CS4046D COMPUTER VISION

Instructor

Lijiya A

Assistant Professor

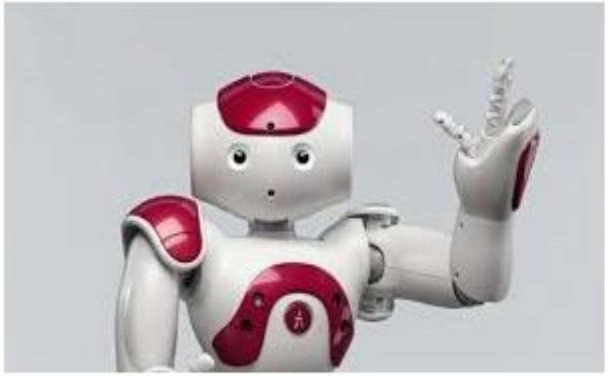
CSED, NIT Calicut

Objective of Computer Vision

- Enabling Computers/Machines to see and extract/interpret information from the visuals.

Self Driving Cars from Google

Unmanned Aerial Vehicles



Humanoid Robots

AI and Computer Vision

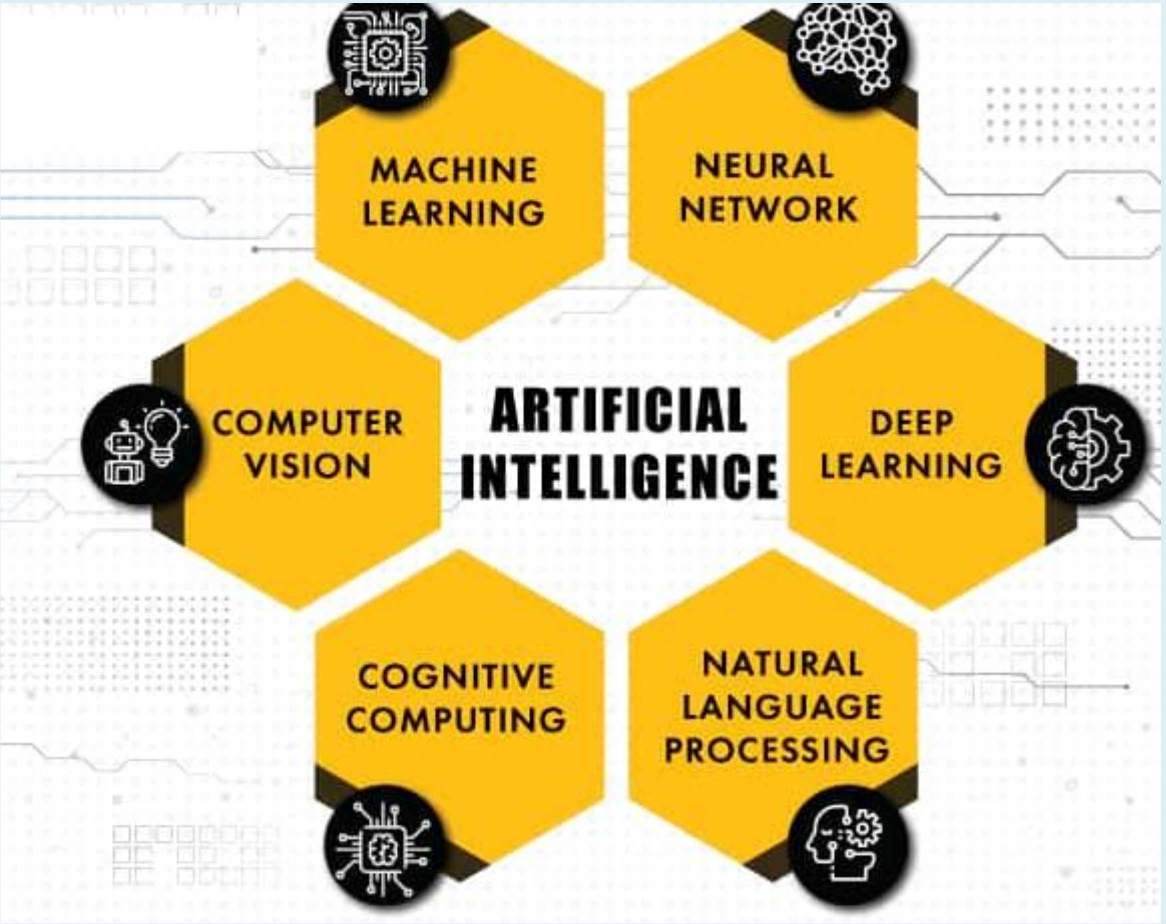


Image Courtesy

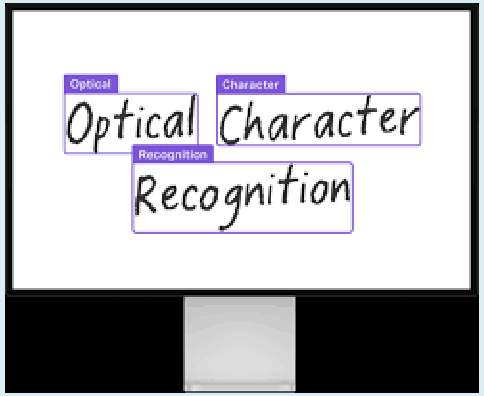
- https://www.innoplexus.com/blog/how-artificial-intelligence-works/

Image Processing Vs Computer Vision

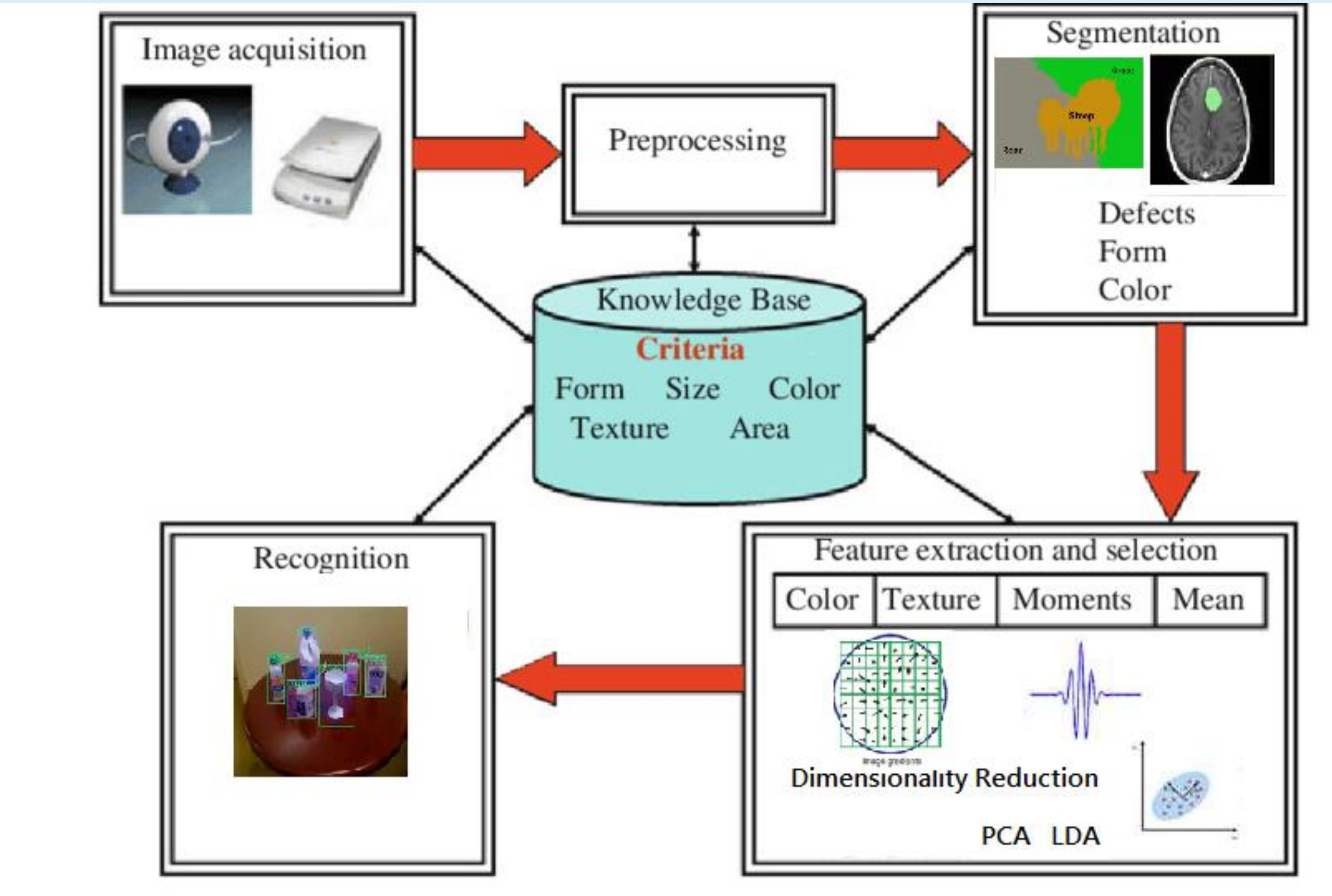
Image Processing	Computer Vision
Input and output are Images	Input may be an image or video and output may be task specific knowledge inferred from the visual.
Focus is on processing the image	Focus is on understanding the scene.
Image Enhancement, segmentation, compression, Watermarking etc.	Verification, detection, labeling, prediction, measurement etc.

Applications of Computer Vision

- Optical character recognition (OCR)
- Machine inspection
- Retail (e.g. automated checkouts)
- 3D model building (photogrammetry)
- Medical imaging
- Automotive safety
- Motion capture
- Image Inpainting
- Surveillance
- Fingerprint recognition and biometrics



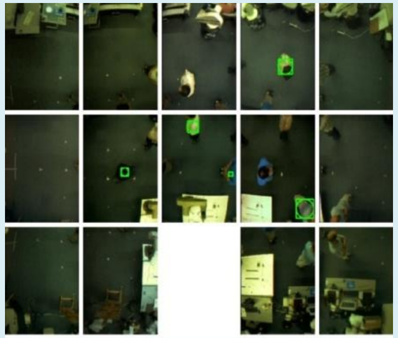
Different Stages in a CV system



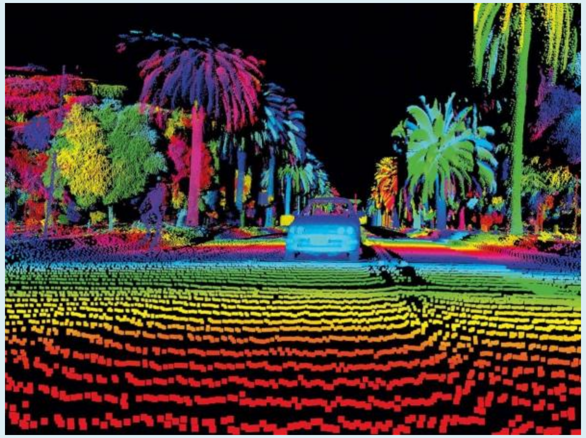
Data Acquisition :Computer vision is more than pictures

Images

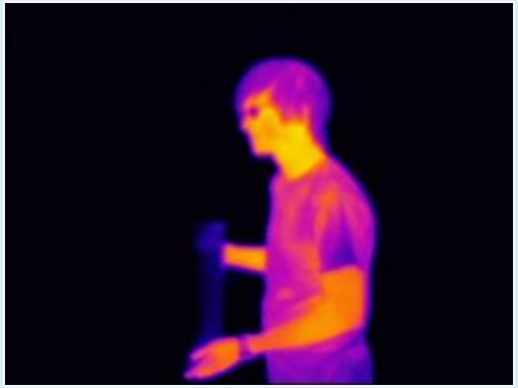
Video



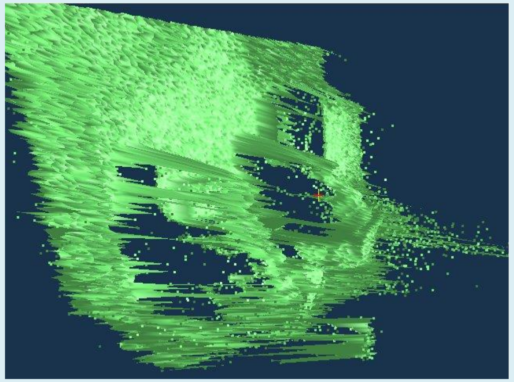
Camera array



LiDaR

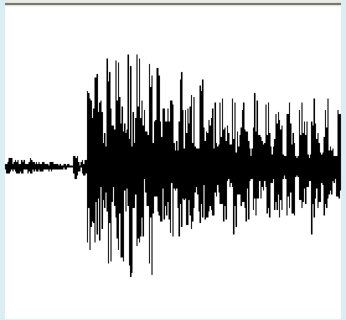


Thermal Infrared



3d range scans
(flash lidar)

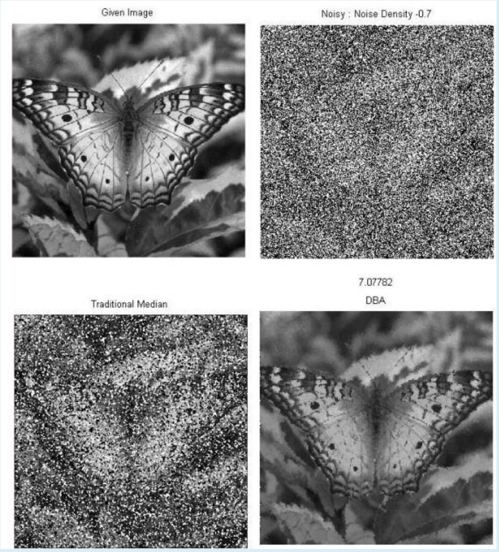
3d range scan
(laser scanner)



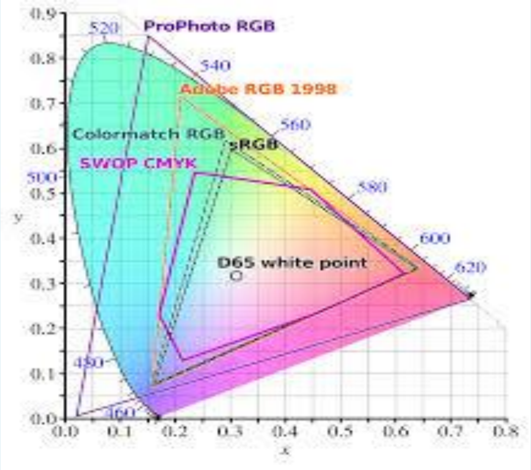
Audio

Preprocessing

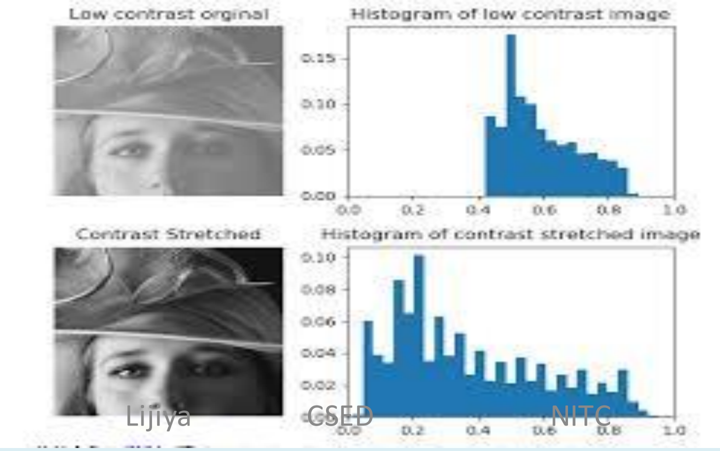
Noise Removal



Color Space Conversion



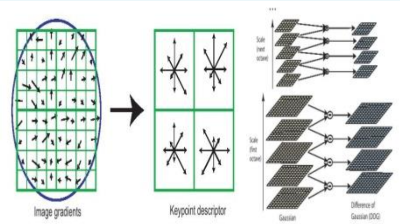
Contrast Strectching



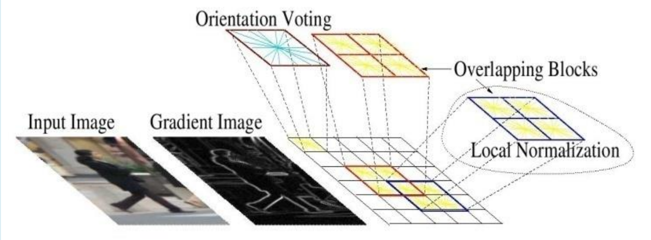
Feature Extraction

Feature Descriptor Should be:

- **Complete**
- **Invariant to affine transformations**
- **Compact**
- **Robust**
- **Low Computational Cost**



