Chapter-2

Digital Imaging Fundamentals

(4th Edition)

Contents

Human Visual System

- The best vision system known to mankind

- Understanding how images form in the eye can help us with processing digital images

Structure of the Human Eye

- **Lens:** Focuses light from objects onto the Retina
- **Retina:** Covered with light receptors
  - *Cones* (6-7 million) and
  - *Rods* (75-150 million)

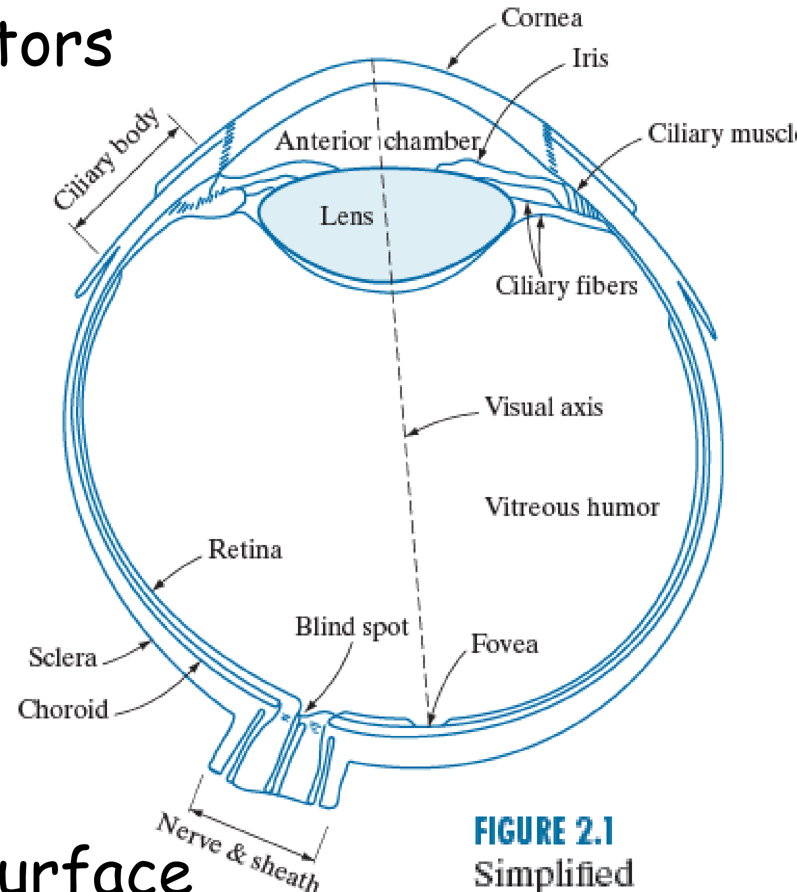


FIGURE 2.1 Simplified diagram of a cross section of the human eye.

- **Cones**
  - Concentrated around Fovea
  - Very sensitive to color
- **Rods**
  - More spread out over retina surface
  - Sensitive to low levels of illumination

Distribution of Rods and Cones in the Retina

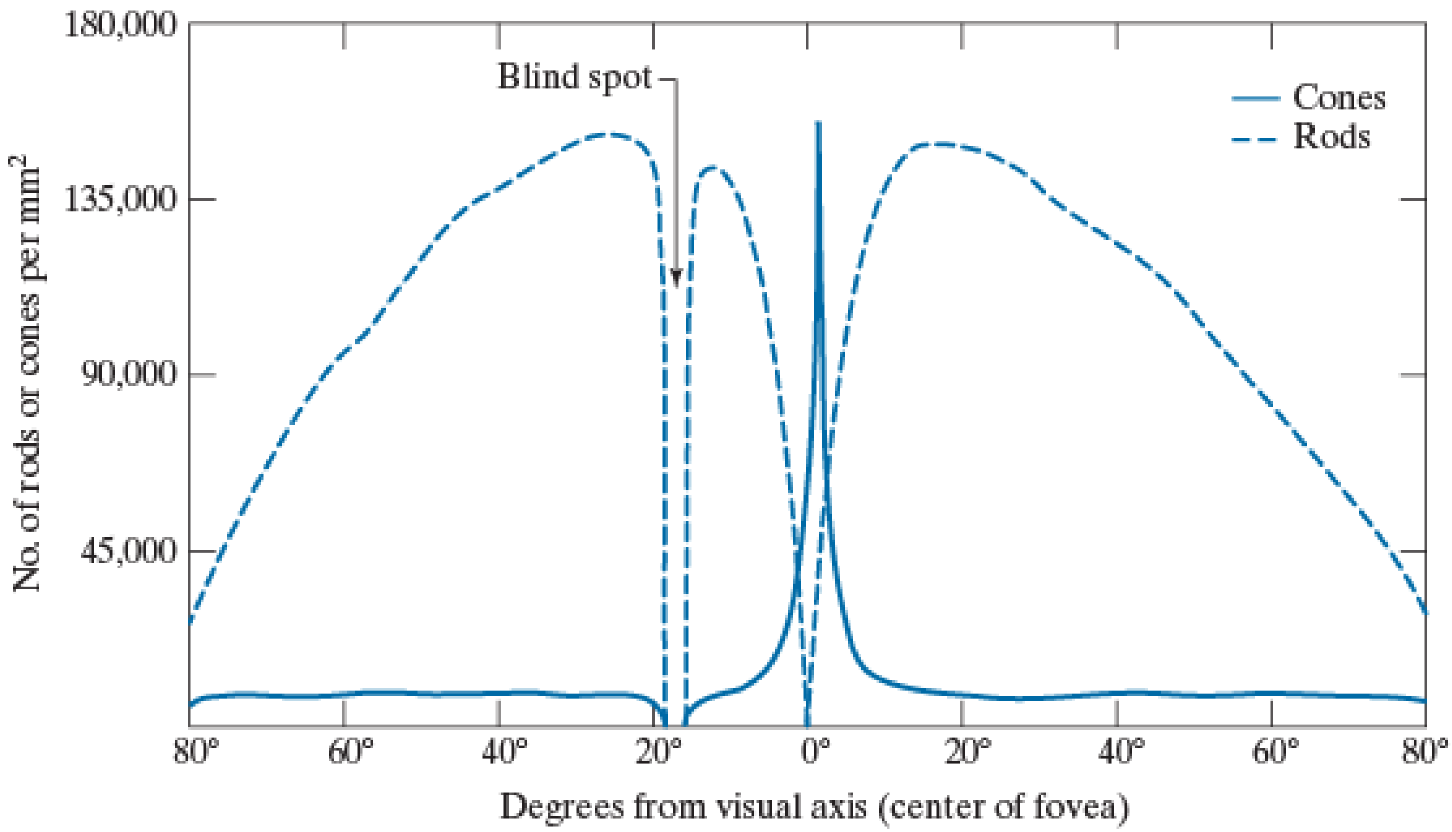


FIGURE 2.2 Distribution of rods and cones in the retina.

Blind-Spot Experiment

- Draw an image similar to that below on a piece of paper (the dot and cross should be about 6 inches apart)

- Close your right eye and focus on the cross with your left eye

- Hold the image about 20 inches away from your face and move it slowly towards you

- The dot should disappear!

Image Formation in the Eye

- **Muscles within the eye**
  - Used to change the shape of the lens
  - Allow us to focus on objects that are near or far away
- **Image focused onto the retina**
  - Causes rods and cones to become excited which ultimately send signals to the brain

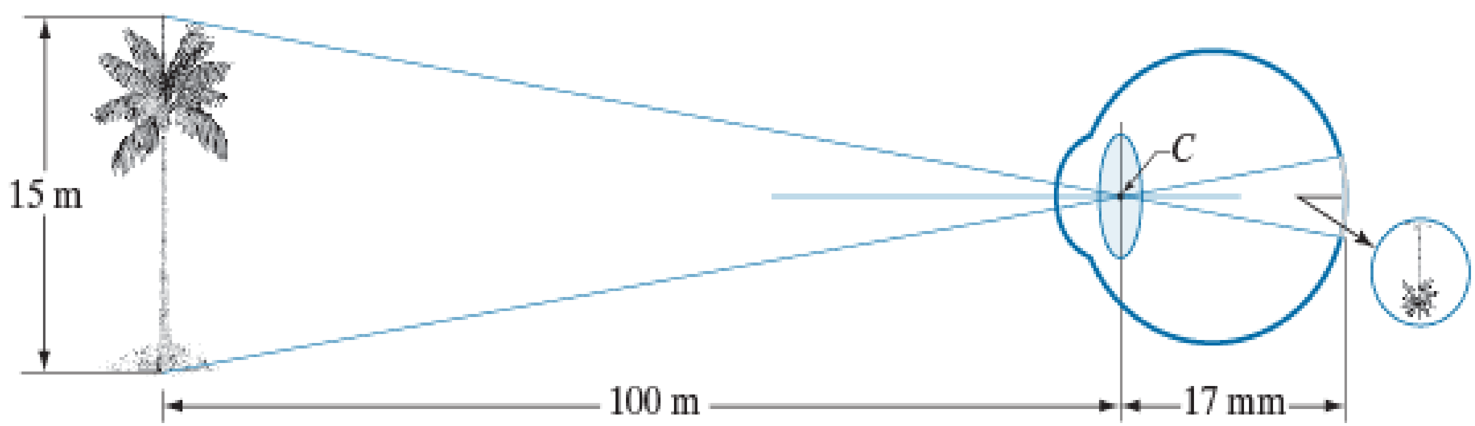
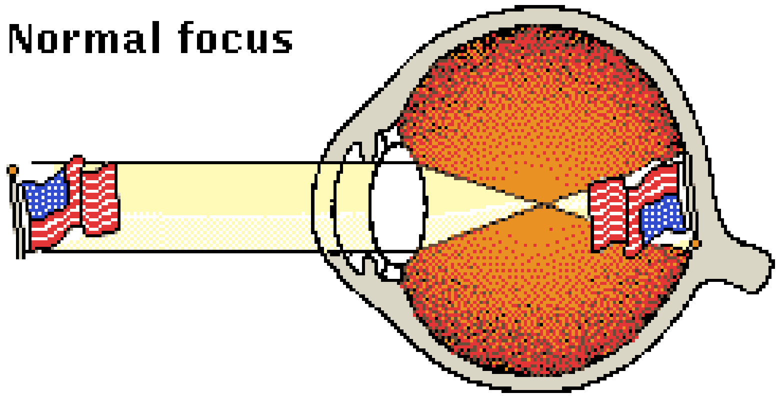
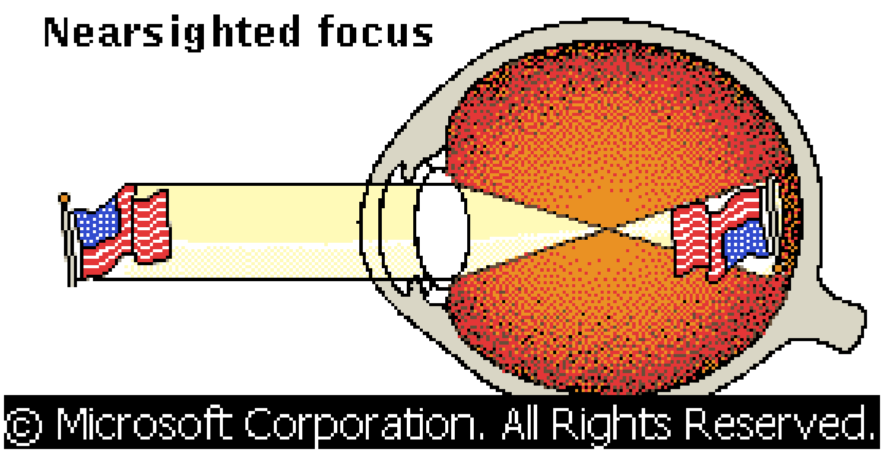


FIGURE 2.3 Graphical representation of the eye looking at a palm tree. Point *C* is the focal center of the lens.

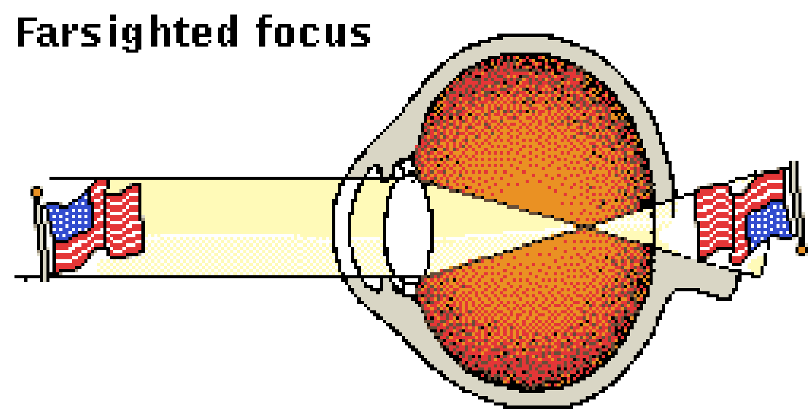
Image Formation in the Eye: Near-Sighted & Far-Sighted



Need negative power for correction

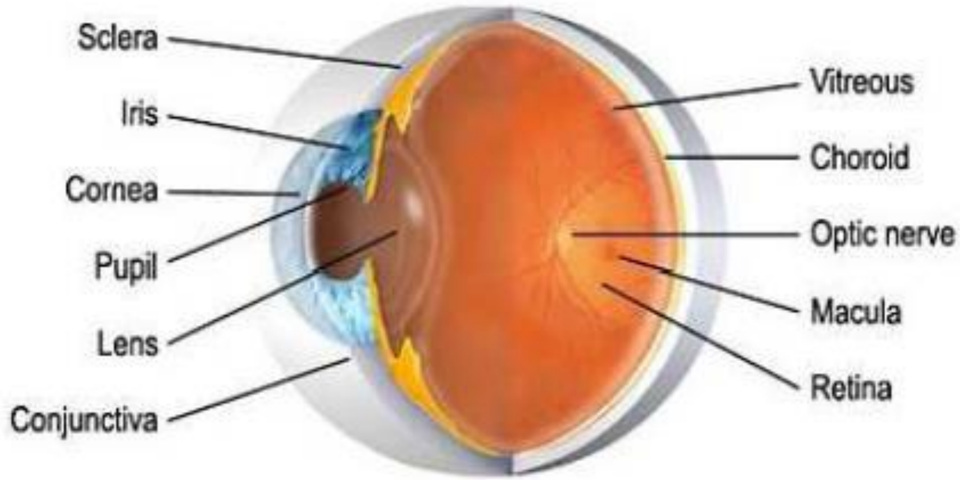


Need positive power for correction



Picture Courtesy of Microsoft Encarta 2000

Eye vs. Camera



Camera components	Eye components
Lens	Lens, cornea
Shutter	Iris, pupil
Film	Retina
Cable to transfer images	Optic nerve send the info to the brain

Contrast Sensitivity

- Response of eye due to changes in intensity of luminance is → Nonlinear.
- **Weber's Law:** Human visual perception is sensitive to luminance contrast rather than absolute luminance values

$$\frac{|I - I_s|}{I} = \frac{\Delta I}{I} = \Delta c \cong d \ \log I \ \cong 0.02$$

- I : Luminance of the object
- I_s : Luminance of its surrounds
- $I - I_s = \Delta I$: Minimum noticeable difference in luminance

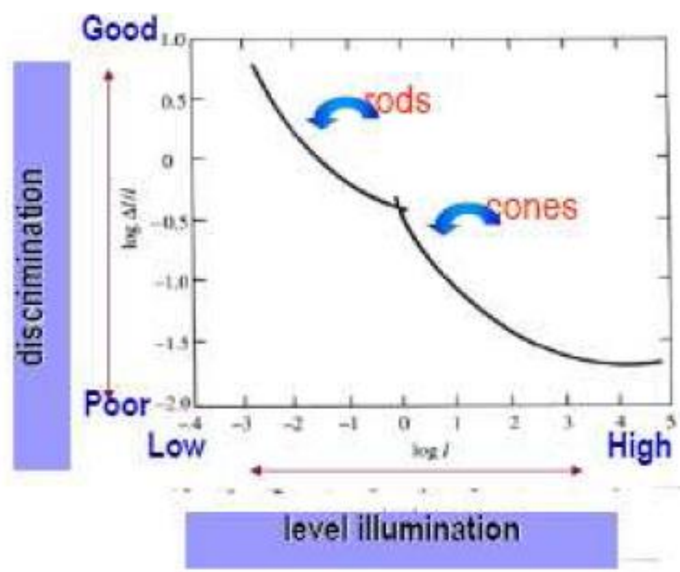
Brightness Adaptation & Discrimination

- **Human Visual System**
  - Can perceive approximately 10^{10} different light intensity levels
- **Brightness Adaptation**
  - At any one time our eyes can only discriminate between a much smaller range of intensity
  - The ability of the eye to discriminate between *changes* in light intensity at any specific brightness adaptation level is governed by

Weber's Ratio : $\dfrac{\Delta I}{I}$

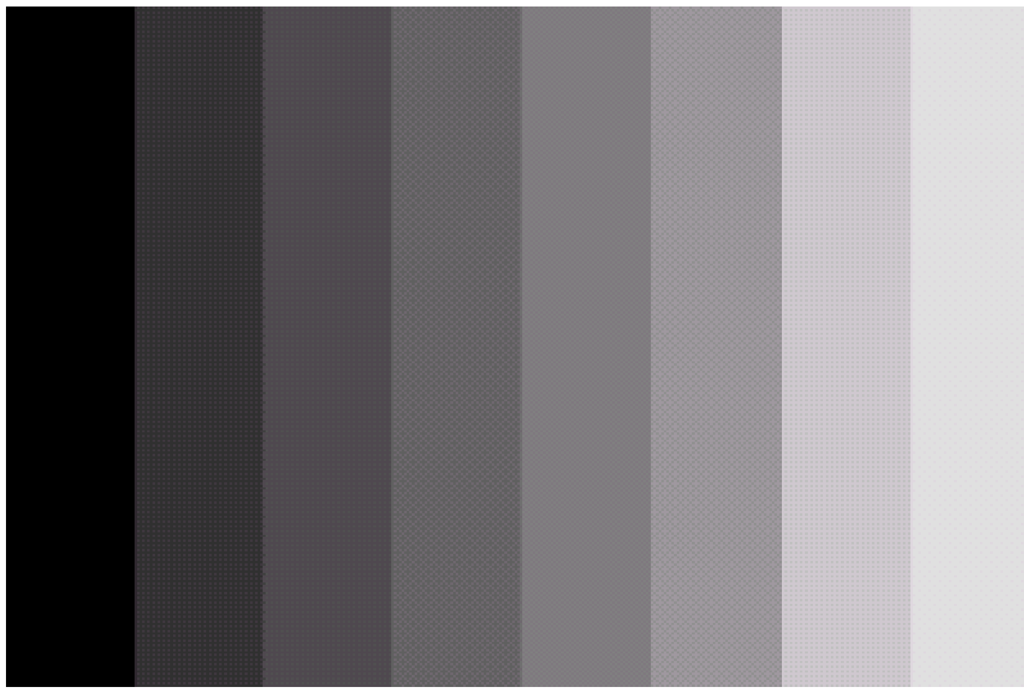
- States that the just noticeable difference ΔI is proportional to the background luminance I
- Hard to distinguish discrimination in a bright area, but easier to discriminate in a dark area

FIGURE 2.6 A typical plot of the Weber ratio as a function of intensity.



Mach Bands

- Visual Systems tend to Overshoot or Undershoot around the boundaries of different intensities
  - Mach Bands



An example of Mach bands

Mach Bands & Perceived Intensity

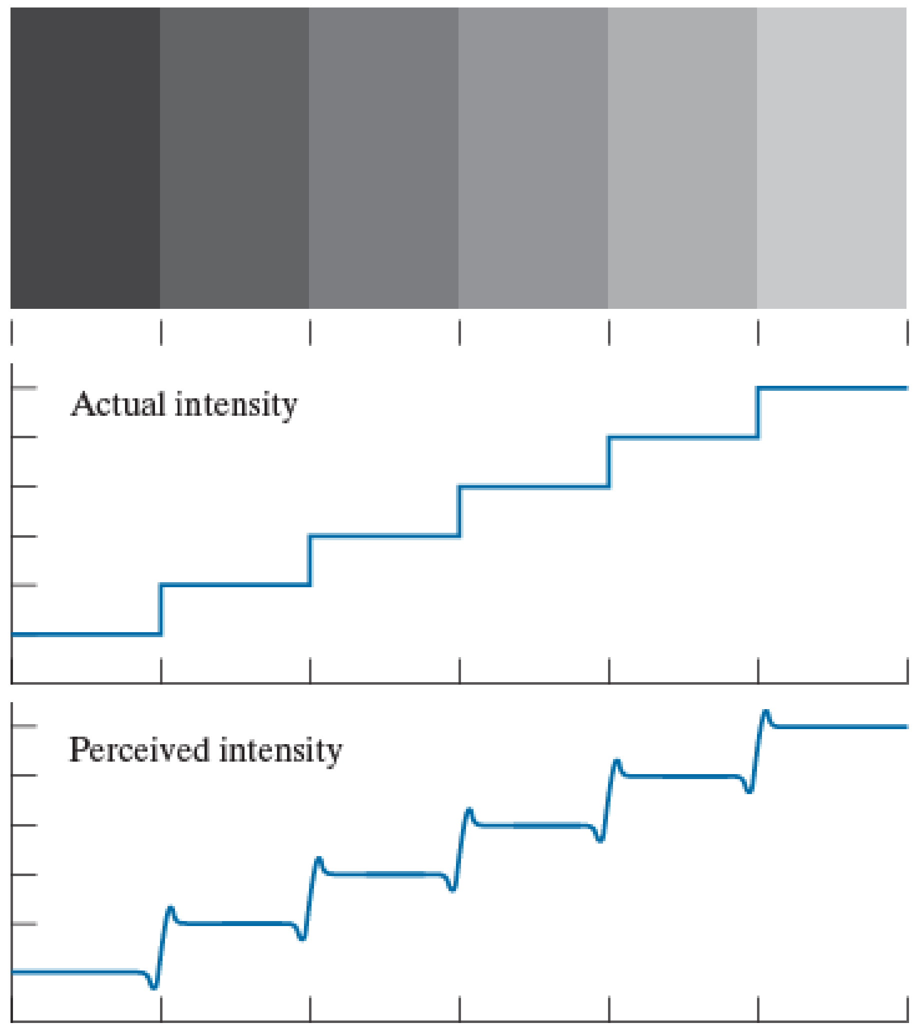
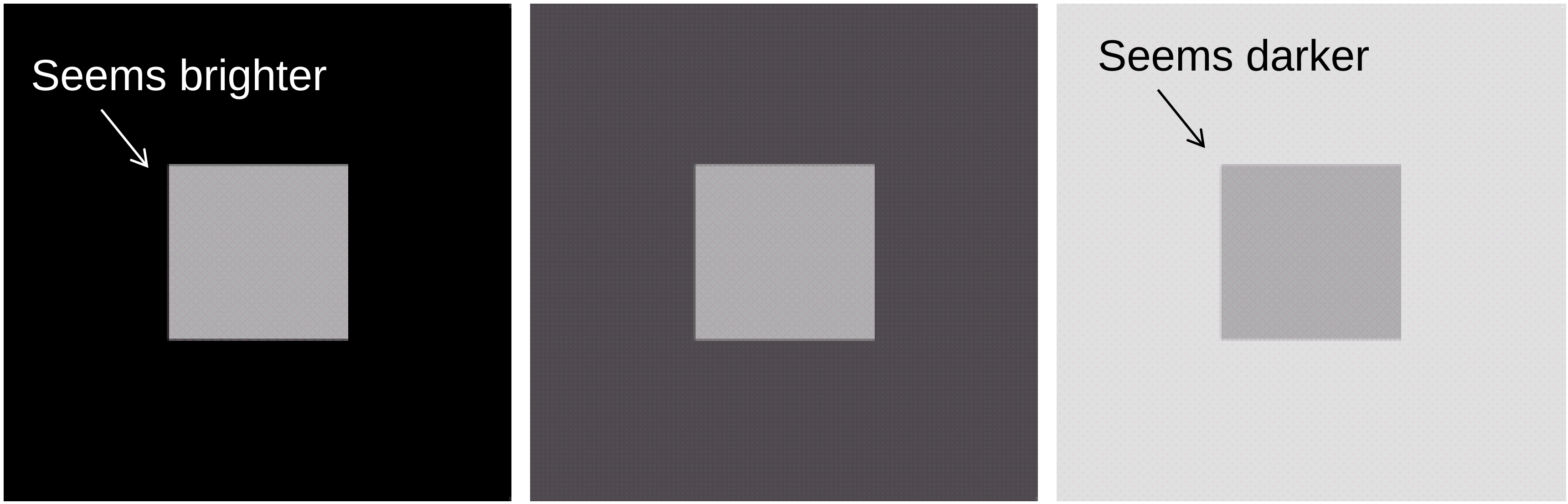


FIGURE 2.7 Illustration of the Mach band effect. Perceived intensity is not a simple function of actual intensity.

Simultaneous Contrast

- **Simultaneous Contrast: Perceived Intensity of a Region**
  - Depends on intensities of regions surrounding it
- White paper looks dark when held in front of a lamp



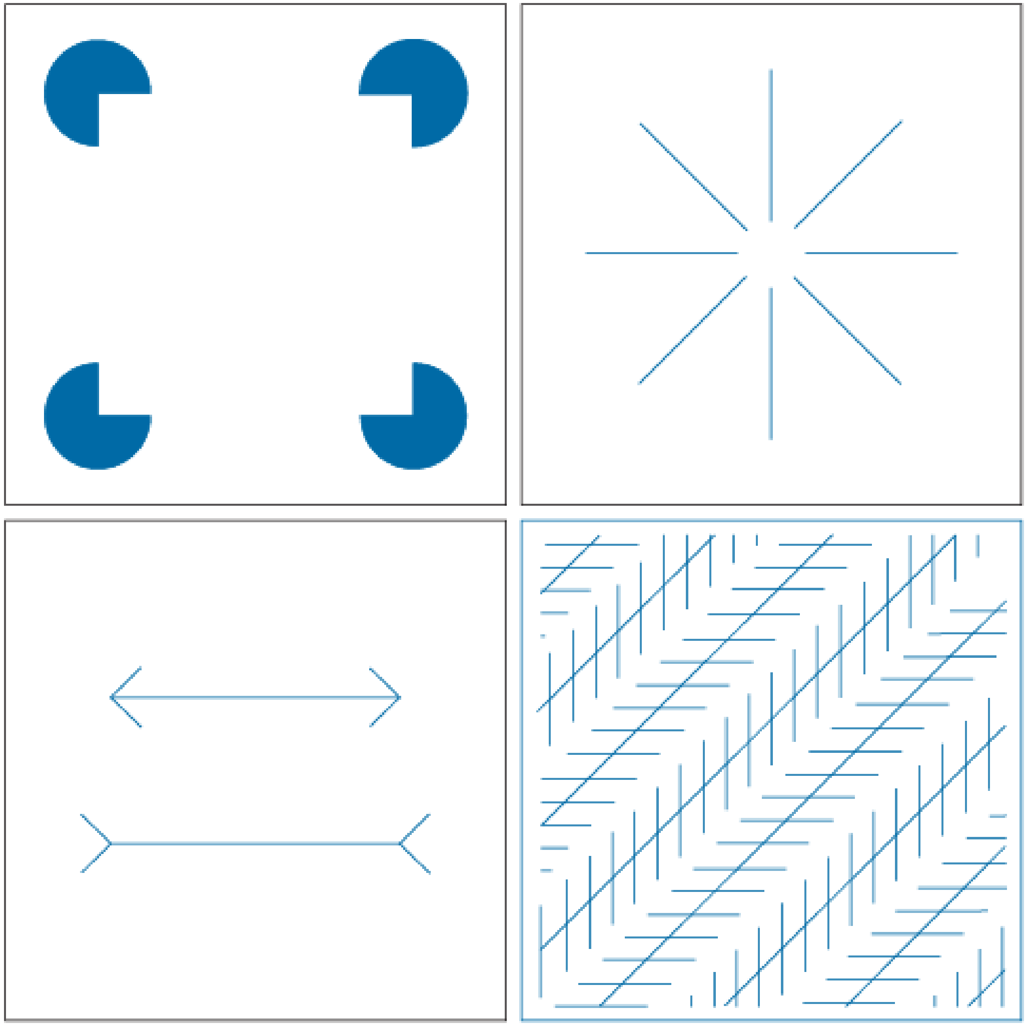
An example of *simultaneous contrast:* All small squares have same intensity

a b c

FIGURE 2.8 Examples of simultaneous contrast. All the inner squares have the same intensity, but they appear progressively darker as the background becomes lighter.

Optical Illusions

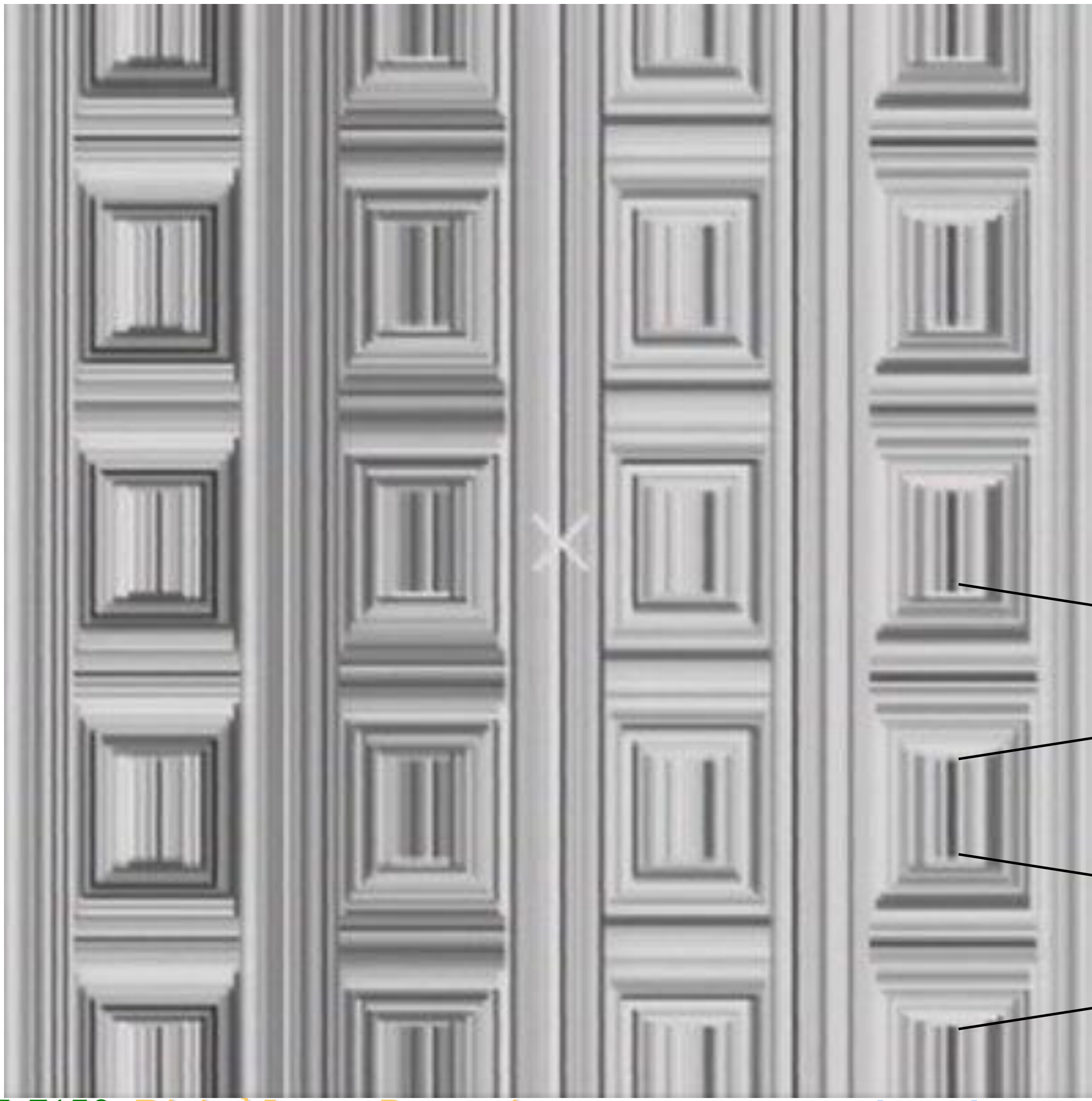
- Our visual system plays interesting tricks on us
- Our eyes can fill in non-existing information



a b
c d

FIGURE 2.9 Some well-known optical illusions.

