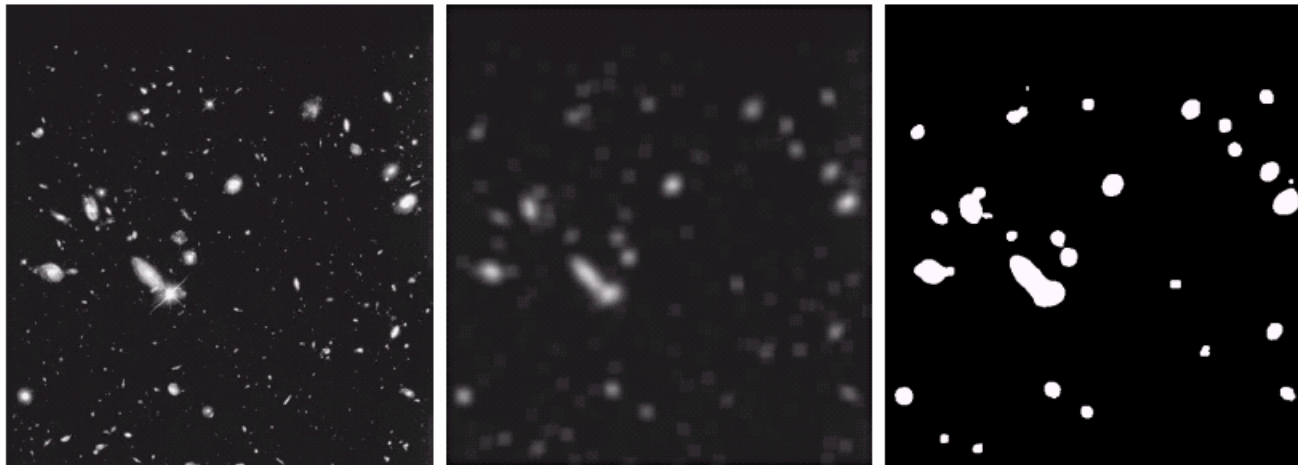
# Smoothing Spatial Filters



a b  
c d  
e f

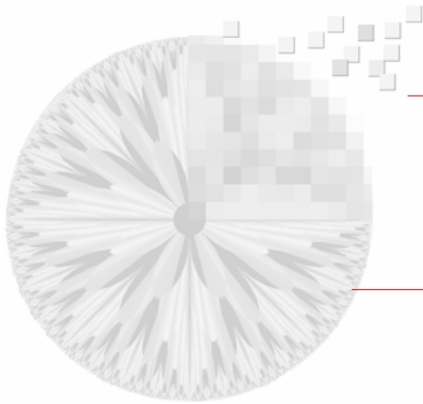
**FIGURE 3.35** (a) Original image, of size  $500 \times 500$  pixels. (b)–(f) Results of smoothing with square averaging filter masks of sizes  $n = 3, 5, 9, 15$ , and  $35$ , respectively. The black squares at the top are of sizes  $3, 5, 9, 15, 25, 35, 45$ , and  $55$  pixels, respectively; their 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  $50 \times 120$  pixels.

# Smoothing Spatial Filters



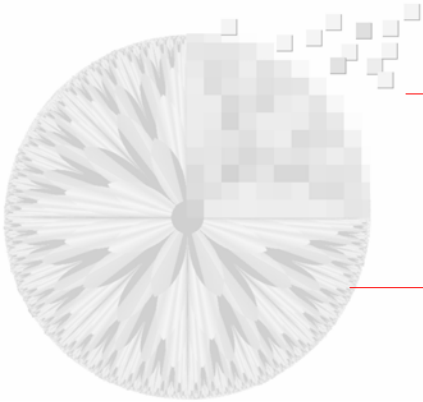
a b c

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



# Order Statistics Filters

- Order statistics filters are nonlinear spatial filters whose response is based on ordering (ranking) the pixels contained in an area covered by the filter
- The best known example in this category is *median filter*
- Median filters replace the value of the pixel by the median of the gray levels in the neighborhood of that pixel