SIFT



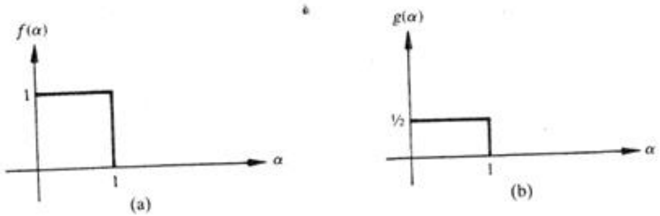
HoG

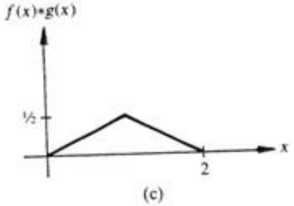
Feature	Description
Energy (ENR)	$ENR = \sum_{i=0}^{n-1}\sum_{j=0}^{n-1} G(i,j)^2$
Entropy (ENT)	$ENT = -\sum_{i=0}^{n-1}\sum_{j=0}^{n-1} G(i,j)\, ln[G(i,j)]$
Contrast (CON)	$CON = \sum_{i=0}^{n-1}\sum_{j=0}^{n-1} G(i,j)(i-j)^2$
Difference entropy (DENT)	$DENT = -\sum_{i=0}^{n-1} G_{x-y}(i)\, ln[G_{x-y}(i)]$
Difference variance (DVAR)	$DVAR = -\sum_{i=0}^{n-1} G_{x-y}(i)(i-DENT)^2$
Maximum probability (MAXP)	$MAXP = MAX_{i,j}G(i,j)$
Sum entropy (SENT)	$SENT = -\sum_{i=2}^{2n} G_{x+y}(i)\, ln[G_{x+y}(i)]$
Sum average (SVAR)	$SVAR = \sum_{i=0}^{n-1} iG_{x+y}(i)$
Homogeneity (HOM)	$HOM = \sum_{i=0}^{n-1}\sum_{j=0}^{n-1} G(i,j)/(1+(i-j)^2$
Correlation (COR)	$COR = \sum_{i=0}^{n-1}\sum_{j=0}^{n-1} \frac{G(i,j)[(i-\mu_x)(j-\mu_y)}{\sigma_x\sigma_y}$

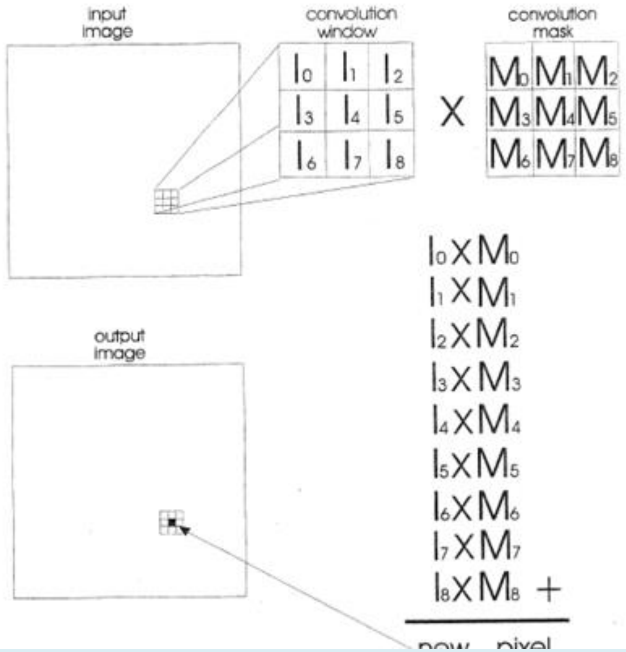
GLCM

Convolution

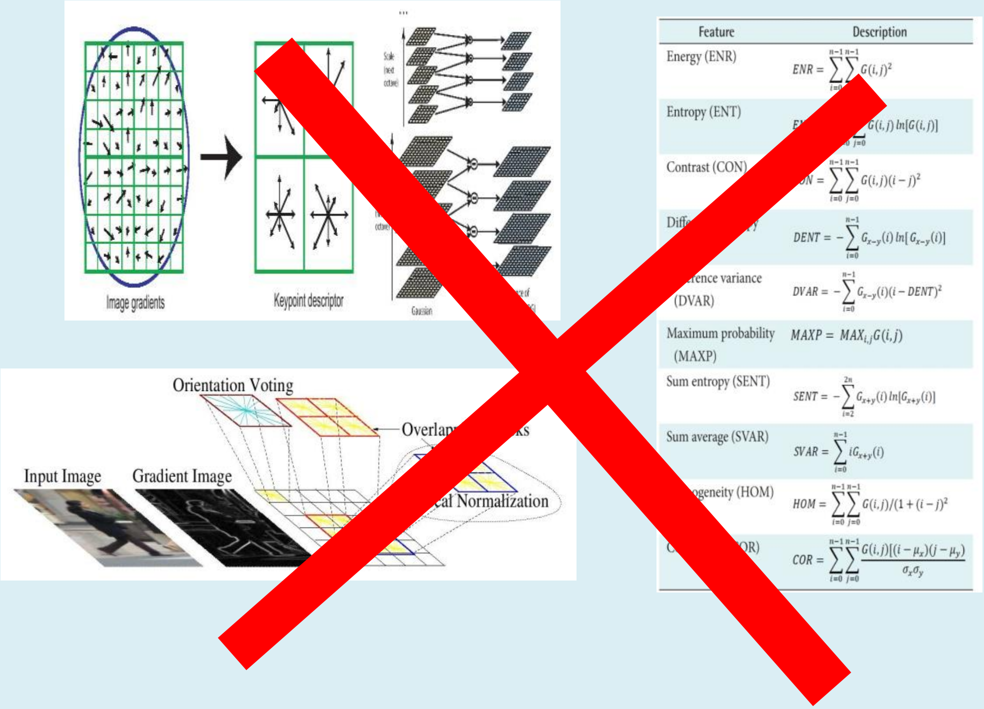
$$f(x) * g(x) = \int_{-\infty}^{\infty} f(a)g(x-a)da$$







The goal of Unsupervised Feature Learning



Unlabeled images

↓

Learning algorithm

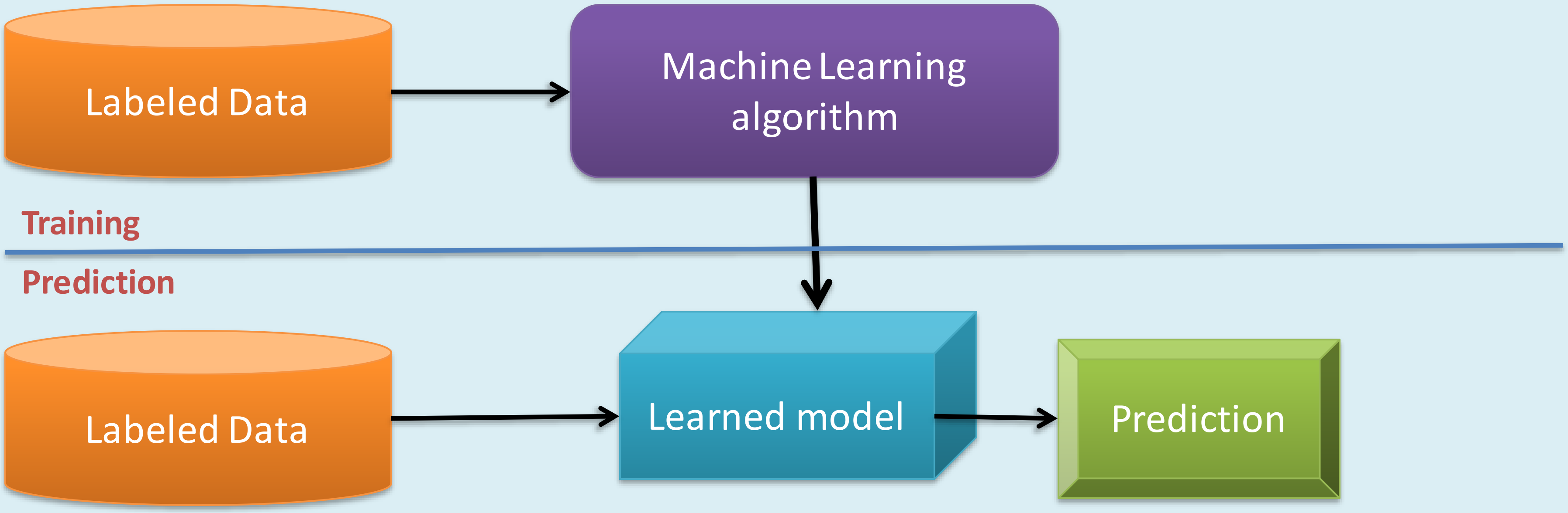
↓

Feature representation

Can we automatically learn good feature representations?

Machine Learning Basics

Machine learning is a field of computer science that gives computers the ability to **learn new concepts mainly from the given set of examples without being explicitly programmed**



Methods that can learn from and make predictions on data

Types of Learning

Supervised: Learning with a **labeled training** set
Example: Object detection and classification with already labeled images as training set.
ANN, Bayes classifier, KNN, SVM, Decision trees.

Unsupervised: Discover **patterns** in **unlabeled** data
Clustering algorithms

Reinforcement learning: learn to **act** based on **feedback/reward**
Q learning

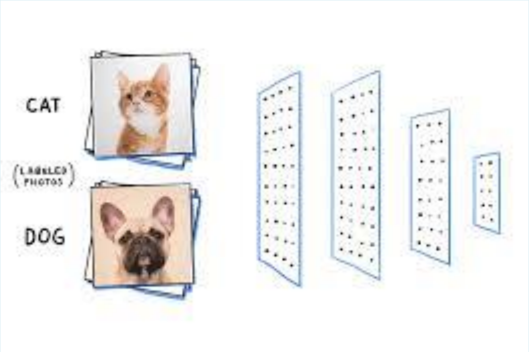


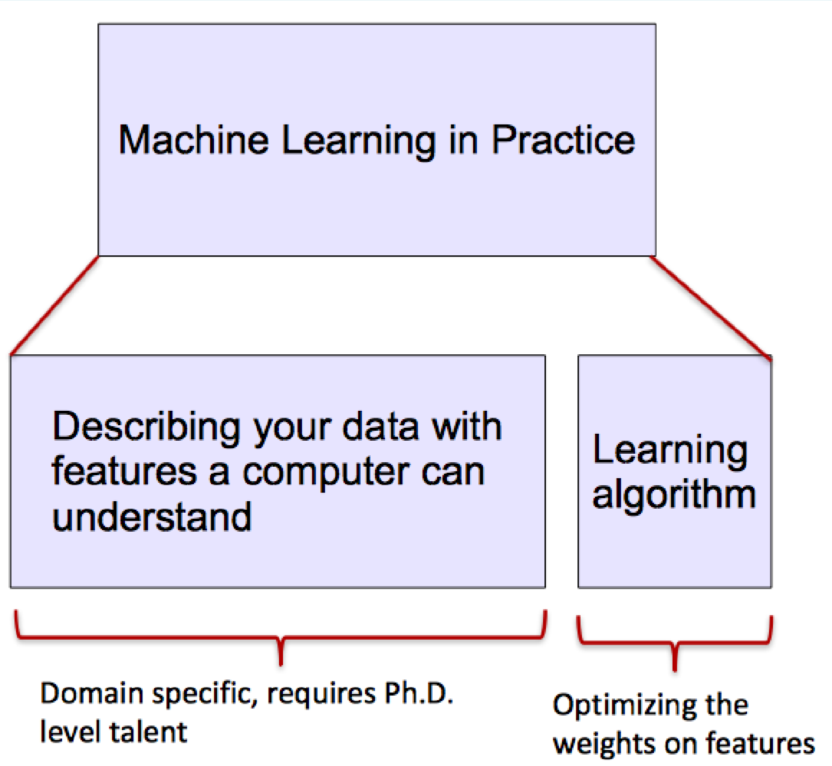
Image Classification

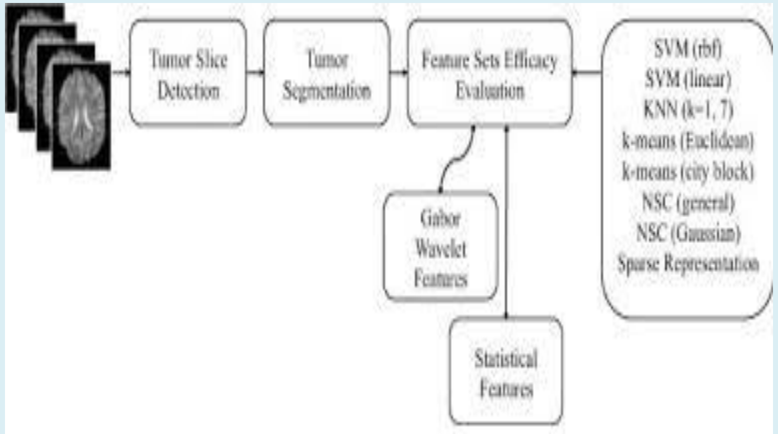
Image segmentation using clustering

[Smart Traffic Signals in India using Deep Reinforcement Learning and Advanced Computer Vision]

ML vs. Deep Learning

Most machine learning methods work well because of **human-designed representations** and **input features (hand crafted)**

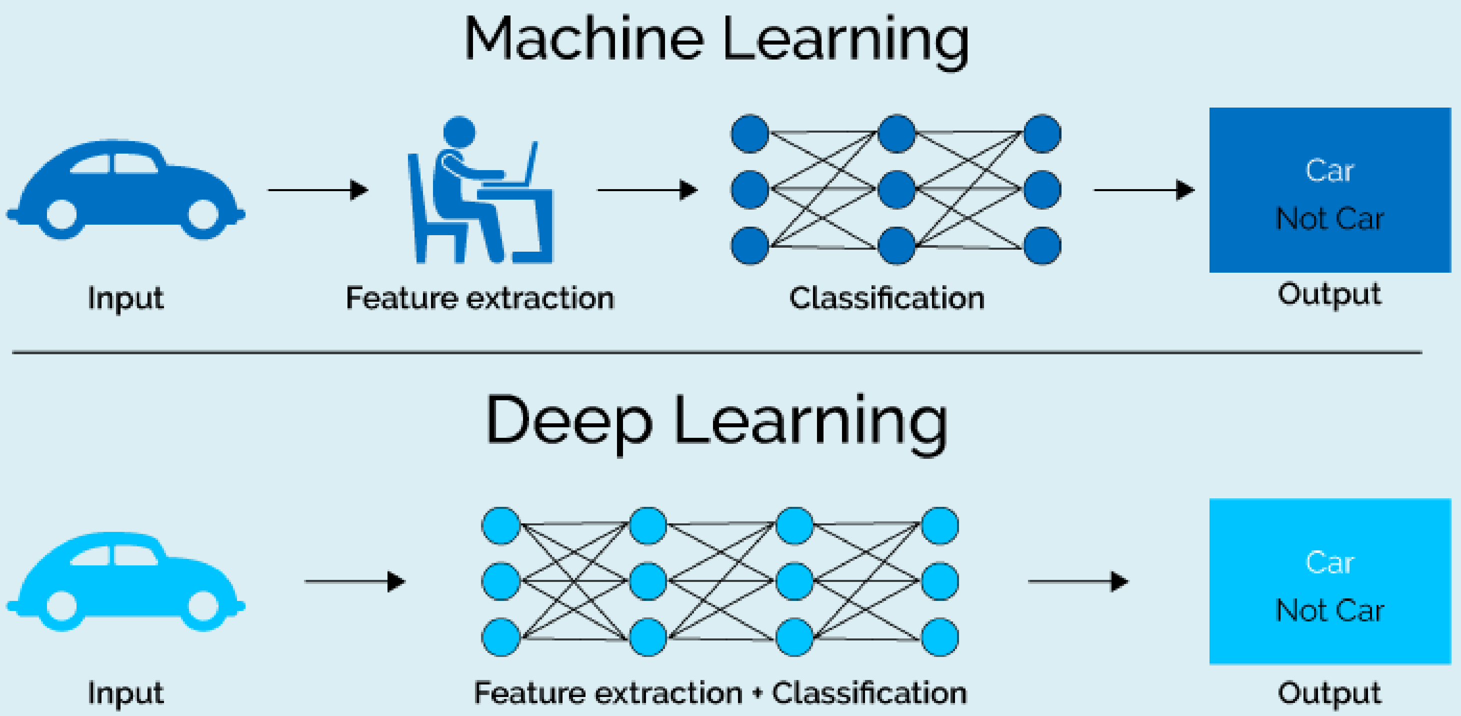




A tumor detection system using ML

What is Deep Learning (DL) ?