Optical Illusions (cont…)

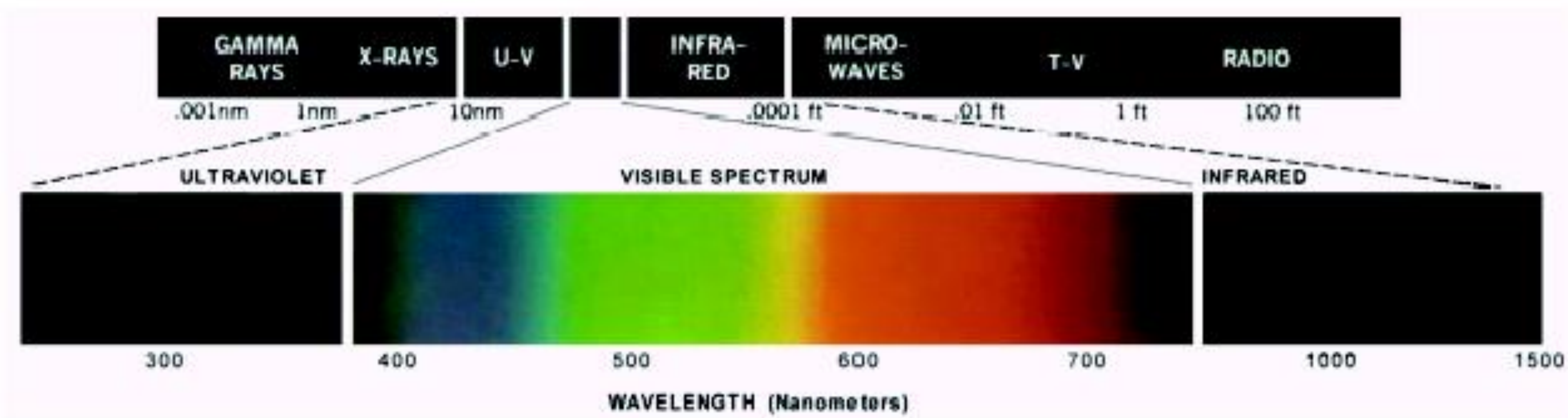


Stare at the cross in the middle of the image and look for circles

Light & the Electromagnetic Spectrum

- Visible Light is only a particular part of the electromagnetic spectrum that human eyes can sense

- The electromagnetic spectrum is split up according to the wavelengths of different forms of energy



Wavelength, Frequency & Energy

The electromagnetic spectrum can be expressed in terms of wavelength (λ), frequency (v), or energy (E). Recall that

$$\lambda = c/v$$

where c is the speed of light (2.998×10^8 m/s).

The energy of the various components is given by:

$$E = hv$$

where h is Planck's constant ($6.62606891 \times 10^{-34}$ Joule-seconds (or m^2kg/s)). E is measured in electron-volt.

Wavelength, Frequency & Energy

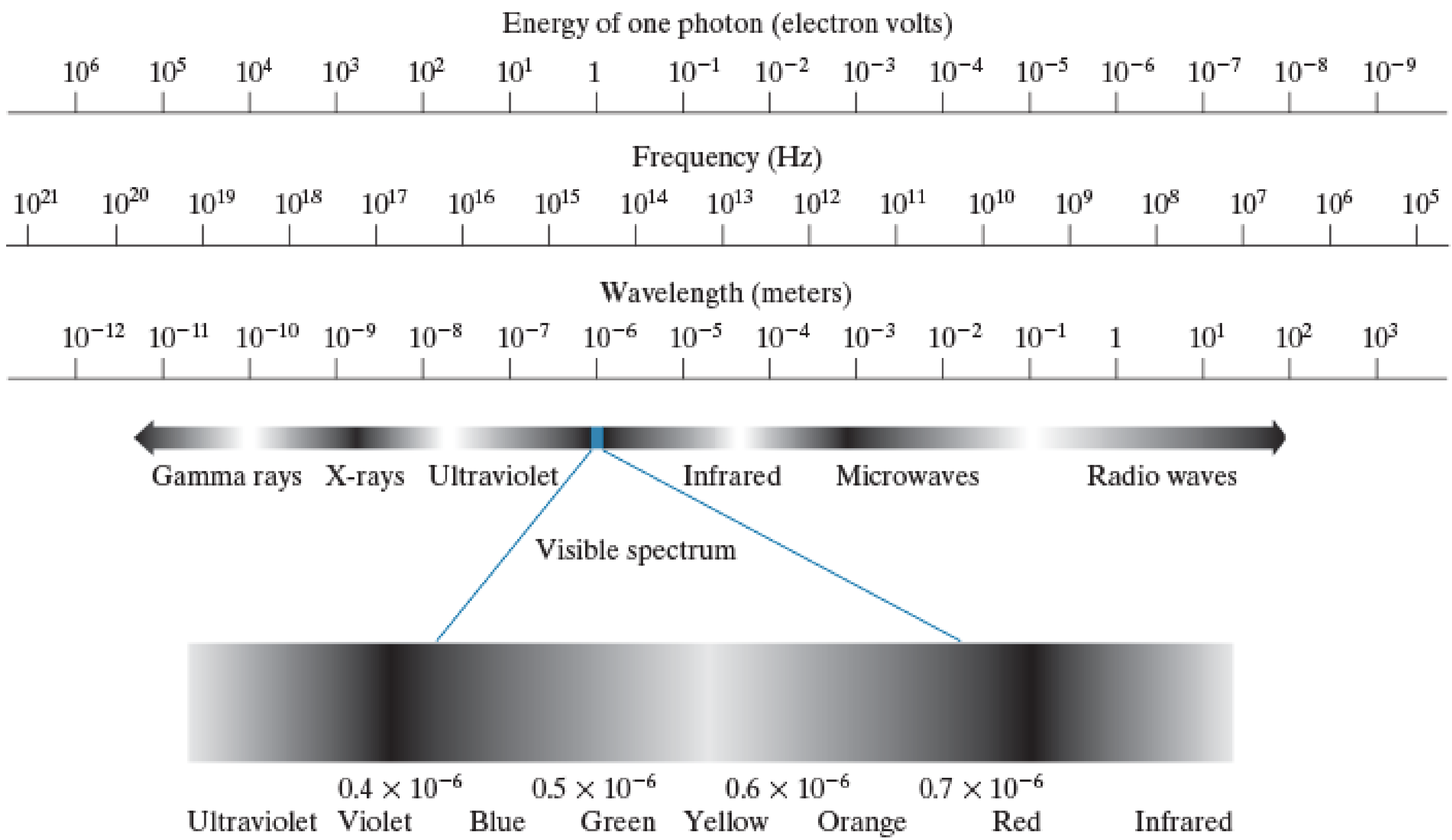
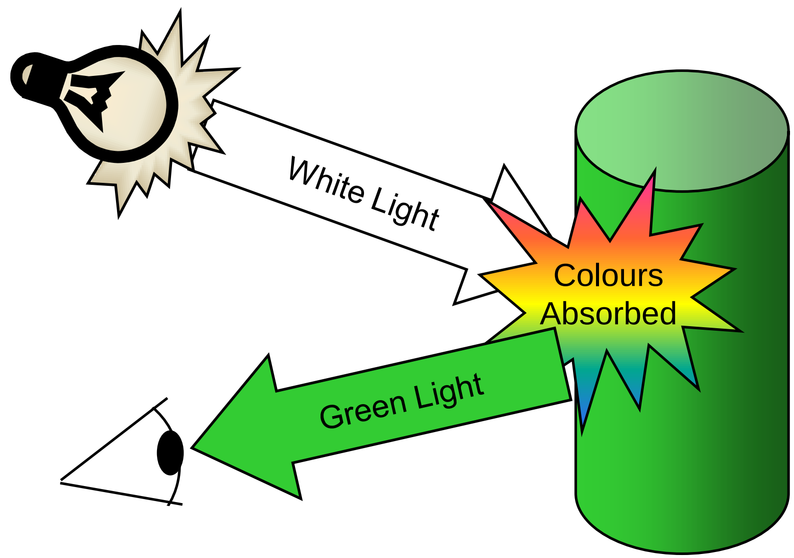


FIGURE 2.10 The electromagnetic spectrum. The visible spectrum is shown zoomed to facilitate explanations, but note that it encompasses a very narrow range of the total EM spectrum.

Reflected Light

•The colors that we perceive are determined by the nature of the light reflected from an object

• For example, if white light is shone onto a green object, most wavelengths are absorbed, while green light is reflected back from the object



Sensing, Sampling, Quantization & Resolution

- Image sensing and representation
- Sampling and Quantization
- Resolution

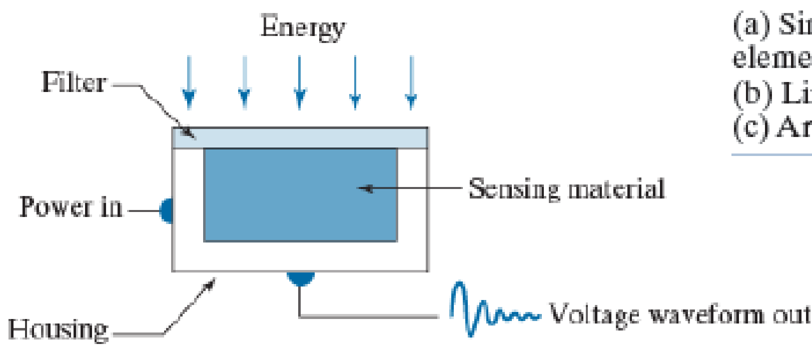
Image Sensing

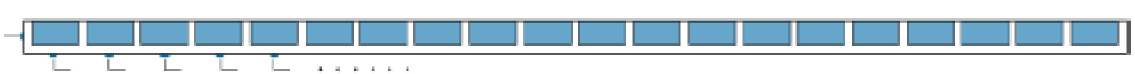
- Incoming energy lands on a sensor material responsive to that type of energy and this generates a voltage
- Collections of sensors are arranged to capture images

Imaging Sensor

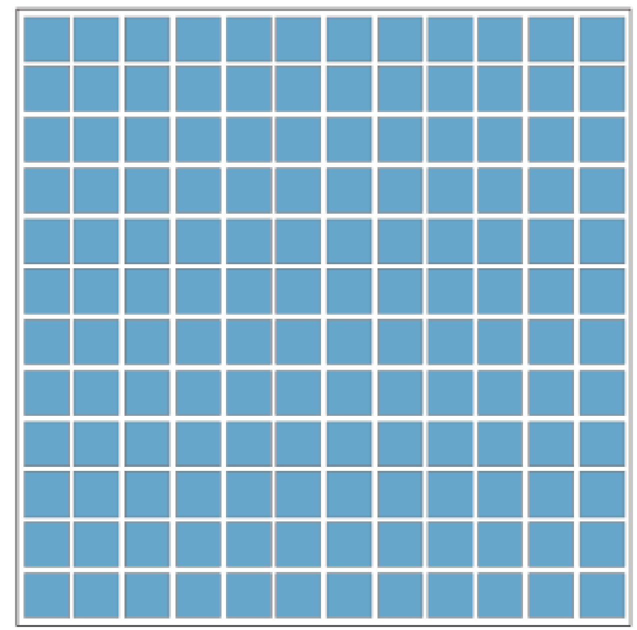
a
b
c

FIGURE 2.12
(a) Single sensing element.
(b) Line sensor.
(c) Array sensor.





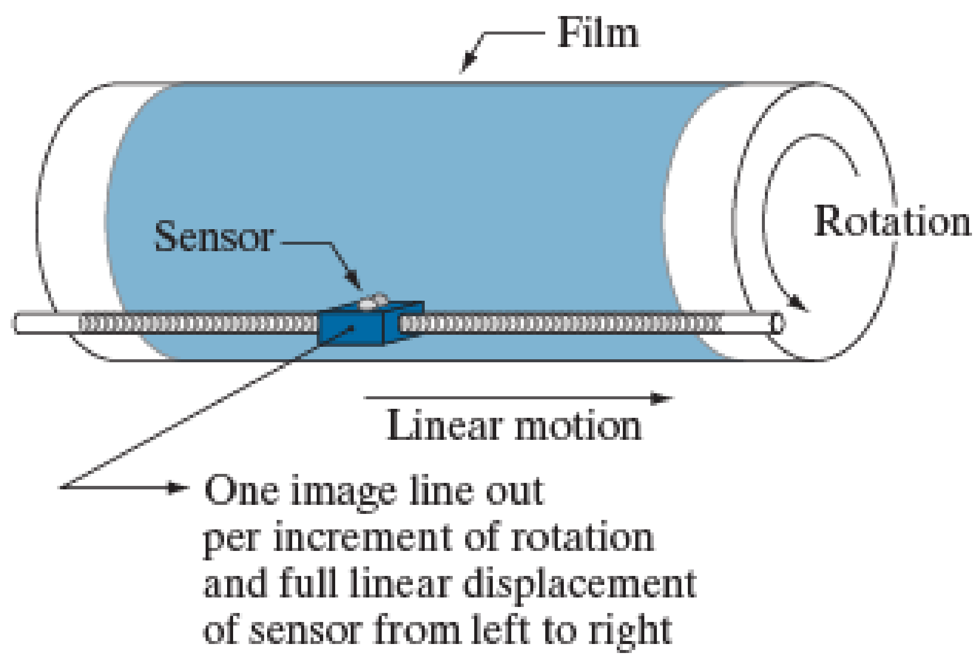
Line of Image Sensors



Array of Image Sensors

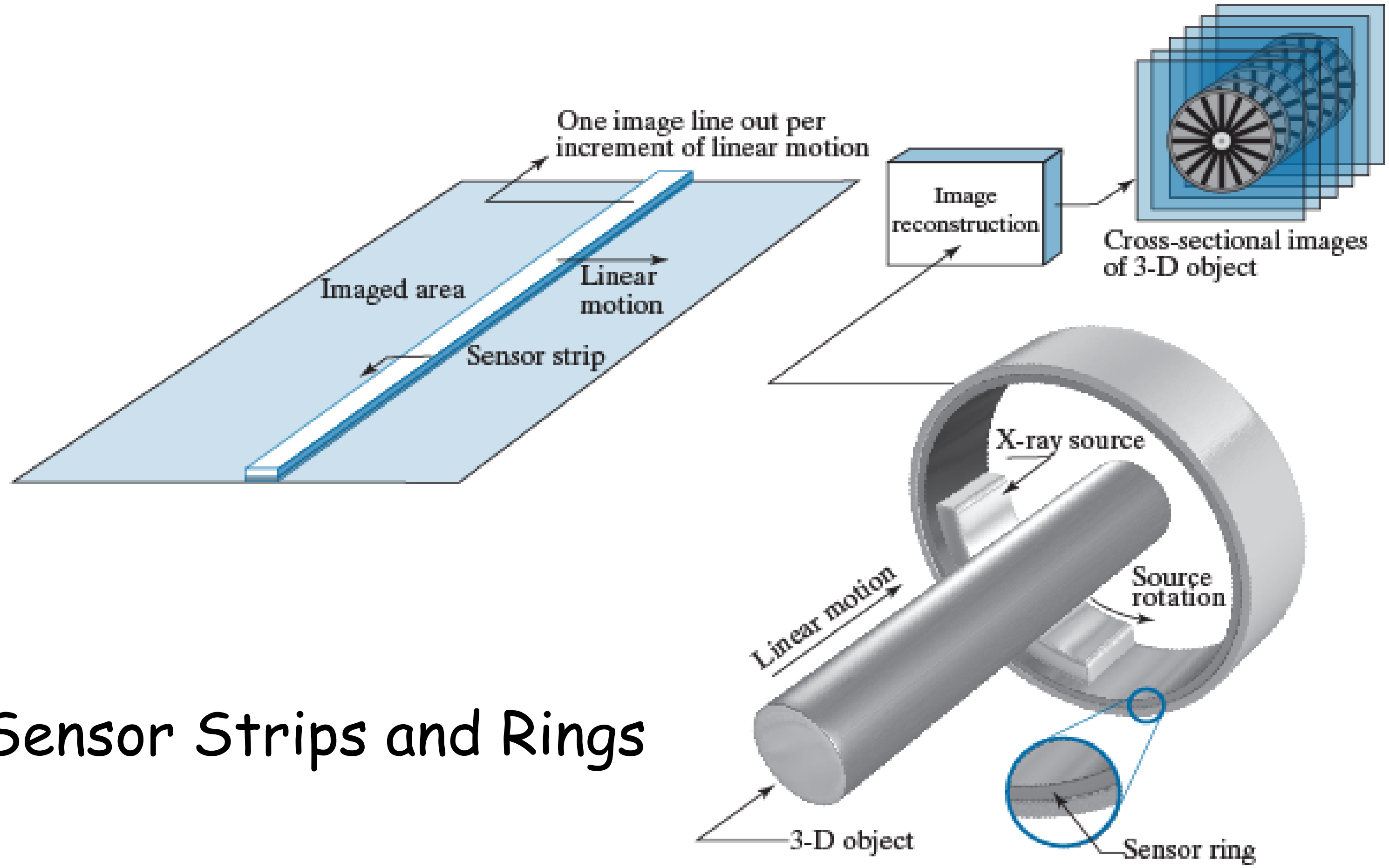
2-D Sensing

FIGURE 2.13
Combining a single sensing element with mechanical motion to generate a 2-D image.



Combine Single Sensor + Motion (linear + circular)

Image Sensing – 2D & 3D



Using Sensor Strips and Rings

a b

FIGURE 2.14 (a) Image acquisition using a linear sensor strip. (b) Image acquisition using a circular sensor strip.

Image Acquisition

Passive Sensing

- Images: Typically generated by *illuminating a scene* and absorbing energy reflected by objects in that scene

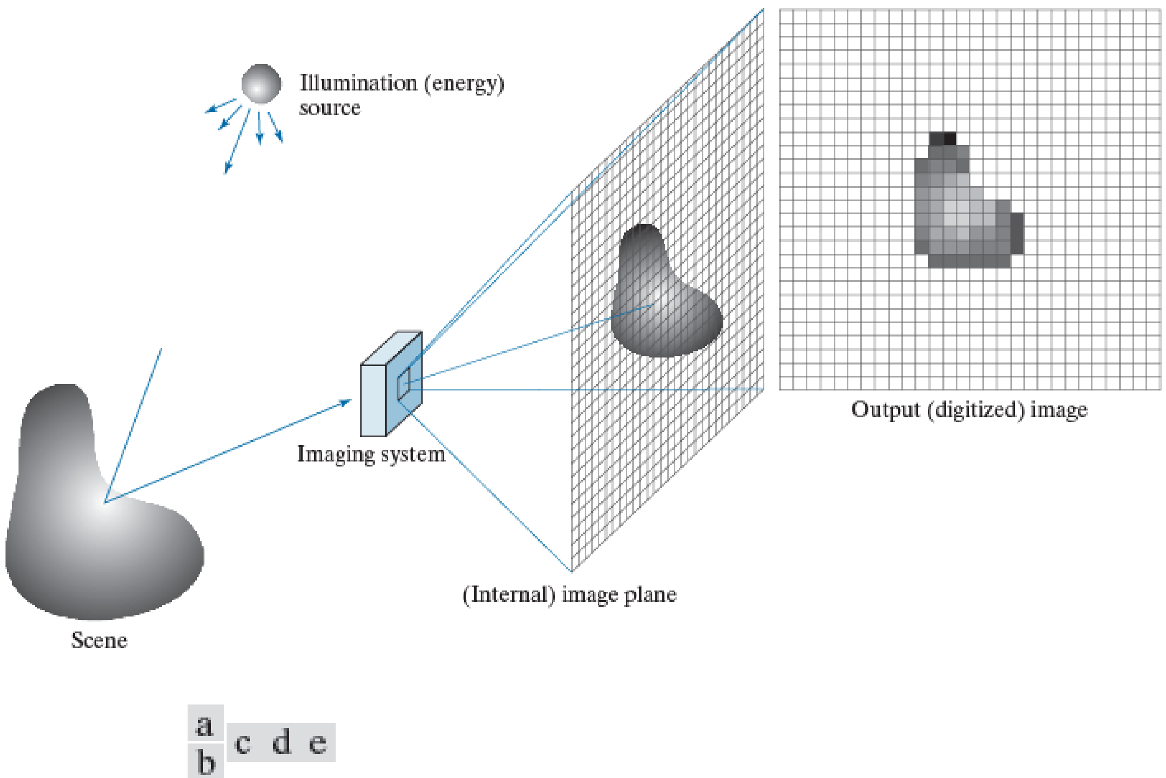


FIGURE 2.15 An example of digital image acquisition. (a) Illumination (energy) source. (b) A scene. (c) Imaging system. (d) Projection of the scene onto the image plane. (e) Digitized image.

- Other notions of illumination and image acquisition:
  - X-rays of a skeleton
  - Ultrasound of an unborn baby
  - Electro-microscopic images of molecules

Gray Level

- **Monochromatic Light**: Light without of color
- Only attribute of such light → **Intensity**
- **Gray level:** Term generally used to describe monochromatic intensity
  - Ranges from Black → Grays and finally → White

Light-Intensity Function

- Image: A 2-D light-intensity function, $f(x, y)$

- Amplitude of f at spatial coordinates (x, y) gives intensity (brightness) of image at that point

- Light (Intensity): A form of energy, thus $f(x, y)$ must be nonzero, nonnegative and finite:

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

Illumination & Reflectance

Nature of $f(x, y)$: Characterized by two components:

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

- **Illumination**: Amount of source light **incident** on the scene being viewed, $i(x, y)$
  - $0 < i(x, y) < \infty$ determined by the nature of the light source

- **Reflectance**: The amount of light **reflected** back by the objects in the scene, $r(x, y)$
  - $0 < r(x, y) < 1$ determined by the nature of the objects in a scene

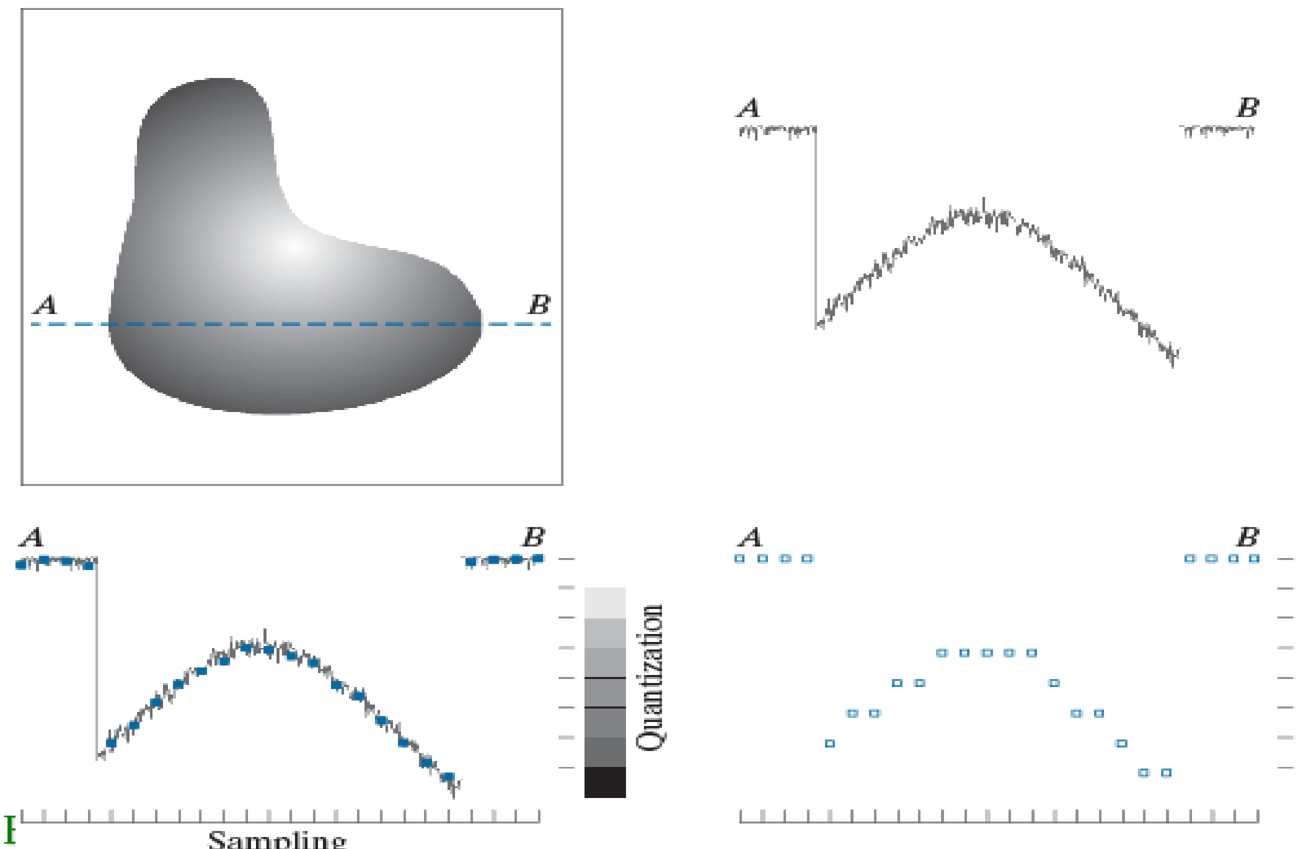
Gray Levels of Monochrome Image

Intensity of a monochrome image f at coordinate (x, y) is the gray level l of the image at that point

- l → Lies in the range $L_{min} \leq l \leq L_{max}$
- L_{max} → Finite
- L_{min} → Positive
- Gray scale = $[L_{min}, L_{max}]$
- Common practice: Shift the interval to $[0, L]$
- Black → 0
- White → L
- Gray → Values in between

2.4 Image Sampling & Quantization

- A digital sensor can only measure a limited number of **samples** at a **discrete** set of energy levels
- **Quantization** is the process of converting a continuous **analog** signal into a digital representation



a b
c d

FIGURE 2.16 (a) Continuous image. (b) A scan line showing intensity variations along line *AB* in the continuous image. (c) Sampling and quantization. (d) Digital scan line. (The black border in (a) is included for clarity. It is not part of the image).

Sampling & Quantization

- **Spatial Resolution (Sampling pixel-size)**
  - Determines the smallest perceivable image detail
  - What is the "best" spatial sampling rate?
- **Gray-Level Resolution (Intensity Quantization)**
  - Smallest discernible change in gray level value
  - Is there an optimal quantizer?

Image Sampling & Quantization

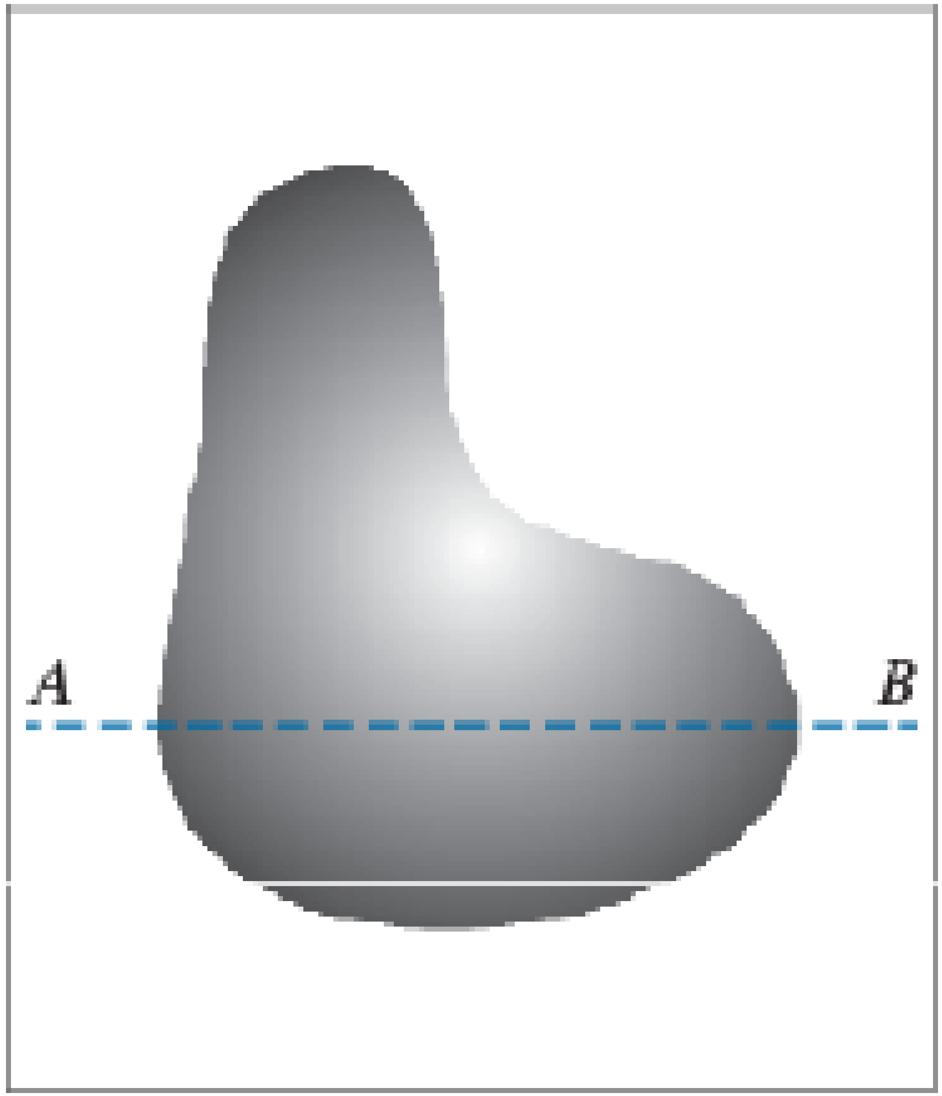
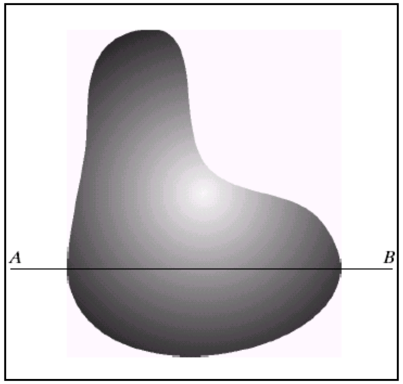


Image Sampling & Quantization



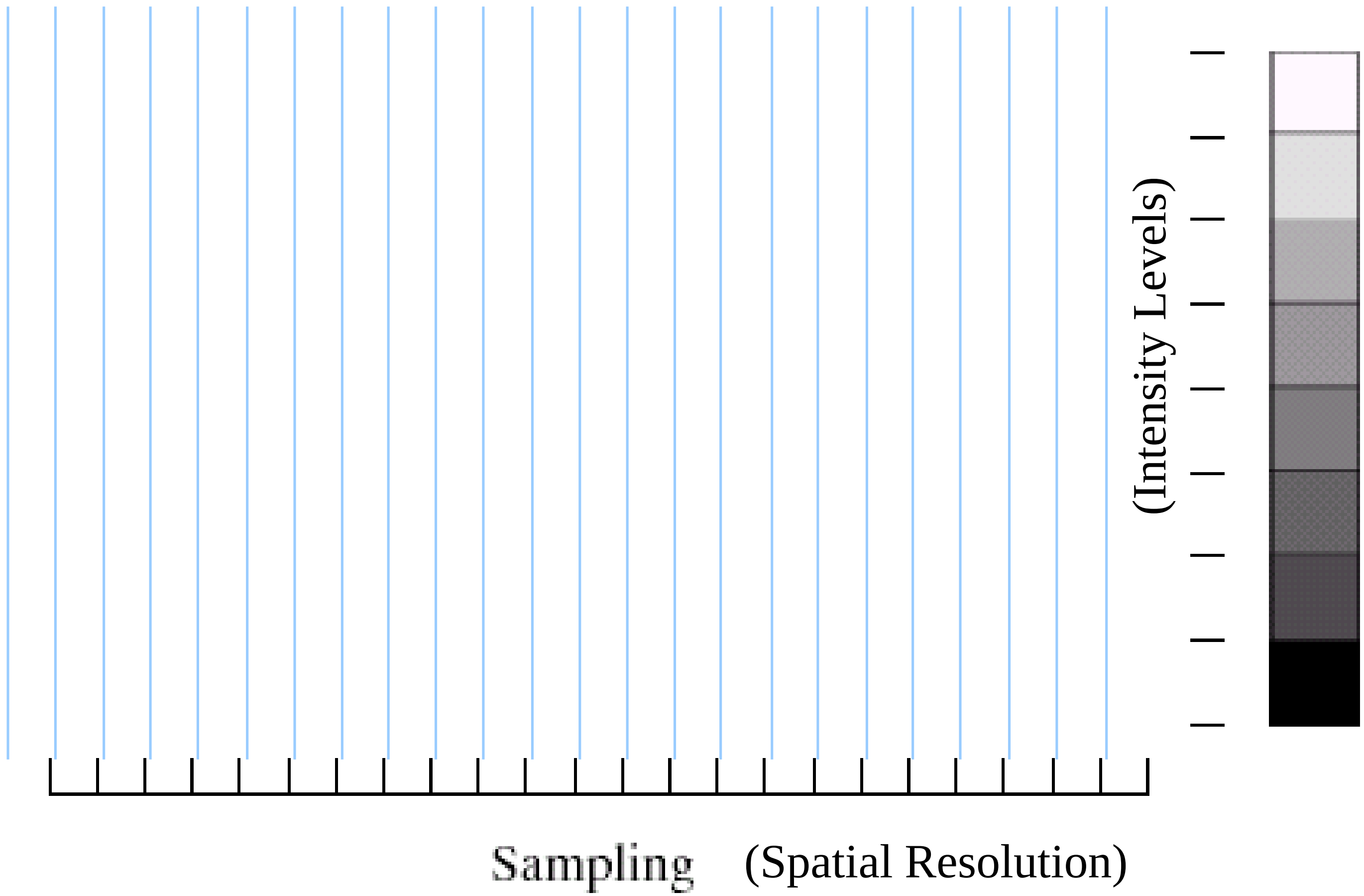
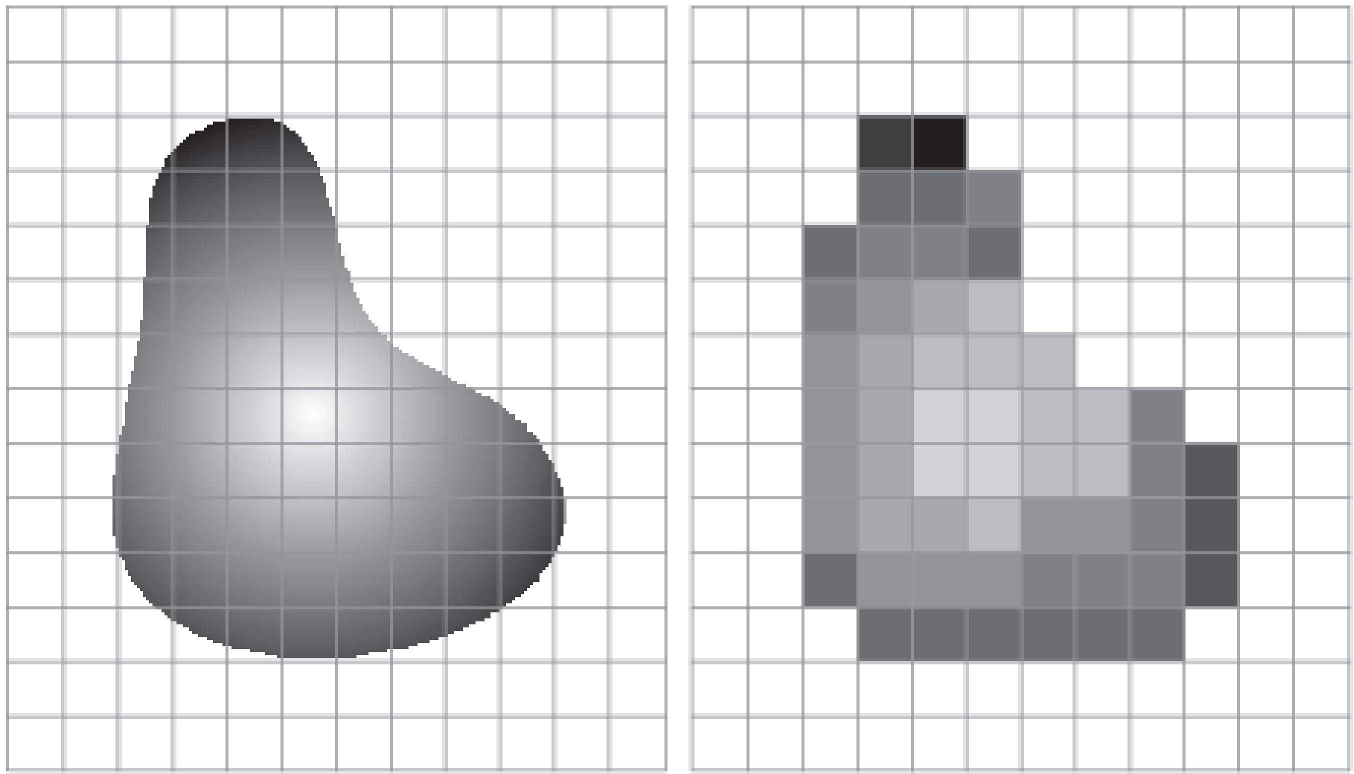


Image Sampling & Quantization (cont...)