# Median Filter

- Example

10	20	20
20	15	20
20	25	100

10	20	20
20	20	20
20	25	100

Order

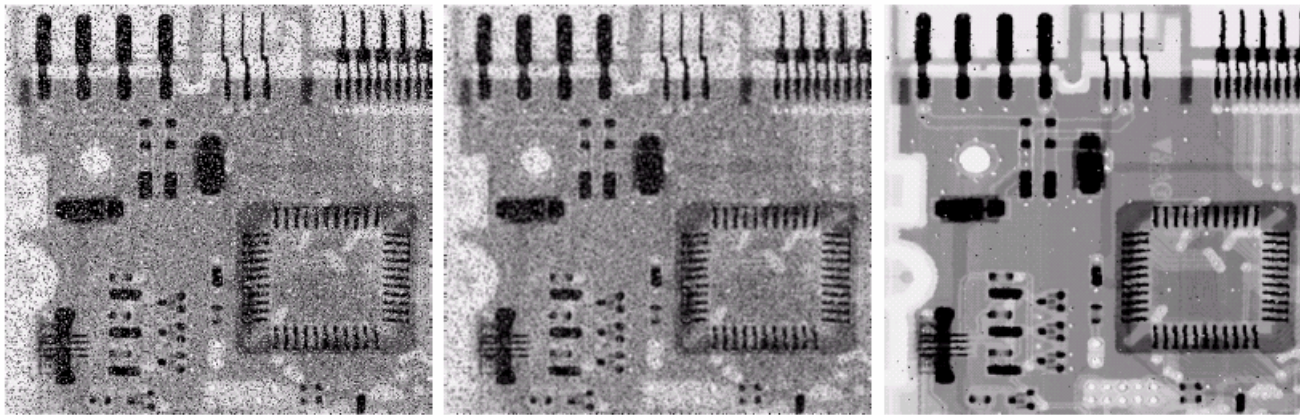


10 15 20 20 20 20 20 25 100



Median value

# 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  $3 \times 3$  averaging mask. (c) Noise reduction with a  $3 \times 3$  median filter. (Original image courtesy of Mr. Joseph E. Pascente, Lixi, Inc.)



# Multi Pass Median Filter



a b  
c d

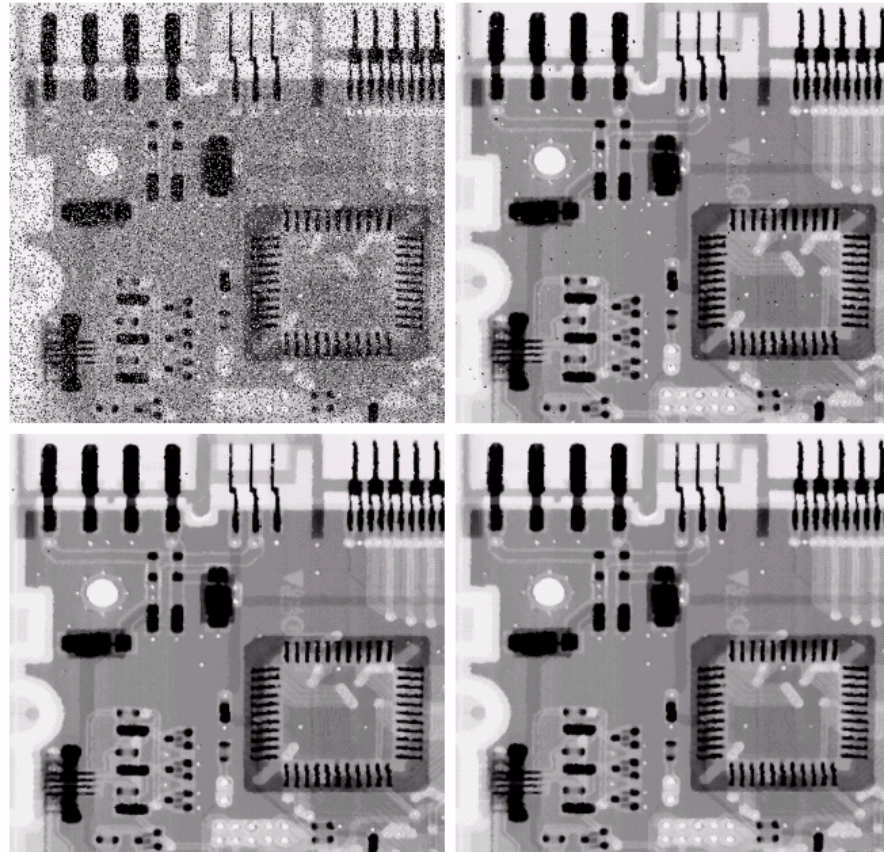
**FIGURE 5.10**

(a) Image corrupted by salt-and-pepper noise with probabilities  $P_a = P_b = 0.1$ .

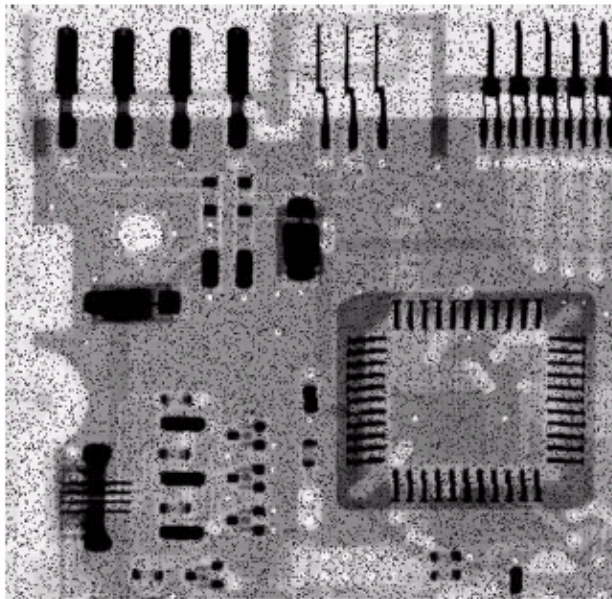
(b) Result of one pass with a median filter of size  $3 \times 3$ .

(c) Result of processing (b) with this filter.