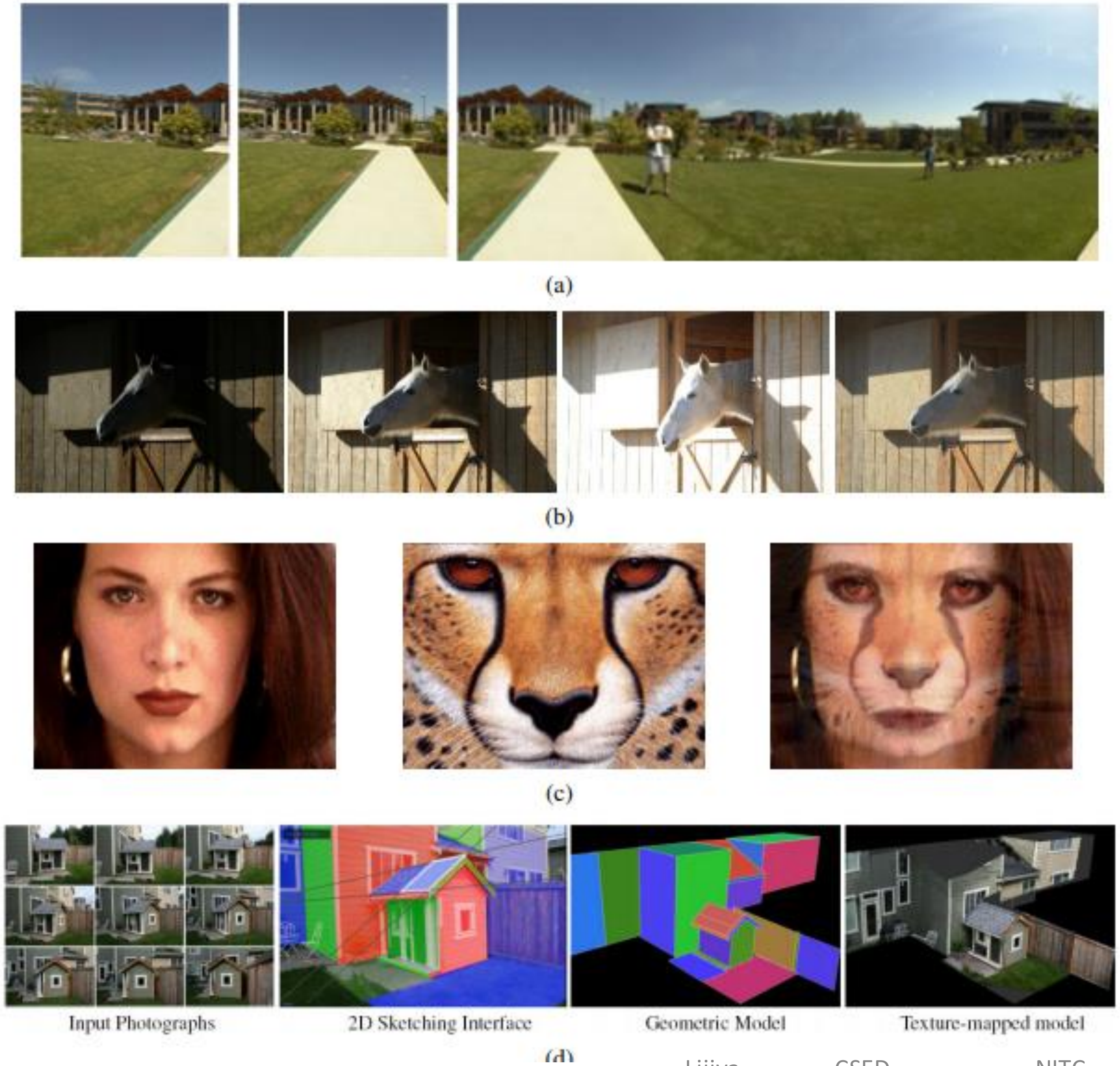
A subfield of machine learning capable of learning **representations** of data.
Exceptionally effective and independent at **learning patterns**.
Deep learning algorithms attempt to learn (multiple levels of) representation by using a **hierarchy of multiple layers.**
If you provide the system **tons of information**, it begins to understand it and respond in useful ways.



https://www.xenonstack.com/blog/static/public/uploads/media/machine-learning-vs-deep-learning.png

Computer Vision

- **Stitching**: turning overlapping photos into a single seamlessly stitched panorama
- **Exposure bracketing:** merging multiple exposures taken under challenging lighting conditions (strong sunlight and shadows) into a single perfectly exposed image
- **Morphing**: turning a picture of one of your friends into another, using a seamless morph transition
- **3D modeling:** converting one or more snapshots into a 3D model of the object or person you are photographing
- **Video match move and stabilization:** inserting 2D pictures or 3D models into your videos by automatically tracking nearby reference points or using motion estimates to remove shake from your videos
- **Photo-based walkthroughs:** navigating a large collection of photographs, such as the interior of your house, by flying between different photos in 3D
- **Face detection:** for improved camera focusing as well as more relevant image searching
- **Visual authentication:** automatically logging family members onto your home computer as they sit down in front of the webcam



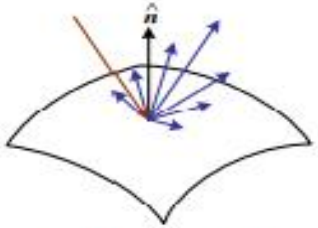
Some consumer applications of computer vision:

(a) image stitching: merging different views (Szeliski and Shum 1997) c 1997 ACM;

(b) exposure bracketing: merging different exposures;

(c) morphing: blending between two photographs (Gomes, Darsa, Costa et al. 1999) c 1999 Morgan Kaufmann;

(d) turning a collection of photographs into a 3D model



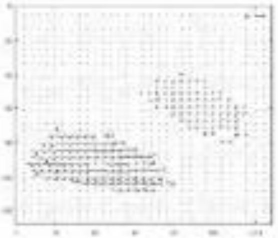
2. Image Formation

3. Image Processing

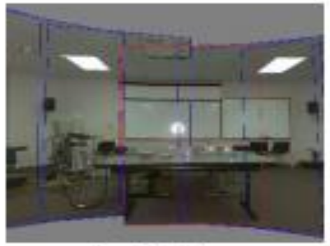
4. Features

5. Segmentation

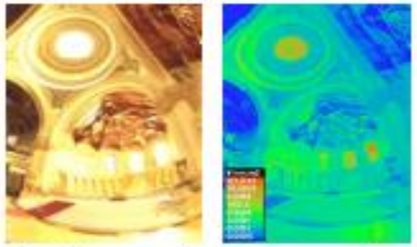
6-7. Structure from Motion



8. Motion



9. Stitching



10. Computational Photography

11. Stereo

12. 3D Shape

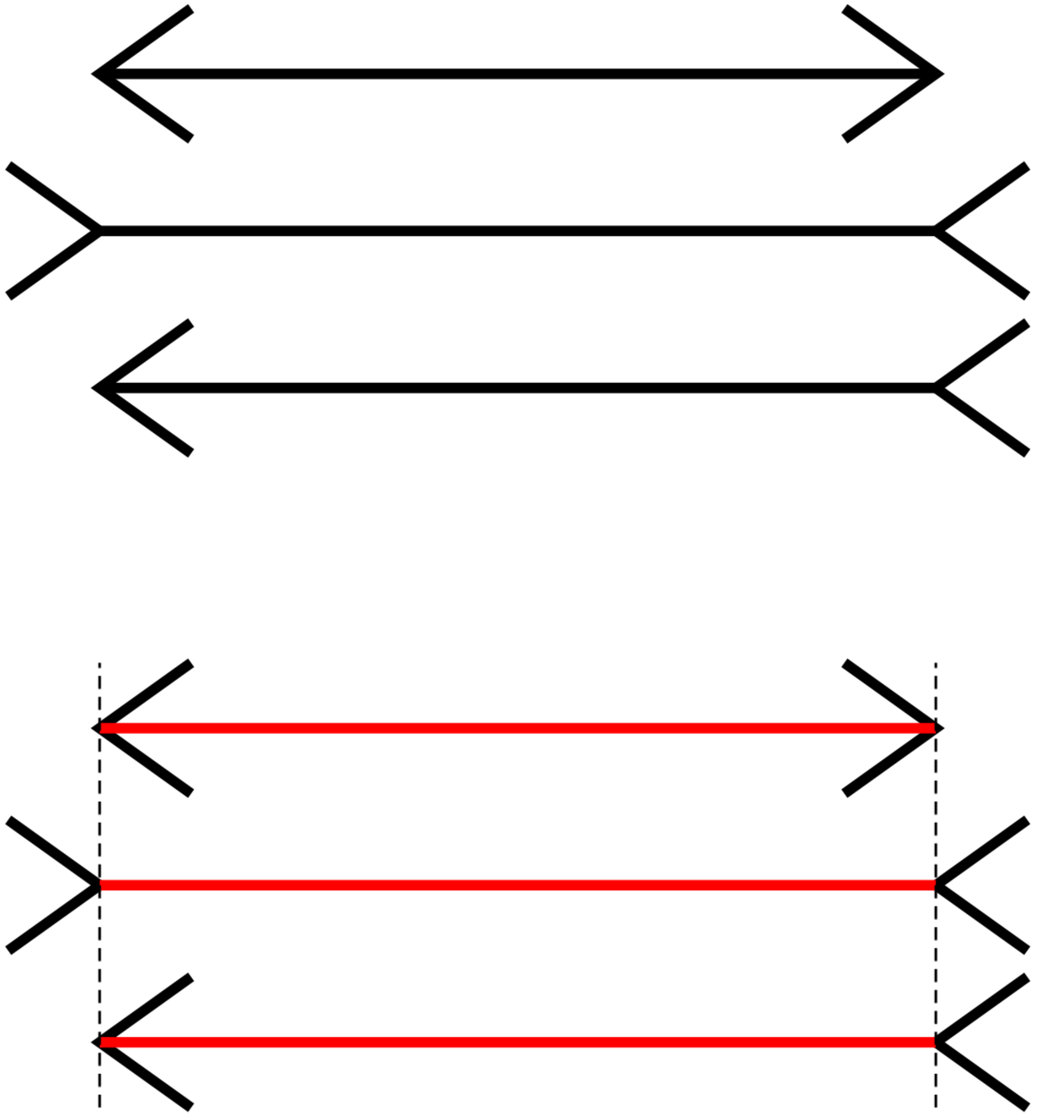
13. Image-based Rendering

14. Recognition

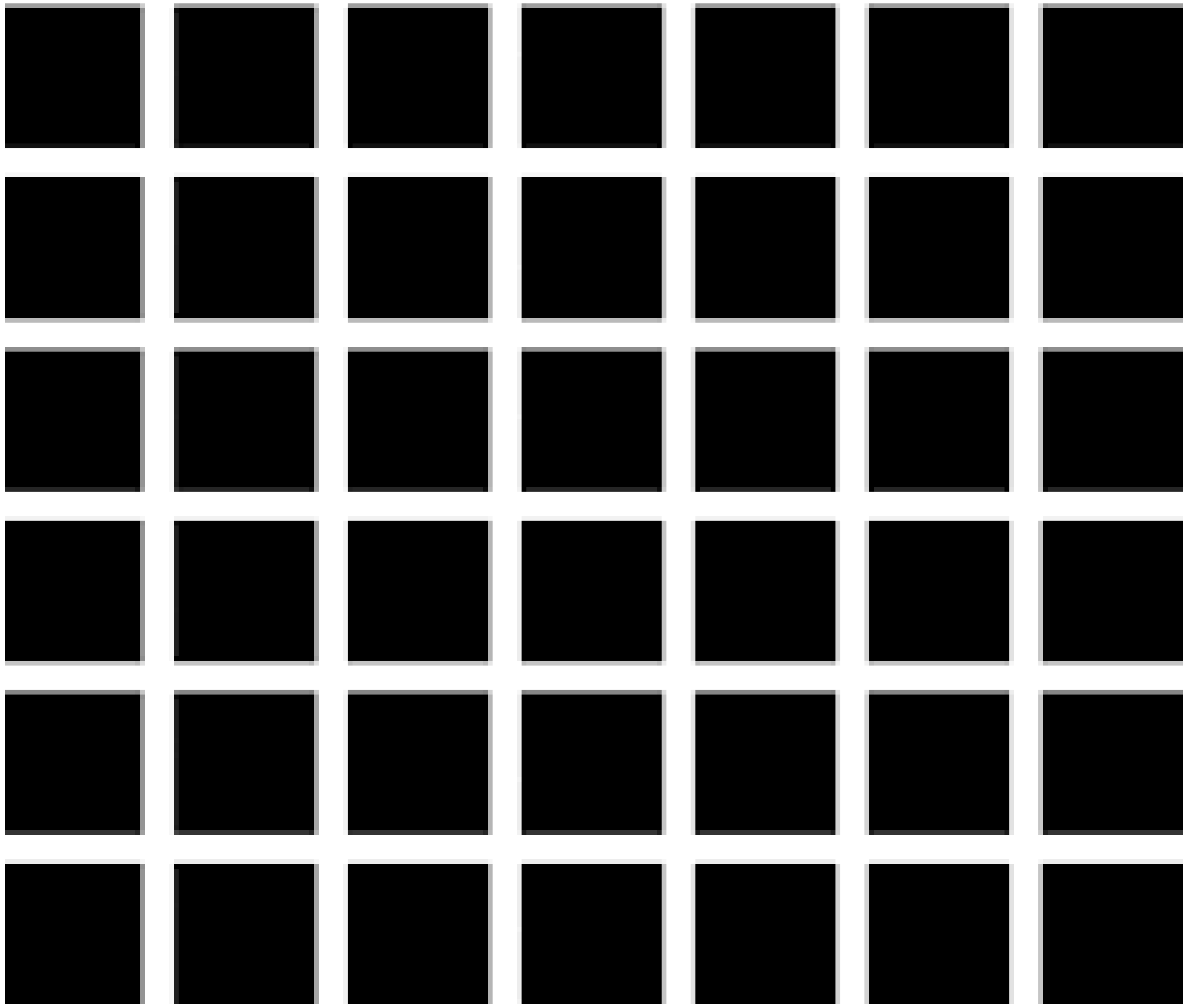
OPTICAL ILLUSIONS

Muller-Lyer illusion

- The Muller-Lyer illusion is a **well-known optical illusion in which two lines of the same length appear to be of different lengths**. The illusion was first created by a German psychologist named Franz Carl Muller-Lyer in 1889.



Hermann Grid Illusion



INVERSE MODEL APPLICATIONS

- NO KNOWLEDGE ABOUT THE CAMERA USED , DISTANCE FROM THE OBJECT AND THE OTHER PARAMETERS BUT WILL HAVE TO MODEL THE REAL WORLD IN WHICH PICTURE WAS TAKEN-ESTIMATE THOSE FROM THE IMAGES.(FROM PARTIAL &NOISY INFORMATION)
- IN FORWARD MODEL APPLICATIONS THESE INFORMATIONS WILL BE AVAILABLE (EG COMPUTER GRAPHICS),PHYSICS(RADIOMETRY, OPTICS & SENSOR DESIGN)

A brief History of CV

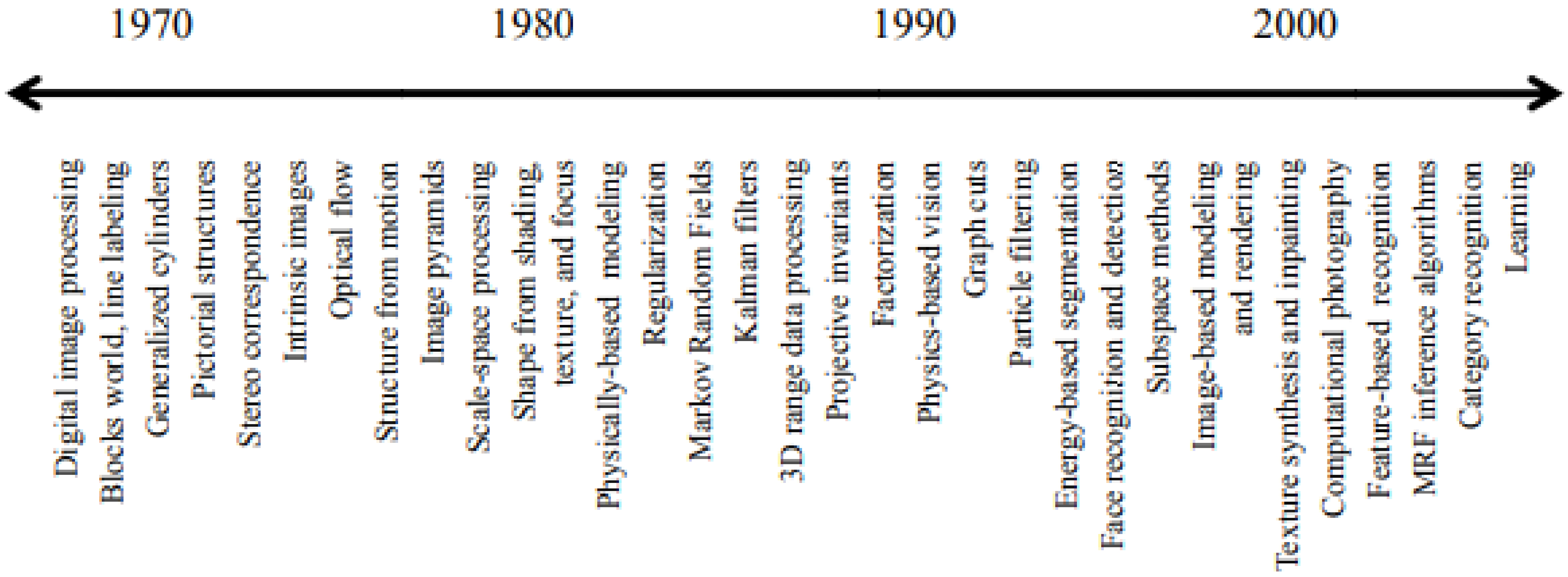


Figure 1.6 A rough timeline of some of the most active topics of research in computer vision.