- A digital image: An **approximation** of a real world scene

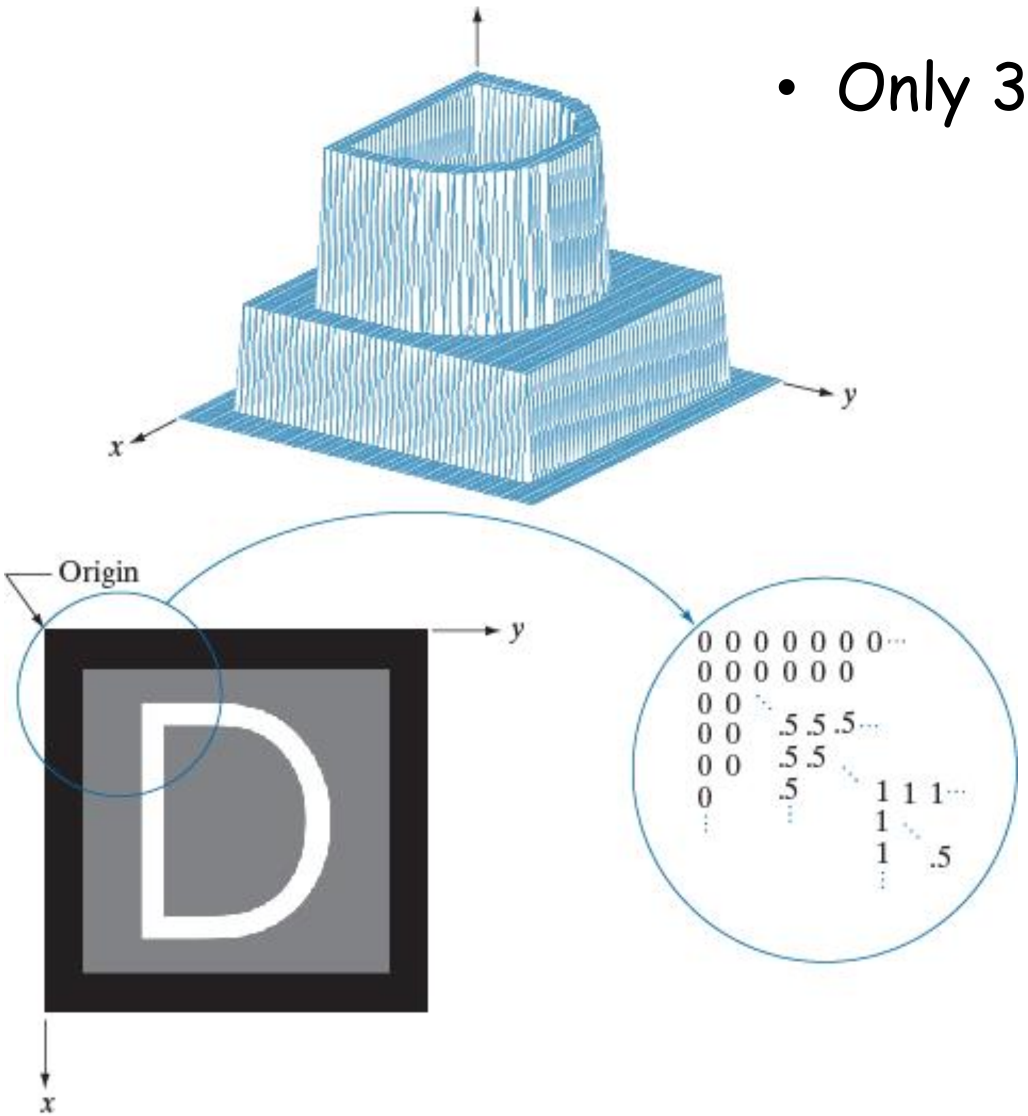


a b

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

Image Representation

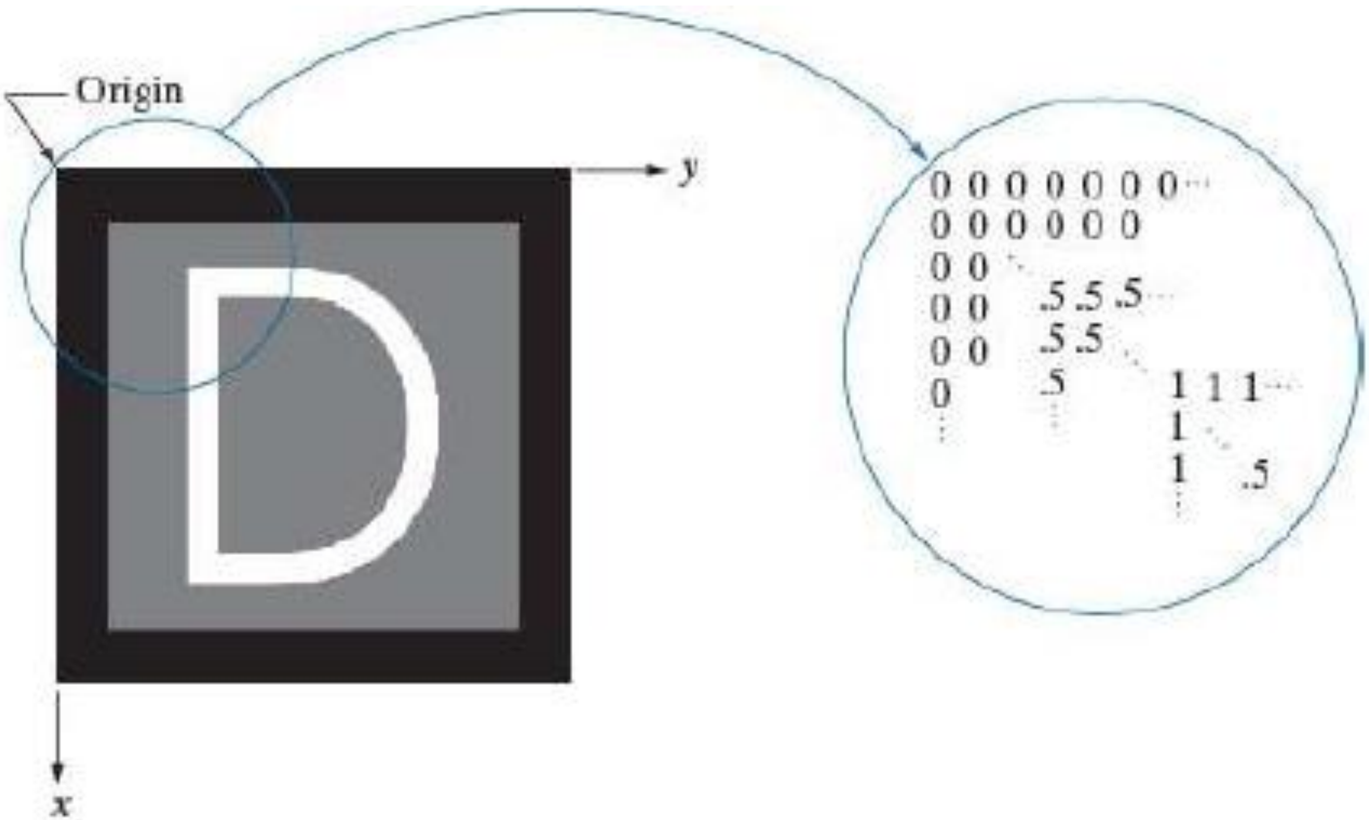
- Only 3 intensity levels



a
b c

FIGURE 2.18
(a) Image plotted as a surface. (b) Image displayed as a visual intensity array. (c) Image shown as a 2-D numerical array. (The numbers 0, .5, and 1 represent black, gray, and white, respectively.)

Image Representation



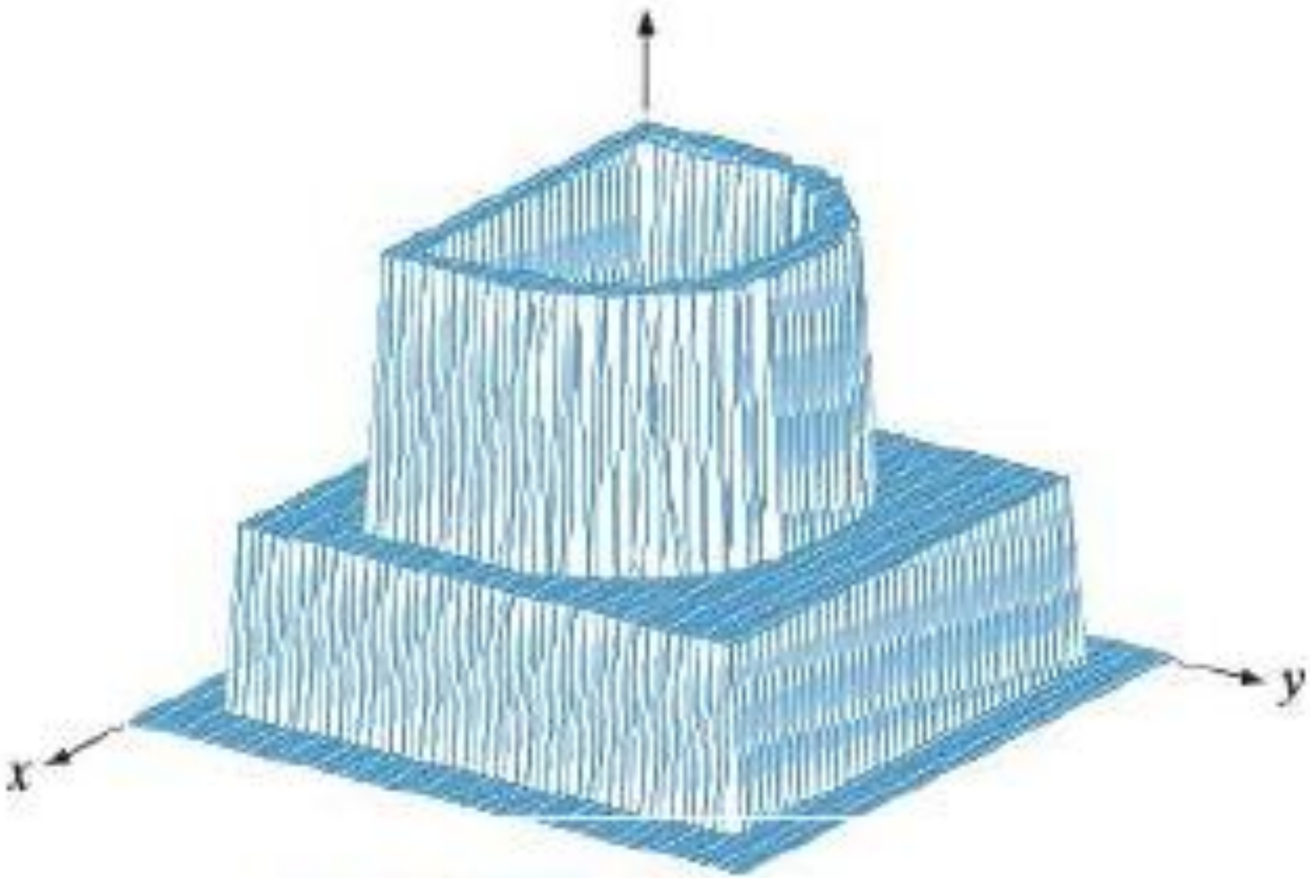
a
b c

FIGURE 2.18 (a) Image plotted as a surface. (b) Image displayed as a visual intensity array. (c) Image shown as a 2-D numerical array. (The numbers 0, .5, and 1 represent black, gray, and white, respectively.)

- Only 3 intensity levels

Image Representation

- Only 3 intensity levels



a
b c

FIGURE 2.18 (a) Image plotted as a surface. (b) Image displayed as a visual intensity array. (c) Image shown as a 2-D numerical array. (The numbers 0, .5, and 1 represent black, gray, and white, respectively.)

Image Representation

- A digital image is composed of M rows and N columns of pixels each storing a value (Intensity)
- Pixel values are most often grey levels in the range 0-255 (black-white)
- It is more convenient to represent images as matrices

Matlab uses this representation

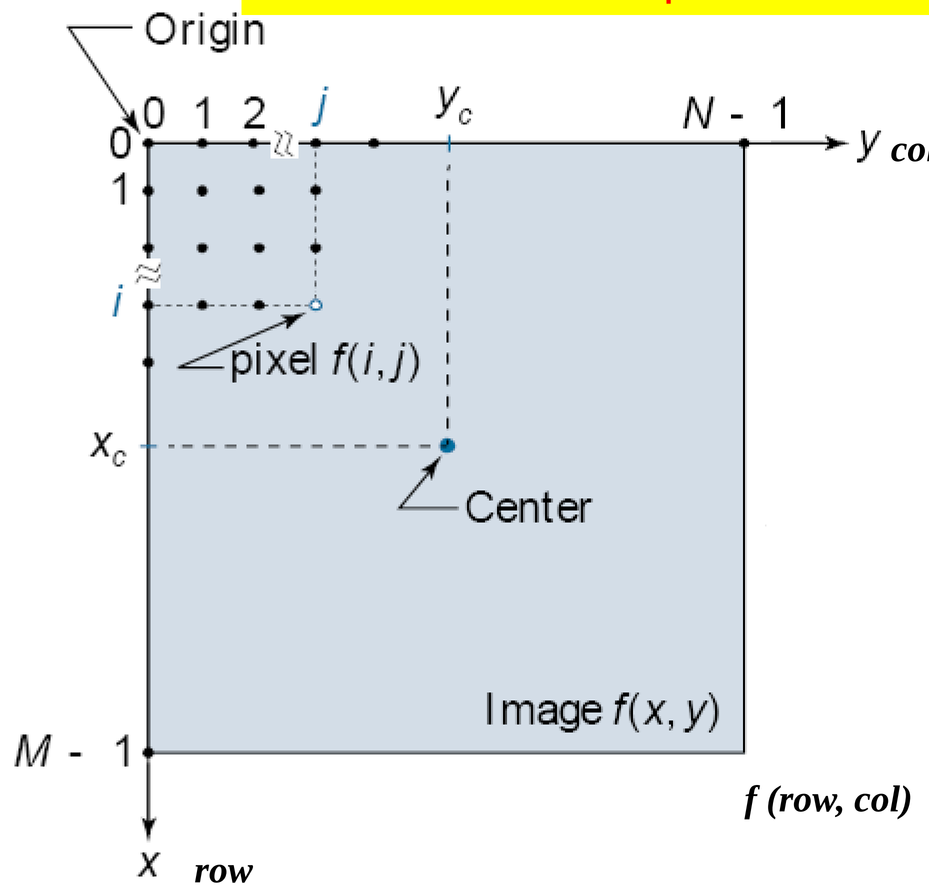


FIGURE 2.19 Coordinate convention used to represent digital images. Because coordinate values are integers, there is a one-to-one correspondence between x and y and the rows (r) and columns (c) of a matrix.

Dynamic Range: Highest-to-Lowest Intensity levels Discernible

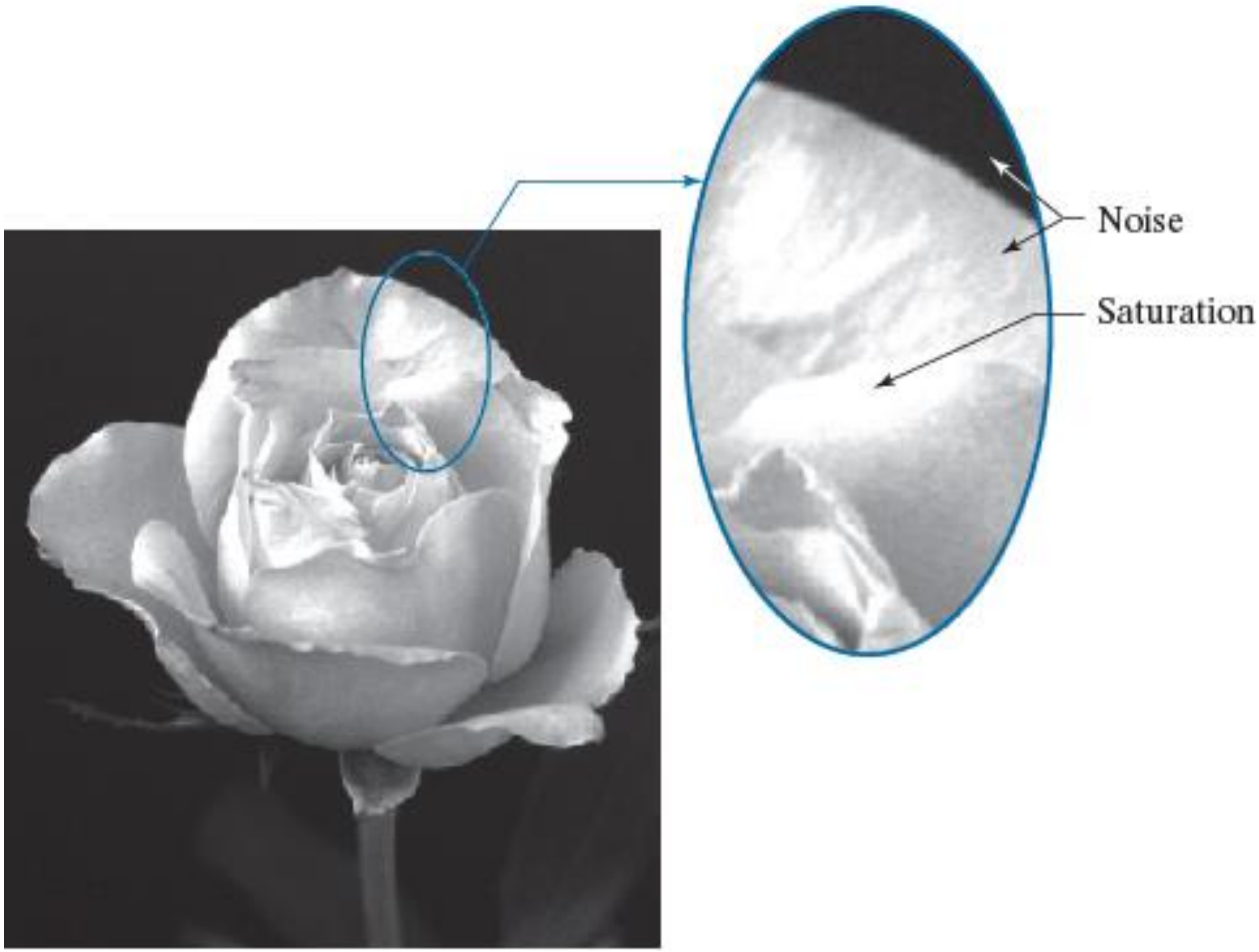
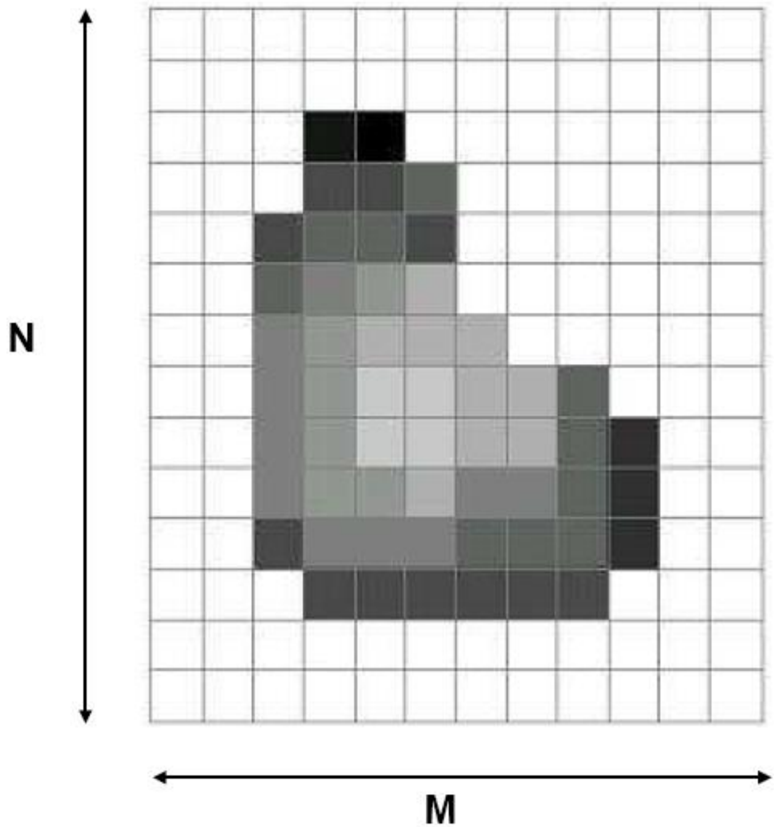


FIGURE 2.20 An image exhibiting saturation and noise. Saturation is the highest value beyond which all intensity values are clipped (note how the entire saturated area has a high, constant intensity level). Visible noise in this case appears as a grainy texture pattern. The dark background is noisier, but the noise is difficult to see.

Number of Bits



- Number of gray levels typically is an integer power of 2:

$$L = 2^k$$

k: Number of bits per pixel

- Number of bits required to store a M x N digitized image:

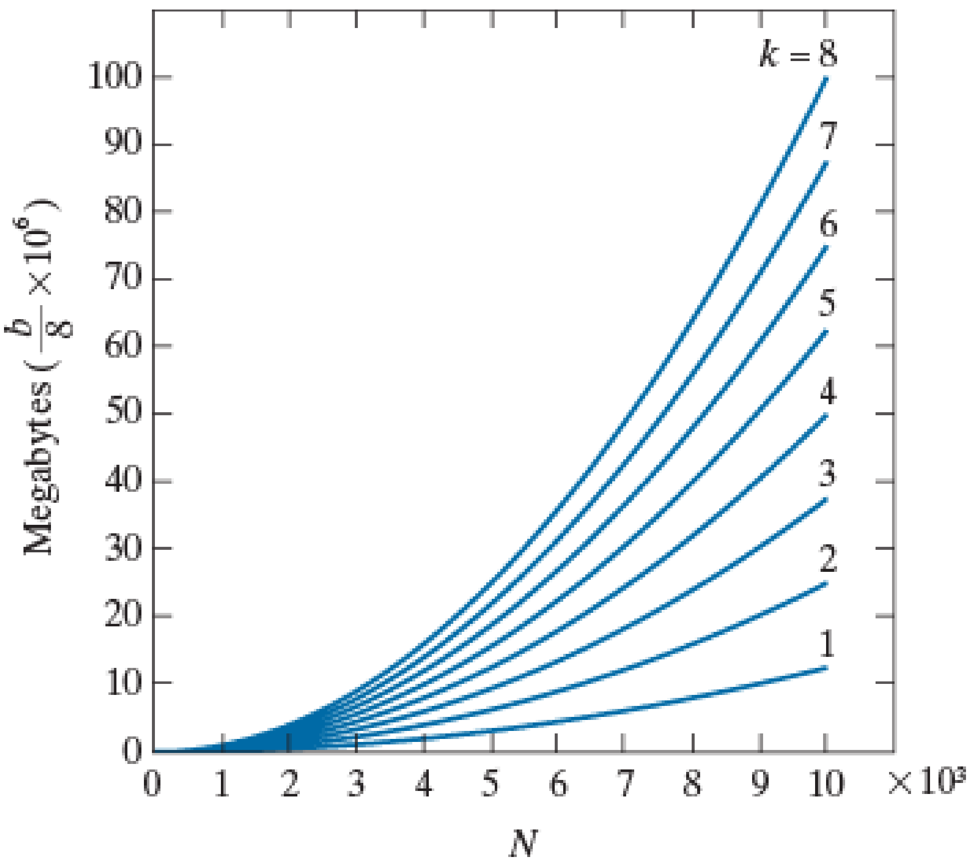
$$b = M \times N \times k$$

- For $M = N$, $b = N^2 * k$

Storage Requirement

- Storage requirement grows exponentially with Image size

FIGURE 2.21 Number of megabytes required to store images for various values of N and k.



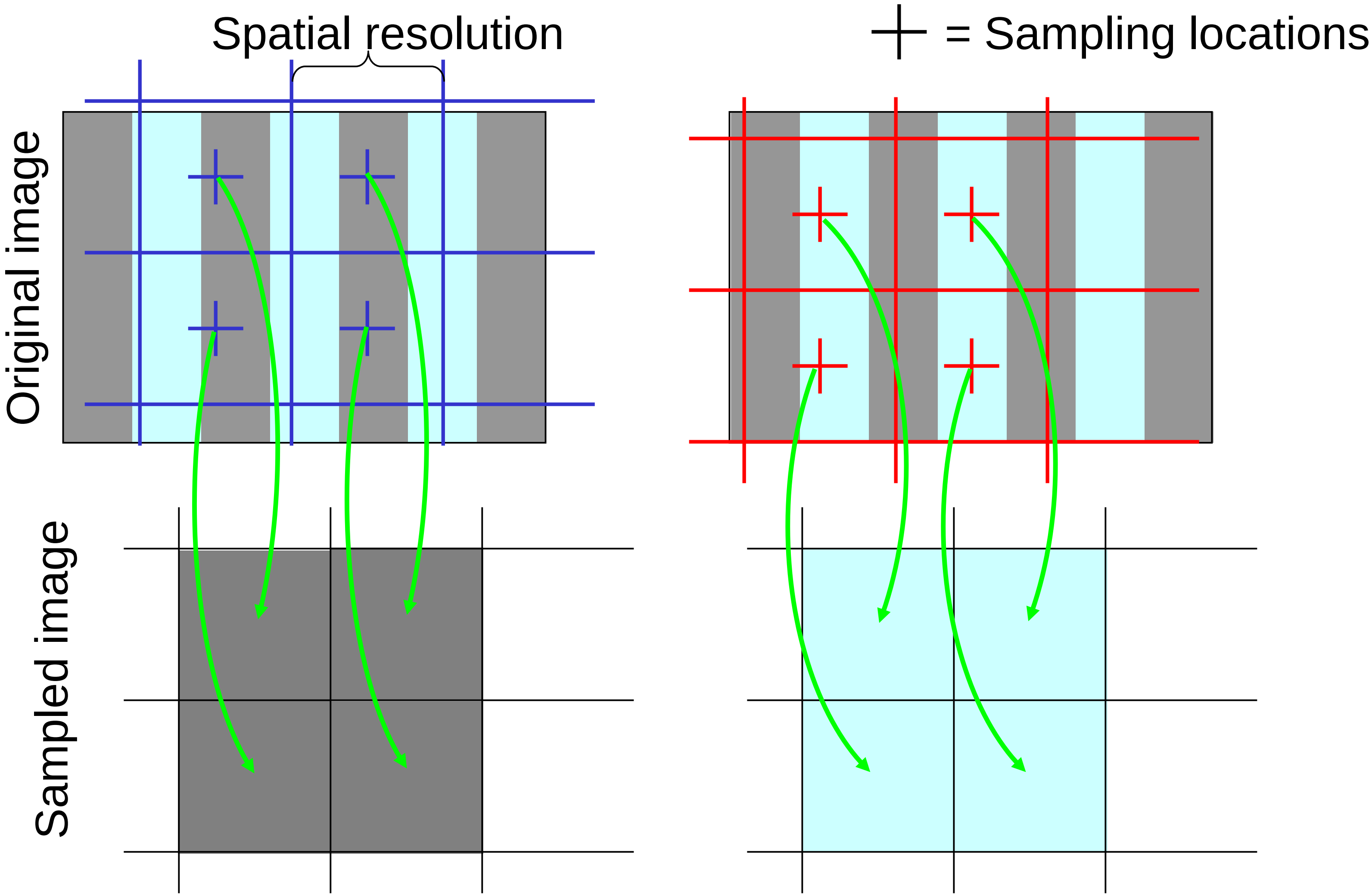
k: Number of bits per pixel

For $M = N$, $b = N^2 * k$

Spatial Resolution

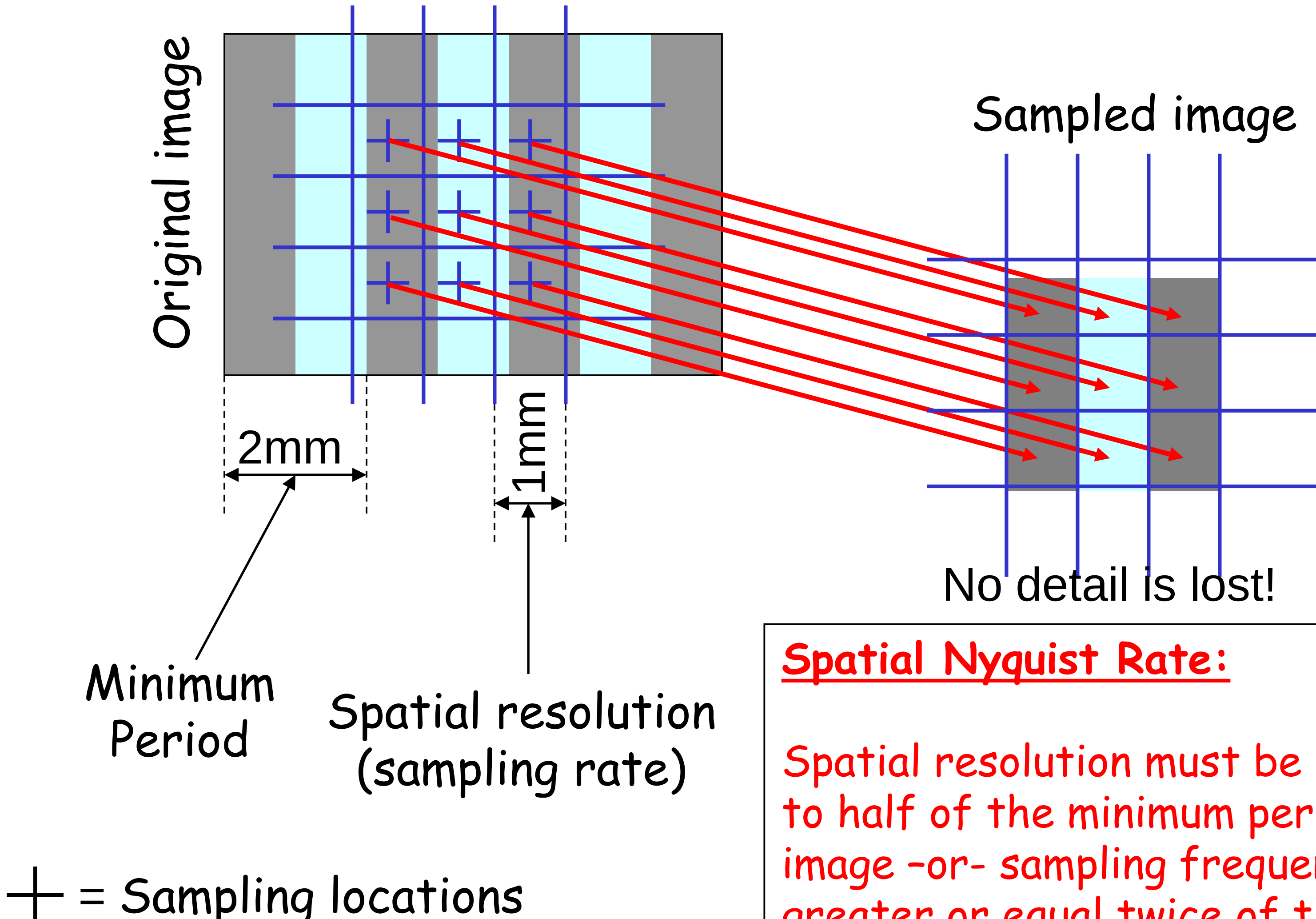
- **Spatial resolution of an image**
  - Determined by how sampling was carried out
  - Refers to smallest discernible detail in an image
  - Vision specialists often talk about pixel size
  - Graphic designers talk about *dots per inch* (DPI)

How to choose Spatial Resolution: Spatial Nyquist Rate



Undersampling: Some image details are lost

How to choose the spatial resolution : Nyquist Rate



Spatial Nyquist Rate:

Spatial resolution must be less or equal to half of the minimum period of the image –or- sampling frequency must be greater or equal twice of the maximum frequency.