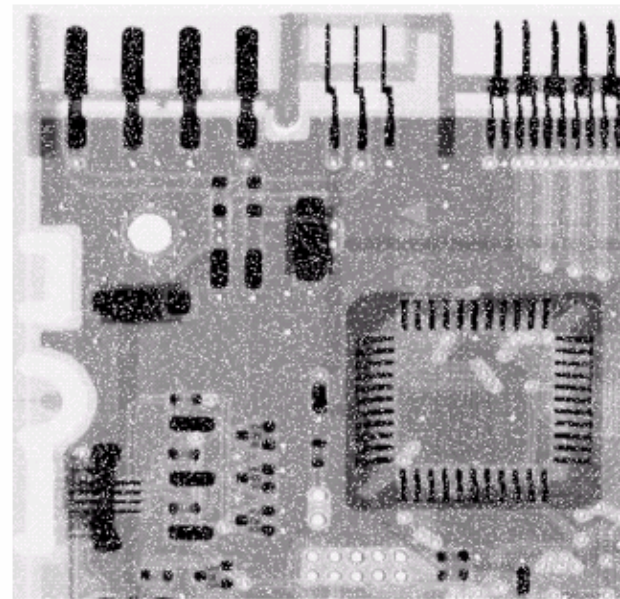
(d) Result of processing (c) with the same filter.



# Other Order Statistics Filters

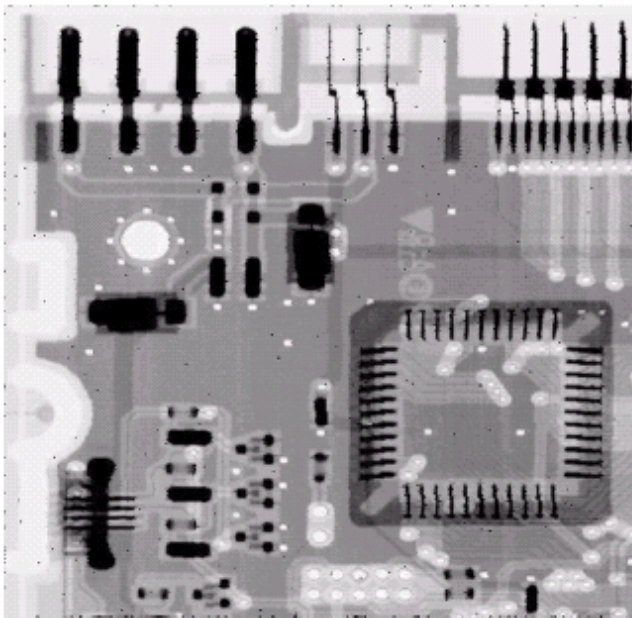


Image+Pepper Noise

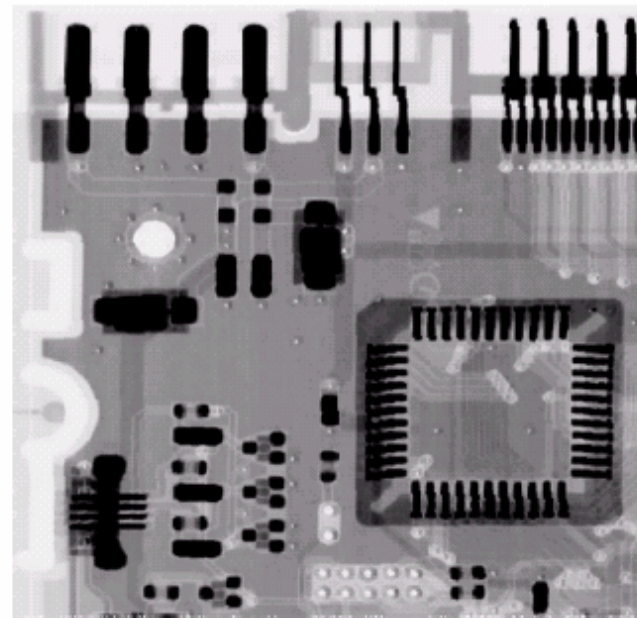


Image+Salt Noise

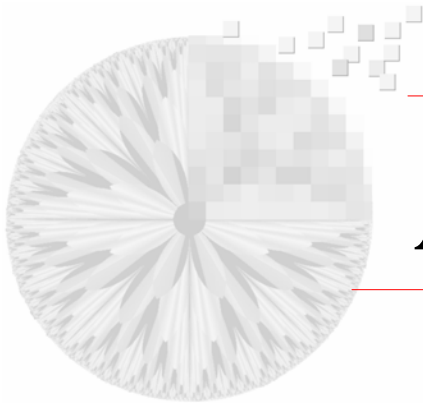
# Other Order Statistics Filters



Max Filter



Min Filter



# Adaptive Median Filter

- We want to preserve the detail while smoothing non impulse noise.
- Vary the size of the window.
- Algorithm:

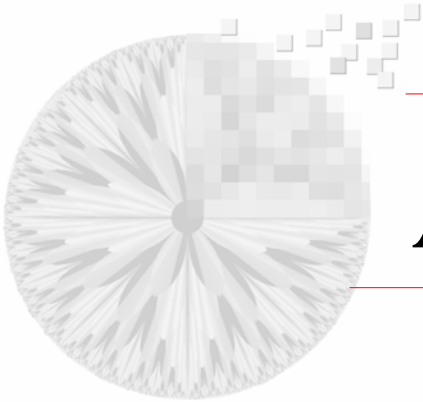
Let

$$z_{\min} = \min \text{ graylevel in } S_{xy}$$

$$z_{\max} = \max \text{ graylevel in } S_{xy}$$

$$z_{\text{med}} = \text{median graylevel in } S_{xy}$$

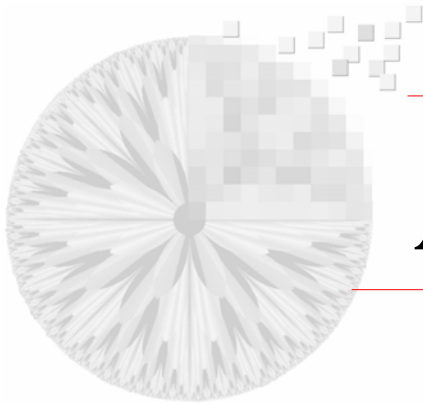




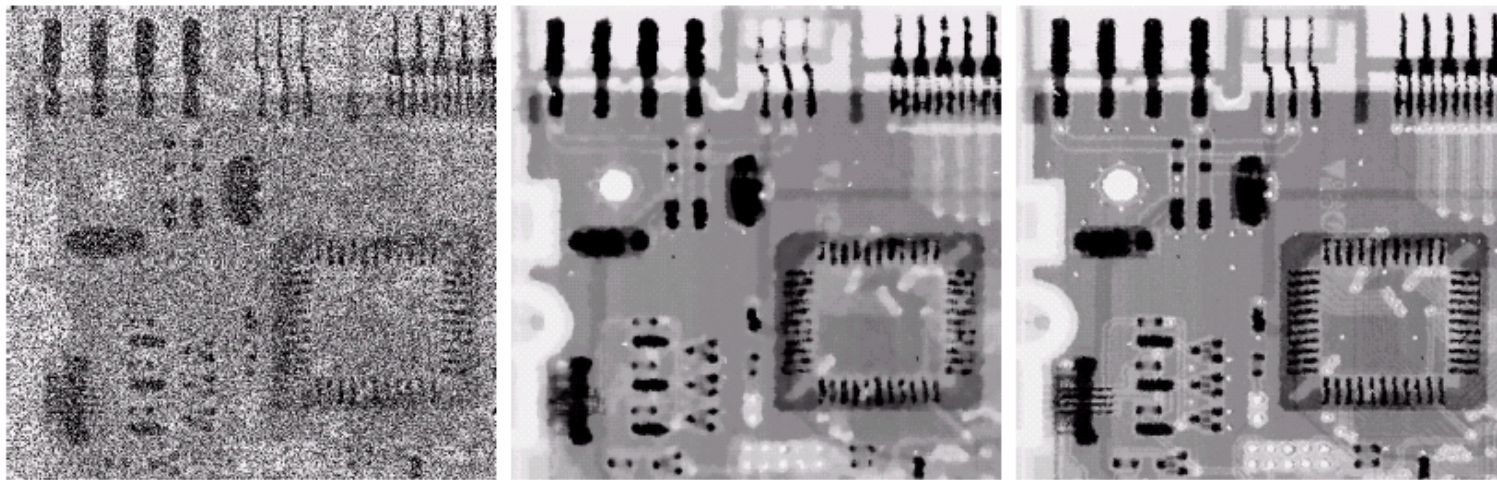
# Adaptive Median Filter

A:  $A1 = z_{med} - z_{min}$   
 $A2 = z_{med} - z_{max}$   
*if  $A1 > 0$  AND  $A2 < 0$  Goto B*  
*Else increase window size*  
*If window size  $< S_{max}$ , Goto A*  
*Else output  $z_{xy}$*

B:  $B1 = z_{xy} - z_{min}$   
 $B2 = z_{xy} - z_{max}$   
*if  $B1 > 0$  AND  $B2 < 0$ , output  $z_{xy}$*   
*Else output  $z_{med}$*



# Adaptive Median Filter



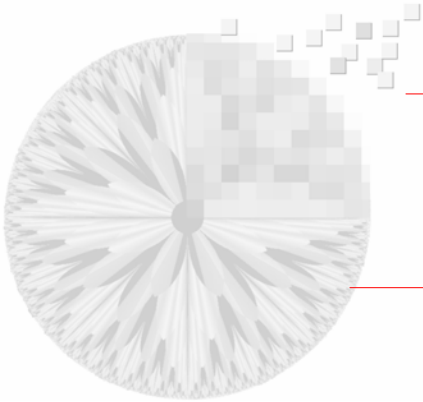
a b c

**FIGURE 5.14** (a) Image corrupted by salt-and-pepper noise with probabilities  $P_a = P_b = 0.25$ . (b) Result of filtering with a  $7 \times 7$  median filter. (c) Result of adaptive median filtering with  $S_{\max} = 7$ .