Some Early CV algorithms

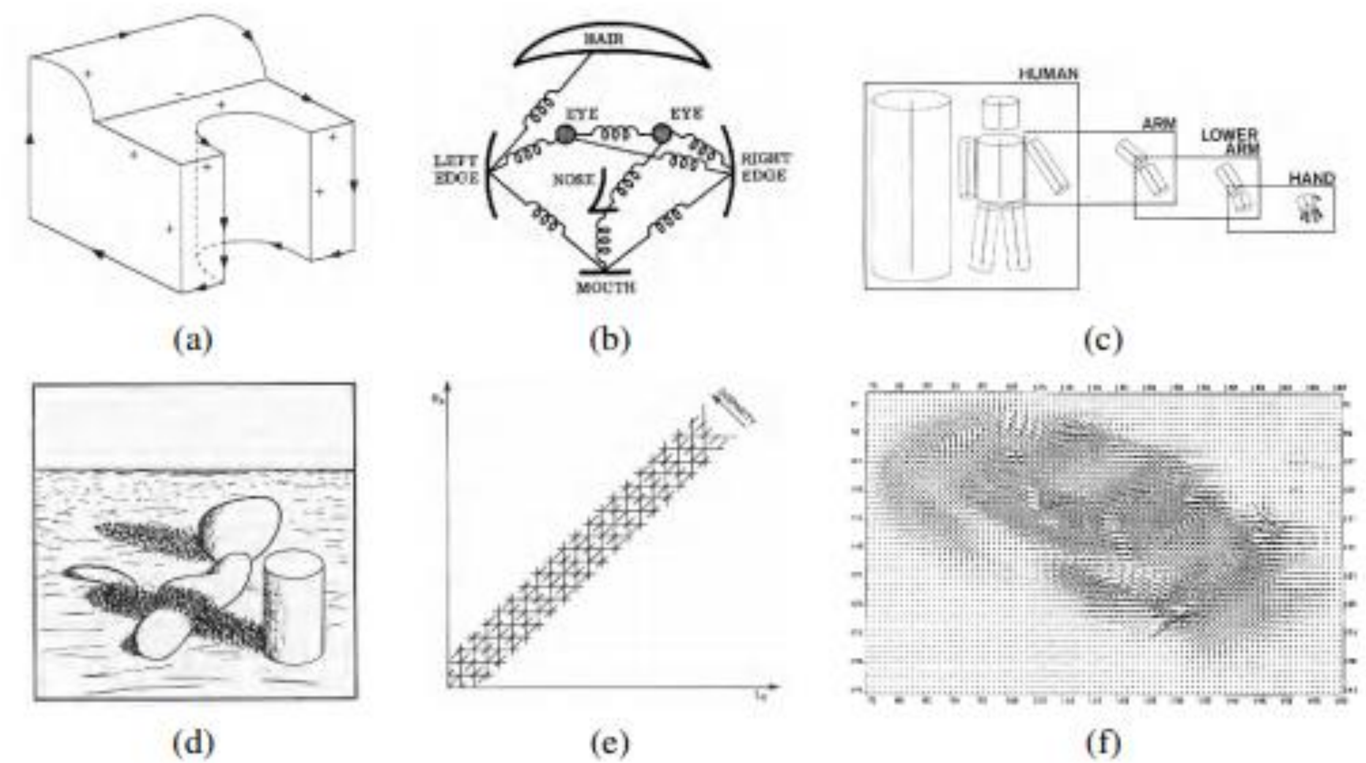


Figure 1.7 Some early (1970s) examples of computer vision algorithms: (a) line labeling (Nalwa 1993) © 1993 Addison-Wesley, (b) pictorial structures (Fischler and Elschlager 1973) © 1973 IEEE, (c) articulated body model (Marr 1982) © 1982 David Marr, (d) intrinsic images (Barrow and Tenenbaum 1981) © 1973 IEEE, (e) stereo correspondence (Marr 1982) © 1982 David Marr, (f) optical flow (Nagel and Enkelmann 1986) © 1986 IEEE.

Image Formation

PINHOLE CAMERA

The first models of the camera obscura (literally, dark chamber) invented in the sixteenth century did not have lenses, but instead used a pinhole to focus light rays onto a wall or translucent plate and demonstrate the laws of perspective discovered a century earlier by Brunelleschi.

FIGURE 1.1: Image formation on the backplate of a photographic camera. *Figure from US NAVY MANUAL OF BASIC OPTICS AND OPTICAL INSTRUMENTS, prepared by the Bureau of Naval Personnel, reprinted by Dover Publications, Inc. (1969).*

PINHOLE CAMERA

- Pinholes were replaced by more and more sophisticated lenses as early as 1550, and the modern photographic or digital camera is essentially a camera obscura capable of recording the amount of light striking every small area of its backplane

Pinhole Imaging Model

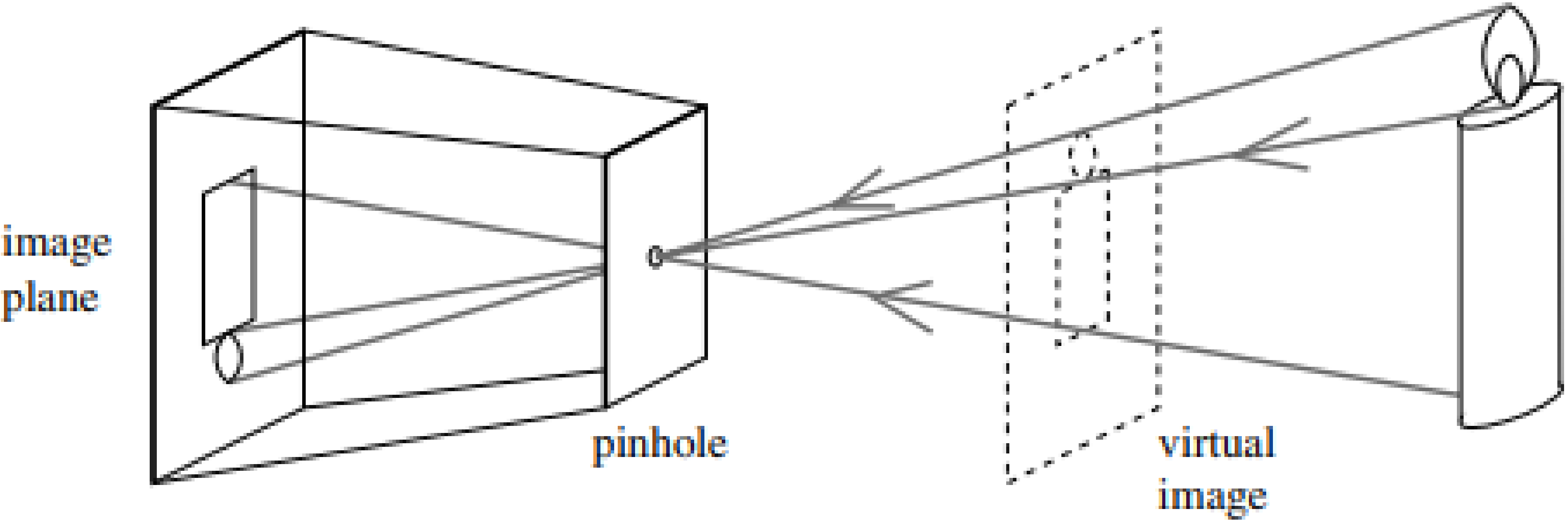
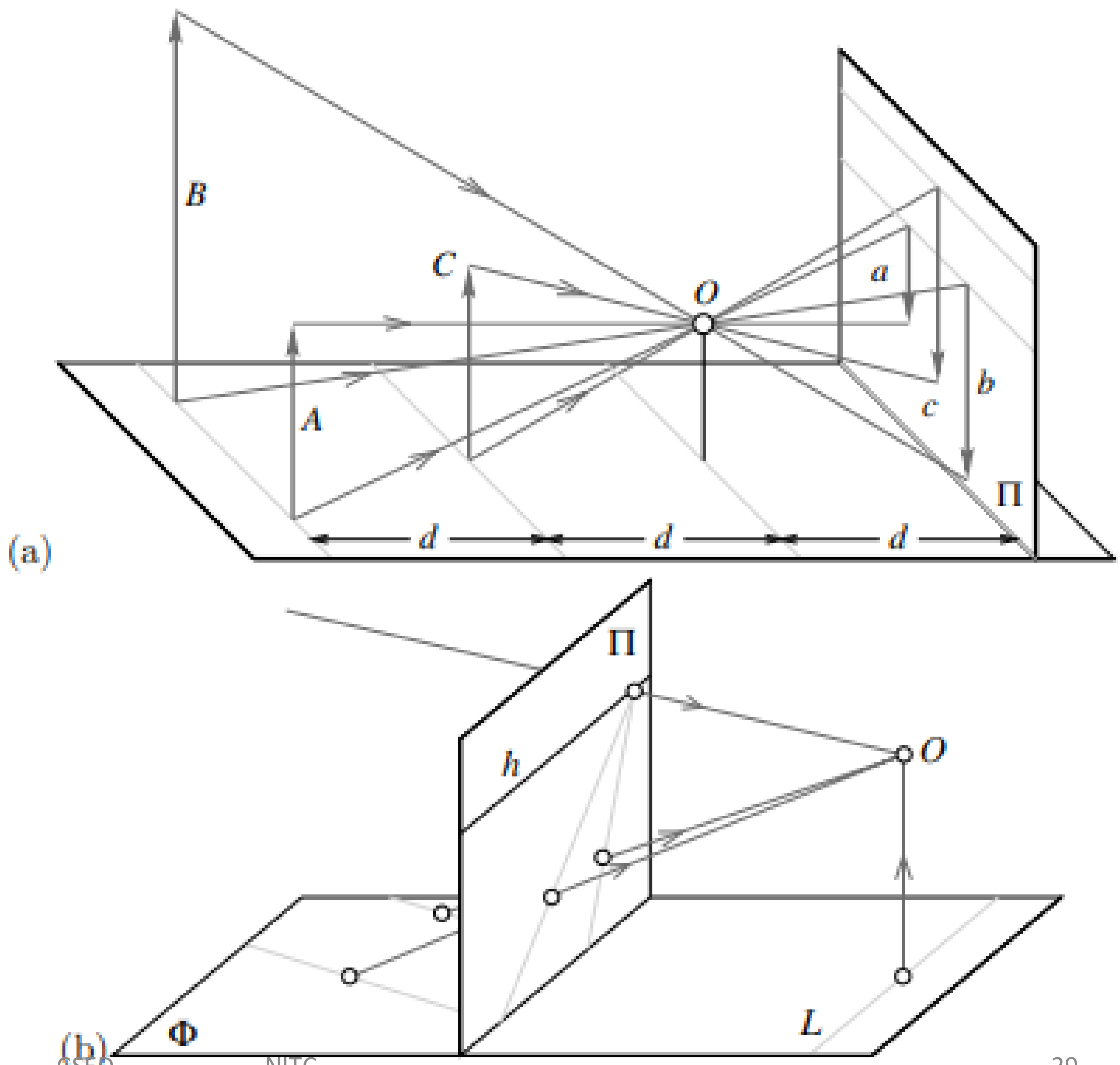
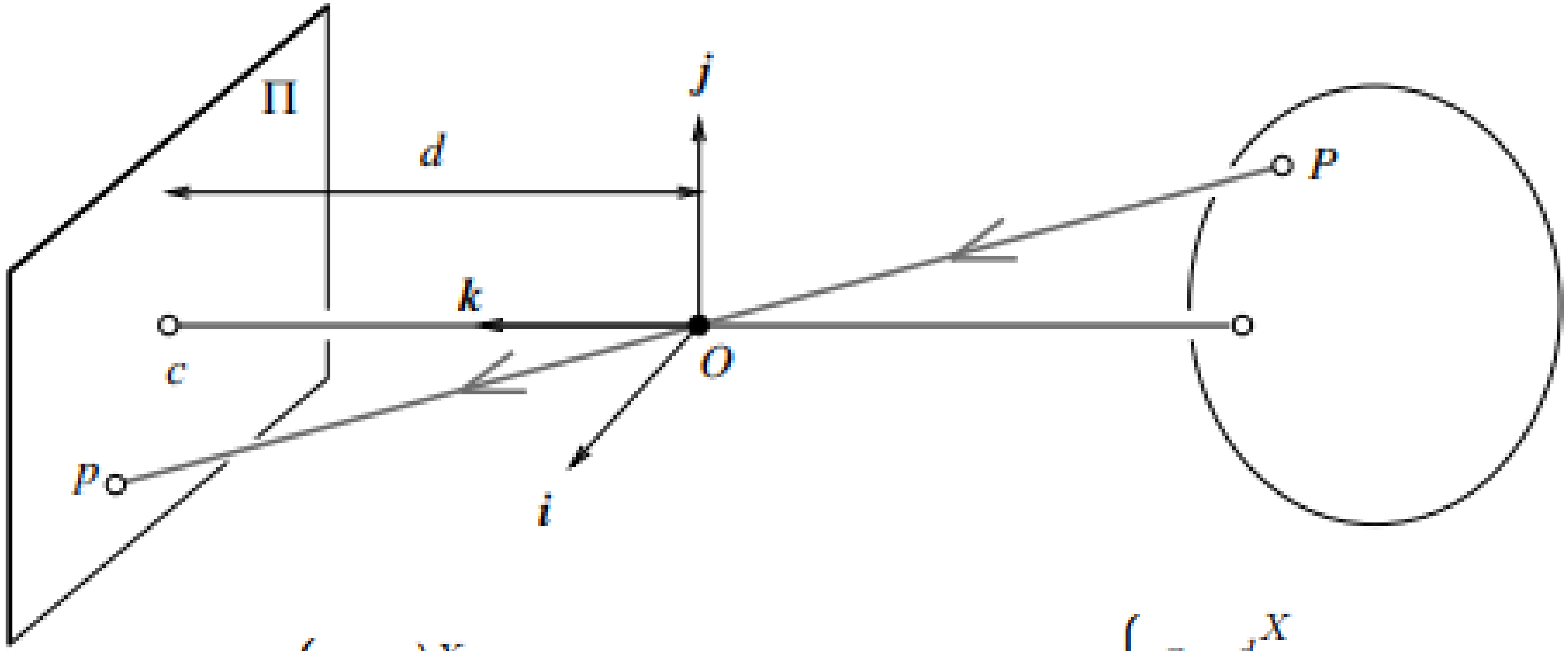


FIGURE 1.2: The pinhole imaging model.

Perspective Effects

- **far objects appear smaller than close ones: The distance d from the pinhole O to the plane containing C is half the distance from O to the plane containing A and B;**
- **(b) the images of parallel lines intersect at the horizon (after Hilbert and Cohn-Vossen, 1952, Figure 127). Note that the image plane Π is behind the pinhole in (a) (physical retina), and in front of it in (b) (virtual image plane).**





$$\begin{cases} x = \lambda X \\ y = \lambda Y \\ d = \lambda Z \end{cases} \iff \lambda = \frac{x}{X} = \frac{y}{Y} = \frac{d}{Z},$$

$$\begin{cases} x = d\dfrac{X}{Z}, \\ y = d\dfrac{Y}{Z}. \end{cases}$$

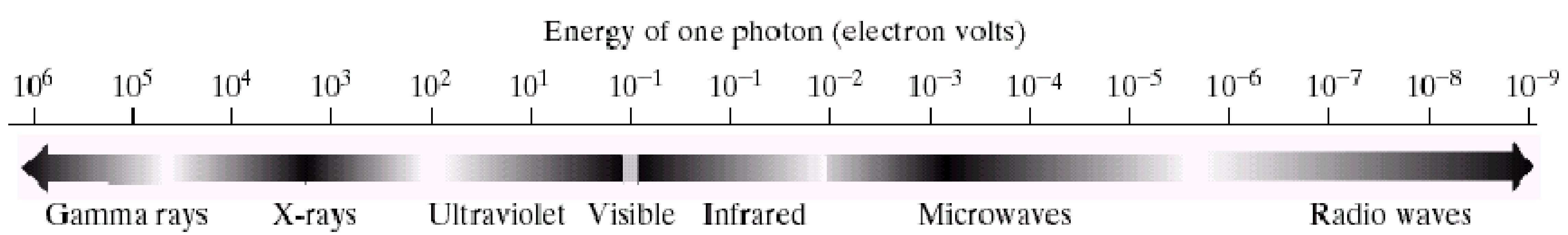
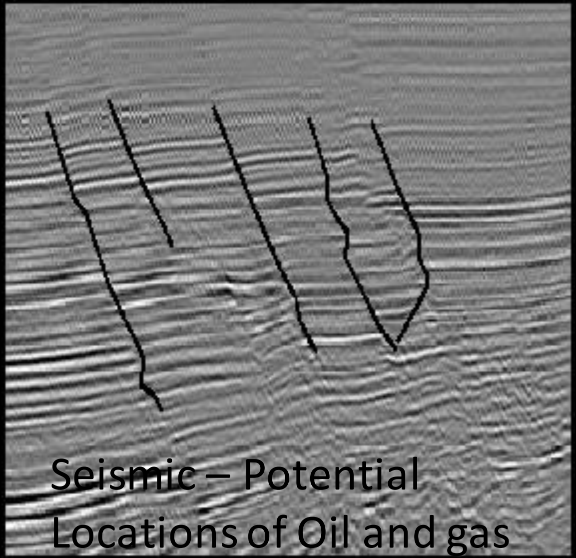


Image Acquisition

- **Requires a device that is sensitive to a band in the electromagnetic energy spectrum (such as x-ray, ultraviolet, visible or infra red bands), and that produces an electrical signal output proportional to the level of energy sensed.**

- **Eg;- film, TV camera, vidicon camera, CCD camera, etc.**

Other Non-ElectroMagnetic Imaging Modalities



Seismic – Potential Locations of Oil and gas

- Acoustic imaging
- Translate “sound waves” into image signals
- Electron microscopy
- Shine a beam of electrons through a speciman
- Synthetic images in Computer Graphics
- Computer generated (non-existent in the real world)