Resolution: How Much Is Enough?

- Big question with resolution: *How much is enough?*
  - Depends on what is in the image and what the user would like to do with it
  - Key questions include
    - Does the image look aesthetically pleasing?
    - Can you see what you need to see within the image?

Resolution: How Much Is Enough? (cont…)

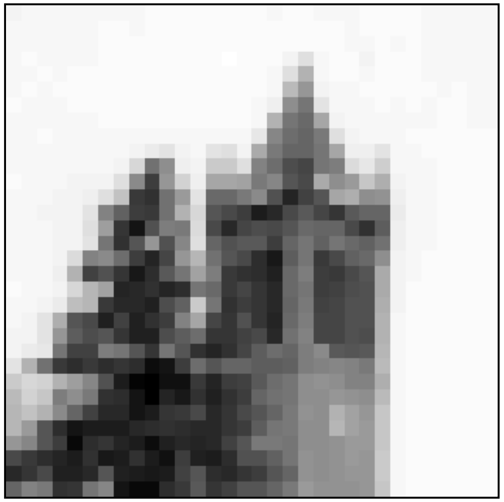
- The picture on the right is fine for counting the number of cars, but not for reading the number plate

Effect of Spatial Resolution


256x256 pixels

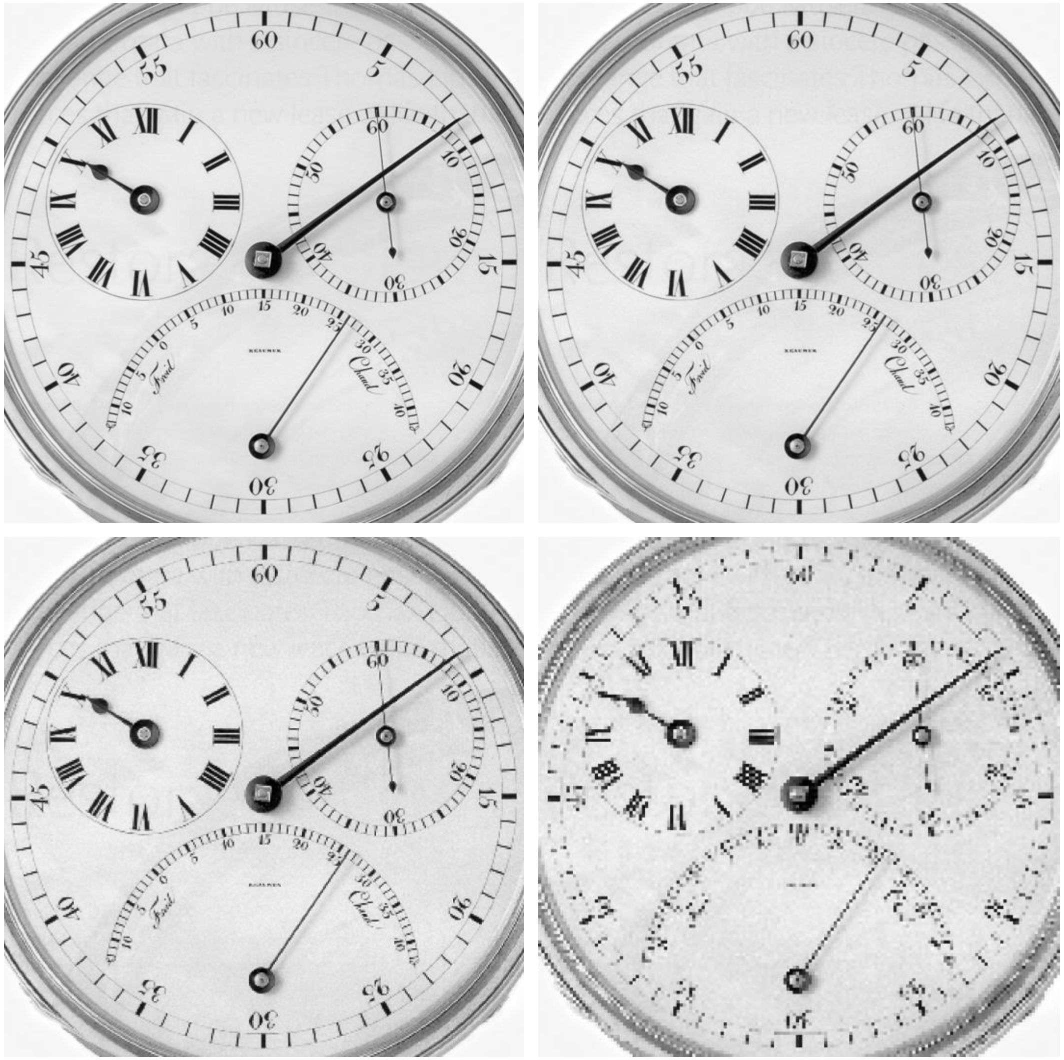

128x128 pixels


64x64 pixels


32x32 pixels

Spatial Resolution Example

- **Effect of reducing Spatial resolution:**
  - All images have same size
  - Only the number of dots per inch (dpi) are different

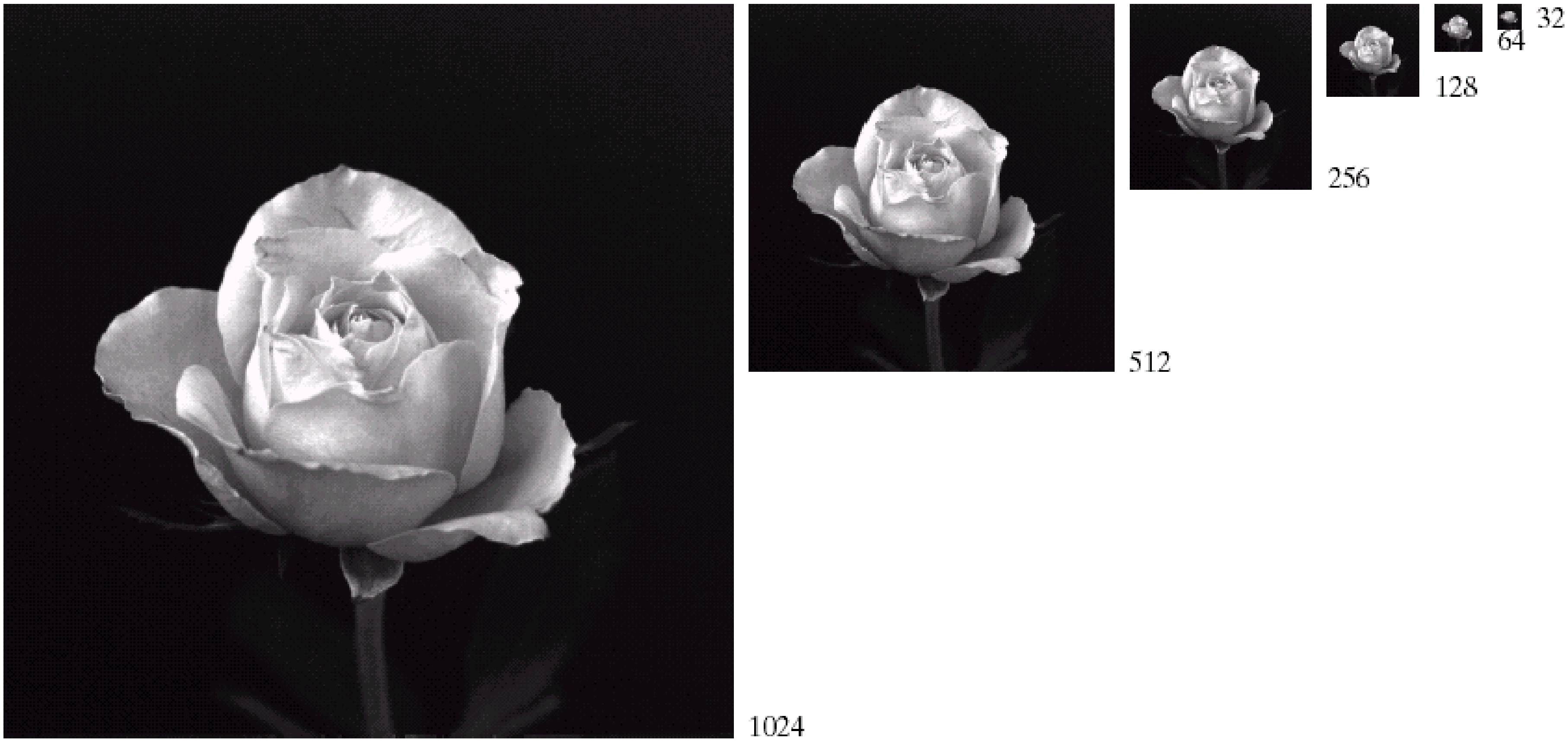


a b
c d

FIGURE 2.23 Effects of reducing spatial resolution. The images shown are at: (a) 930 dpi, (b) 300 dpi, (c) 150 dpi, and (d) 72 dpi.

Spatial Resolution (cont…)

- Only the number of dots per inch (dpi) are different



2nd Edition

Spatial Resolution (cont…)

1024 X 1024

2nd Edition

Spatial Resolution (cont…)

512 X 512

2nd Edition

Spatial Resolution (cont…)

256 X 256

2nd Edition

Spatial Resolution (cont…)

128 X 128

2nd Edition

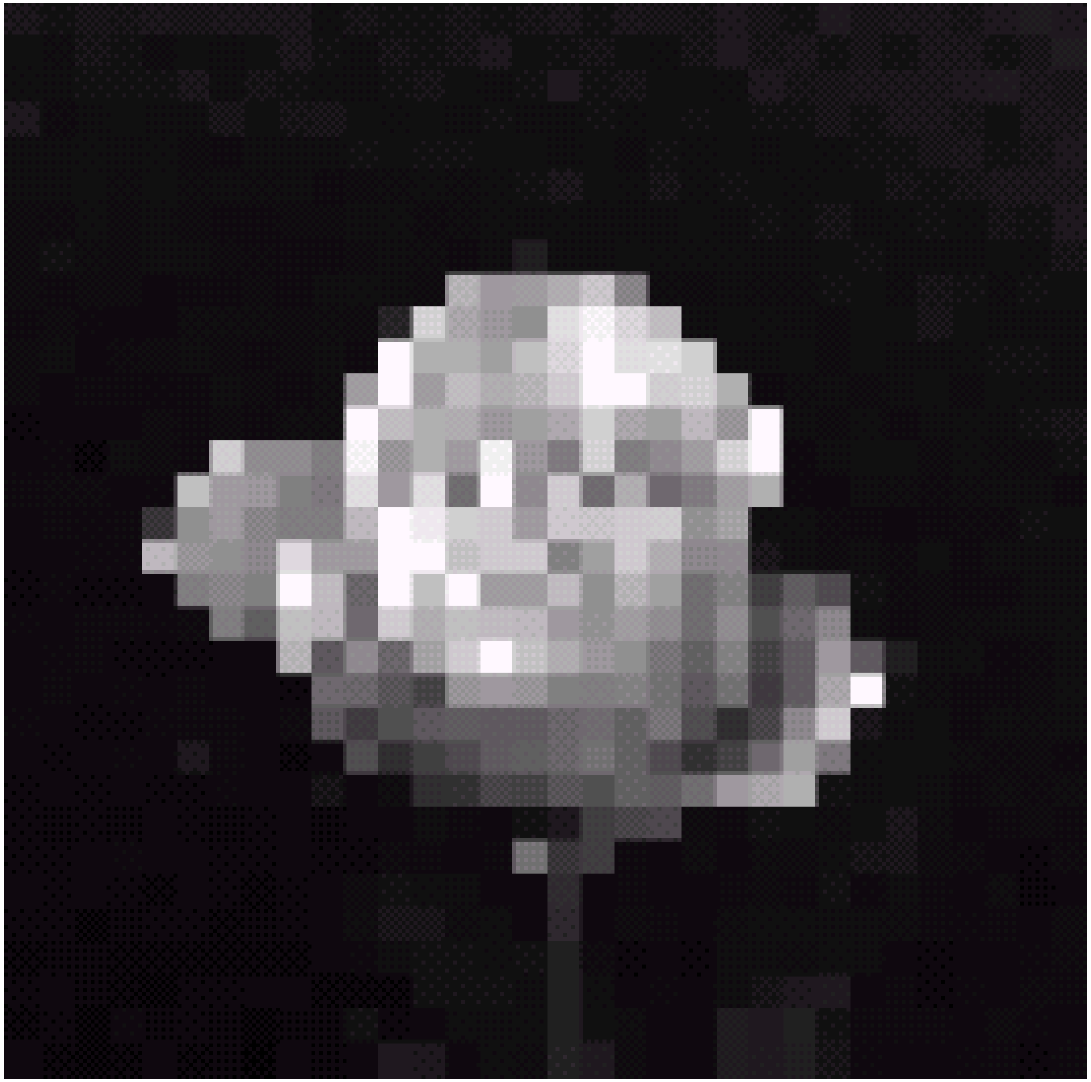
Spatial Resolution (cont…)

64 X 64

2nd Edition

Spatial Resolution (cont…)

32 X 32



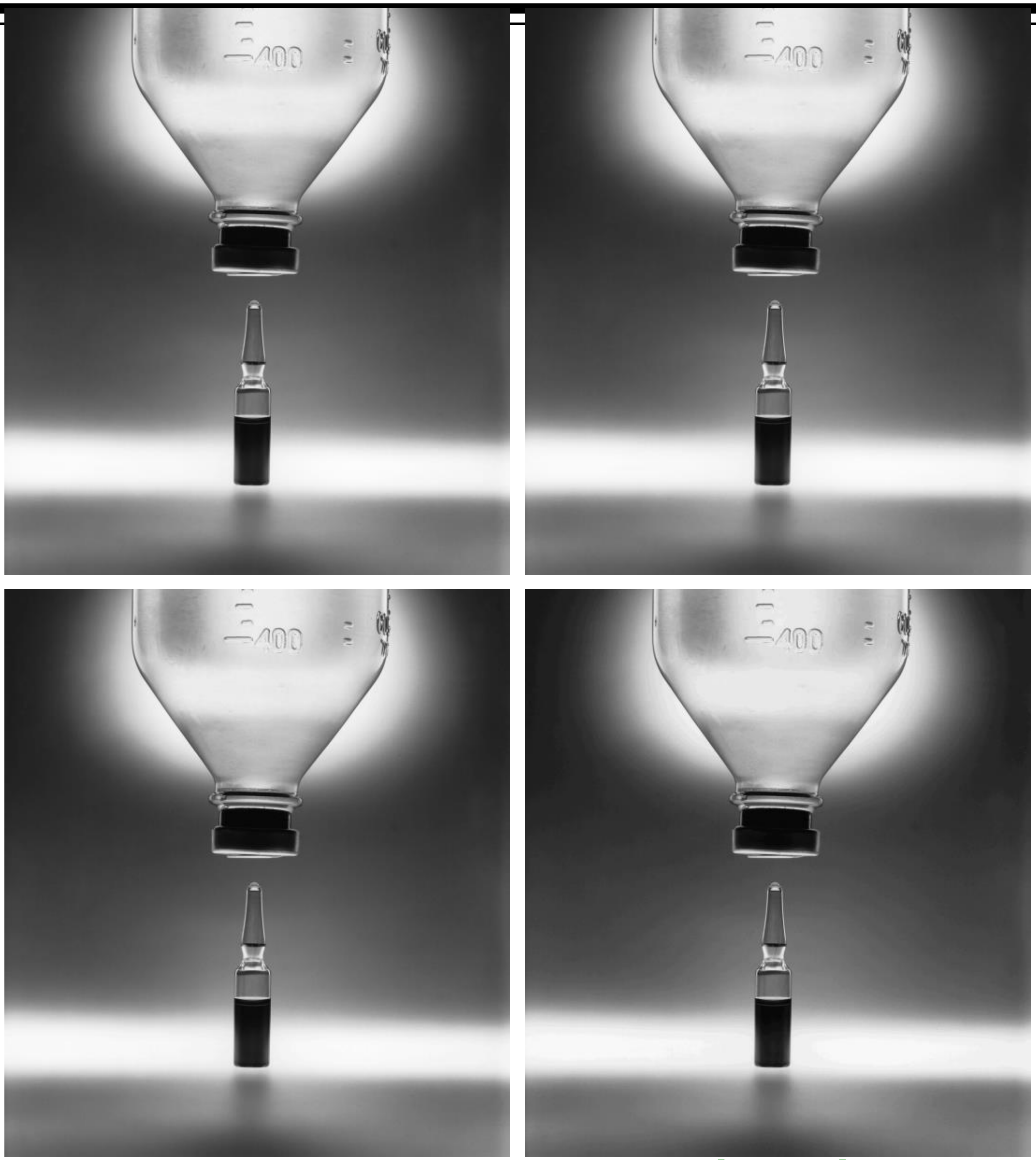
2nd Edition

Intensity Level Resolution

- *Intensity level resolution* refers to the number of intensity levels used to represent the image
  - The more intensity levels used, the finer the level of detail discernible in an image
  - Intensity level resolution is usually given in terms of the number of bits used to store each intensity level

Number of Bits	**Number of Intensity Levels**	**Examples**
1	2	0, 1
2	4	00, 01, 10, 11
4	16	0000, 0101, 1111
8	256	00110011, 01010101
16	65,536	1010101010101010

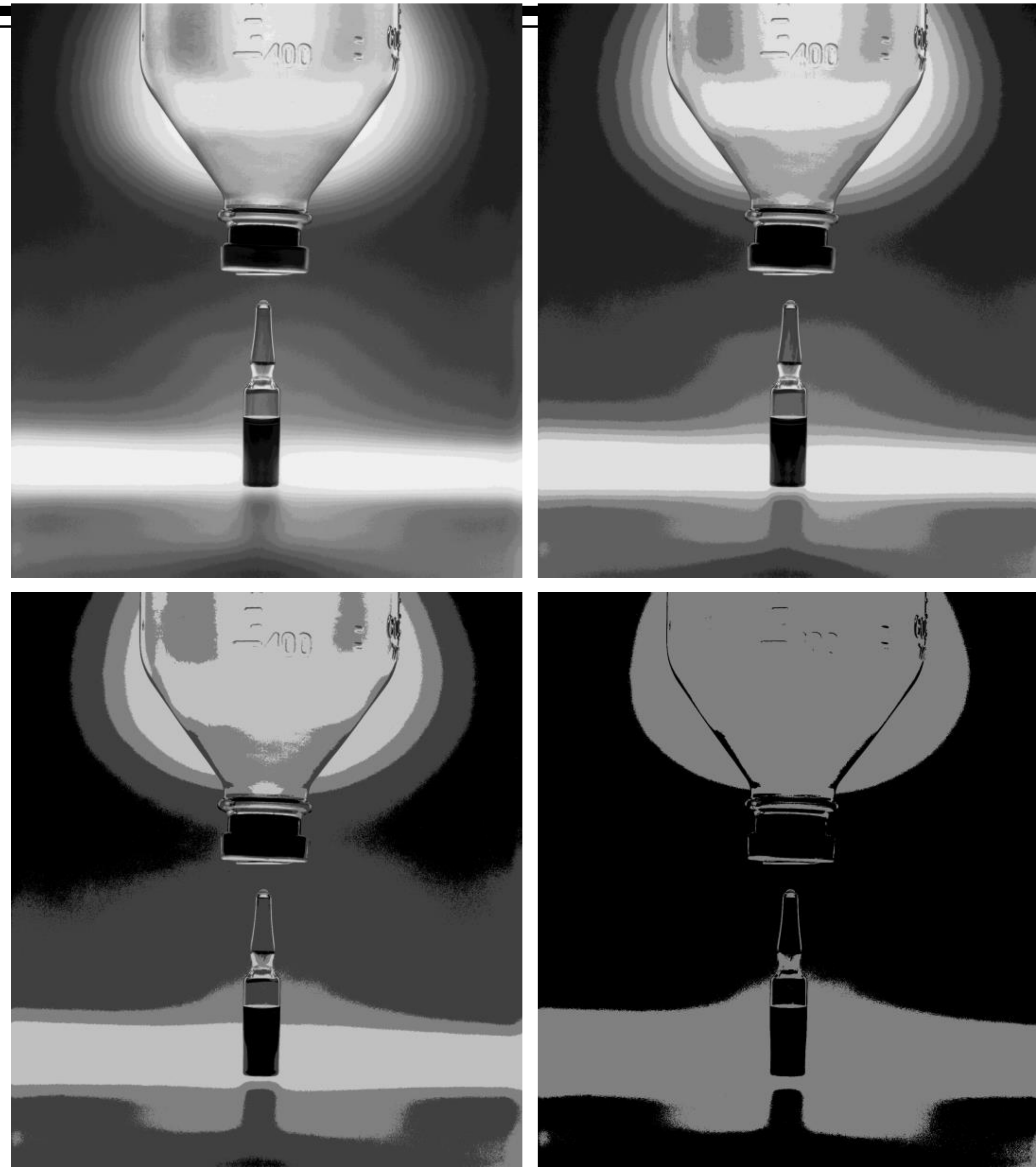
Intensity Level Resolution (cont…)



a b
c d

FIGURE 2.24
(a) 2022 × 1800, 256-level image. (b)-(d) Image displayed in 128, 64, and 32 intensity levels, while keeping the image size constant. (Original image courtesy of the National Cancer Institute.)

Intensity Level Resolution (cont…)



e f
g h

FIGURE 2.24 (*Continued*) (e)-(h) Image displayed in 16, 8, 4, and 2 intensity levels. (Original image courtesy of the National Cancer Institute.)

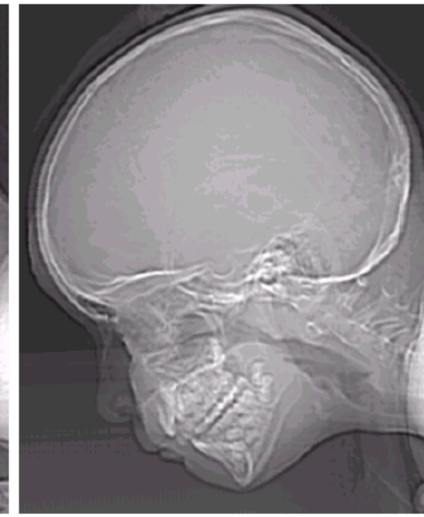
Intensity Level Resolution (cont…)

256 grey levels (8 bits per pixel)

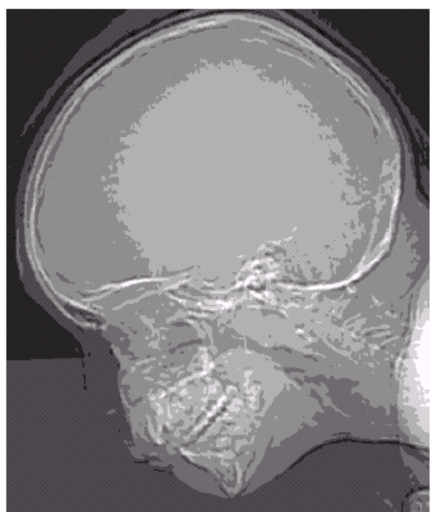
128 grey levels (7 bpp)

64 grey levels (6 bpp)

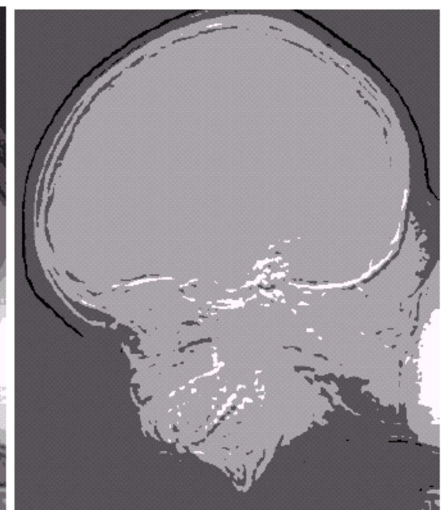
32 grey levels (5 bpp)



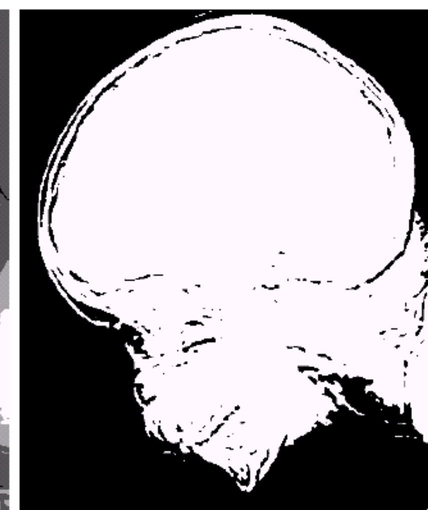
16 grey levels (4 bpp)



8 grey levels (3 bpp)



4 grey levels (2 bpp)



2 grey levels (1 bpp)

Intensity Level Resolution (cont…)

256 grey levels (8 bits per pixel)

Intensity Level Resolution (cont…)

128 grey levels (7 bpp)

Intensity Level Resolution (cont…)

64 grey levels (6 bpp)

Intensity Level Resolution (cont…)

32 grey levels (5 bpp)

Intensity Level Resolution (cont…)

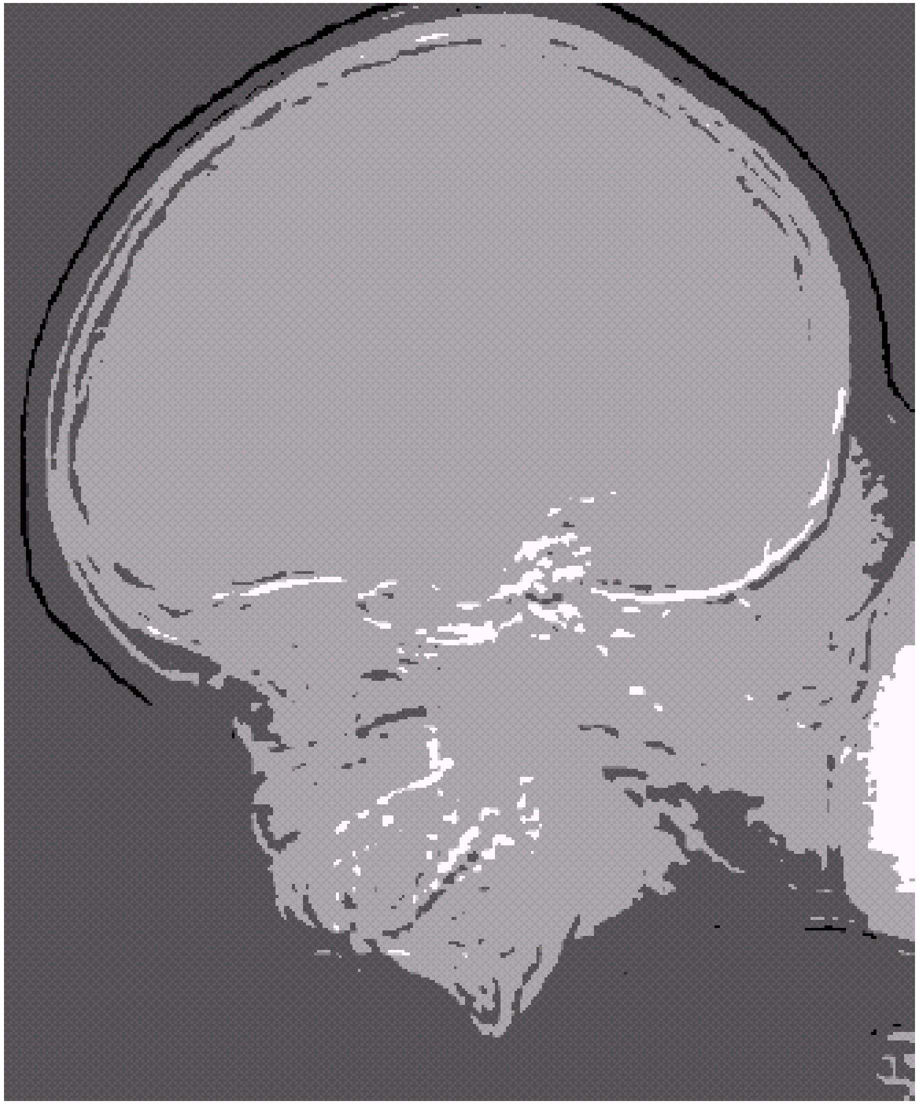
16 grey levels (4 bpp)

Intensity Level Resolution (cont…)

8 grey levels (3 bpp)

Intensity Level Resolution (cont…)

4 grey levels (2 bpp)



Intensity Level Resolution (cont…)

2 grey levels (1 bpp)

Intensity Level Resolution (cont…)

Low Detail

Need less resolution

Medium Detail

High Detail

Need more resolution

a b c

FIGURE 2.25 (a) Image with a low level of detail. (b) Image with a medium level of detail. (c) Image with a relatively large amount of detail. (Image (b) courtesy of the Massachusetts Institute of Technology.)