# Sharpening Spatial Filters

- The principal objective of sharpening is to highlight fine details in an image or to enhance details that has been blurred.
- We saw before that image blurring could be accomplished by pixel averaging, which is analogous to *integration*.
- Sharpening could be accomplished by *spatial differentiation*
- In this section, we will define operators for sharpening by digital differentiation
- Fundamentally, the strength of the response of the operator should be proportional to the degree of discontinuity (presence of edges).



# Digital Differentiation

- A basic definition of the first-order derivative at one dimensional function  $f(x)$  is the difference

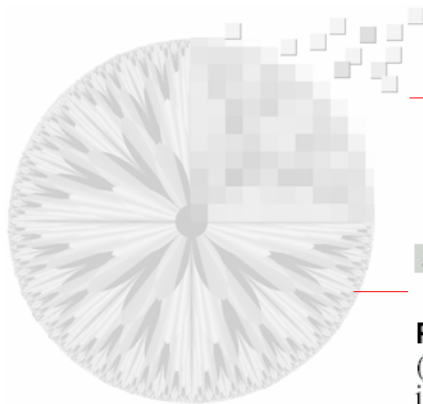
$$\frac{\partial f}{\partial x} = f(x+1) - f(x)$$

- The second order derivative

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



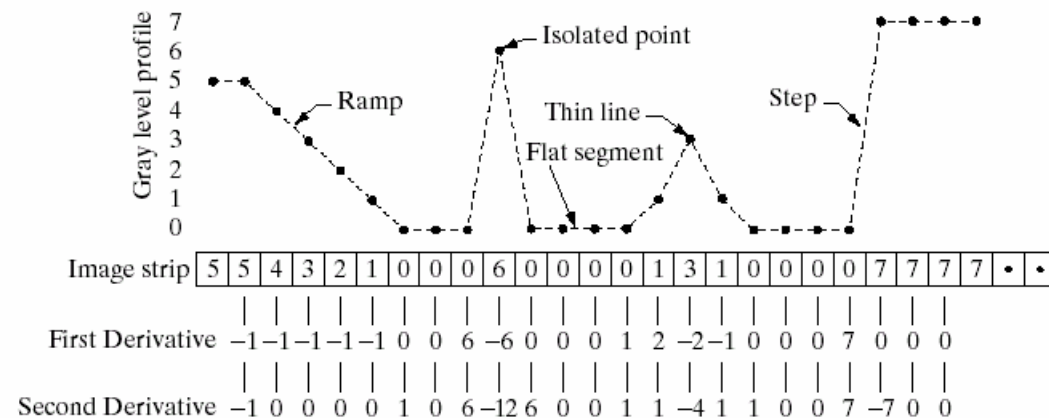
# Digital Differentiation

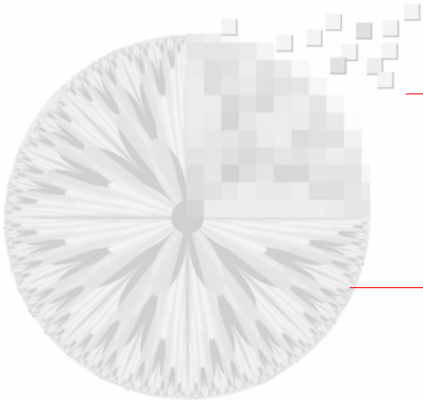


a b  
c

**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).





# The Laplacian $\nabla^2 f$

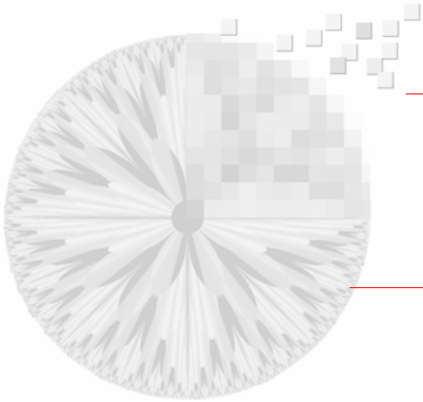
- The *Laplacian* of an image is define as

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

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

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

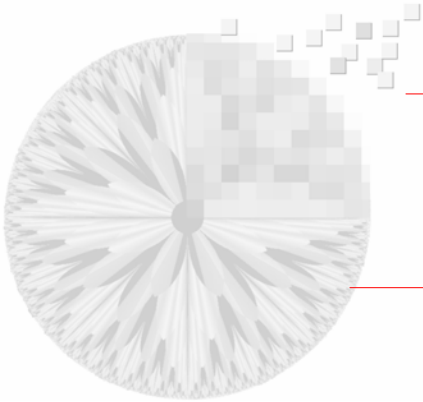
$$\nabla^2 f = [f(x+1, y) + f(x-1, y) + f(x, y+1) + f(x, y-1)] - 4f(x, y)$$