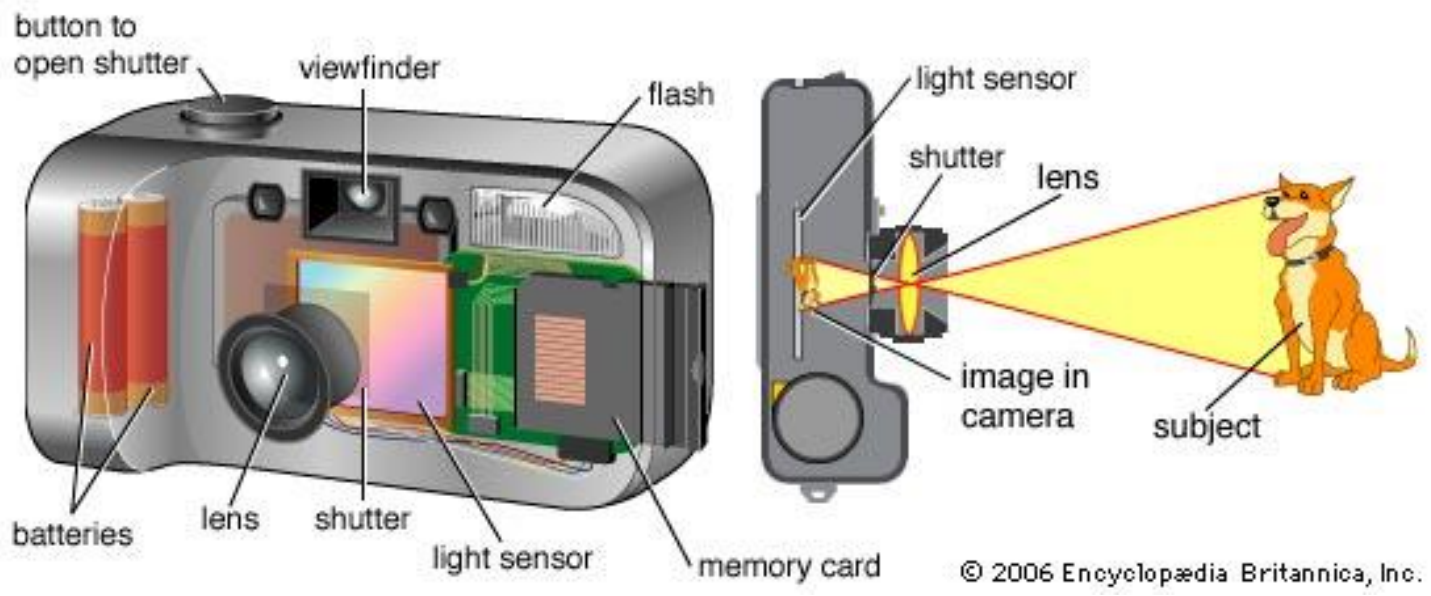


Image Acquisition - Devices

- Image sensors
- monochrome/ color camera
- CCD Camera
- line-scan camera
- area scan camera
- A/D Converter & frame buffers

- A charge-coupled device is an integrated circuit containing an array of linked, or coupled, capacitors. Under the control of an external circuit, each capacitor can transfer its electric charge to a neighboring capacitor. CCD sensors are a major technology used in digital imaging

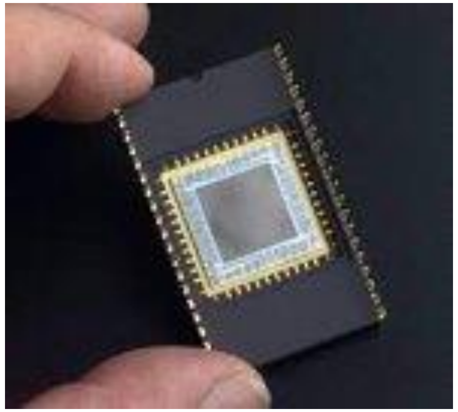


Image Formation

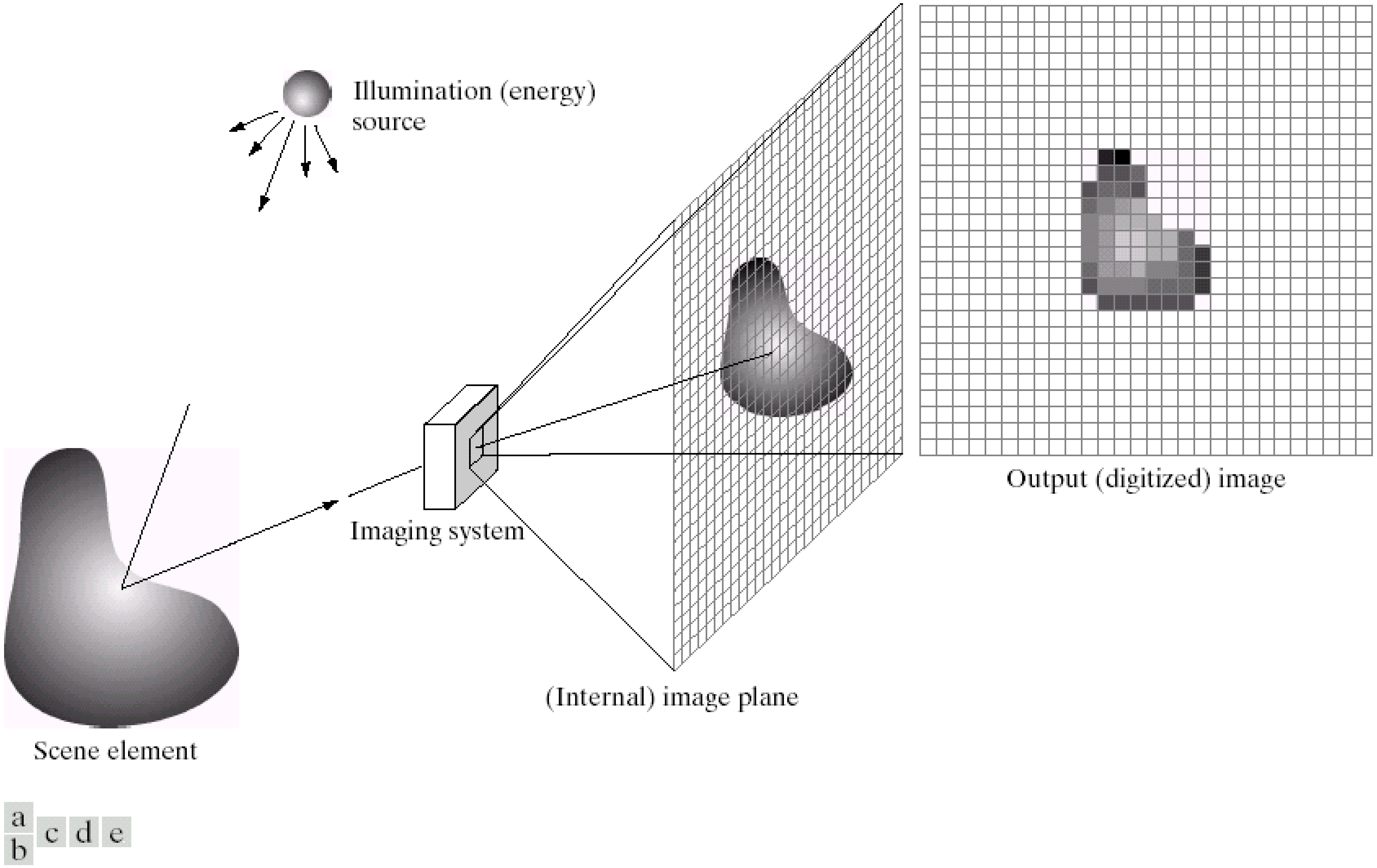


FIGURE 2.15 An example of the digital image acquisition process. (a) Energy ("illumination") source. (b) An element of a scene. (c) Imaging system. (d) Projection of the scene onto the image plane. (e) Digitized image.

Sampling and Quantization

- For computer processing, the image function f(x,y) must be digitized both spatially and in magnitude

- Digitization of the spatial coordinates (x,y) is called image or spatial sampling

- Digitization of the gray level (magnitude of f(x,y)) value is called gray-level quantization or sampling

Sampling and Quantization

A continuous image f(x,y) may be approximated by equally spaced samples arranged in the form of N X M array :

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

Sampling and Quantization

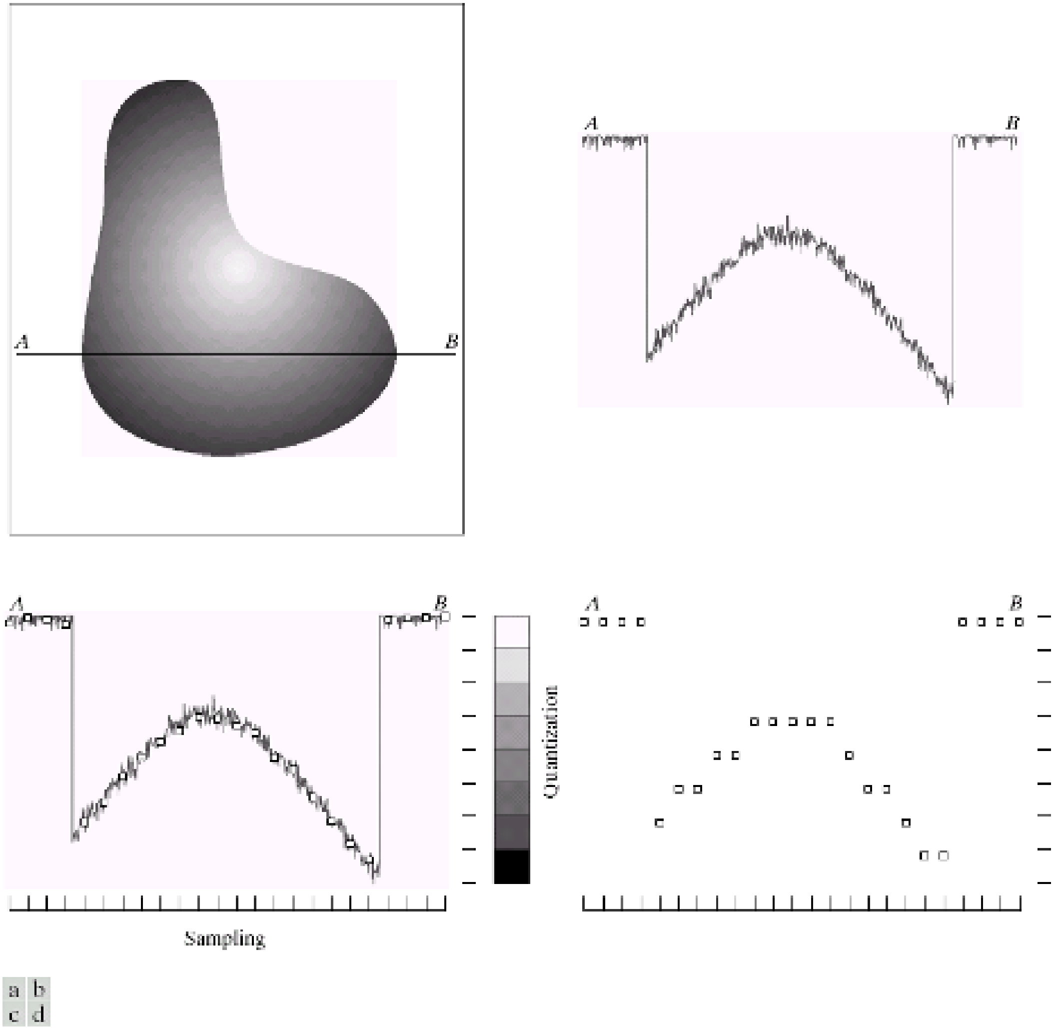
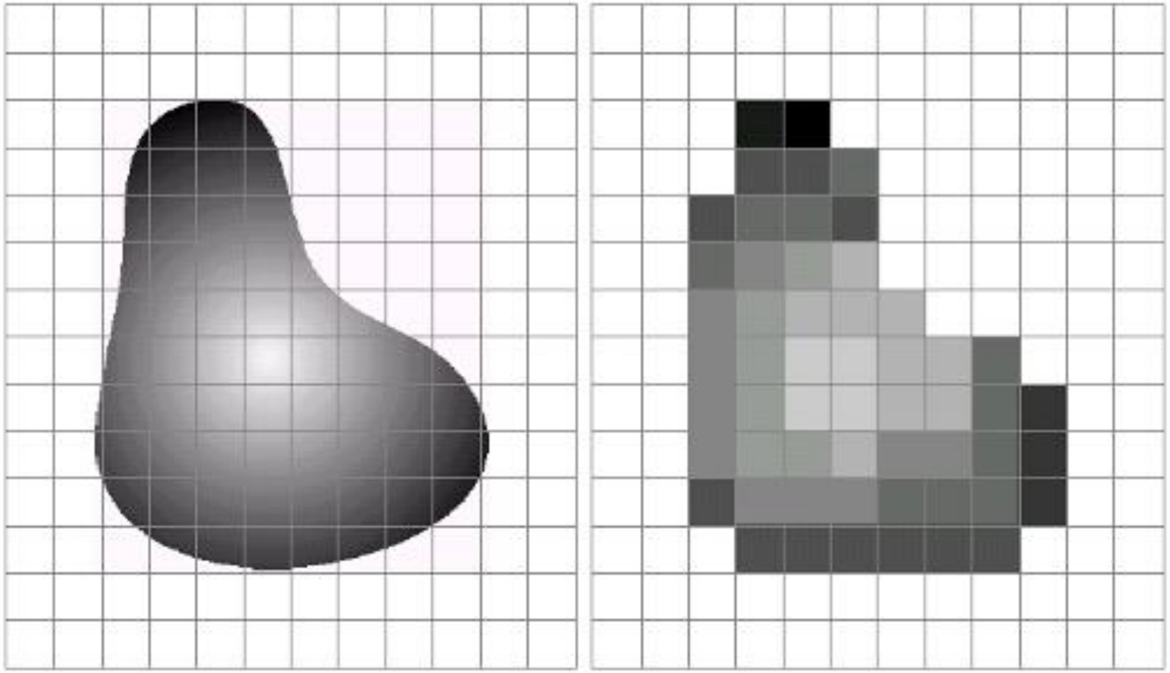


FIGURE 2.16 Generating a digital image. (a) Continuous image. (b) A scan line from A to B in the continuous image, used to illustrate the concepts of sampling and quantization. (c) Sampling and quantization. (d) Digital scan line.

Sampling and Quantization



a b

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

Non-uniform sampling & quantization

- Non-uniform sampling & quantization in general improves the appearance of images
- Fine spatial sampling is required in the neighborhood of sharp gray-level transitions
- Coarse sampling may be used in relatively smooth regions
- Ex:- consider a face image superimposed on a uniform background
- Background need to be sampled only coarsely
- Regions of face which contain more details are to be sampled finely
- Greater sampling concentration should be used in the boundaries betn. the face and the background

What is an Image?

- We can think of an image as a function, f, from R2 to R:

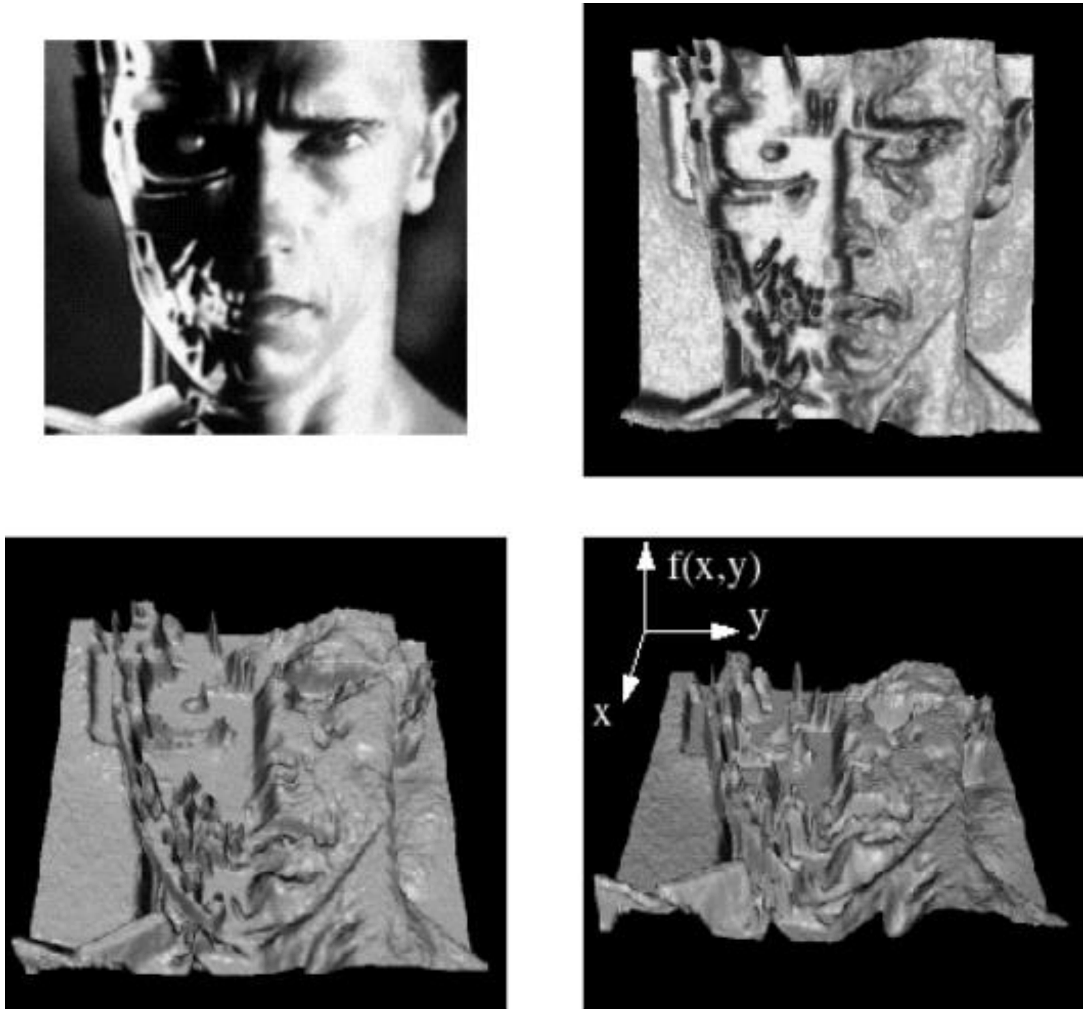
f(x, y) gives the intensity at position (x, y) Click to add text