Image Interpolation

- Use known data to estimate values at unknown locations
- Useful for zooming, shrinking, rotating, and geometric corrections
- Nearest Neighbor Interpolation
  - Use closest pixel to estimate the intensity
  - Simple but has tendency to produce artifacts
- Bilinear Interpolation
  - Use 4 nearest neighbors to estimate the intensity
  - Much better result
- Bicubic Interpolation
  - Use 16 nearest neighbors of a point

Image Interpolation

Nearest Neighbor | Bilinear | Bicubic

a b c

FIGURE 2.27 (a) Image reduced to 72 dpi and zoomed back to its original 930 dpi using nearest neighbor interpolation. This figure is the same as Fig. 2.23(d). (b) Image reduced to 72 dpi and zoomed using bilinear interpolation. (c) Same as (b) but using bicubic interpolation.

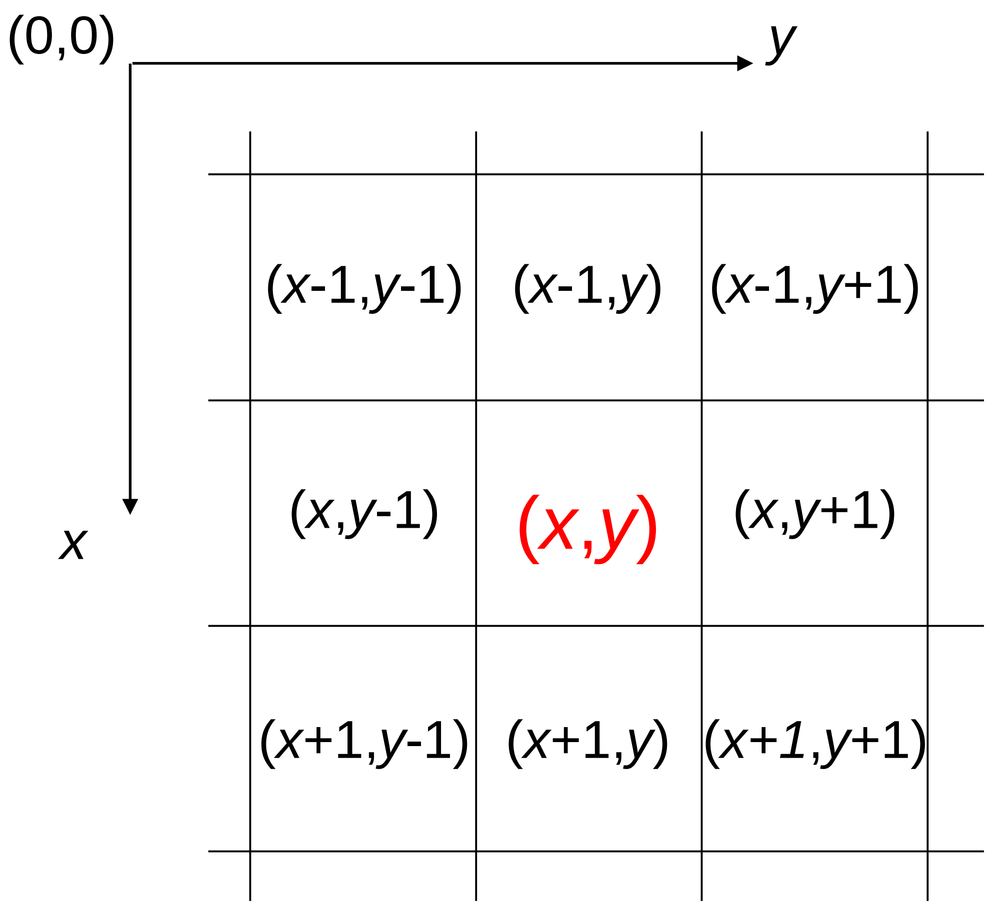
- Interpolation was used after sampling to go back to original size (3692x2812)

  - Nearest Neighbor
  - Bilinear
  - Bicubic

Original at 930 dpi
Fig. 2.23(a)

Matlab commands: imresize or interp2

2.5 Basic Relationship of Pixels



Conventional indexing method

Adjacency of Pixels

- Adjacency is adapted from neighborhood relations
- Two pixels are adjacent if they are in the same class (V)
  - Binary: V = {1} or V = {0}
  - Gray-scale: V = {0} or {1} or {2} . . . {255} or any subset of these
  - Same color or gray-level or same range of intensity (V) AND they are neighbors of one another
- For p and q from the same class
  - 4-Adjacency: p and q are 4-adjacent if $q \in N_4(p)$
  - 8-Adjacency: p and q are 8-adjacent if $q \in N_8(p)$
  - Mixed-Adjacency (m-Adjacency): p and q are m-adjacent IF

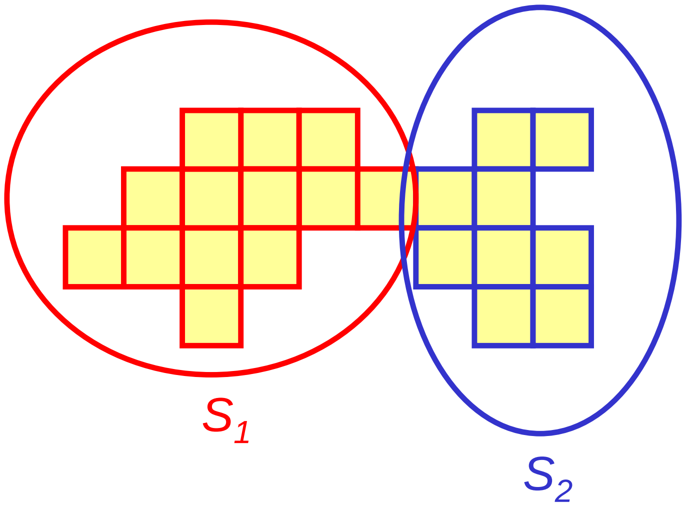
$$\text{(i) } q \in N_4(p) \quad \text{OR} \quad \text{(ii) } q \in N_D(p) \text{ and } [N_4(p) \cap N_4(q) = \varnothing]$$

Diagonal

Adjacency of Pixels

- A pixel *p* is *adjacent* to pixel *q* is they are connected

- Two image subsets S_1 and S_2 are adjacent if some pixel in S_1 is adjacent to some pixel in S_2



WRIGHT STATE UNIVERSITY

4-Neighbors of a Pixel

	$(x\text{-}1,y)$	
$(x,y\text{-}1)$	(x,y)	$(x,y\text{+}1)$
	$(x\text{+}1,y)$	

4-neighbors of *p*:

$$N_4(p) = \left\{ \begin{array}{c} (x-1,y) \\ (x+1,y) \\ (x,y-1) \\ (x,y+1) \end{array} \right\}$$

- 4-neighborhood: Only vertical and horizontal neighbors

Note: $q \in N_4(p)$ implies $p \in N_4(q)$

WRIGHT STATE UNIVERSITY

8-Neighbors of a Pixel

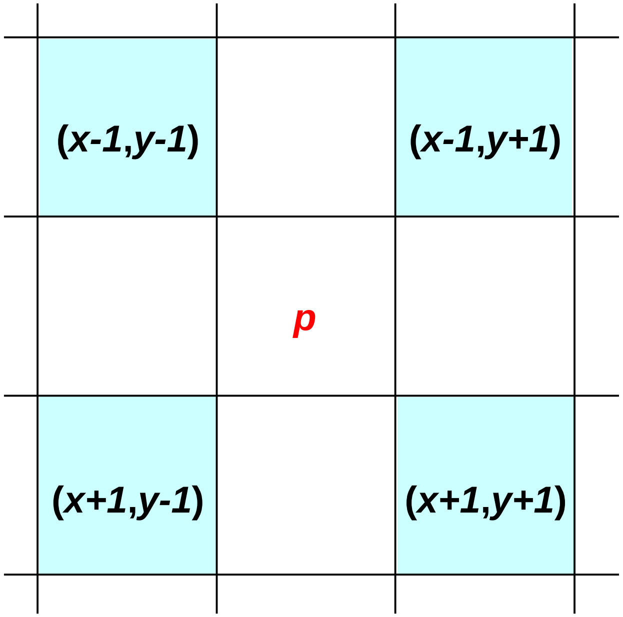
$(x\text{-}1,y\text{-}1)$	$(x\text{-}1,y)$	$(x\text{-}1,y+1)$
$(x,y\text{-}1)$	(x,y)	$(x,y+1)$
$(x+1,y\text{-}1)$	$(x+1,y)$	$(x+1,y+1)$

8-neighbors of *p*:

$$N_8(p) = \left\{ \begin{array}{c} (x-1,y-1) \\ (x,y-1) \\ (x+1,y-1) \\ (x-1,y) \\ (x+1,y) \\ (x-1,y+1) \\ (x,y+1) \\ (x+1,y+1) \end{array} \right\}$$

8-Neighborhood: *ALL* neighboring pixels

Diagonal Neighbors of a Pixel



Diagonal neighbors of *p*:

$$N_D(p) = \left\{ \begin{array}{c} (x-1,y-1) \\ (x+1,y-1) \\ (x-1,y+1) \\ (x+1,y+1) \end{array} \right\}$$

Diagonal – Neighborhood: Only *Diagonal* neighbor pixels

Connectivity Path Between Two Pixels

- A ***path*** from pixel p at (x, y) to pixel q at (s, t) is a sequence of distinct pixels with similar gray levels:

$$(x_0, y_0), (x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n),$$

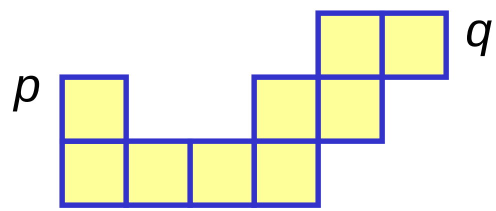
such that

$$(x_0, y_0) = (x, y) \text{ and } (x_n, y_n) = (s, t)$$

and

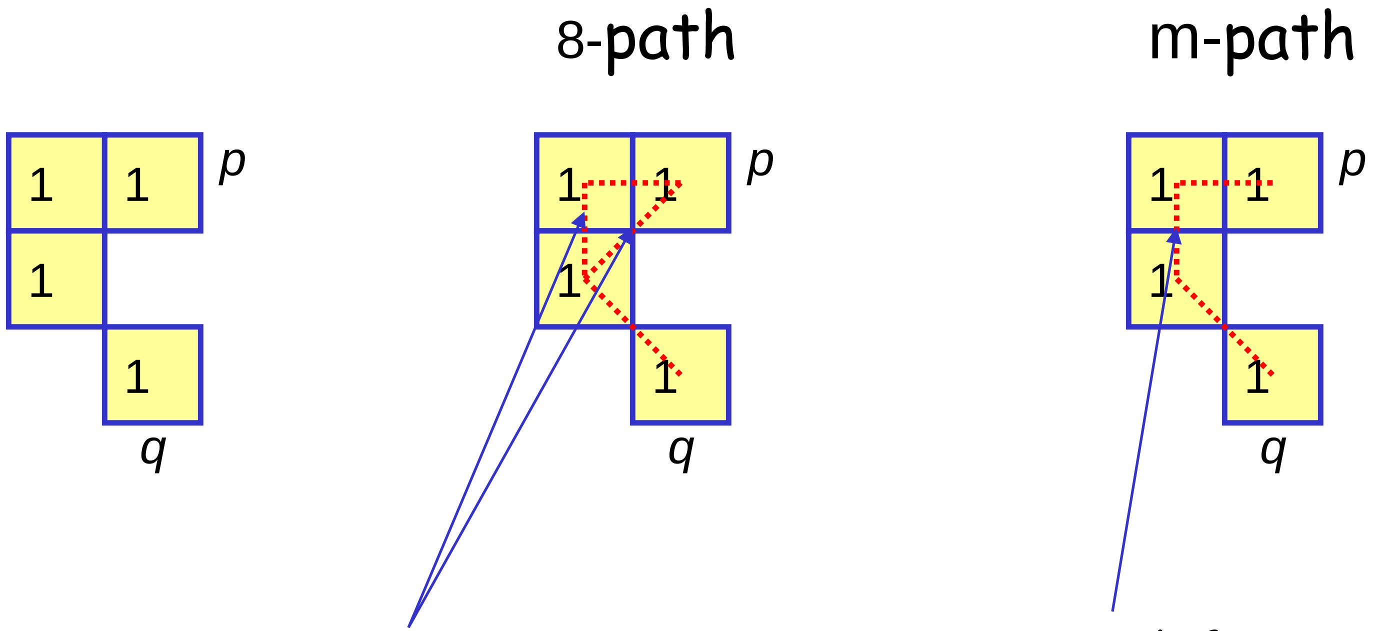
$$(x_i, y_i) \text{ is } \underline{\text{adjacent}} \text{ to } (x_{i-1}, y_{i-1}), \qquad i = 1, \ldots, n$$

n = length of path



- We can define type of path: 4-path, 8-path or m-path depending on type of adjacency

Path Determination



8-path from p to q results in some ambiguity

m-path from p to q solves this ambiguity

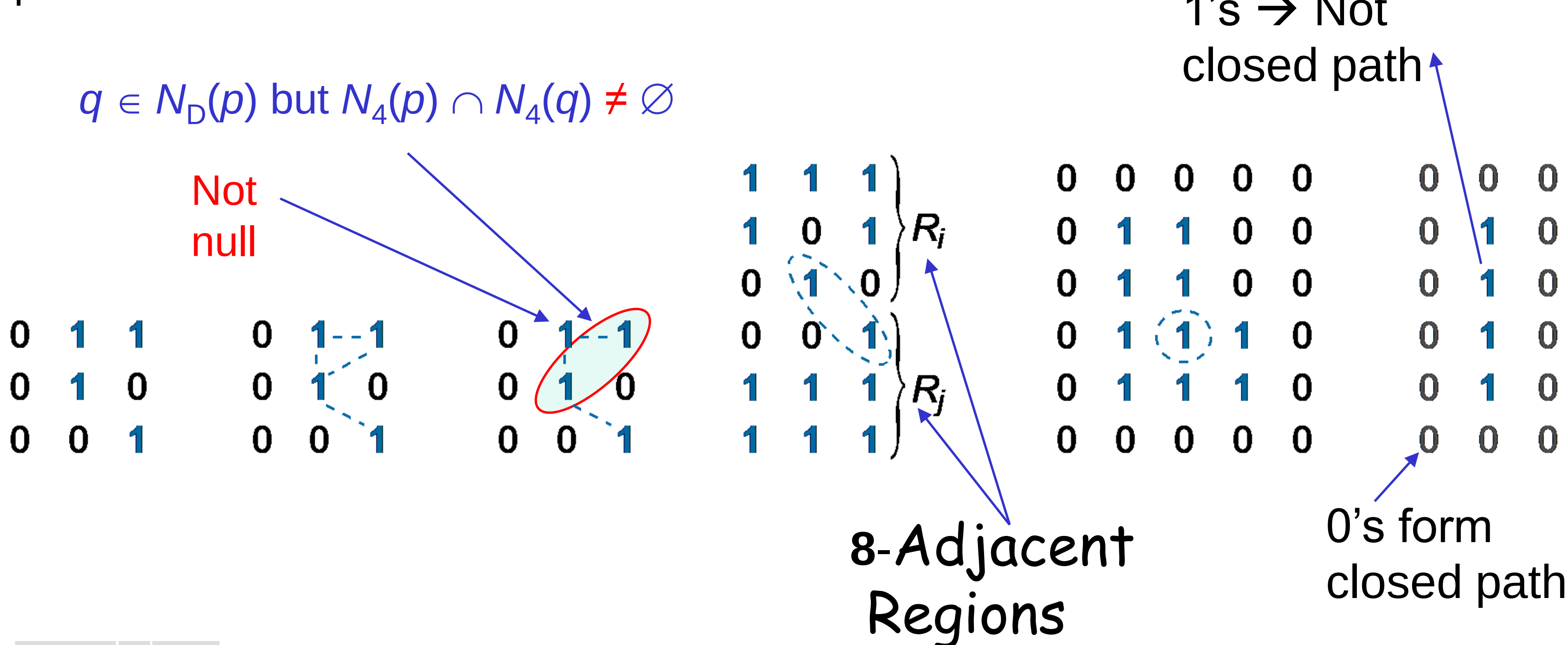
(i) $q \in N_4(p)$ OR (ii) $q \in N_D(p)$ and $N_4(p) \cap N_4(q) = \varnothing$

See Next Page for details →

Path Determination

Two image subset S1and S2 are adjacent if some pixel in S1 is adjacent to some pixels in S2.

$q \in N_D(p)$ but $N_4(p) \cap N_4(q) \neq \varnothing$



a b c d e f

FIGURE 2.28 (a) An arrangement of pixels. (b) Pixels that are 8-adjacent (adjacency is shown by dashed lines). (c) *m*-adjacency. (d) Two regions (of 1's) that are 8-adjacent. (e) The circled point is on the boundary of the 1-valued pixels only if 8-adjacency between the region and background is used. (f) The inner boundary of the 1-valued region does not form a closed path, but its outer boundary does.

2.5.3 Distance Measures

- Euclidean, city-block (D_4), chessboard (D_8) and D_m-distance.

- Euclidian: $D_e(p,q) = \sqrt{(x-s)^2 + (y-t)^2}$
- City-Block: $D_4(p,q) = |x-s| + |y-t|$
- Chessboard: $D_8(p,q) = \max(|x-s|, |y-t|)$

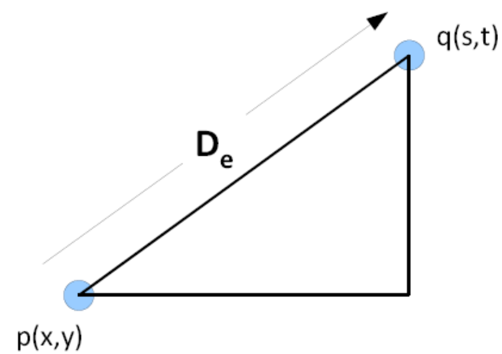
Properties of Distance Metric

D : A ***distance function*** or ***metric*** if for pixel p, q, and z with coordinates (x,y), (s,t) and (u,v), following 3 constraints hold

- $D(p, q) \geq 0$ ($D(p, q) = 0$ if and only if $p = q$) (non-negative)
- $D(p, q) = D(q, p)$ (symmetric)
- $D(p, z) \leq D(p, q) + D(q, z)$ (Triangular inequality)

Example: Euclidean distance

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



NOT Integer valued → Could be Fractions

City-Block (Manhattan) Distance

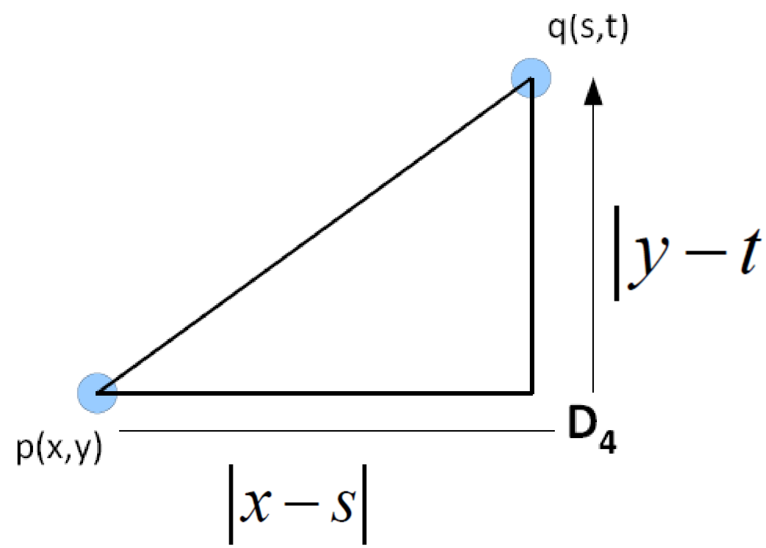
D_4-distance (City-Block distance):

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

(Integer valued)

- Diamond-centered at (x, y)
- Pixels with $D_4(p) = 1$ are 4-neighbors of p

		2		
	2	1	2	
2	1	0	1	2
	2	1	2	
		2		



Chessboard Distance

D_8-distance *(chessboard distance)* is defined as

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

(Integer valued)

- Square-centered at (x, y)
- Pixels with $D_8(p) = 1$ is 8-neighbors of *p*

2	2	2	2	2
2	1	1	1	2
2	1	0	1	2
2	1	1	1	2
2	2	2	2	2

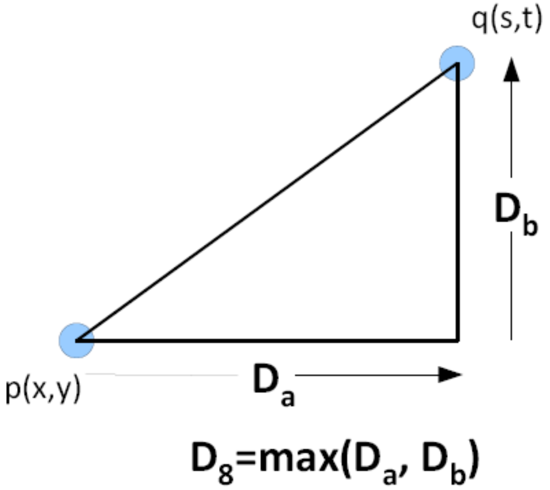


Image Range Calibration to Utilize All Bits

- Arithmetic operation can produce pixel values outside of the range [0 - 255] (For 8-bit Pixels)
- Need to ensure that image is displayed properly AND
- Guarantee full range of values from arithmetic operations
- **Solution:** Convert ALL values back to the range [0 - 255]
- Resulting values (g_s) "captured" as → Fixed number of bits
- Step-1 (Shift): Subtract the minimum: $g_m = g - \min(f)$
  - Creates image with minimum value = 0
- Step-2: Scale the image: $g_s = K\,[\,g_m / max(g_m)]$
- Final Image: In the range $[0, K]$ → K=255 for 8-bit image

Resulting Range: $[g_{\min} - g_{\max}]$ → $[\,0 - 255]$

2.6.3 Image Enhancement using Arithmetic/Logic Operations

- **Arithmetic/ Logic operations** involving images are performed on a pixel-by-pixel basis between two or more images [Except NOT operation → Requires One image].
- **Logic Operations**: AND, OR and NOT - Functionally complete and used usually for logical masking for selecting sub-images (ROI processing) or morphological operations.
- **Arithmetic Operations**: Subtraction, Addition, Division, Multiplication.
- **Image Multiplication**: Used in enhancement primarily as a masking operation that is more general than logical masks.
  - Used to implement gray-level, rather than binary masks.
- **Division** of two images is equivalent to multiplication of one image by the reciprocal of the other.

Arithmetic Operations

- Array operations between images
  - Carried out between corresponding pixel pairs
  - Arithmetic Operations

$$s(x, y) = f(x, y) + g(x, y)$$

$$d(x, y) = f(x, y) - g(x, y)$$

$$p(x, y) = f(x, y) \times g(x, y)$$

$$v(x, y) = f(x, y) \div g(x, y)$$

- e.g., Averaging K different noisy images can decrease noise
  - Used in the field of astronomy

Point Processing Via Arithmetic/Logic Operations

- Two images of same size can be combined using operations of addition, subtraction, multiplication, division, logical AND, OR, XOR and NOT
- Such operations are done on pairs of their corresponding pixels
- Often only one of the images is a real picture while the other is a machine generated Mask
- Mask: Often a binary image consisting only of pixel values 0 and 1

Image Averaging

- A noisy image $g(x,y)$ can be defined by

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

where, $f(x,y)$: Original image

$\eta(x,y)$: Random noise

- One simple way to reduce granular noise is to take several identical pictures and average them, thus smoothing out the randomness.

Image Averaging

- Consider a series of noisy image:

$$g_i(x,y) = f(x,y) + \eta_i(x,y)$$

- Where $\eta_i(x, y)$ is the noise image and assume that the noise at any point is uncorrelated and has zero average value. Let us average K different noisy images

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

- Then it follows that

$$E\left\{\bar{g}(x,y)\right\} = f(x,y)$$

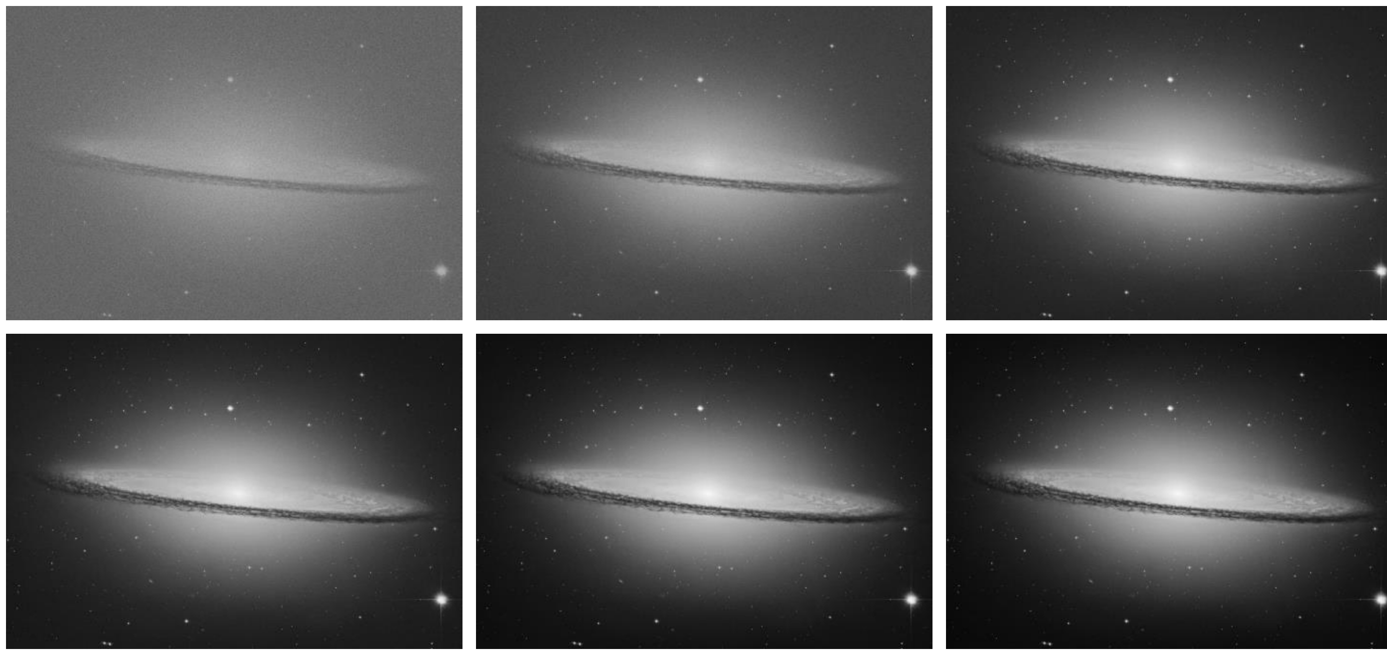
$$\sigma^2_{\bar{g}(x,y)} = \frac{1}{K}\sigma^2_{\eta_i(x,y)}$$

- The signal-to-noise ratio (SNR) will be decreased to $1/K$ times of the original output image

Noise Reduction by Image Averaging

- Image quality can be improved by averaging a number of images together (Very useful in astronomy applications)

- **Effectively decreases noise**
- **Assumes images are fully registered**



a b c
d e f

FIGURE 2.29 (a) Sample noisy image of the Sombrero Galaxy. (b)-(f) Result of averaging 10, 50, 100, 500, and 1,000 noisy images, respectively. All images are of size 1548 × 2238 pixels, and all were scaled so that their intensities would span the full [0, 255] intensity scale. (Discovered in 1767, the Sombrero Galaxy is 28 light years from Earth. Original image courtesy of NASA.)

WRIGHT STATE UNIVERSITY

Another Application in Astronomy

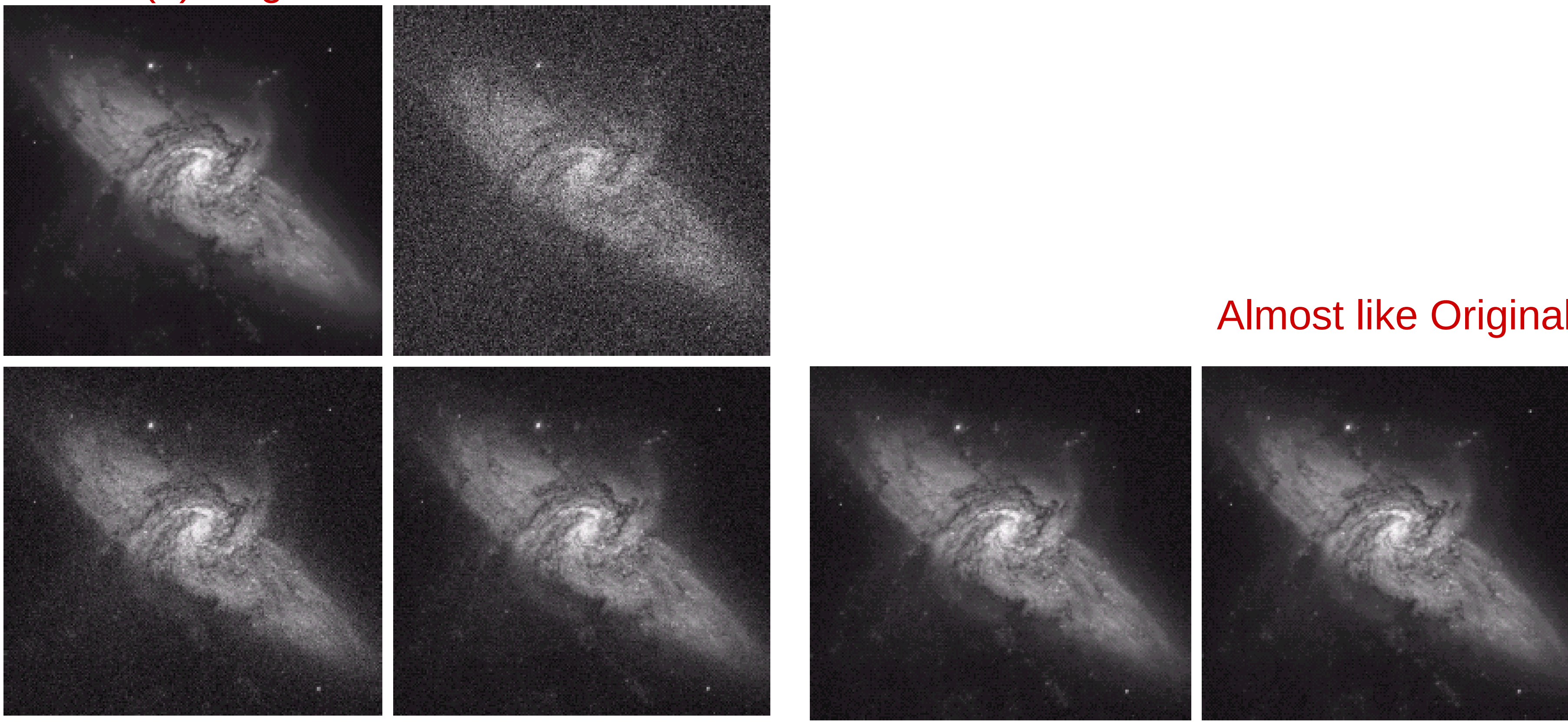
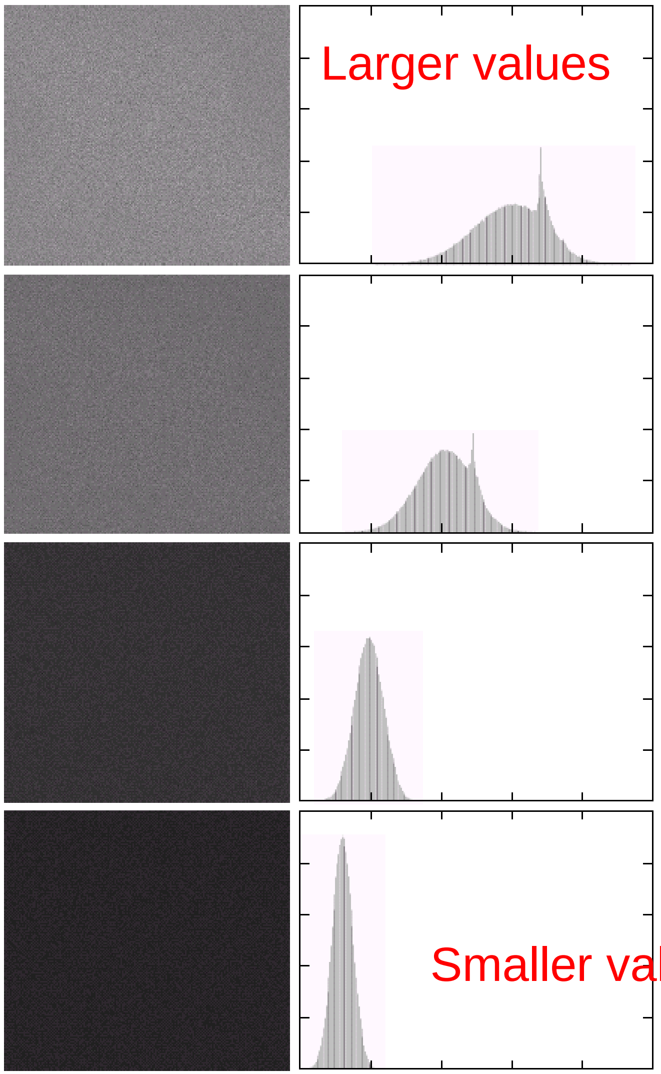


FIGURE 3.30 (a) Image of Galaxy Pair NGC 3314. (b) Image corrupted by additive Gaussian noise with zero mean and a standard deviation of 64 gray levels. (c)–(f) Results of averaging $K = 8, 16, 64,$ and 128 noisy images. (Original image courtesy of NASA.)