# The Laplacian $\nabla^2 f$

0	1	0	1	1	1
1	-4	1	1	-8	1
0	1	0	1	1	1

0	-1	0	-1	-1	-1
-1	4	-1	-1	8	-1
0	-1	0	-1	-1	-1

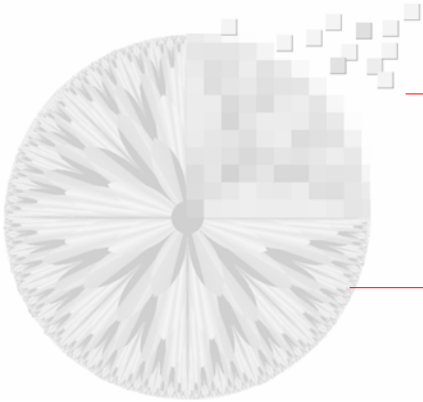


# Sharpening Mask

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

-1	-1	-1
-1	9	-1
-1	-1	-1

$$g(x+y) = \begin{cases} \nabla^2 f - f(x+y) \\ \nabla^2 f + f(x+y) \end{cases}$$



a b  
c d

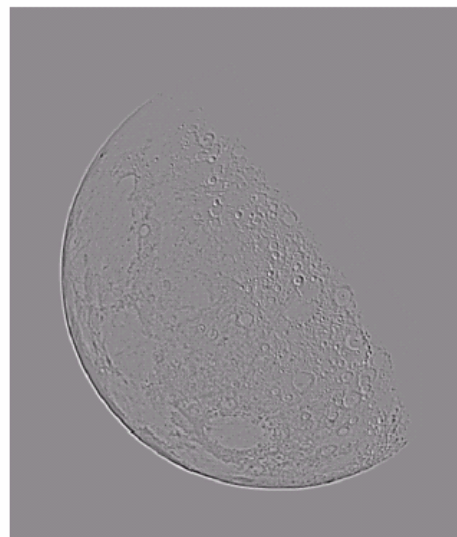
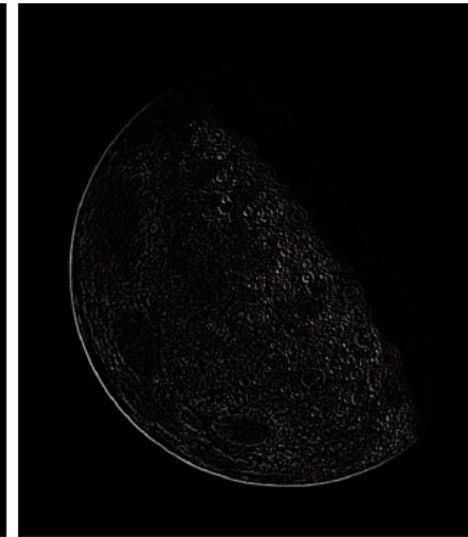
**FIGURE 3.40**

(a) Image of the North Pole of the moon.

(b) Laplacian-filtered image.

(c) Laplacian image scaled for display purposes.

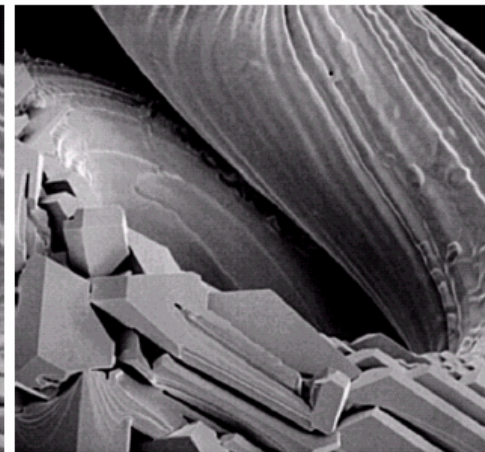
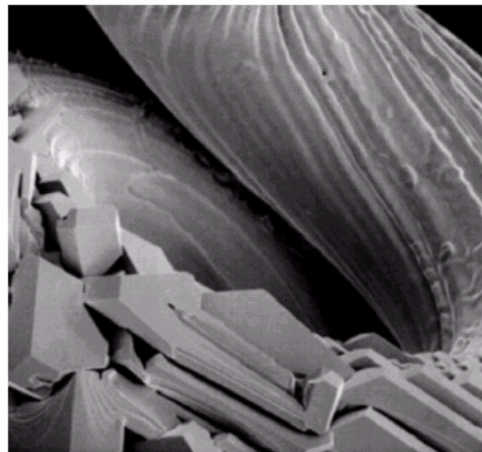
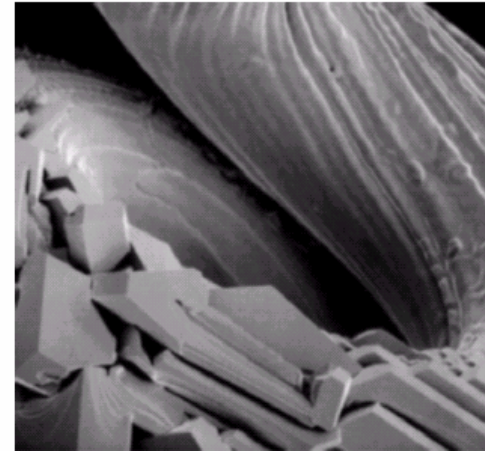
(d) Image enhanced by using Eq. (3.7-5). (Original image courtesy of NASA.)