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

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

illumination Reflectance

$0<I(x,y)<\infty$ $0<r(x,y)<1$



Grayscale Image Storage Requirements

- The number of quantization levels are: L^k
- For a NxN image the number of storage bits is

$$B=N^2k$$

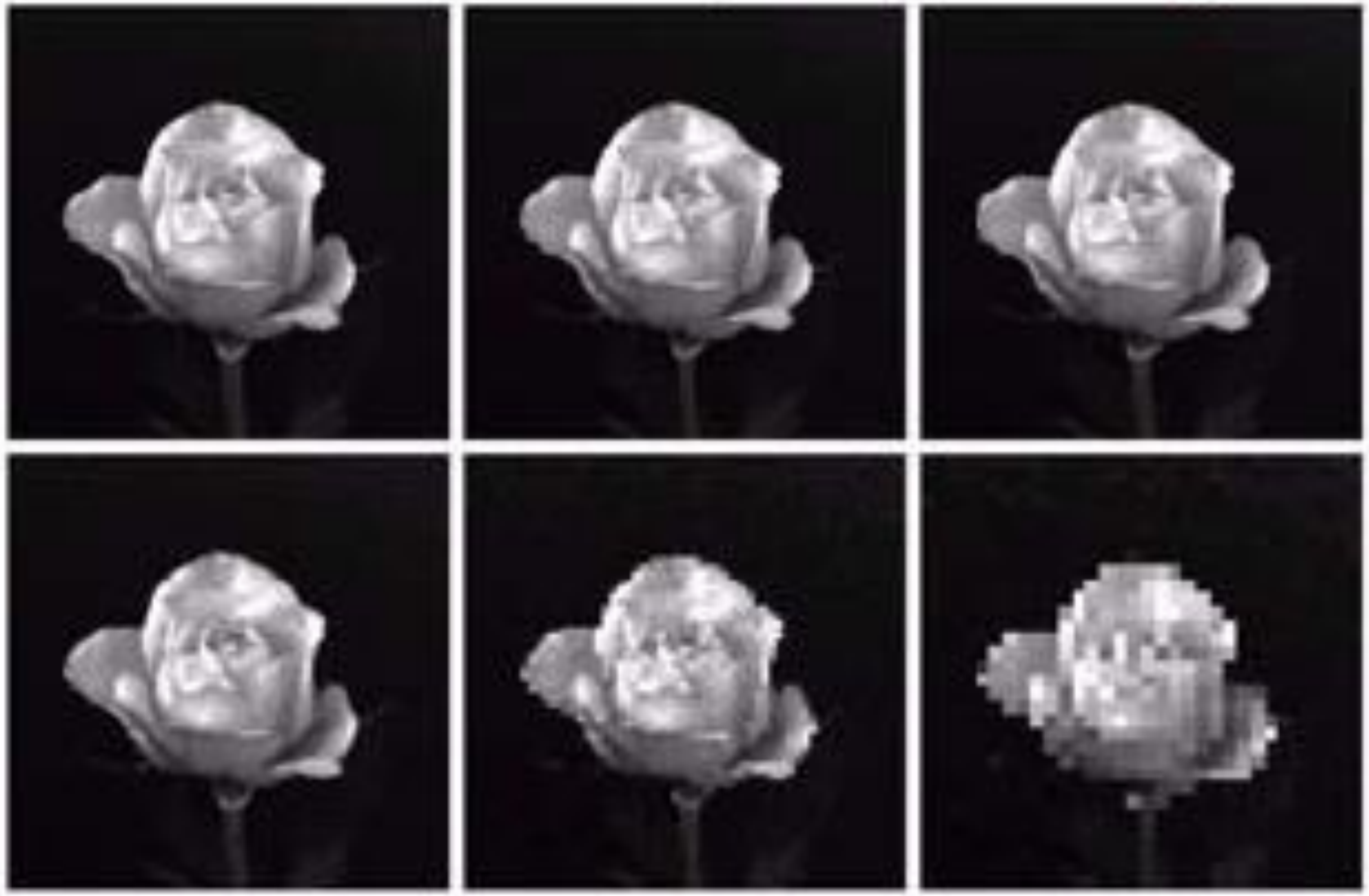
TABLE 2.1
Number of storage bits for various values of N and k.

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	3072	4,096	5,120	6,144	7,168	8,192
64	4,096	8,192	12288	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	196608	262,144	327,680	393,216	458,752	524,288
512	262,144	524,288	786432	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	29,360,128	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

Spatial Resolution (Fixed Pixel Size)

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

Spatial Resolution (Fixed Image Size)



a b c
d e f

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

Intensity Resolution

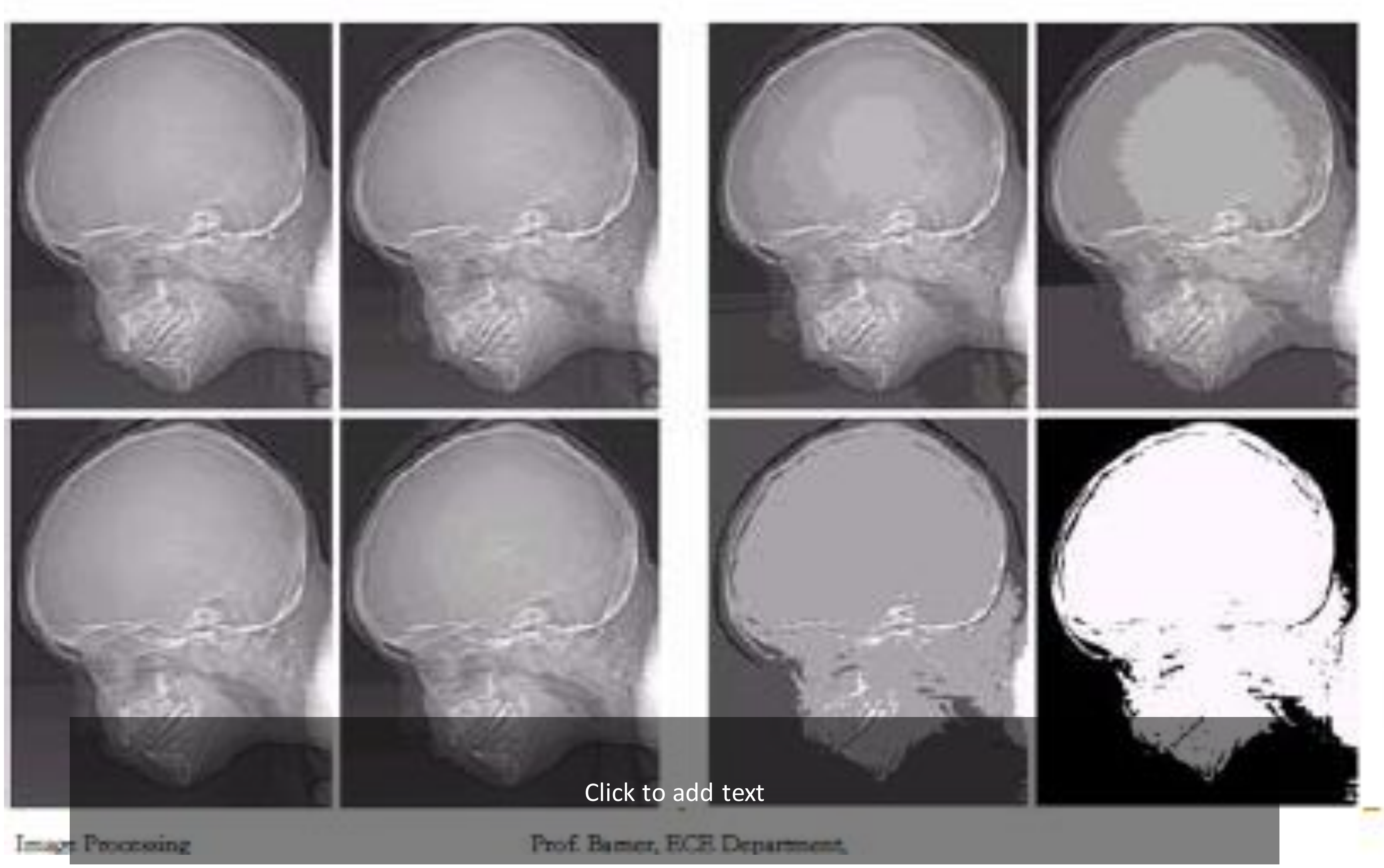
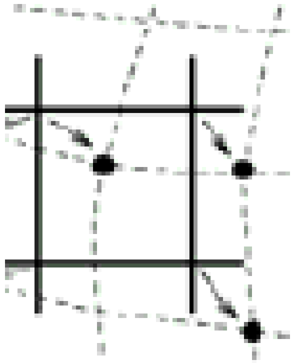
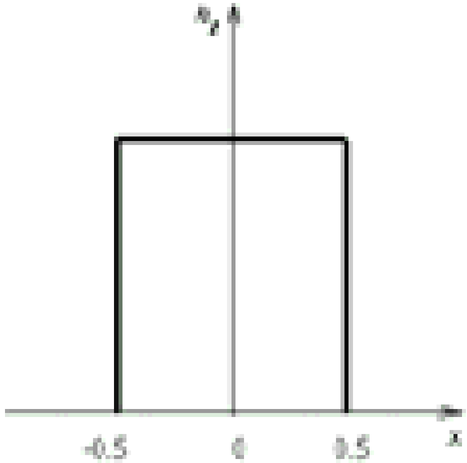


Image Zooming and Shrinking

- Both operations involve resampling
  - Zooming is oversampling
  - Shrinking is undersampling





- Nearest neighbor interpolation
  - Overlay two sampling grids (known and unknown)
  - Populate unknown grid with the closest sample from unknown grid
  - Special case: pixel replication
    - Integer increases in sampling rate
    - Repeat rows, columns, etc.

Pixel Neighborhoods

- Let the pixel *p* have coordinates *(x,y)*
- $N_4(p)$ is the 4-neighborhood of *p* consisting of horizontal and vertical neighbor pixels at locations

$$(x+1, y), (x-1, y), (x, y+1), (x, y-1)$$

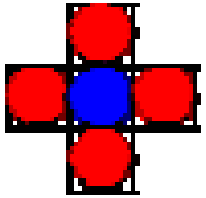
- $N_D(p)$ is the diagonal neighborhood of *p*:

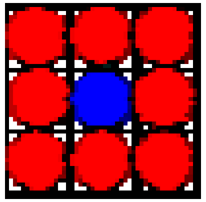
$$(x+1, y+1), (x+1, y-1), (x-1, y+1), (x-1, y-1)$$

- The 8-neighborhood of *p* is

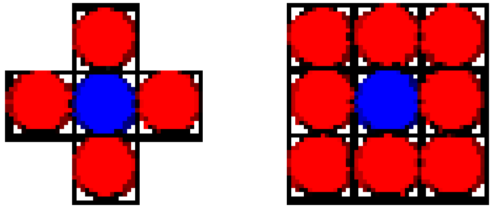
$$N_8(p) = N_D(p) \cup N_4(p)$$

- Consider border affects



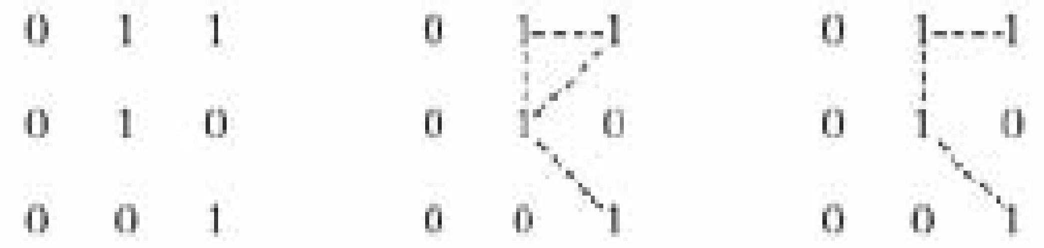


Adjacency



- A set of gray levels, V, is used to define adjacency
  - Examples:
    - Binary, $V=\{1\}$
    - Grayscale, $V=\{128, 129, \ldots, 255\}$
- Let $\boldsymbol{p}_c$ be the coordinates of a pixel p
- p and q are 4-*adjacent* if $p, q \in V$ and $\mathbf{q}_c \in N_4(p)$
- p and q are 8-*adjacent* if $p, q \in V$ and $\mathbf{q}_c \in N_8(p)$
- p and q are (mixed) *m-adjacent* if $p, q \in V$ and
  - $\mathbf{q}_c \in N_4(p)$ or
  - $\mathbf{q}_c \in N_D(p)$ and $N_4(p) \cap N_4(q)$ has no pixel values from V

Adjacency Example



a b c

FIGURE 2.26 (a) Arrangement of pixels; (b) pixels that are 8-adjacent (shown dashed) to the center pixel; (c) *m*-adjacency.

- The example above is binary, with $V=\{1\}$
- A path from p to q is a sequence of coordinates

$$\mathbf{p}_C^0, \mathbf{p}_C^1, \ldots, \mathbf{p}_C^N$$

  where $p^0=p$, $p^N=q$ and p^i and p^{i-1} are adjacent
- The length of the path is N
- We can do find 4-, 8-, or m-paths depending on adjacency

Paths, Regions, and Boundaries

- A path is *closed* if $p^0=p^N$
- Let S be a subset of pixels in an image
  - p and q are connected in S if there exists a path between them consisting entirely of pixels in S
  - For any p in S, the set of pixels connected to p in S is the *connected component* of S
  - S is a *connected set* if it has only one connected component
    - Connected subsets are referred to as *regions*
  - The *boundary* of a region R is the set of pixels in the region that have one or more neighbors outside R
    - Boundaries form closed paths (different concept than edge)

Distance Measures

- Let p, q, and z be pixels. D is a distance functions (metric) if
  - $D(p,q) \geq 0$ $(D(p,q)=0$ iff $p=q)$,
  - $D(p,q)= D(q,p)$, and
  - $D(p,z) \leq D(p,q)+D(q,z)$
- Euclidean distance: $D_e(p,q)=[(p_{c1}-q_{c1})^2+(p_{c2}-q_{c2})^2]^{½}$
- City-block distance: $D_4(p,q)=|p_{c1}-q_{c1}|+|p_{c2}-q_{c2}|$
- Chessboard distance: $D_8(p,q)=\max(|p_{c1}-q_{c1}|,|p_{c2}-q_{c2}|)$

Distance Matrices

Distance examples

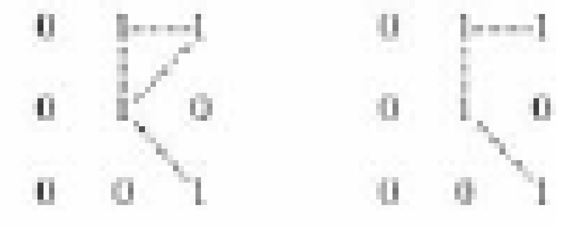
- D_4 example:

```
    2
  2 1 2
2 1 0 1 2
  2 1 2
    2
```

- D_8 example:

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

- D_m distance is defined as the length of the shortest *m*-path



Definitions: Some basic relationships between pixels

- Arithmetic/ logic operations
- Arithmetic operations between two pixels – p & q are denoted by
  - ❖ Addition : p + q
  - ❖ Subtraction : p – q
  - ❖ Multiplication : p*q, or pq, or p X q
  - ❖ Division : p / p
- Arithmetic operations are carried out pixel by pixel