2nd Edition

Application in Astronomy (contd.)

Differences Histograms



a b

FIGURE 3.31 (a) From top to bottom: Difference images between Fig. 3.30(a) and the four images in Figs. 3.30(c) through (f), respectively. (b) Corresponding histograms.

Larger

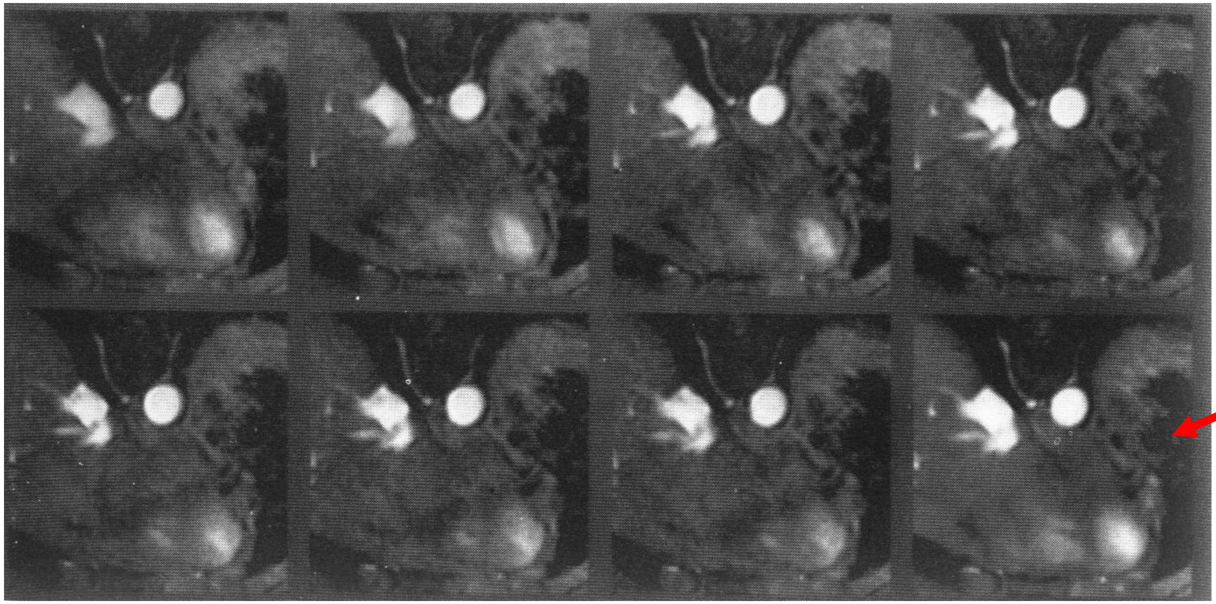
Smaller

Standard deviation and mean of the difference image

Conclusion:
Averaging reduces both mean and variance of noise!

2nd Edition

Application in Medical Imaging

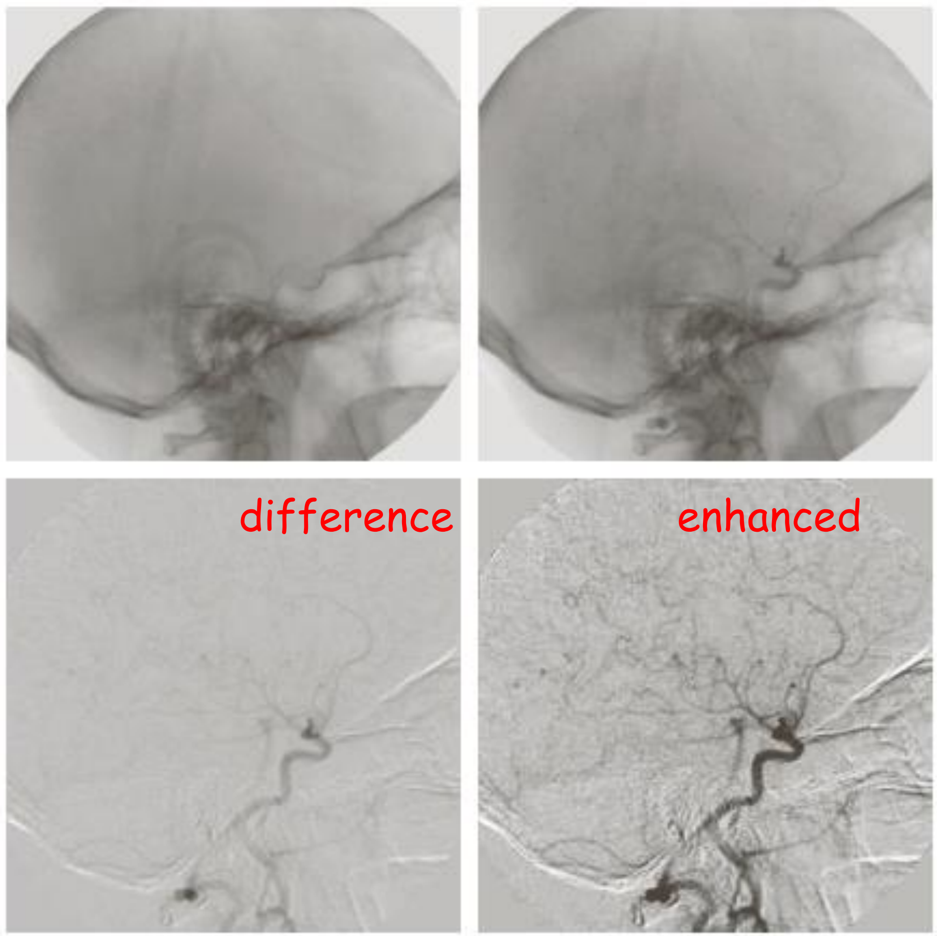


Average of the other seven images

- Gated MR images of the heart taken at the same time point in seven successive heartbeats (first seven images). The bottom right image is the sum of the other seven to minimize the reconstruction artifacts in each original image.
- Important: All images being averaged must be registered (aligned) in order to avoid the introduction of blurring and other artifacts in the output image.

Image Subtraction – Medical Application

- Iodine medium injected into the bloodstream

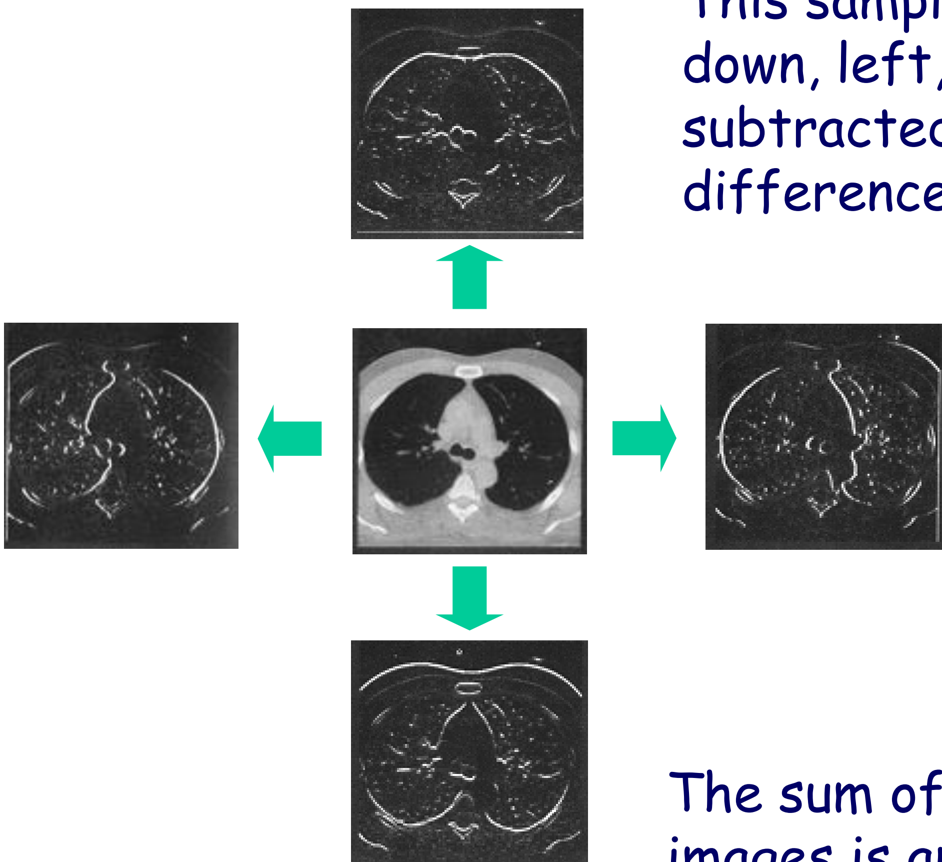


a b
c d

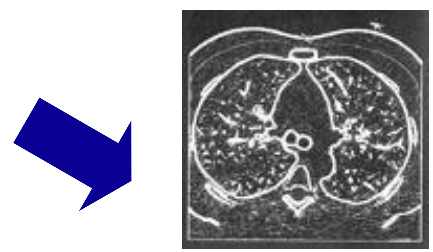
FIGURE 2.32 Digital subtraction angiography. (a) Mask image. (b) A live image. (c) Difference between (a) and (b). (d) Enhanced difference image. (Figures (a) and (b) courtesy of the Image Sciences Institute, University Medical Center, Utrecht, The Netherlands.)

Image Subtraction in Edge-Strength Image

This sample image is shifted one pixel up, down, left, and right respectively, then subtracted from the original to create four difference images.



Difference → Equivalent to Taking Derivative (Studied Later)



The sum of the four shifted difference images is an estimate of grayscale gradient magnitude

Multiplication/Division – Shading Removal

- Suppose a sensor introduces some shading in the form

$$g(x,y)=f(x,y)\,h(x,y)$$

- Estimate $h(x,y)$ and remove shading by division

Original　　Shade Pattern　　Shade Correction

a b c

FIGURE 2.33 Shading correction. (a) Shaded test pattern. (b) Estimated shading pattern. (c) Product of (a) by the reciprocal of (b). (See Section 3.5 for a discussion of how (b) was estimated.)

Multiplication/Division – Shading Removal

- Suppose a sensor introduces some shading in the form

$$g(x,y)=f\,(x,y)\;h(x,y)$$

- Estimate $h(x,y)$ and remove shading by division

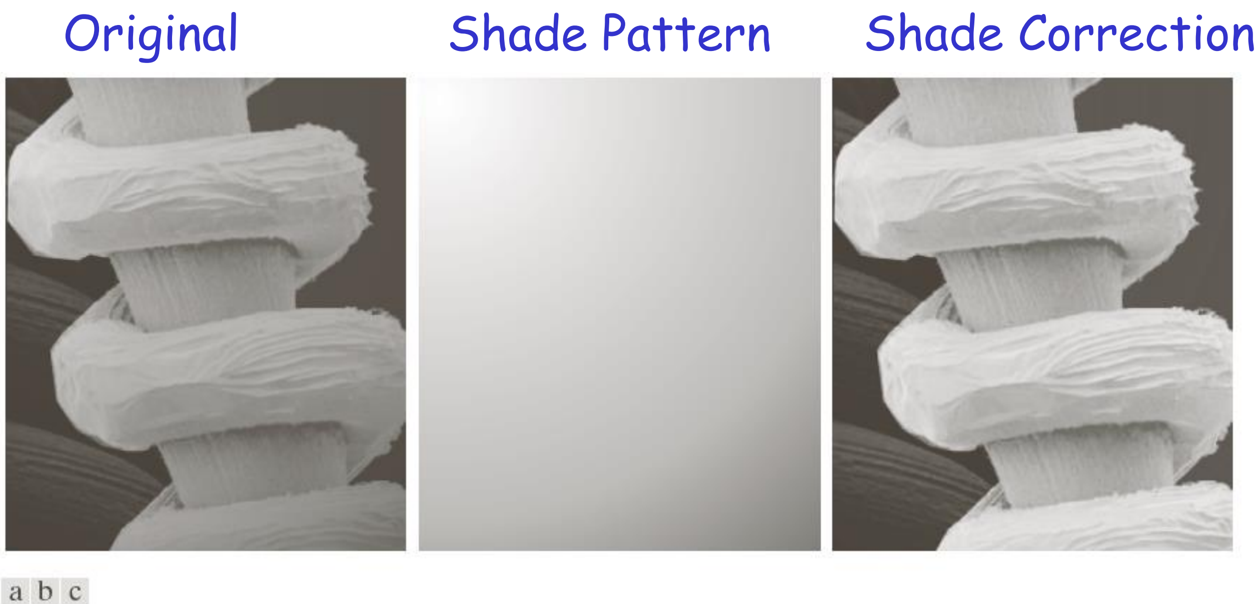
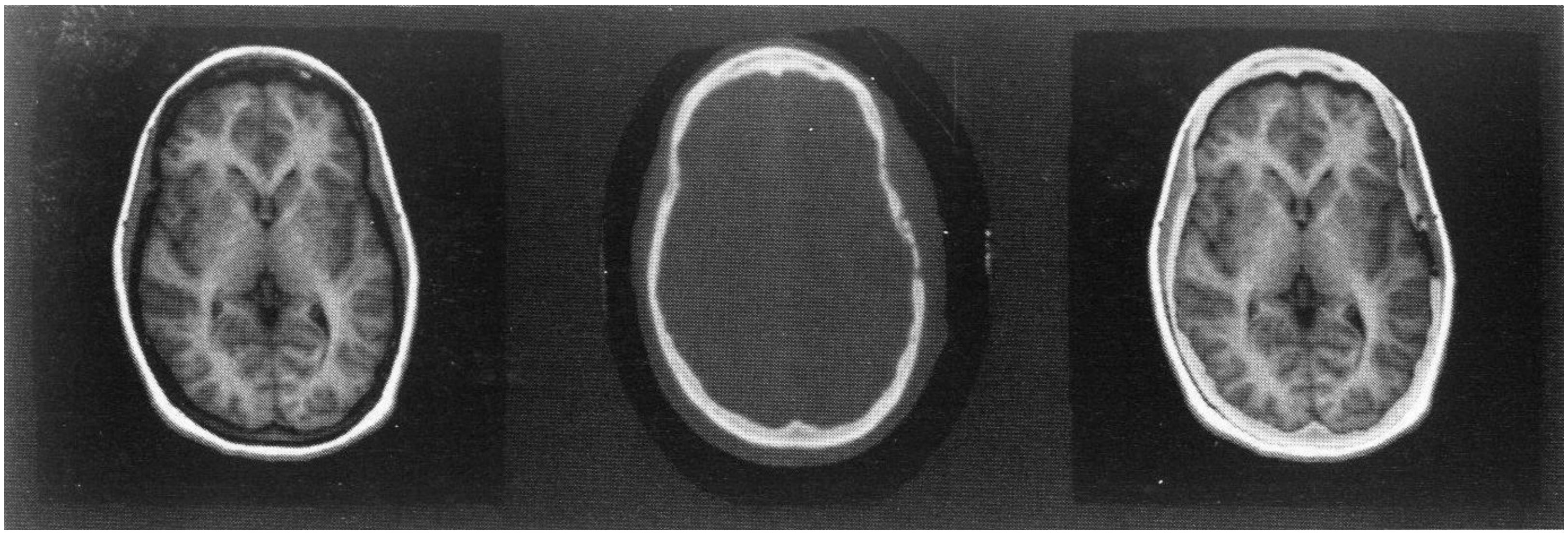


FIGURE 2.29 Shading correction. (a) Shaded SEM image of a tungsten filament and support, magnified approximately 130 times. (b) The shading pattern. (c) Product of (a) by the reciprocal of (b). (Original image courtesy of Mr. Michael Shaffer, Department of Geological Sciences, University of Oregon, Eugene.)

3rd Edition

Image Multiplication and Division

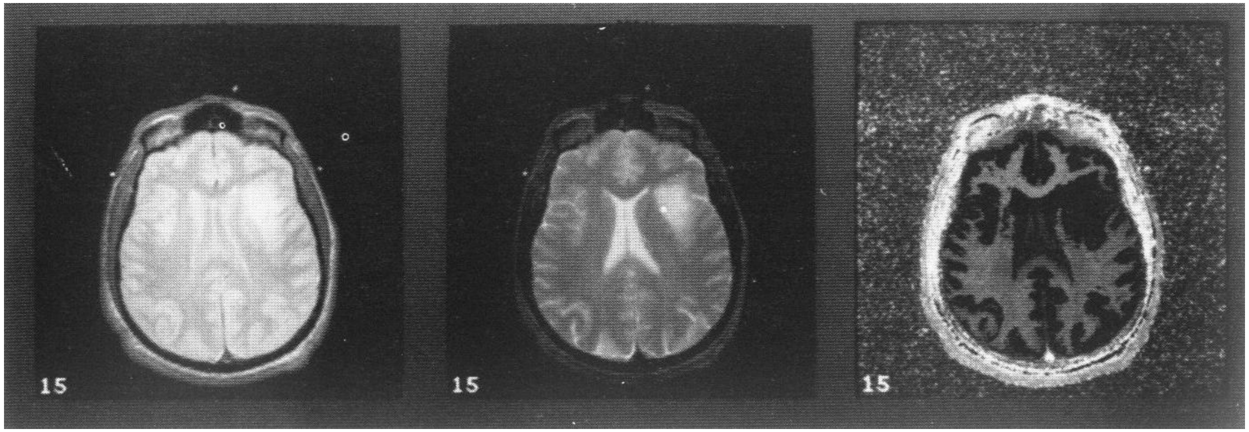
- Image multiplication finds use in enhancement primarily as a masking operation



- MR image combined with bone field of co-registered CT image produces composite output image

Image Division

- Image division is most often used to combine complementary medical image modalities



- In MRI, the T1 image and T2 image of the brain show white and gray matter in different ways, but the T1 $\div$T2 image shows this difference more dramatically.

- In general, division of two images is considered simply as multiplication of one image by the reciprocal of the other.

Set and Logical Operations

- Union, Intersection, Complement & Difference Operations
  - Elements of sets: Coordinates of pixels (ordered pairs of integers) representing regions (objects) in an image

TABLE 2.1
Some important set operations and relationships.

Description	Expressions
Operations between the sample space and null sets	$\Omega^c = \varnothing;\ \varnothing^c = \Omega;\ \Omega \cup \varnothing = \Omega;\ \Omega \cap \varnothing = \varnothing$
Union and intersection with the null and sample space sets	$A \cup \varnothing = A;\ A \cap \varnothing = \varnothing;\ A \cup \Omega = \Omega;\ A \cap \Omega = A$
Union and intersection of a set with itself	$A \cup A = A;\ A \cap A = A$
Union and intersection of a set with its complement	$A \cup A^c = \Omega;\ A \cap A^c = \varnothing$
Commutative laws	$A \cup B = B \cup A$ $A \cap B = B \cap A$
Associative laws	$(A \cup B) \cup C = A \cup (B \cup C)$ $(A \cap B) \cap C = A \cap (B \cap C)$
Distributive laws	$(A \cup B) \cap C = (A \cap C) \cup (B \cap C)$ $(A \cap B) \cup C = (A \cup C) \cap (B \cup C)$
DeMorgan's laws	$(A \cup B)^c = A^c \cap B^c$ $(A \cap B)^c = A^c \cup B^c$

Set and Logical Operations

- Union, Intersection, Complement & Difference Operations
  - Elements of sets: Coordinates of pixels (ordered pairs of integers) representing regions (objects) in an image

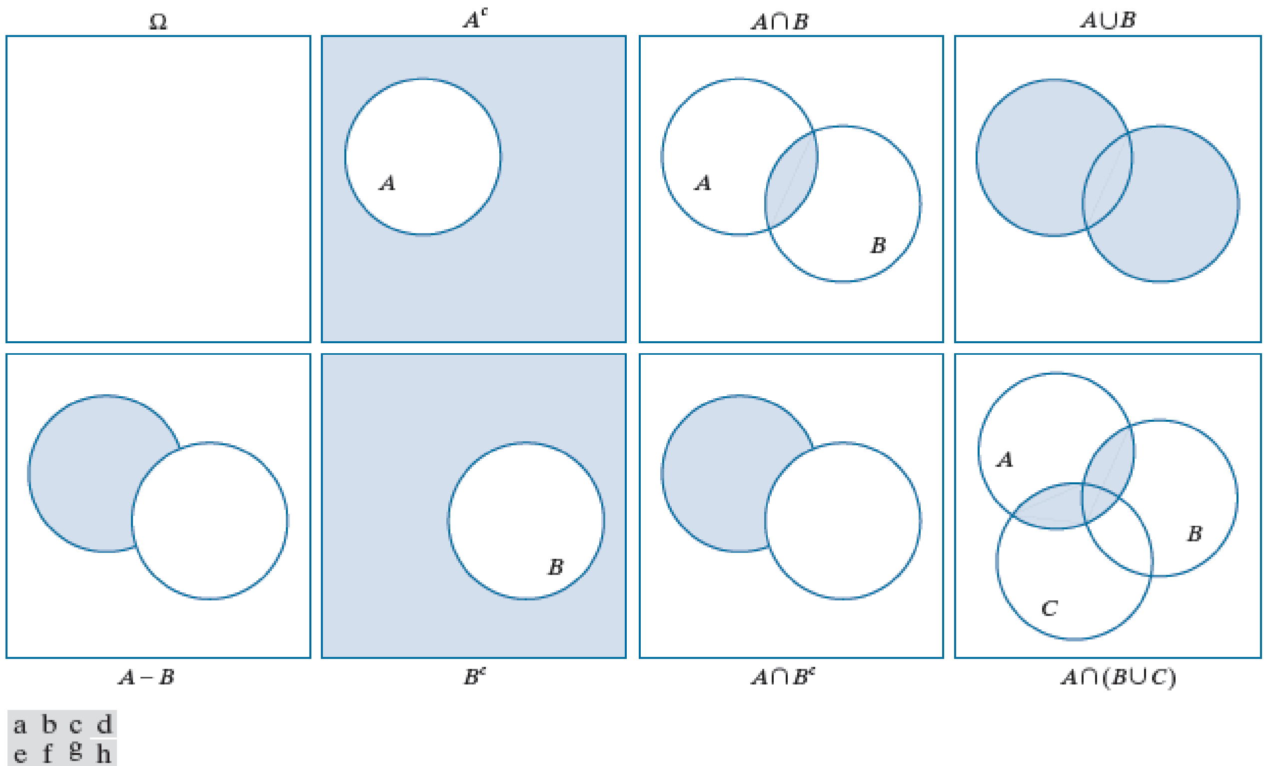


FIGURE 2.35 Venn diagrams corresponding to some of the set operations in Table 2.1. The results of the operations, such as A^c, are shown shaded. Figures (e) and (g) are the same, proving via Venn diagrams that $A - B = A \cap B^c$ [see Eq. (2-40)].

Logical Operations

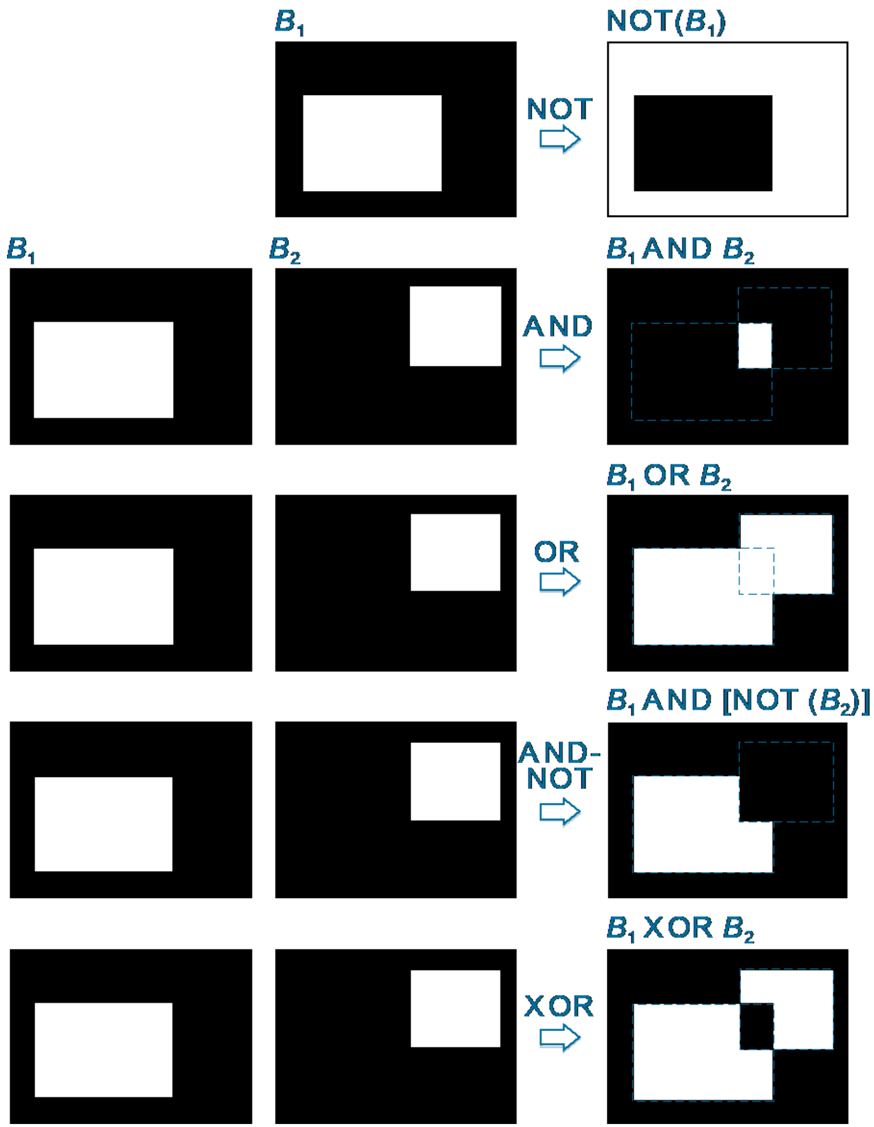


FIGURE 2.37 Illustration of logical operations involving foreground (white) pixels. Black represents binary 0's and white binary 1's. The dashed lines are shown for reference only. They are not part of the result.

WRIGHT STATE UNIVERSITY

Spatial Operations

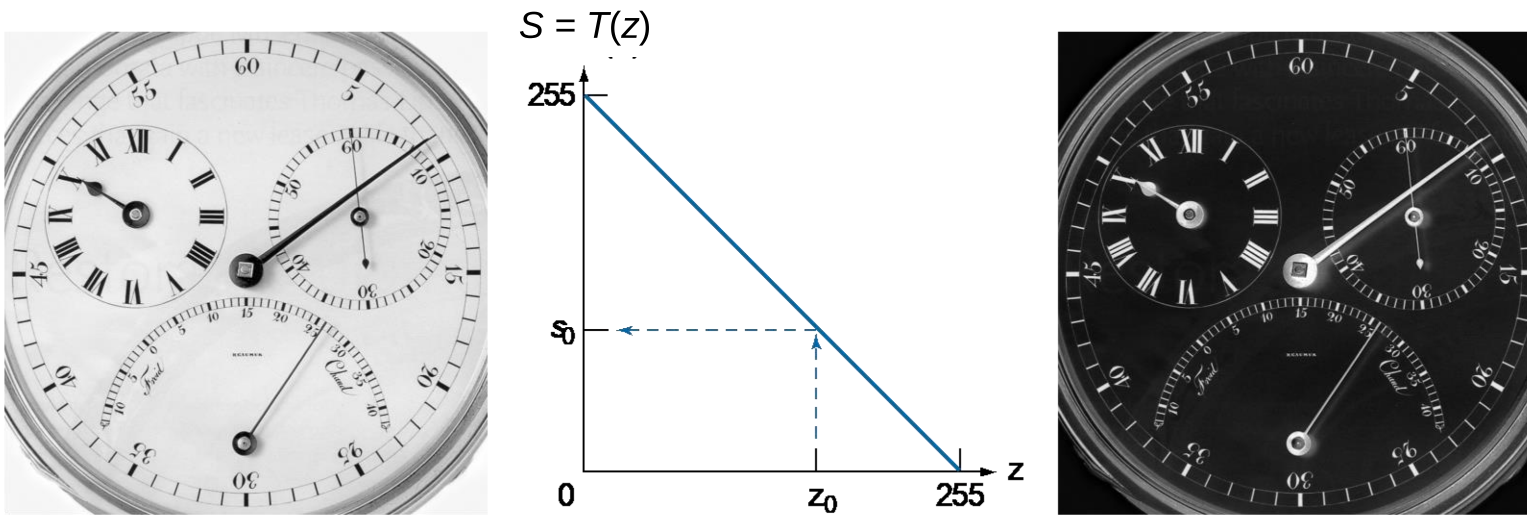
- Geometric spatial transformations $(x, y) = T\{(v, w)\}$
  - Also known as: "Rubber-sheet" transformations
  - Consists of two operations
    - Spatial transformation of coordinates

      e.g. $(x, y) = T\{(v, w)\} = (v/2, w/2)$ → shrinks to half-size
    - Affine Transforms: scale, rotate, transform, or shear a set of points

$$[x \quad y \quad 1] = [v \quad w \quad 1]\mathbf{T} = [v \quad w \quad 1]\begin{bmatrix} t_{11} & t_{12} & 0 \\ t_{21} & t_{22} & 0 \\ t_{31} & t_{32} & 1 \end{bmatrix}$$

    - Intensity interpolation (discussed earlier)

Spatial Operations – Image Negative



a b c

Figure 2.38 (a) An 8-bit image. (b) Intensity transformation function used to obtain the digital equivalent of a "photographic" negative of an 8-bit image. The arrows show transformation of an arbitrary input intensity value z_0 into its corresponding output value s_0. (c) Negative of (a), obtained using the transformation function in (b).