# Sharpening Spatial Filters

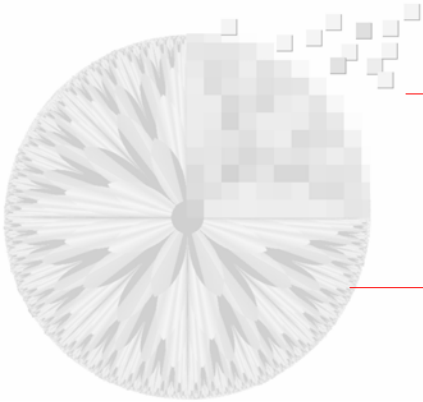
0	-1	0
-1	5	-1
0	-1	0

-1	-1	-1
-1	9	-1
-1	-1	-1



a b c  
d e

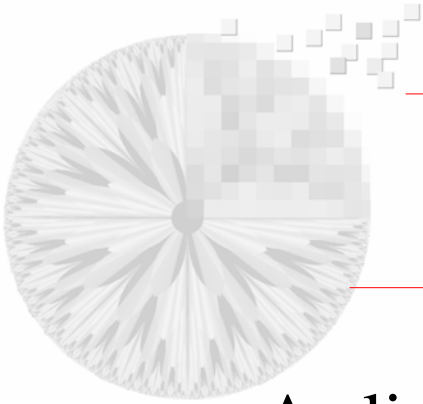
**FIGURE 3.41** (a) Composite Laplacian mask. (b) A second composite mask. (c) Scanning electron microscope image. (d) and (e) Results of filtering with the masks in (a) and (b), respectively. Note how much sharper (e) is than (d). (Original image courtesy of Mr. Michael Shaffer, Department of Geological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



# Unsharp Masking

- A process used for many years in the publishing industry to sharpen images.
- It consists of subtracting a blurred version of the image from the image itself

$$f_s(x, y) = f(x, y) - \tilde{f}(x, y)$$



# High Boost Filters

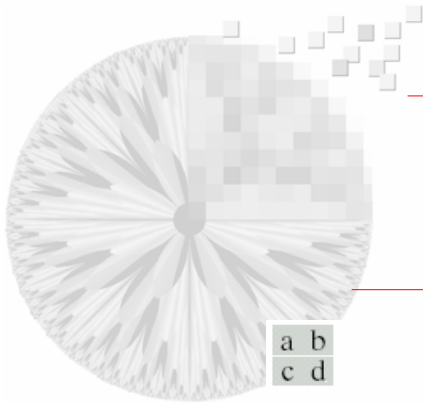
A slight generalization of unsharp masking is called high boost filters