Image Addition

Original Image with salt and pepper noise

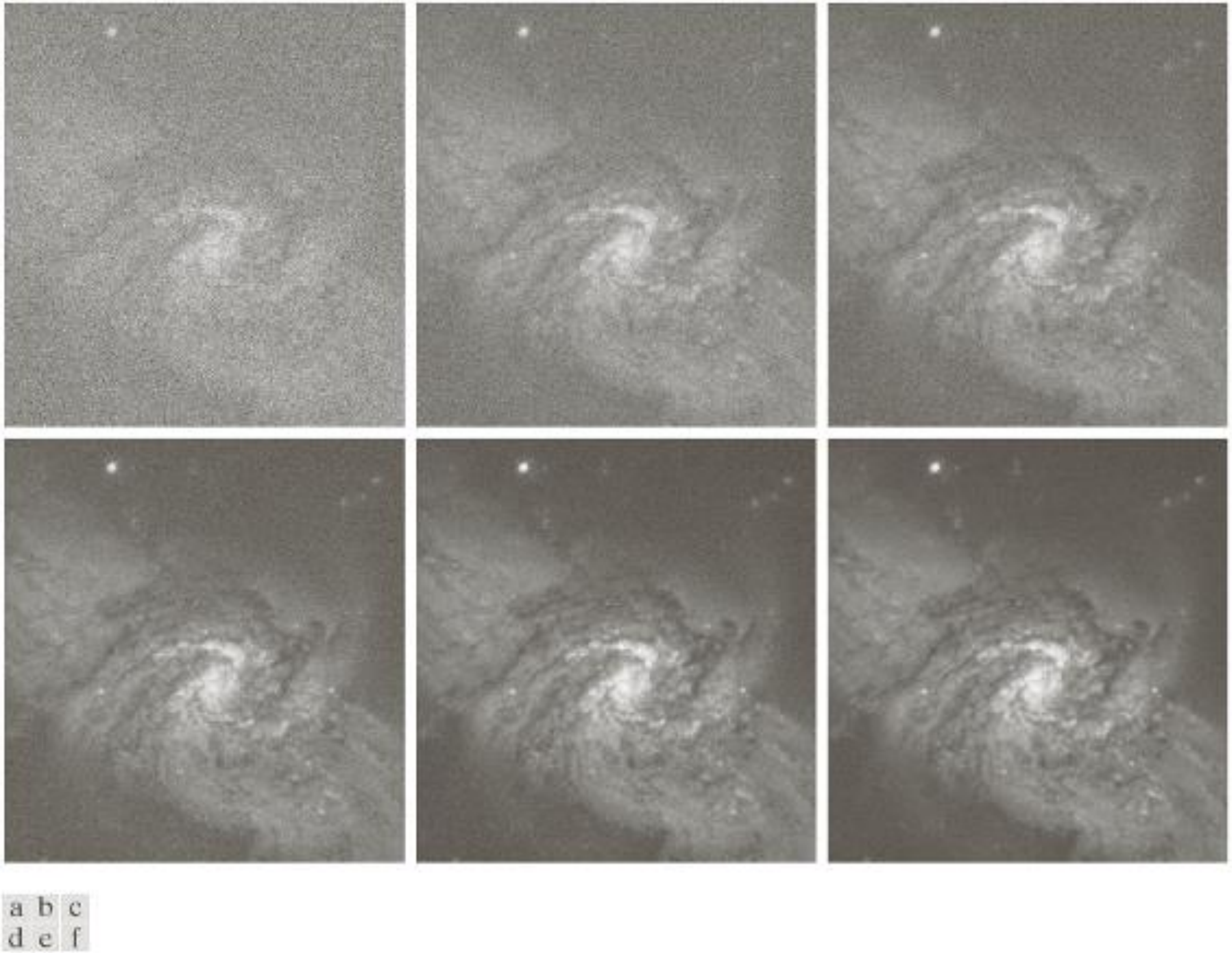
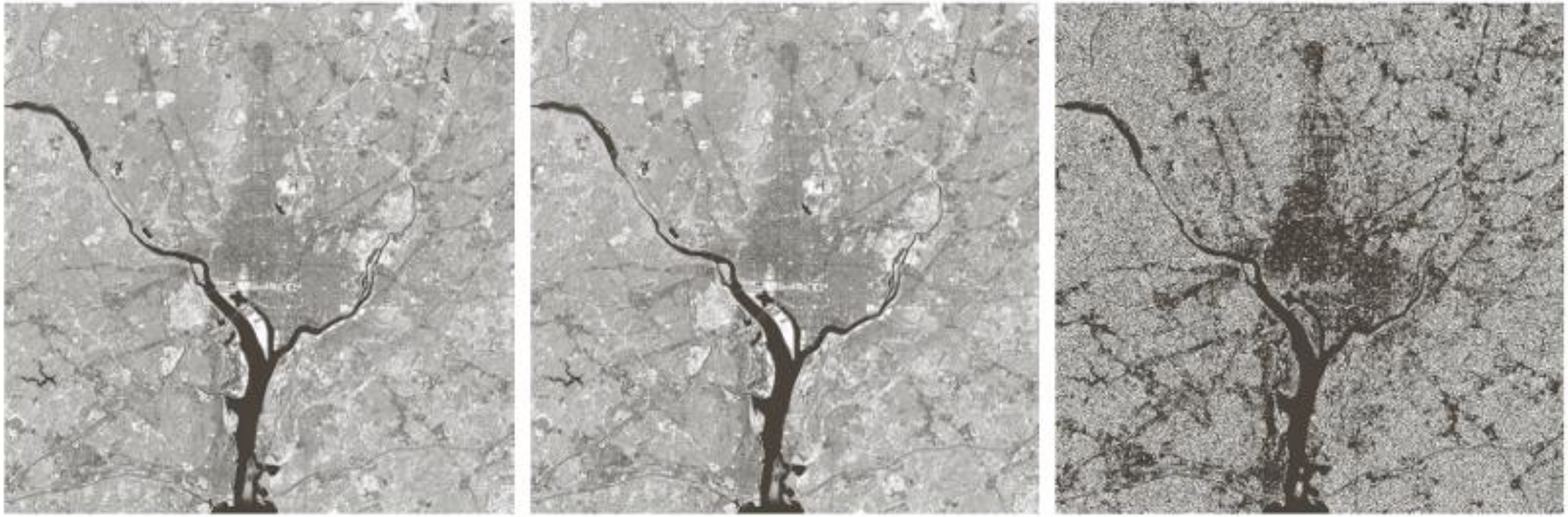


FIGURE 2.26 (a) Image of Galaxy Pair NGC 3314 corrupted by additive Gaussian noise. (b)–(f) Results of averaging 5, 10, 20, 50, and 100 noisy images, respectively. (Original image courtesy of NASA.)

Image Subtraction



a b c

FIGURE 2.27 (a) Infrared image of the Washington, D.C. area. (b) Image obtained by setting to zero the least significant bit of every pixel in (a). (c) Difference of the two images, scaled to the range [0, 255] for clarity.

Mask Mode Radiography

- **Image subtraction** is used in medical imaging called mask mode. radiography. • The initial image is captured and used as the mask image, h(x,y).
- A contrast material is injected into the bloodstream and the image is taken say f(x,y)
- The mask image is subtracted from the resulting image f(x,y) to give an enhanced output image g(x,y).

$G(x,y) = f(x,y) - h(x,y)$

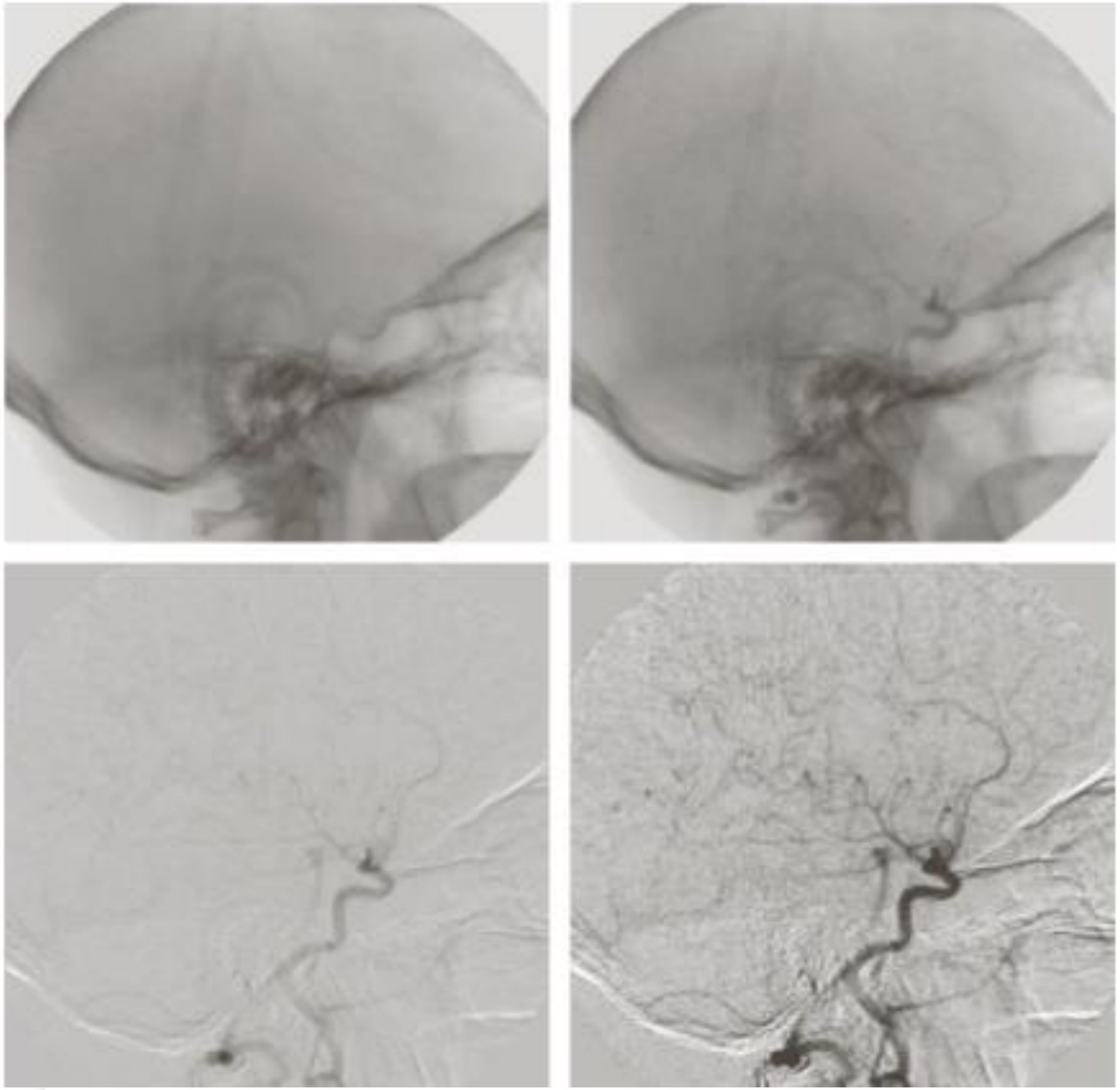


Image Multiplication – Shading Correction

a b c

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.)

Masking

a b c

FIGURE 2.30 (a) Digital dental X-ray image. (b) ROI mask for isolating teeth with fillings (white corresponds to 1 and black corresponds to 0). (c) Product of (a) and (b).

Definitions: Some basic relationships between pixels

- Logic operations
  - AND : p AND q
  - OR : p OR q
  - COMPLEMENT : not q
- Neighborhood operations

	z1	z2	z3		
	z4	z5	z6		
	z7	z8	z9		

subarea of an image

w1	w2	w3
w4	w5	w6
w7	w8	w9

3 X 3 mask with general coefficients

Logical Operations

- **AND, OR, NOT, XOR**

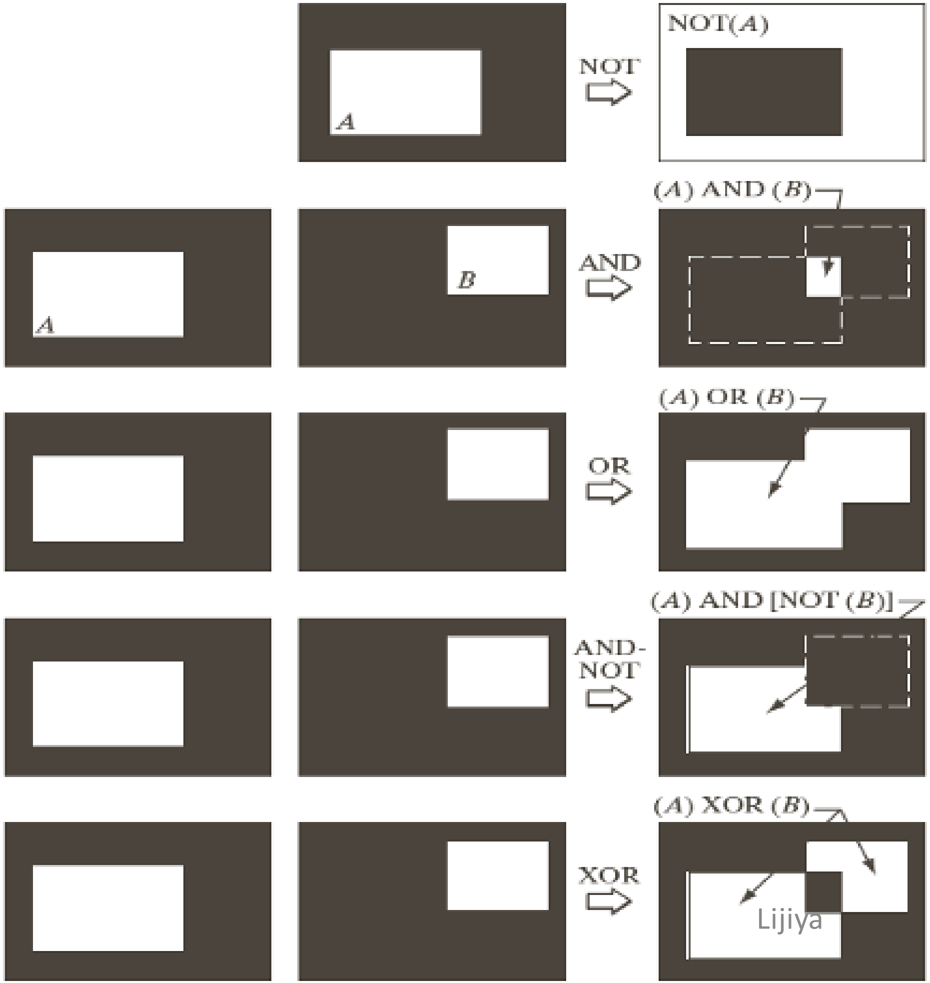
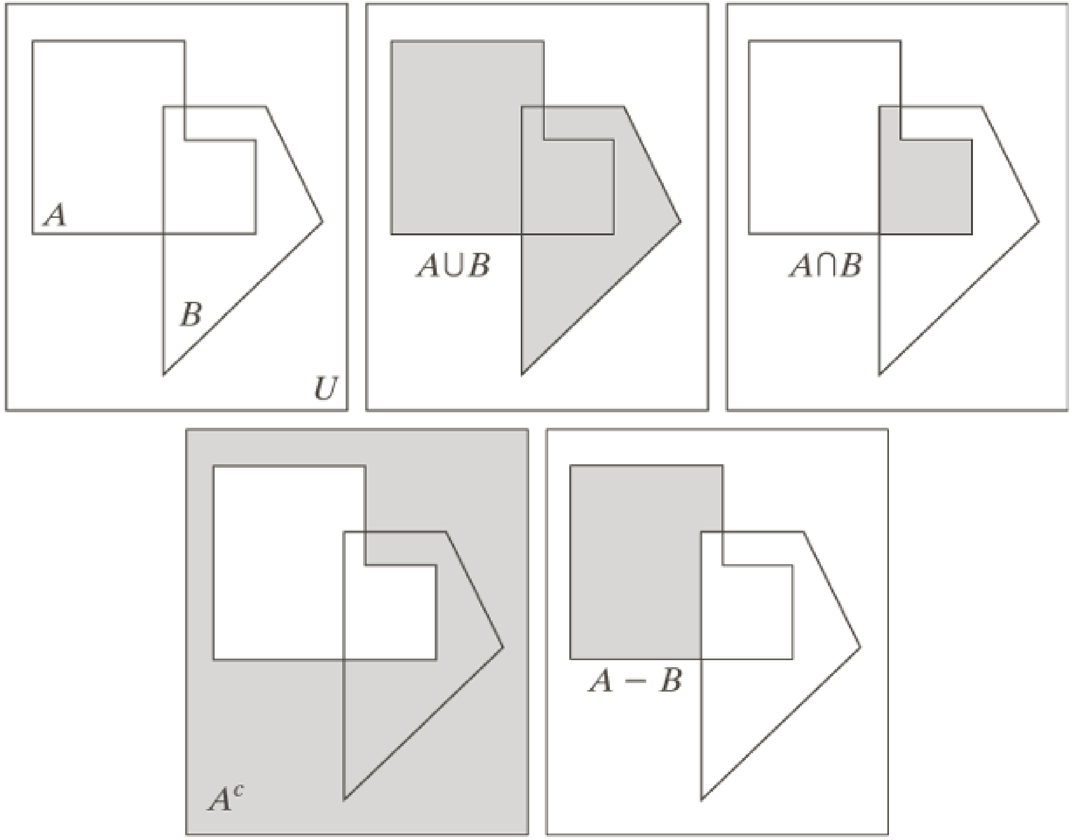


FIGURE 2.33 Illustration of logical operations involving foreground (white) pixels. Black represents binary 0s and white binary 1s. The dashed lines are shown for reference only. They are not part of the result.

a b c
d e

FIGURE 2.31
(a) Two sets of coordinates, A and B, in 2-D space. (b) The union of A and B. (c) The intersection of A and B. (d) The complement of A. (e) The difference between A and B. In (b)–(e) the shaded areas represent the member of the set operation indicated.