Set Operations – Image Negative

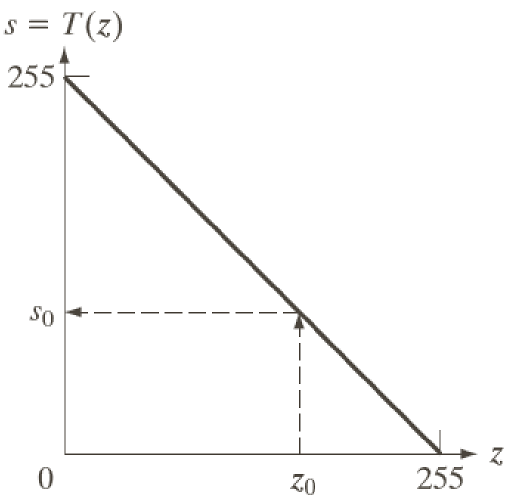
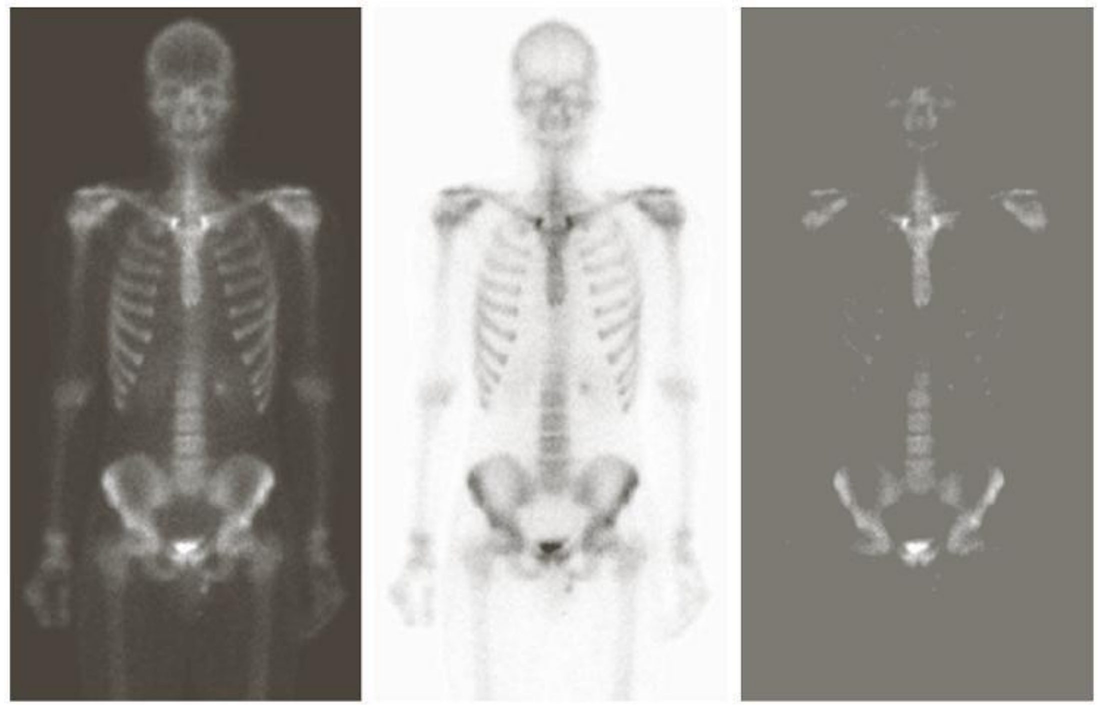


Figure 2.38b Intensity transformation function used to obtain the negative of an 8-bit image. The dashed arrows show transformation of an arbitrary input intensity value z_0 into its corresponding output value s_0.

Original | Negative | Original ∪ Constant



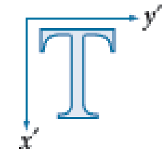
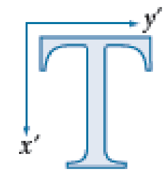
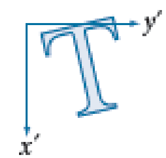
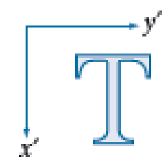
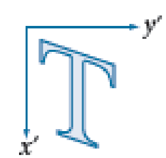
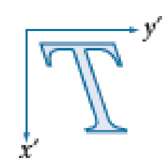
a b c

FIGURE 2.36 Set operations involving grayscale images. (a) Original image. (b) Image negative obtained using grayscale set complementation. (c) The union of image (a) and a constant image. (Original image courtesy of G.E. Medical Systems.)

WRIGHT STATE UNIVERSITY

Affine Transformations

TABLE 2.3
Affine transformations based on Eq. (2-45).

Transformation Name	Affine Matrix, A	Coordinate Equations	Example
Identity	$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$x' = x$ $y' = y$	
Scaling/Reflection (For reflection, set one scaling factor to −1 and the other to 0)	$\begin{bmatrix} c_x & 0 & 0 \\ 0 & c_y & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$x' = c_x x$ $y' = c_y y$	
Rotation (about the origin)	$\begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$x' = x\cos\theta - y\sin\theta$ $y' = x\sin\theta + y\cos\theta$	
Translation	$\begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix}$	$x' = x + t_x$ $y' = y + t_y$	
Shear (vertical)	$\begin{bmatrix} 1 & s_v & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$x' = x + s_v y$ $y' = y$	
Shear (horizontal)	$\begin{bmatrix} 1 & 0 & 0 \\ s_h & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$x' = x$ $y' = s_h x + y$	

Transformation + Interpolation

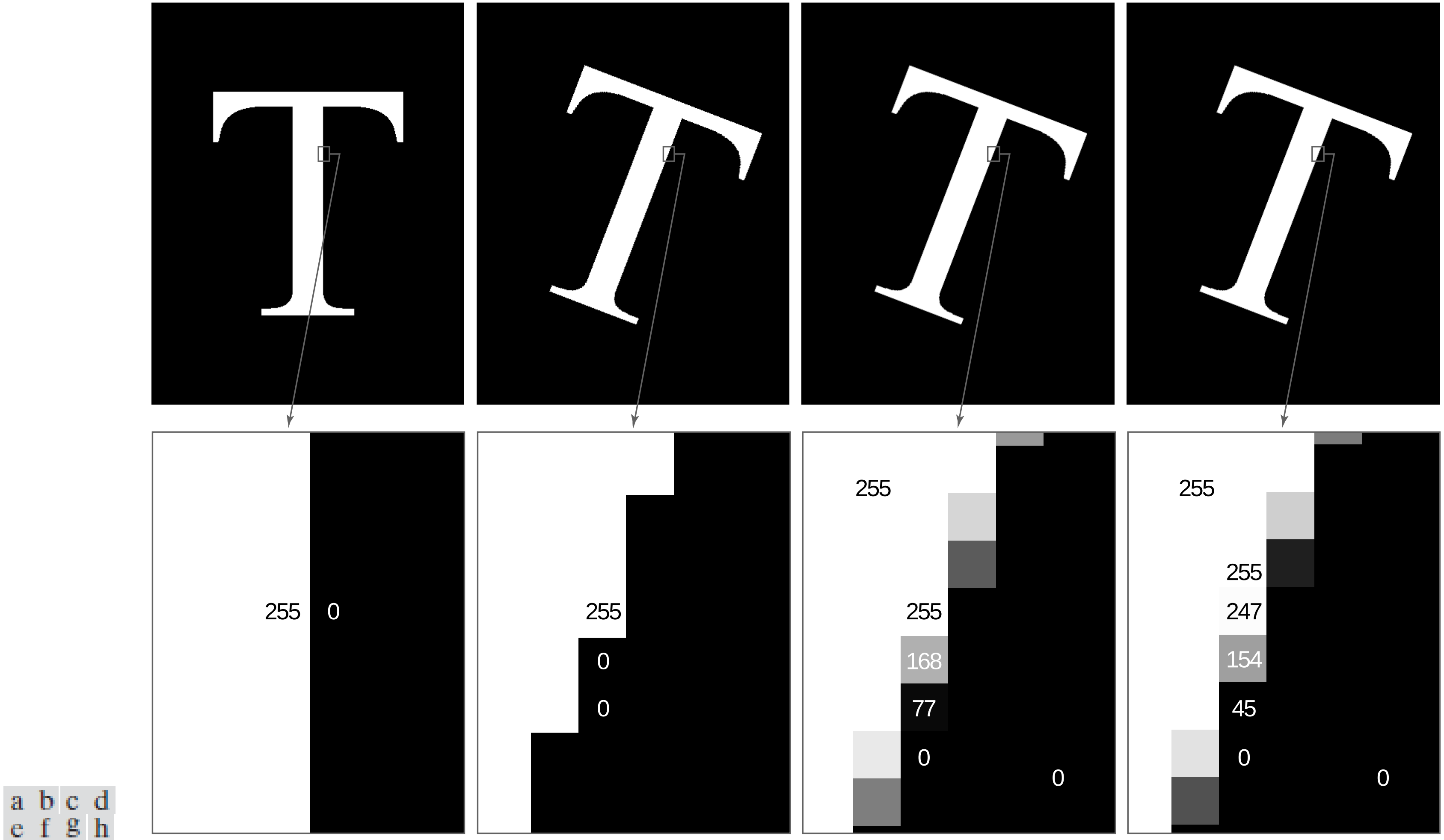


FIGURE 2.40 (a) A 541 × 421 image of the letter T. (b) Image rotated –21° using nearest-neighbor interpolation for intensity assignments. (c) Image rotated –21° using bilinear interpolation. (d) Image rotated –21° using bicubic interpolation. (e)-(h) Zoomed sections (each square is one pixel, and the numbers shown are intensity values).

Image Rotation

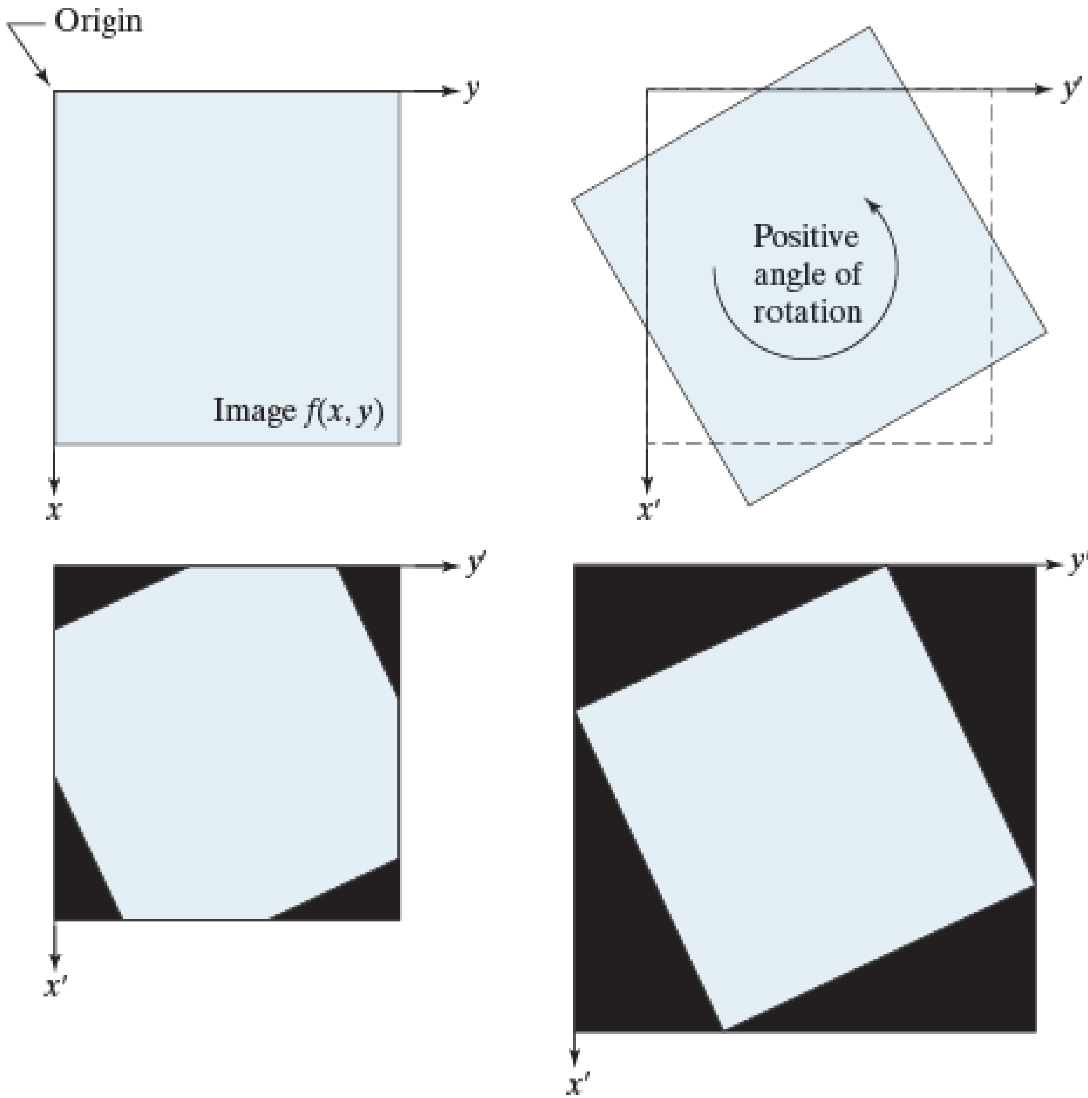


FIGURE 2.41
(a) A digital image.
(b) Rotated image (note the counterclockwise direction for a positive angle of rotation).
(c) Rotated image cropped to fit the same area as the original image.
(d) Image enlarged to accommodate the entire rotated image.

Image Registration

- Input and output images are available but the transformation function is unknown.

Goal: Estimate the transformation function and use it to register the two images.

- One of the principal approaches for image registration is to use ***tie points*** (also called ***control points***)

➢ The corresponding points are known precisely in the input and output (**reference**) images.

A Potential Topic for Final Project

Image Registration

- A simple model based on bilinear approximation:

$$\begin{cases} x = c_1 v + c_2 w + c_3 vw + c_4 \\ y = c_5 v + c_6 w + c_7 vw + c_8 \end{cases}$$

Where (v, w) and (x, y) are the coordinates of tie points in the input and reference images.

Image Registration

a b
c d

FIGURE 2.42 Image registration. (a) Reference image. (b) Input (geometrically distorted image). Corresponding tie points are shown as small white squares near the corners. (c) Registered (output) image (note the errors in the border). (d) Difference between (a) and (c), showing more registration errors.

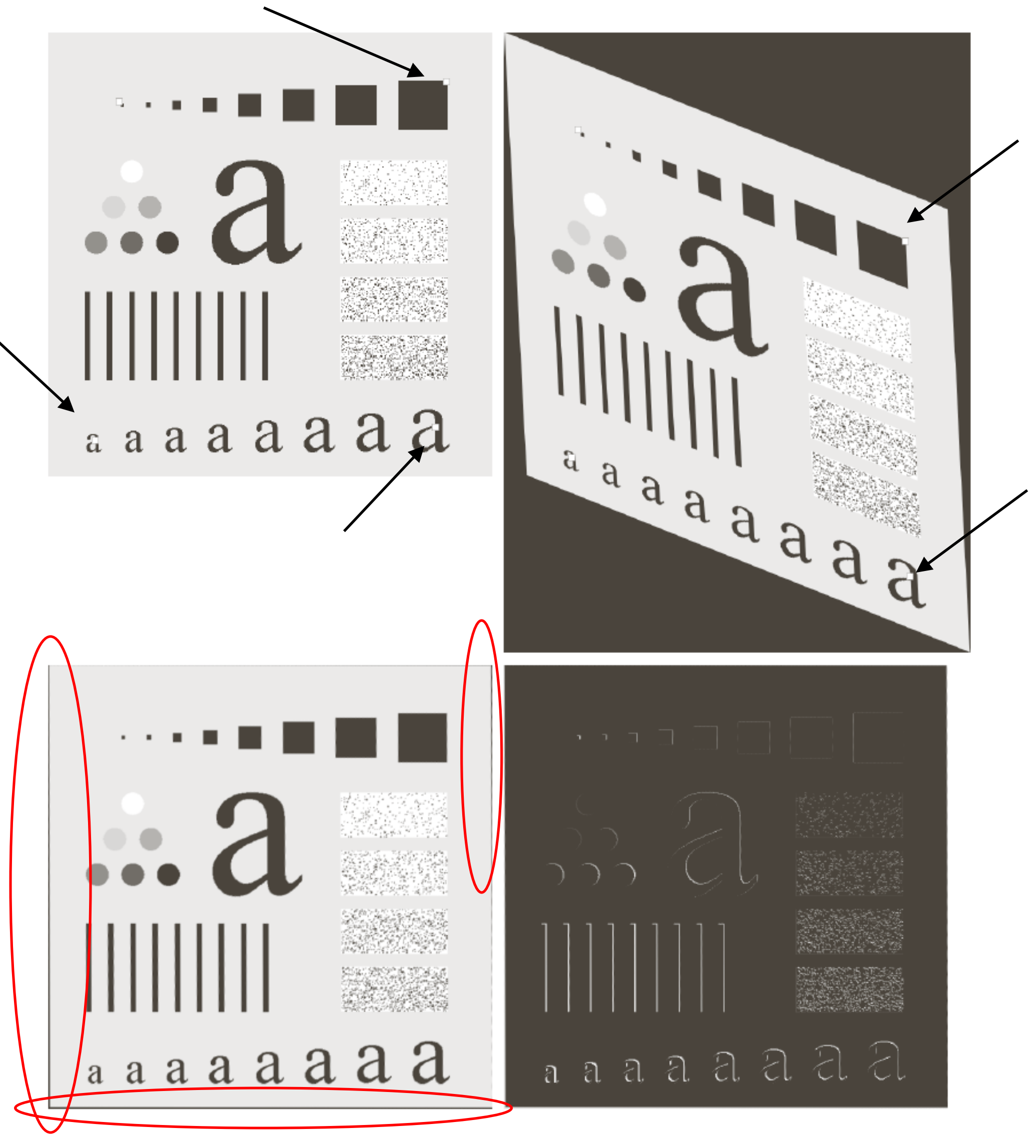


Image Transform

- A particularly important class of 2-D linear transforms, denoted by $T(u, v)$

$$T(u,v)=\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x,y)r(x,y,u,v)$$

where, $f(x,y)$ is the input image,
$r(x,y,u,v)$ is the *forward transformation* kernel,
variables u and v are the transform variables,
$u = 0, 1, 2, \ldots, M-1$ and $v = 0, 1, \ldots, N-1$.

Image Transform

- Given $T(u, v)$, the original image $f(x, y)$ can be recoverd using the inverse transformation of $T(u, v)$.

$$f(x,y) = \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} T(u,v)s(x,y,u,v)$$

where $s(x,y,u,v)$ is the *inverse transformation* kernel,

x = 0, 1, 2, ..., M-1 and y = 0, 1, ..., N-1.

Forward Transform Kernel

$$T(u,v) = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x,y) r(x,y,u,v)$$

The kernel $r(x,y,u,v)$ is said to be SEPERABLE if

$$r(x,y,u,v) = r_1(x,u) r_2(y,v)$$

In addition, the kernel is said to be SYMMETRIC if

$r_1(x,u)$ is functionally equal to $r_2(y,v)$, so that

$$r(x,y,u,v) = r_1(x,u) r_1(y,u)$$

The Kernels for 2-D Fourier Transform

The *forward* kernel

$$r(x, y, u, v) = e^{-j2\pi(ux/M + vy/N)}$$

where, $j = \sqrt{-1}$

The *inverse* kernel

$$s(x, y, u, v) = \frac{1}{MN} e^{j2\pi(ux/M + vy/N)}$$

WRIGHT STATE UNIVERSITY

2-D Fourier Transform

$$T(u,v) = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x,y)\, e^{-j2\pi(ux/M + vy/N)}$$

$$f(x,y) = \frac{1}{MN} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} T(u,v)\, e^{j2\pi(ux/M + vy/N)}$$

Vector and Matrix Operations

- RGB images
- Multispectral images
- Hyperspectral

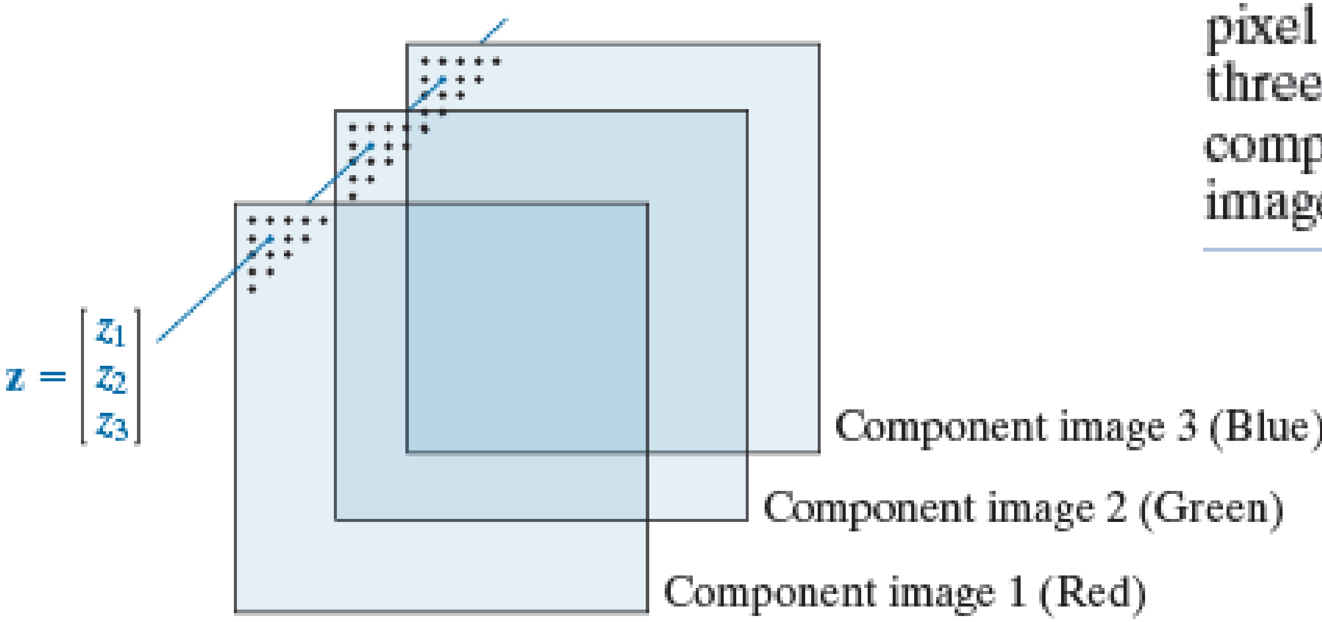
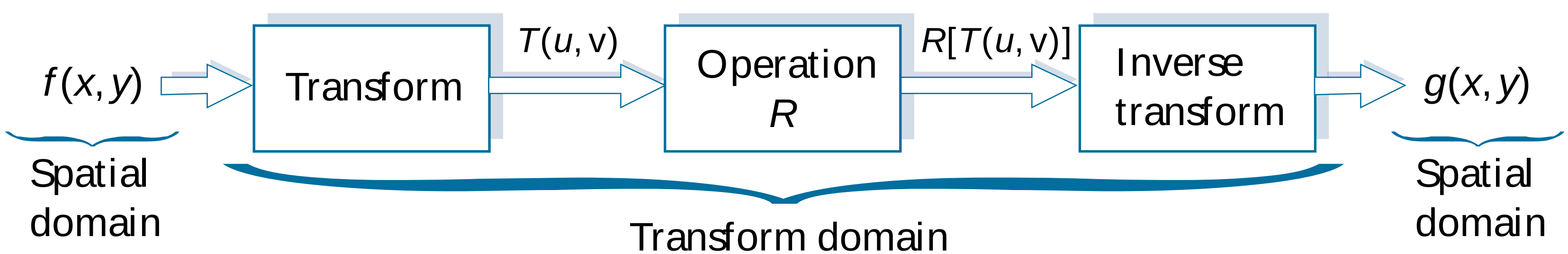


FIGURE 2.43 Forming a vector from corresponding pixel values in three RGB component images.

2.6.7 Image Transforms

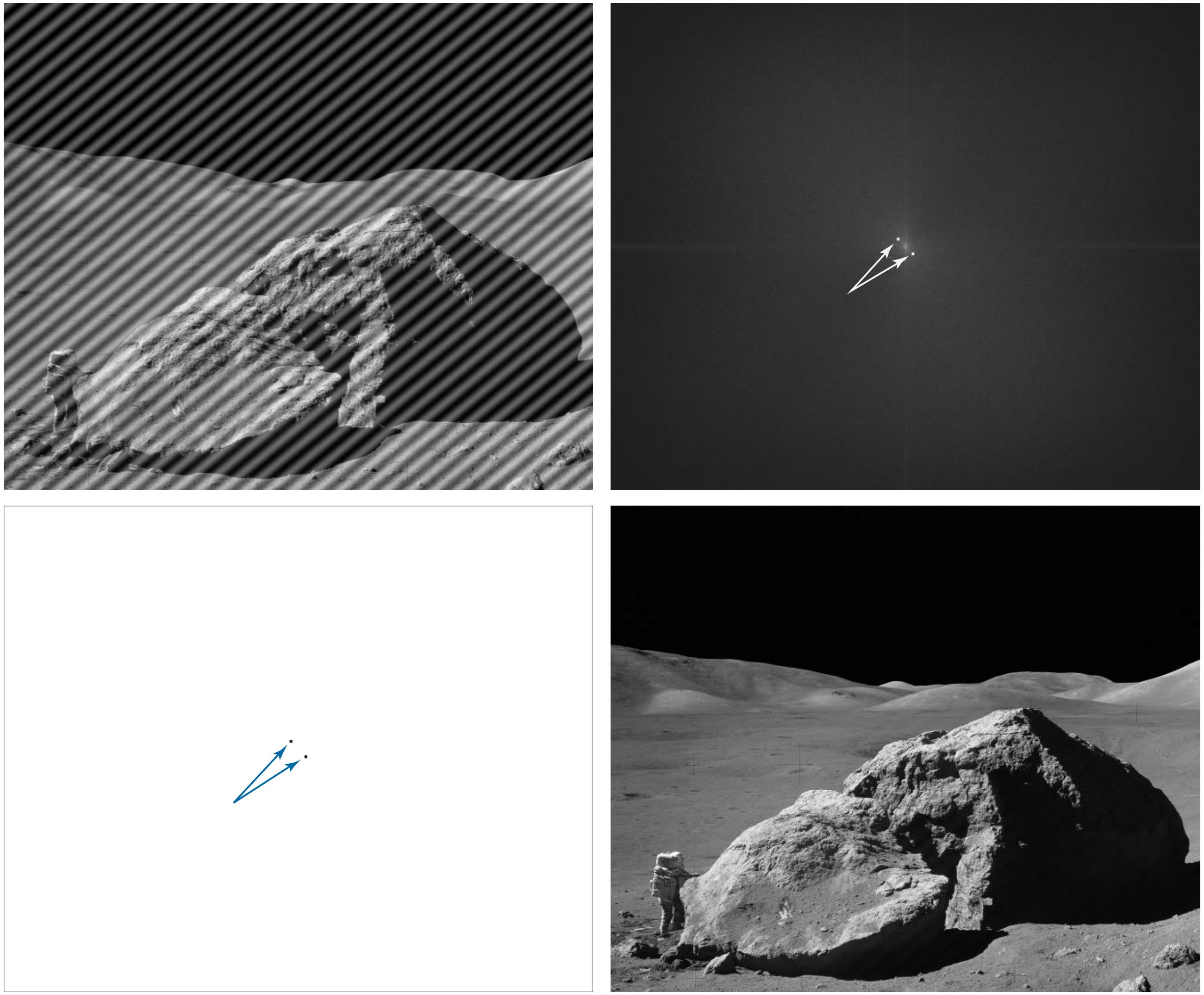
- Image processing tasks are best formulated by
  - Transforming the images
  - Carrying the specified task in a transform domain
  - Applying the inverse transform



More Details in Chapter 4 - Frequency-Domain Filtering

FIGURE 2.44 General approach for working in the linear transform domain.

Example: Image Transforms

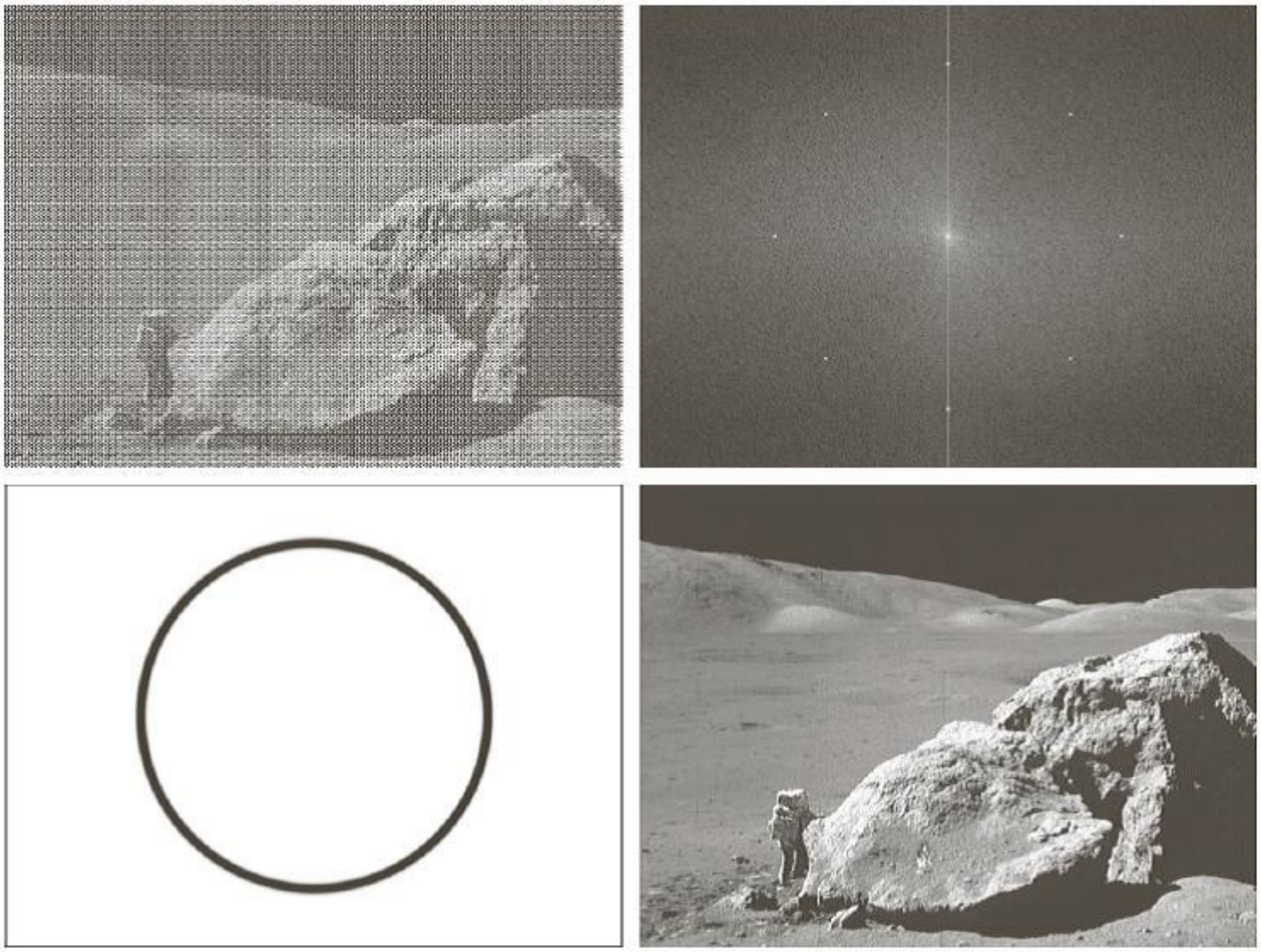


a b
c d

FIGURE 2.45 (a) Image corrupted by sinusoidal interference. (b) Magnitude of the Fourier transform showing the bursts of energy caused by the interference (the bursts were enlarged for display purposes). (c) Mask used to eliminate the energy bursts. (d) Result of computing the inverse of the modified Fourier transform. (Original image courtesy of NASA.)

More Details in Chapter 4 - Frequency-Domain Filtering

Example: Image Transforms



a b
c d

FIGURE 2.40 (a) Image corrupted by sinusoidal interference. (b) Magnitude of the Fourier transform showing the bursts of energy responsible for the interference. (c) Mask used to eliminate the energy bursts. (d) Result of computing the inverse of the modified Fourier transform. (Original image courtesy of NASA.)