$$f_{hb}(x, y) = A f(x, y) - \tilde{f}(x, y)$$

0	-1	0	-1	-1	-1
-1	$A + 4$	-1	-1	$A + 8$	-1
0	-1	0	-1	-1	-1



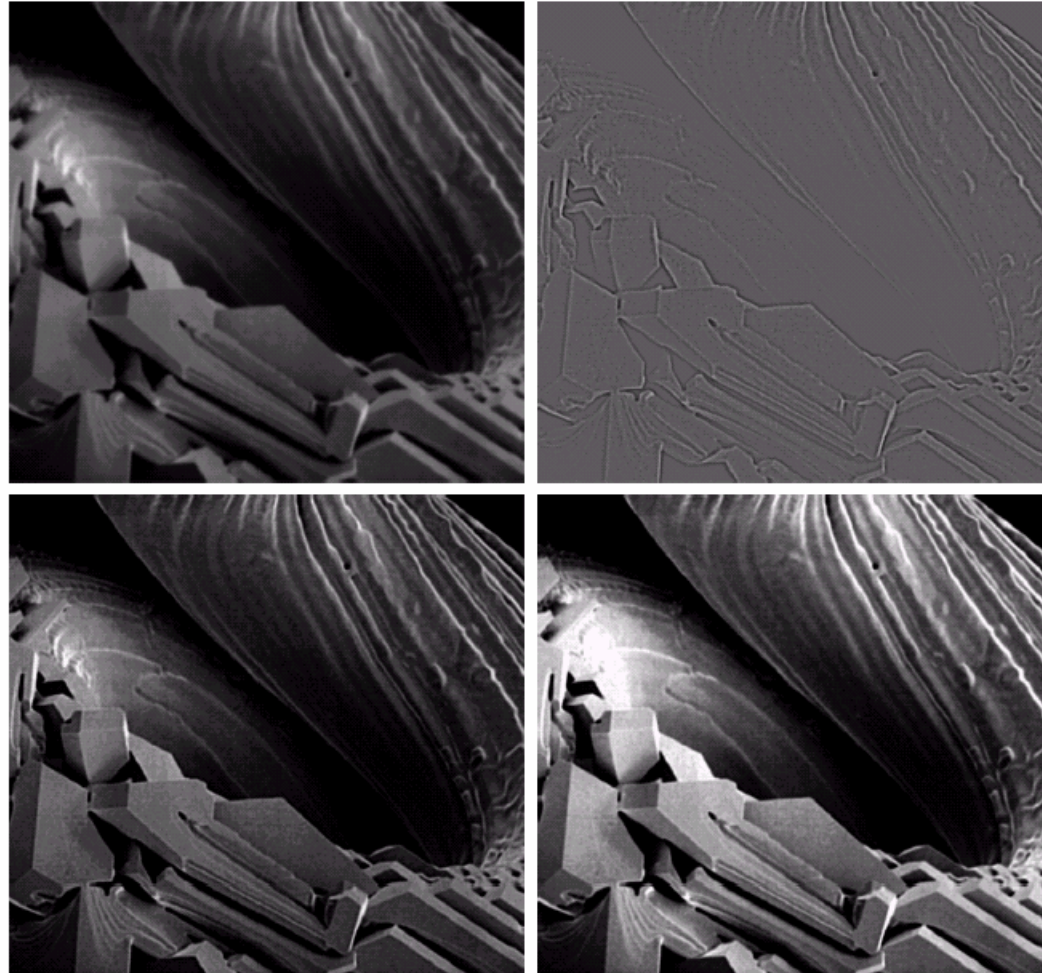
# High Boost Filters



a b  
c d

**FIGURE 3.43**

(a) Same as Fig. 3.41(c), but darker.  
(b) Laplacian of (a) computed with the mask in Fig. 3.42(b) using  $A = 0$ .  
(c) Laplacian enhanced image using the mask in Fig. 3.42(b) with  $A = 1$ . (d) Same as (c), but using  $A = 1.7$ .



a	
b	c
d	e

A  $3 \times 3$  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.

$z_1$	$z_2$	$z_3$
$z_4$	$z_5$	$z_6$
$z_7$	$z_8$	$z_9$

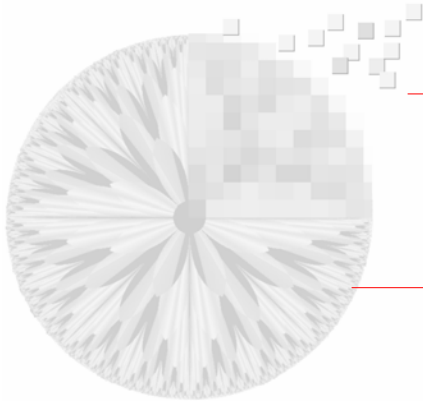
-1	0
0	1

0	-1
1	0

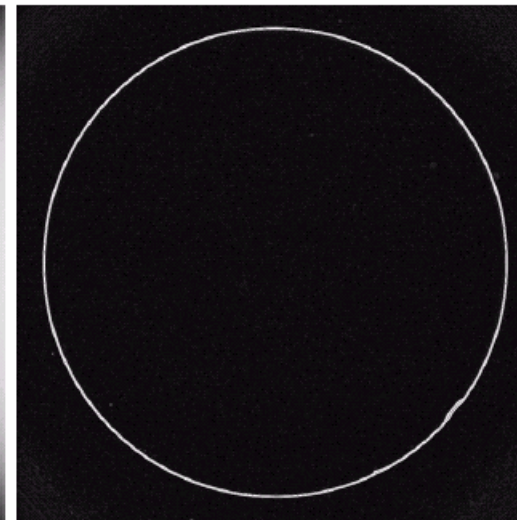
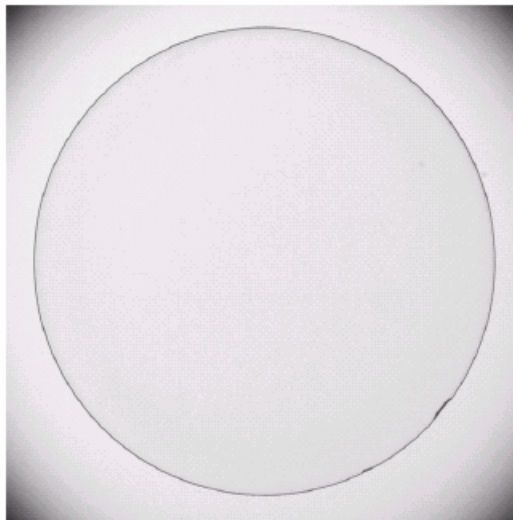
  

-1	-2	-1
0	0	0
1	2	1

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



# Edge Detection



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.)



# Anisotropic Diffusion Filter

The idea is to filter within the object not across boundaries

Therefore, image details remain unblurred while achieving Smoothness within objects

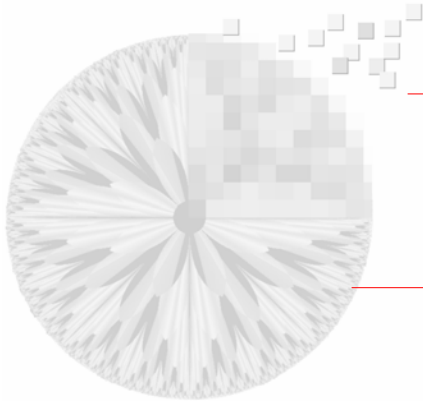
The filtering is modeled as a diffusion process that stops at image boundaries



# Anisotropic Diffusion Filter

$$\frac{\partial f(\bar{x}, t)}{\partial t} = \text{div}(c(\bar{x}) \cdot \nabla f(\bar{x}, t))$$

$$c(\bar{x}) = e^{-\left[\frac{\|\nabla f(\bar{x}, t)\|^2}{K}\right]}$$



# Thank You