3rd Edition

More Details in Chapter 4 - Frequency-Domain Filtering

Probabilistic Methods

Let $z \in [0,\ 1,\ 2,\ \ldots,\ L-1]$, denote the values of all possible intensities in an $M \times N$ digital image. The probability, $p(z)$, of intensity level z occurring in a given image is estimated as

$$p(z) = \frac{h(z)}{MN}, \quad z \in [0,\ 1,\ 2,\ \ldots,\ L-1],$$

where $h(z)$ is the number of times that intensity z occurs in the image.

$$\sum_{z \in R} p(z) = 1$$

Sample Mean : For K measurements, $z_k, k = 1,\ 2,\ \ldots,\ K$,

$$\overline{z} = \frac{1}{K} \sum_{k=1}^{K} z_k$$

Probabilistic Methods

Sample Variance of the intensities is given by

$$\sigma^2 = \frac{1}{K}\sum_{k=1}^{K}(z_k - \overline{z})^2$$

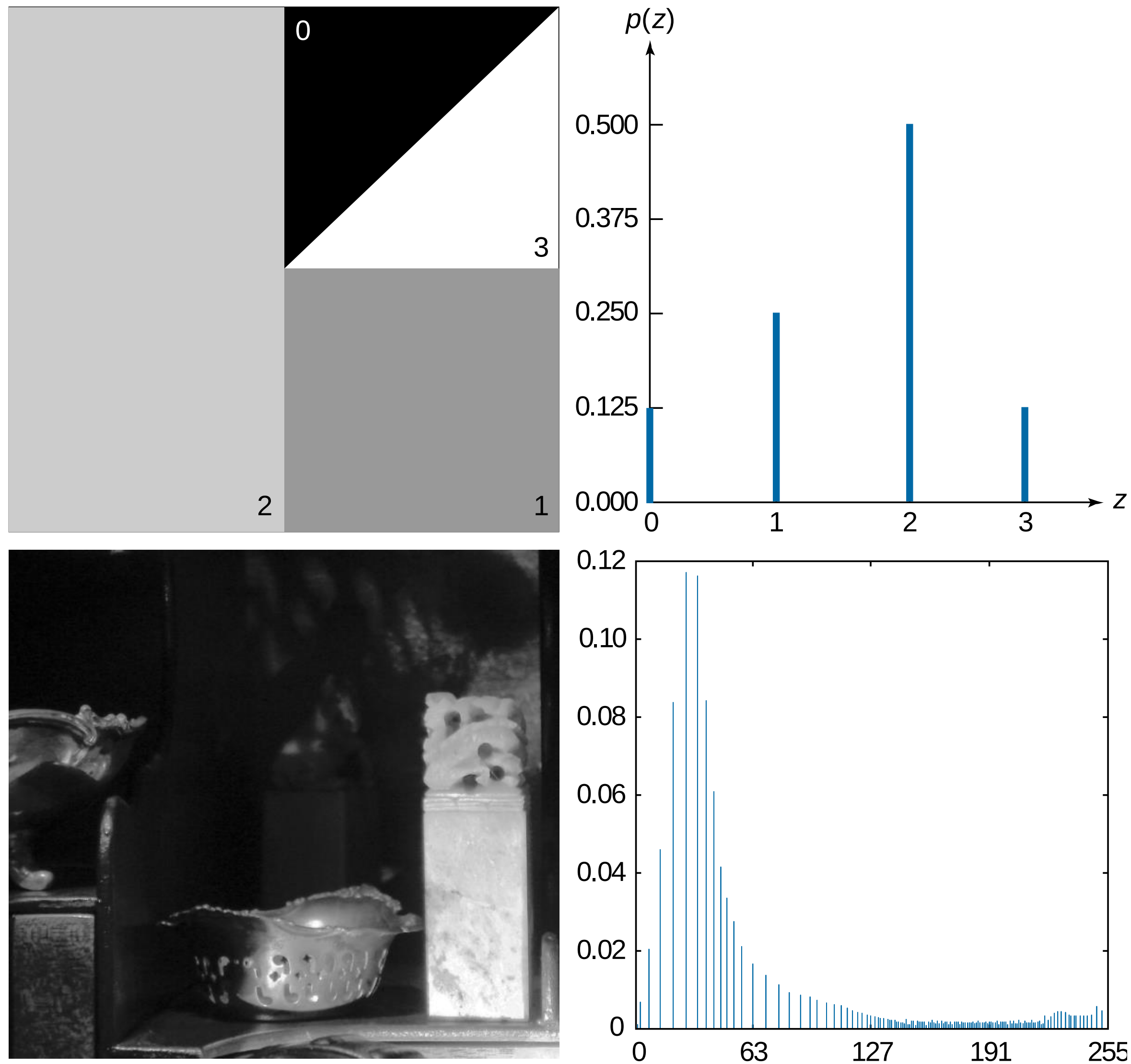
The n^{th} moment of the intensity variable z is

$$\mu_n = \frac{1}{K}\sum_{k=1}^{K}(z_k - \overline{z})^n$$

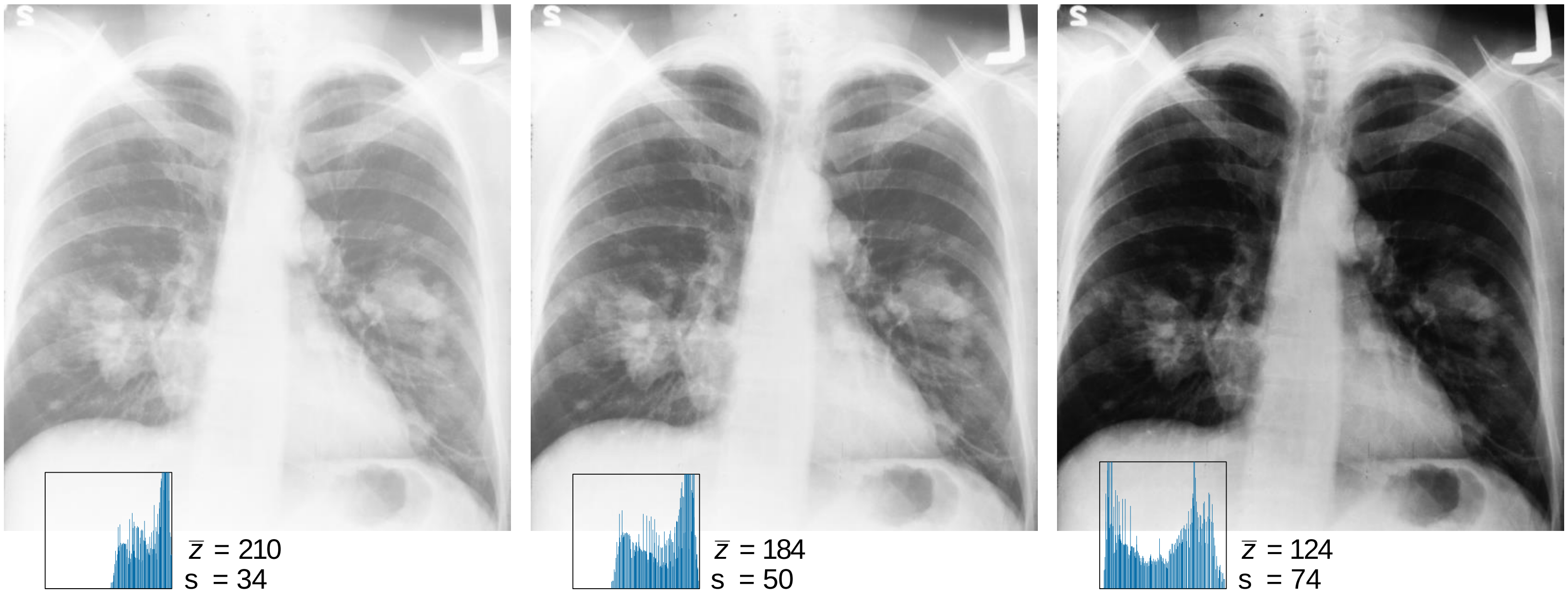
Histogram Examples

a b
c d

FIGURE 2.49 (a) A synthetic image, and (b) its histogram. (c) A natural image, and (d) its histogram.



Contrast, Mean & Std. Deviation



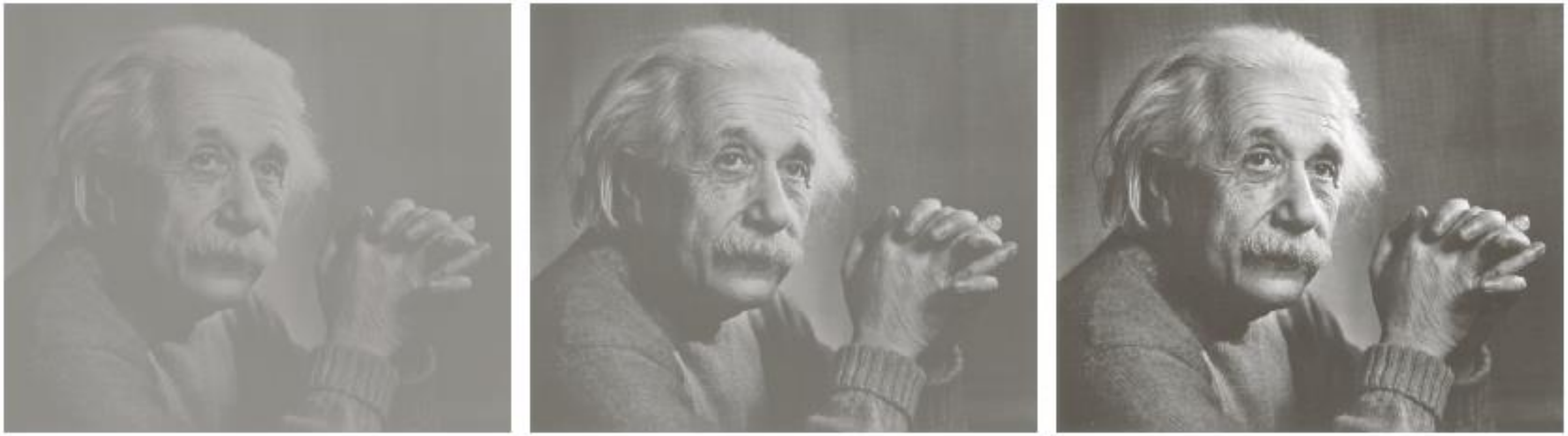
$\sigma = 34$ $\sigma = 50$ $\sigma = 74$

a b c

FIGURE 2.51 Illustration of the mean and standard deviation as functions of image contrast. (a)-(c) Images with low, medium, and high contrast, respectively. (Original image courtesy of the National Cancer Institute.)

More Details in Chapter 3 - Spatial-Domain Filtering

Contrast & Std. Deviation



$\sigma = 14.3$ $\sigma = 31.6$ $\sigma = 49.2$

a b c

FIGURE 2.41 Images exhibiting (a) low contrast, (b) medium contrast, and (c) high contrast.

More Details in Chapter 3 - Spatial-Domain Filtering

3rd Edition

Greyscale from RGB

- Sometimes a single value at each pixel makes processing easier
- This value is usually the intensity or 'grey value'
- We can convert an RGB image to greyscale using
  - Simple average of red, green and blue, i.e.

    $i = (r+g+b)/3$ or
  - Weighted average, i.e.

    $i = a_1 r + a_2 g + a_3 b$ where $a_1 + a_2 + a_3 = 1$

  where, i is the grey value and r, g, and b are the red green and blue values

Average Original Weighted

- Since our eyes are more sensitive to green light one commonly used formula:

$$i = 0.30r + 0.59g + 0.11b$$

Acknowlegements

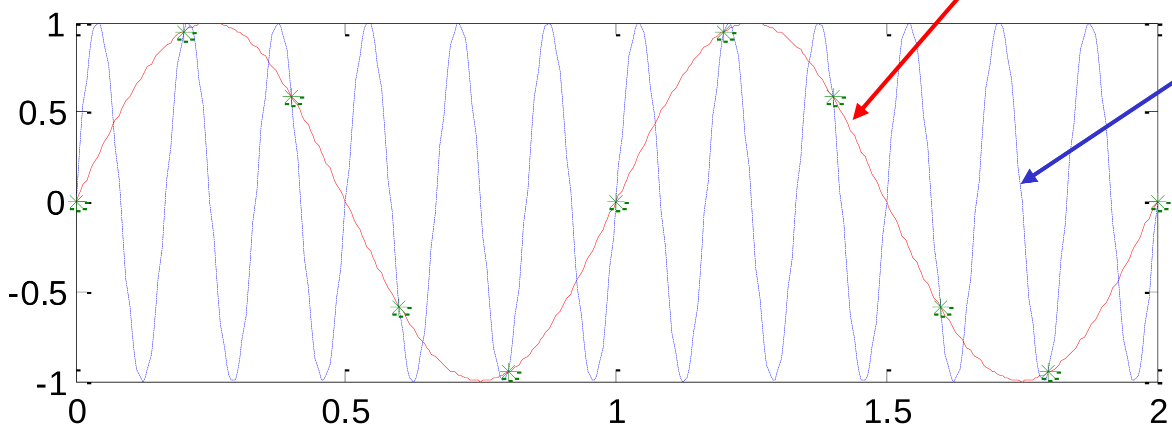
- Slides are primarily based on the figures and images in the Digital Image Processing textbook by Gonzalez and Woods (4th, 3rd and 2nd Editions):

- http://www.imageprocessingplace.com/DIP-3E/dip3e_book_images_downloads.htm

- In addition, slides have been adopted and modified from the following excellent sources:

  - http://www.comp.dit.ie/bmacnamee/gaip.htm
  - http://baggins.nottingham.edu.my/~hsooihock/G52IIP/
  - http://gear.kku.ac.th/~nawapak/178353.html

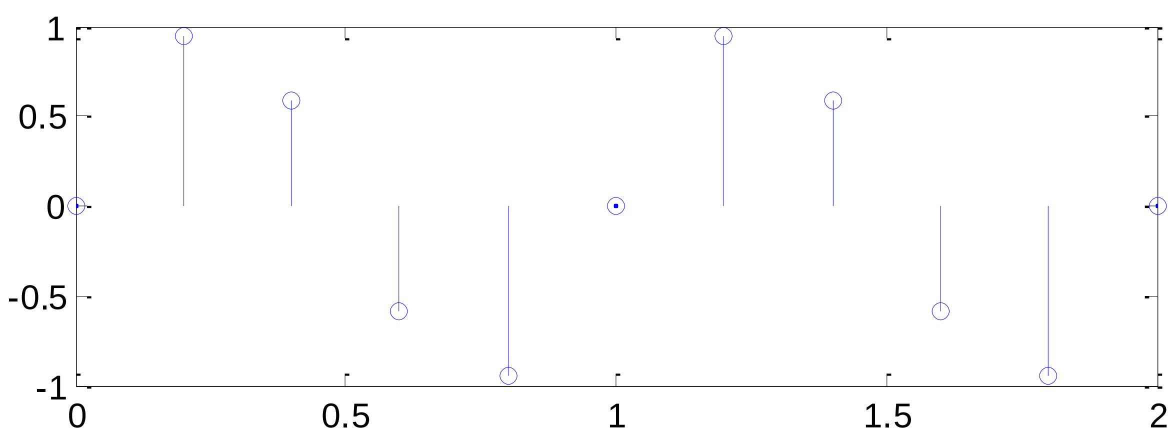
Aliased Frequency



$$x_1(t) = \sin(2\pi t), \qquad f = 1$$

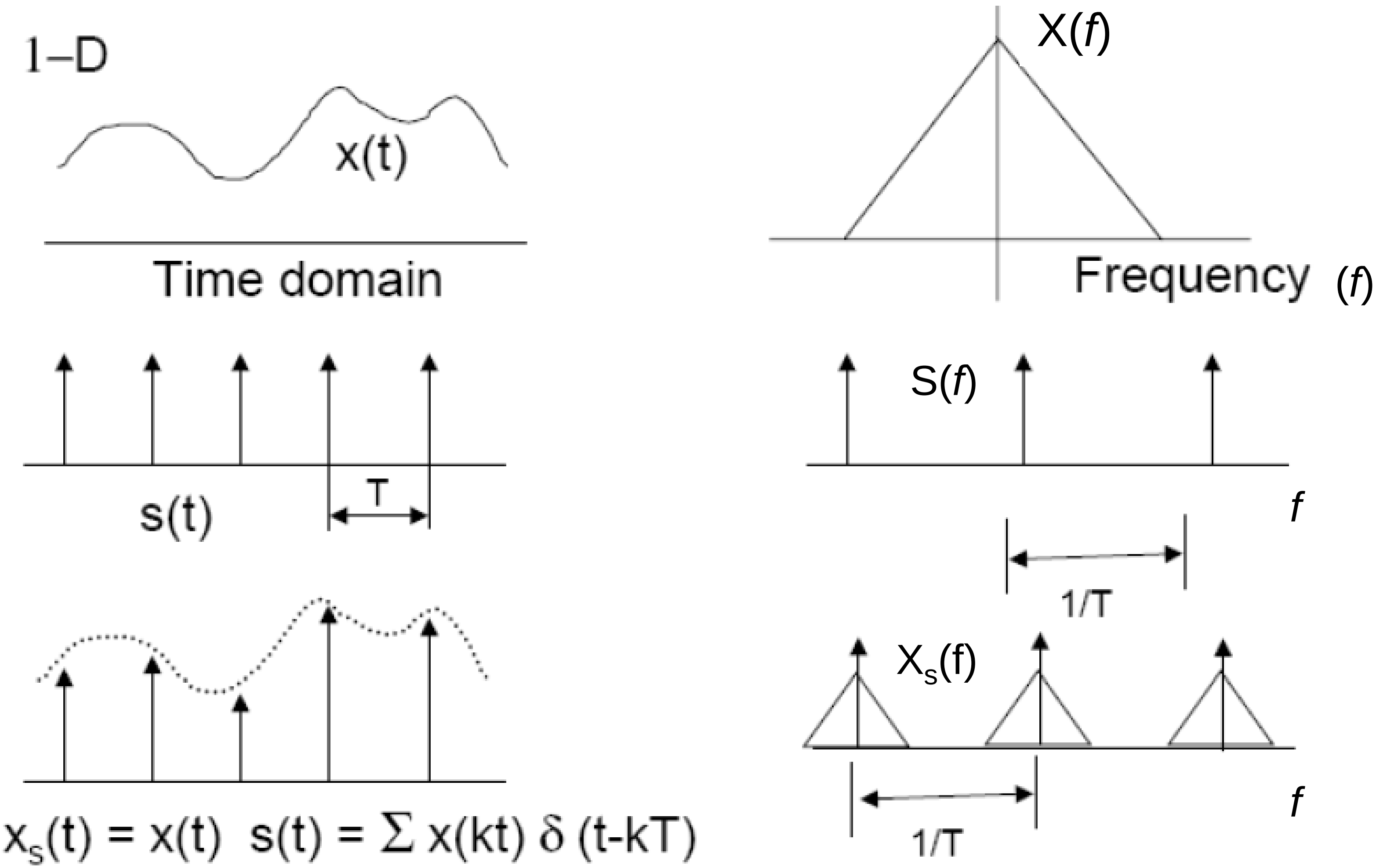
$$x_2(t) = \sin(12\pi t), \qquad f = 6$$

Sampling rate:
5 samples/sec



Two different frequencies but the same results !

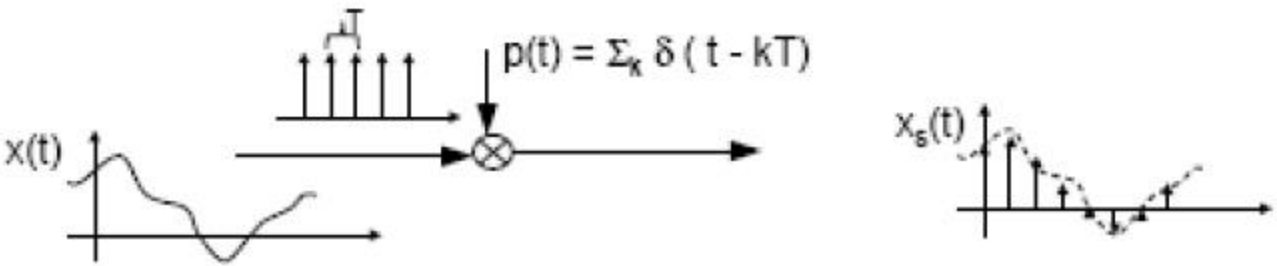
Sampling in 1-Dimension



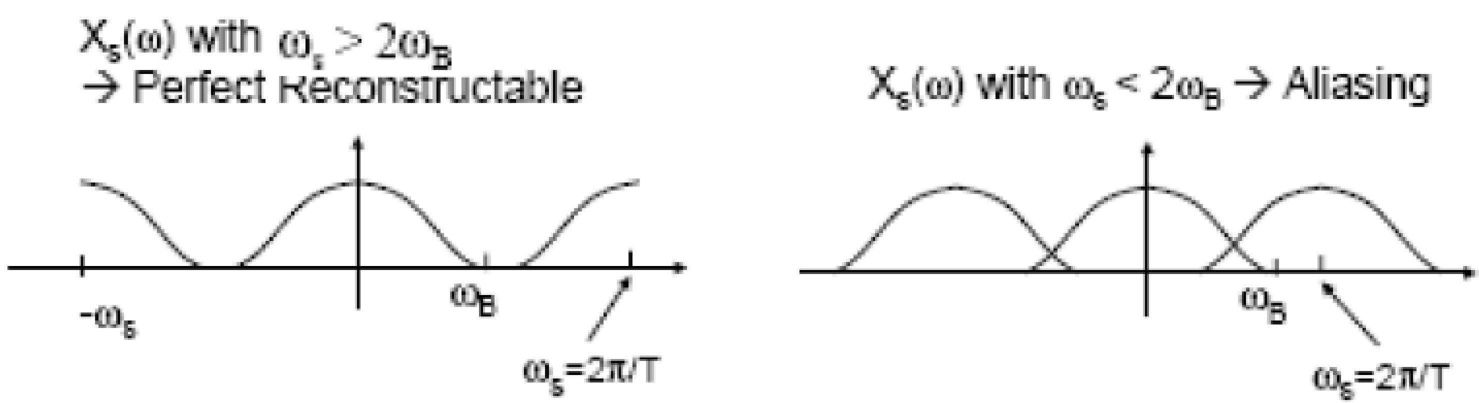
$x_s(t) = x(t)\ \ s(t) = \Sigma\, x(kt)\, \delta\, (t\text{-}kT)$

Sampling in 1-Dimension

- Time domain
  - Multiply continuous-time sig. with periodic impulse train

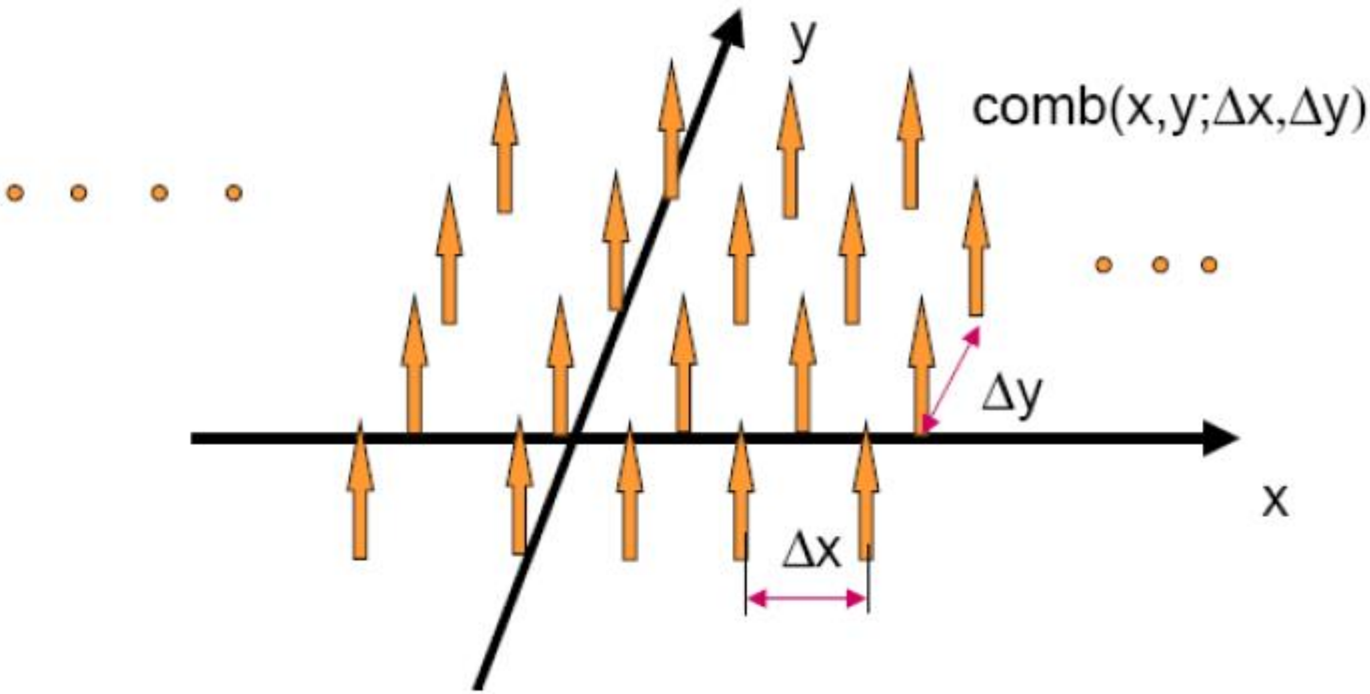


- Sampling below Nyquist rate ($2\omega_B$) cause *Aliasing*



Sampling Images: 2-D Comb Function

$$\mathrm{Comb}(x,y;\Delta x,\Delta y) \cong \sum_{m=-\infty}^{\infty}\sum_{n=-\infty}^{\infty}\delta(x-m\Delta x, y-n\Delta y)$$



Sampling Images

$$f_s(x,y) = f(x,y)\ \mathrm{comb}\ (x,y\ ;\ \Delta x, \Delta y)$$

$$= \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} f(m\Delta x, n\Delta y)\ \delta(x - m\Delta x, y - n\Delta y)$$

$$\mathrm{comb}(x,y;\Delta x,\Delta y) \xleftrightarrow{\Im} \mathrm{COMB}\ (u,v) =$$

$$\frac{1}{\Delta x \Delta y}\mathrm{comb}(u,v\ ;\frac{1}{\Delta x},\frac{1}{\Delta y})$$

Sampled 2-D Spectrum

$$F_s(u,v) = F(u,v) * \mathrm{COMB}(u,v)$$

$$= \frac{1}{\Delta x \Delta y} \sum_{k,l=-\infty}^{\infty}\sum F(u,v) * \delta\left(u - \frac{k}{\Delta x}, v - \frac{l}{\Delta y}\right)$$

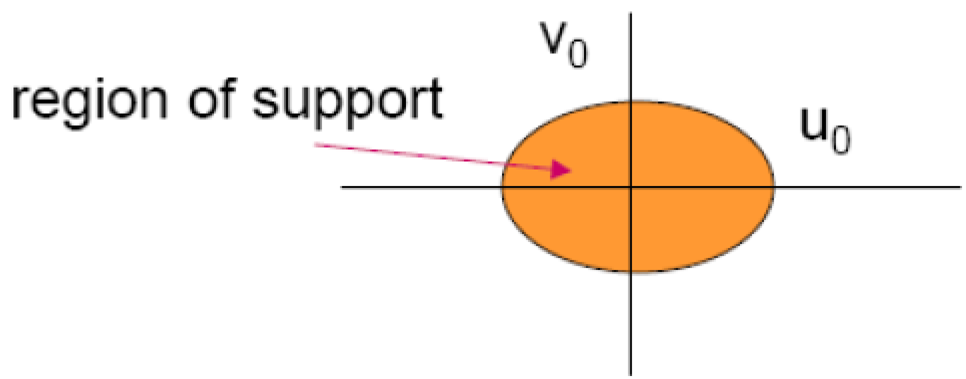
$$= \frac{1}{\Delta x \Delta y} \sum_{k,l=-\infty}^{\infty}\sum F\left(u - \frac{k}{\Delta x}, v - \frac{l}{\Delta y}\right)$$

Need: Band-limited Image

A function f(x,y) is said to be band limited if the Fourier transform

$F(u,v) = 0$ for $|u| > u_0, \; |v| > v_0$

u_0, v_0 → Band width of the image in the x- and y- directions



2-D Sampling Theorem

A band limited image f(x,y) with F(u,v) as its Fourier transform; and F(u,v) = 0 $|u| > u_0$ $|v| > v_0$; and sampled uniformly on a rectangular grid with spacing Δx and Δy, can be recovered without error from the sample values $f(m\,\Delta x, n\,\Delta y)$ provided the sampling rate is greater than the nyquist rate.

i.e $1/\Delta x = u_s > 2u_0$, $1/\Delta y = v_s > 2v_0$

The reconstructed image is given by the interpolation formula:

$$f(x,y) = \sum_{m,n=-\infty}^{\infty}\sum f(m\,\Delta x, n\,\Delta y)\ \frac{\sin(x u_s - m)\pi}{(x u_s - m)\pi}\ \frac{\sin(y v_s - n)\pi}{(y v_s - n)\pi}$$

Aliasing in 2-D Images

Note: If u_s and v_s are below the Nyquist rate, the periodic replications will overlap, resulting in a distorted spectrum.

This overlapping of successive periods of the spectrum causes the foldover frequencies in the original image to appear as frequencies below $u_s/2$, $v_s/2$ in the sampled image. This is called aliasing.

Aliasing effect (Moiré) for an image

Aliasing effect (Moiré) for an image

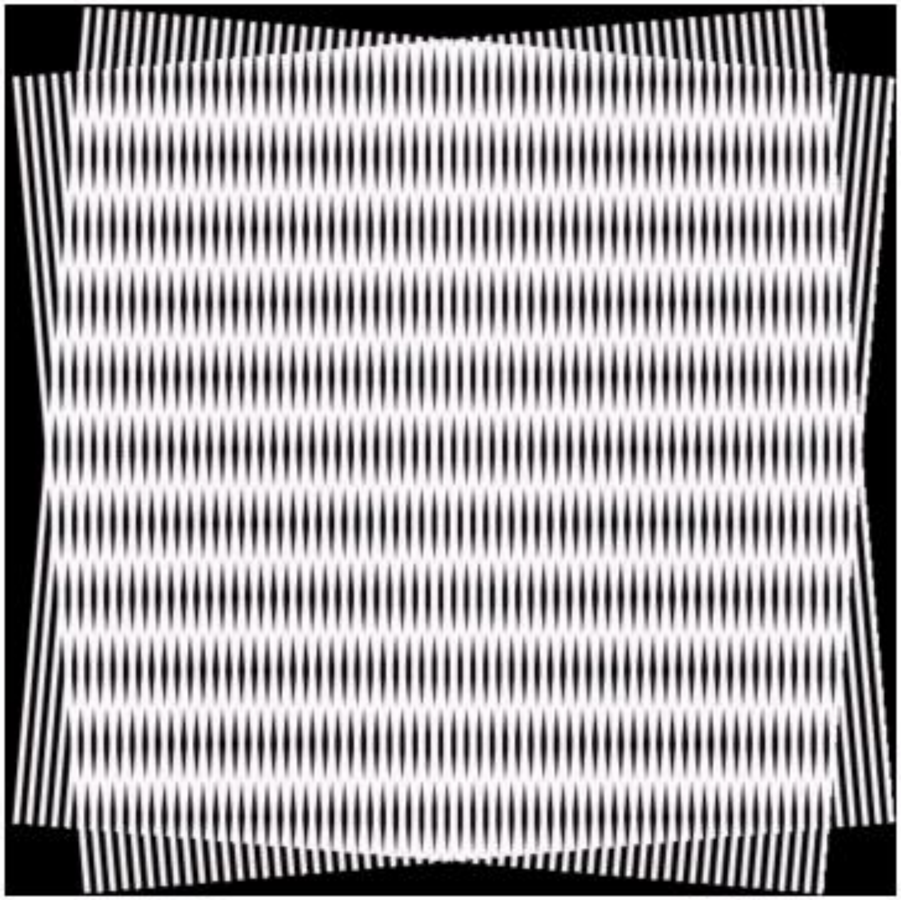
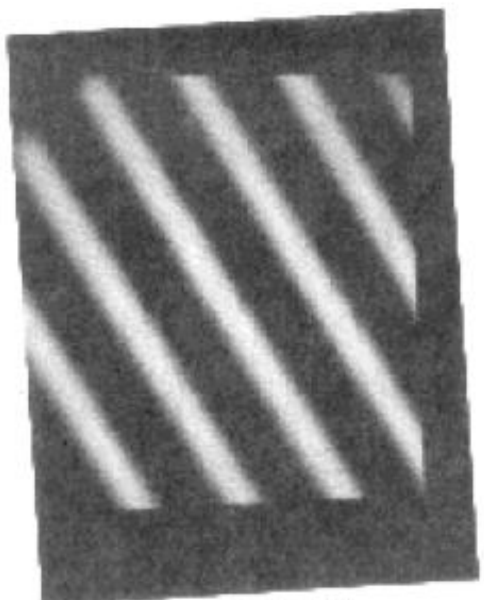


FIGURE 2.24 Illustration of the Moiré pattern effect.

Aliasing effect (Moiré) for an image

Examples



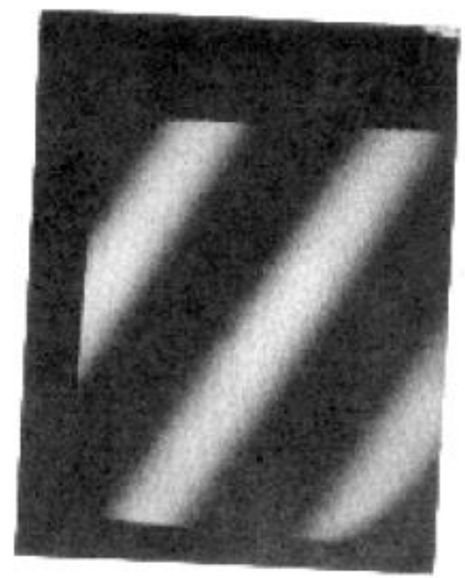


Fig. 7 The reconstructed picture from the samples of the picture in Fig. 5. Note the change in frequency and orientation.

Original and the reconstructed image from samples.