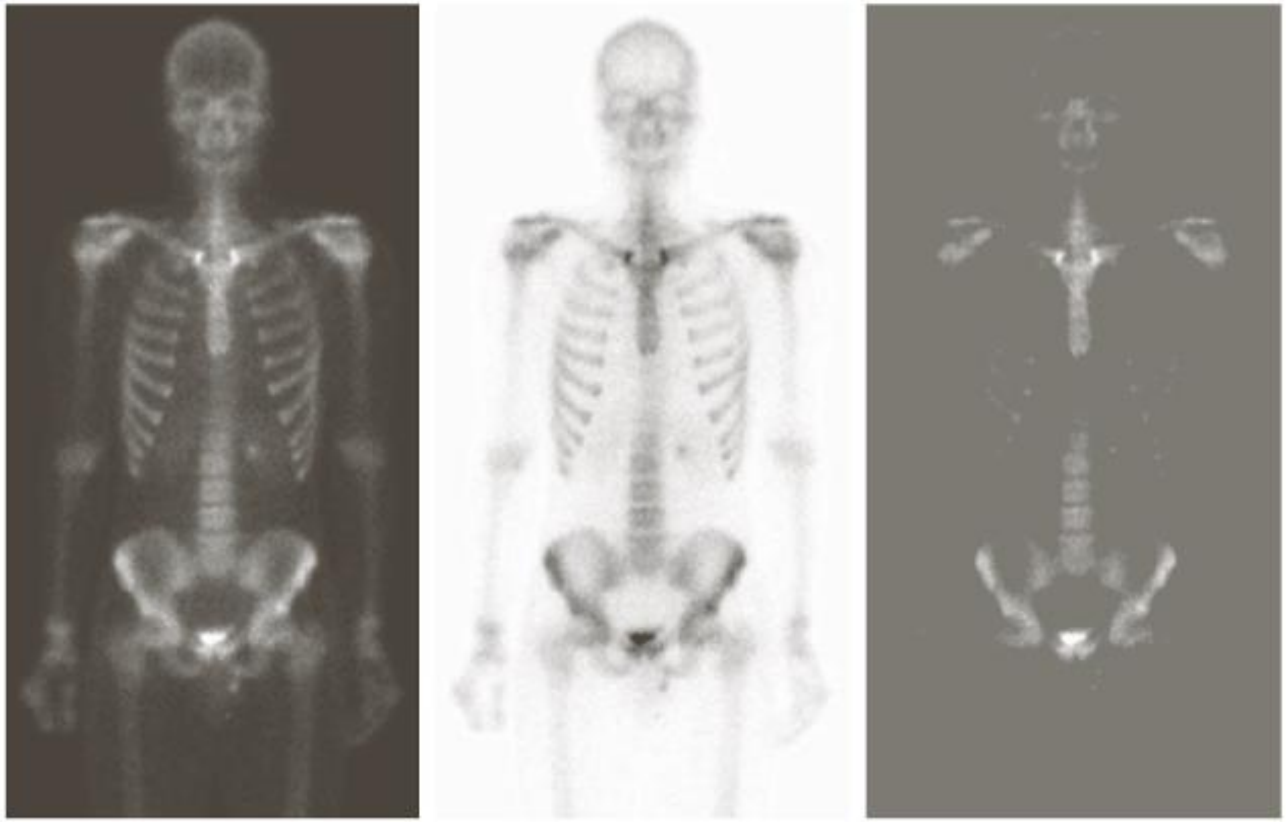


Set Operations

Set Operations

a b c

FIGURE 2.32 Set operations involving gray-scale images. (a) Original image. (b) Image negative obtained using set complementation. (c) The union of (a) and a constant image. (Original image courtesy of G.E. Medical Systems.)



Spatial Operations

- Spatial operations are performed directly on the pixels of a given image.
- Spatial operations can be classified into three broad categories:
- Single-pixel operations
- Neighborhood operations
- Geometric spatial operations

Single-Pixel Operations

- The simplest operation performed on a digital image is altering the values of individual pixels based on their intensity.
- $s = T(z)$
- where, z is the intensity of the pixels in the original image, s is the pixel intensity in the processed image and T is the transformation function.

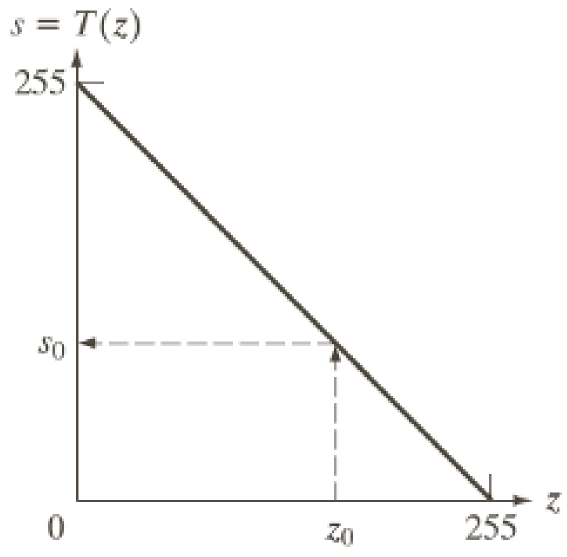
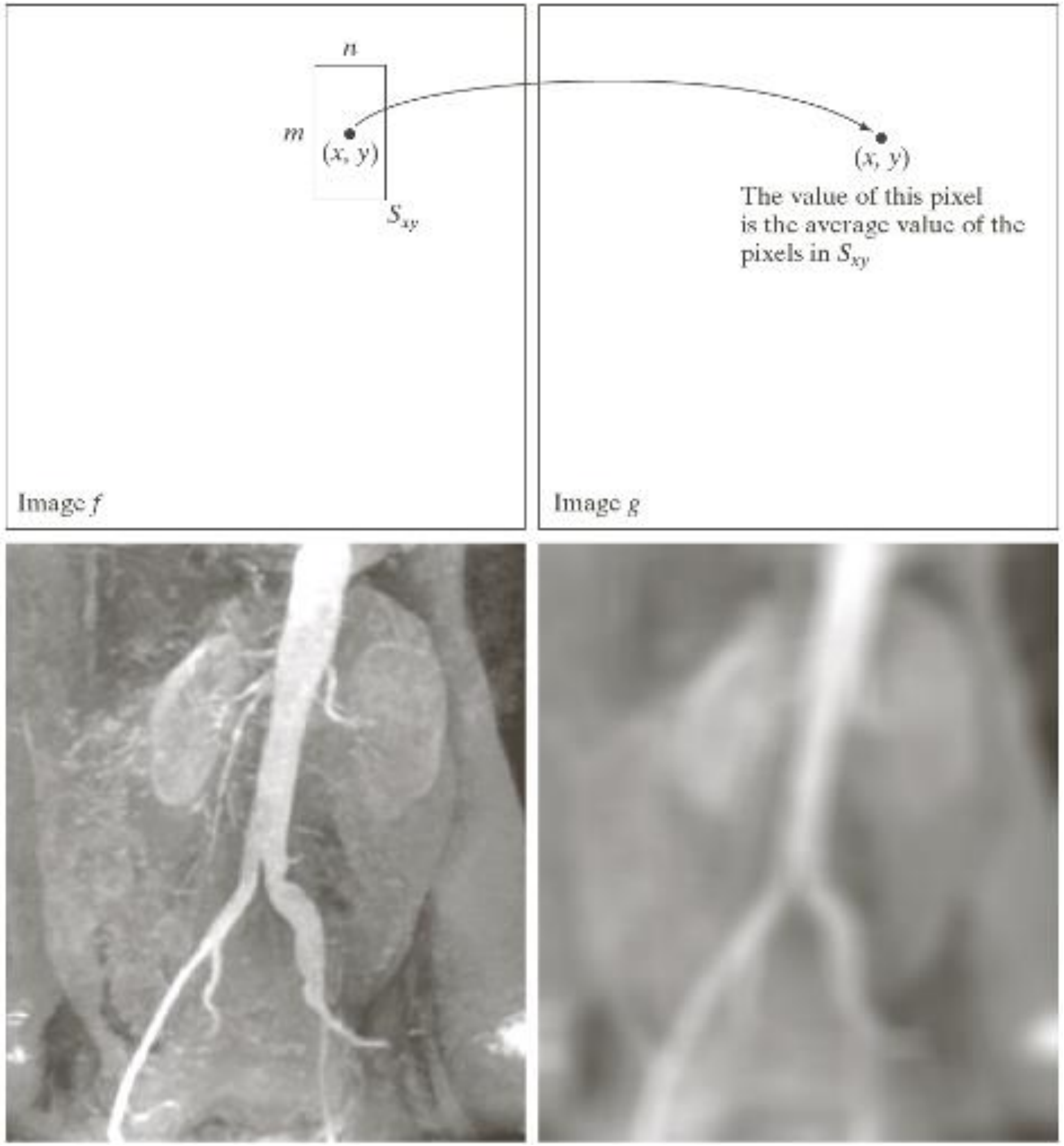


FIGURE 2.34 Intensity transformation function used to obtain the negative of an 8-bit image. The dashed arrows show transformation of an arbitrary input intensity value z_0 into its corresponding output value s_0.

Neighborhood operation:Averaging



a b
c d

FIGURE 2.35 Local averaging using neighborhood processing. The procedure is illustrated in (a) and (b) for a rectangular neighborhood. (c) The aortic angiogram discussed in Section 1.3.2. (d) The result of using Eq. (2.6-21) with $m = n = 41$. The images are of size 790×686 pixels.

Definitions: Neighborhood operations

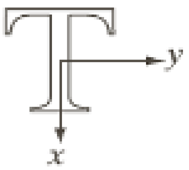
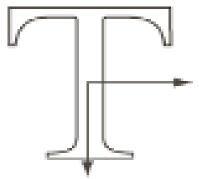
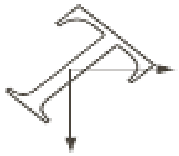
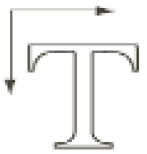
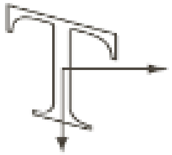
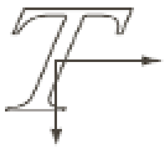
- Averaging
  - Un-weighted averaging
    - $z = 1/9(z1 + z2 + \ldots + z9)$
  - Weighted averaging
    - $Z = w1.z1+w2.z2+ \ldots + w9.z9$

1

Geometric Spatial transformations

- Geometric transformations modify the spatial relationship between pixels in an image.
- Also known as Rubber-sheet transformations.
- A geometric transformation consists of two basic operations:
- A spatial transformation of coordinates.
- Intensity interpolation that assigns intensity values to the spatially transformed pixels.
- The transformation of coordinates may be expressed as
- $(x,y) = T\{(v,w)\}$
- where (v,w) are the pixels coordinates in the original image, and (x,y) are the coordinates in the new image.

TABLE 2.2
Affine transformations based on Eq. (2.6.–23).

Transformation Name	Affine Matrix, T	Coordinate Equations	Example
Identity	$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$x = v$ $y = w$	
Scaling	$\begin{bmatrix} c_x & 0 & 0 \\ 0 & c_y & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$x = c_x v$ $y = c_y w$	
Rotation	$\begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$x = v\cos\theta - w\sin\theta$ $y = v\cos\theta + w\sin\theta$	
Translation	$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ t_x & t_y & 1 \end{bmatrix}$	$x = v + t_x$ $y = w + t_y$	
Shear (vertical)	$\begin{bmatrix} 1 & 0 & 0 \\ s_v & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$x = v + s_v w$ $y = w$	
Shear (horizontal)	$\begin{bmatrix} 1 & s_h & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$x = v$ $y = s_h v + w$	

Imaging Geometry:
Basic transformations in imaging

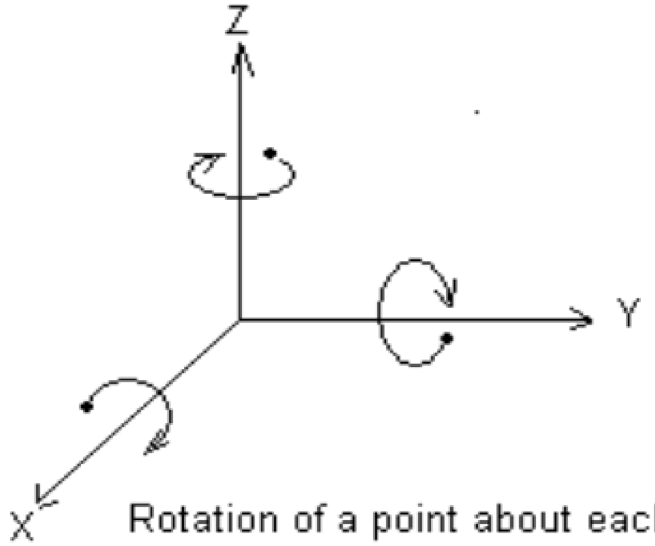
- Scaling:
  - Scaling by factors Sx, Sy, and Sz along the X, Y, and Z axes is given by

$$\mathbf{S} = \begin{bmatrix} S_x & 0 & 0 & X_0 \\ 0 & S_y & 0 & Y_0 \\ 0 & 0 & S_z & Z_0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

1

Rotation

- Rotation:
  - to rotate a point about another arbitrary point in space requires three transformations:

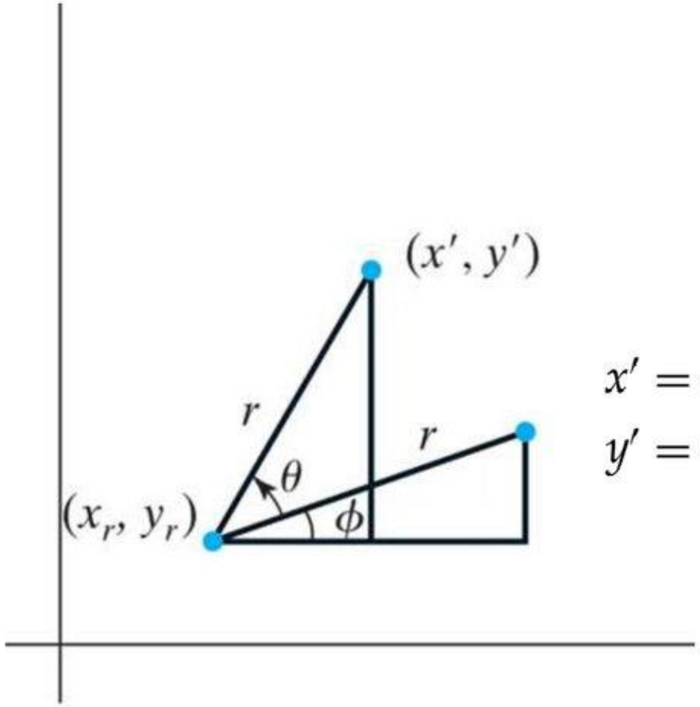


Rotation of a point about each of the co-ordinate axes

1

Rotation about a general pivot point

Figure 7-5 Rotating a point from position (x, y) to position (x', y') through an angle θ about rotation point (x_r, y_r).



$$x' = x_r + (x - x_r)\cos\theta - (y - y_r)\sin\theta$$
$$y' = y_r + (x - x_r)\sin\theta + (y - y_r)\cos\theta$$

Imaging Geometry: Basic transformations in imaging

- Rotation:
  - to rotate a point about the z-coordinate axis by an angle theta is achieved by using the following transformation:

$$R_\theta = \begin{bmatrix} \cos\theta & \sin\theta & 0 & 0 \\ -\sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

1

Imaging Geometry:
Basic transformations in imaging

- Rotation about x-axis:
  - to rotate a point about the x-coordinate axis by an angle alpha is achieved by using the following transformation:

$$R_\alpha = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\alpha & \sin\alpha & 0 \\ 0 & -\sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Imaging Geometry: Basic transformations in imaging

- Rotation about Y-axis:
  - to rotate a point about the y-coordinate axis by an angle Beta is achieved by using the following transformation:

$$R_\beta = \begin{bmatrix} \cos\beta & 0 & -\sin\beta & 0 \\ 0 & 1 & 0 & 0 \\ \sin\beta & 0 & \cos\beta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

1

Intensity Transformations

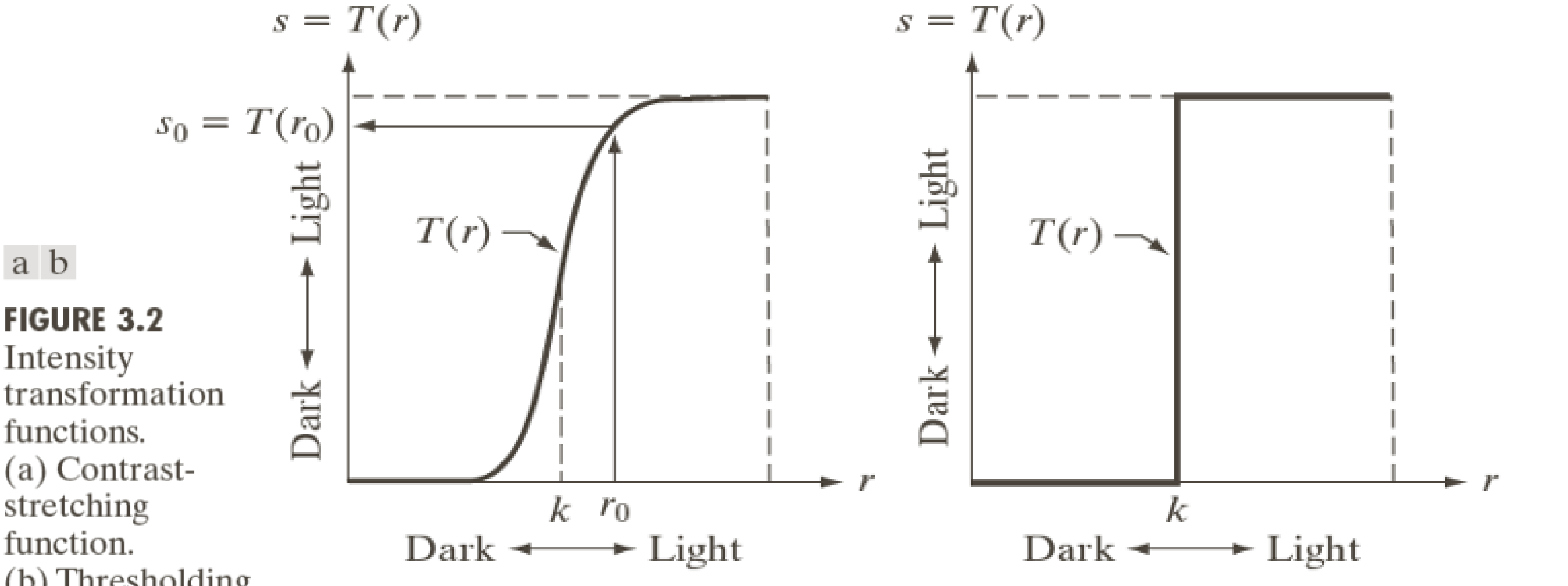


FIGURE 3.2 Intensity transformation functions. (a) Contrast-stretching function. (b) Thresholding function.

Intensity Transformations

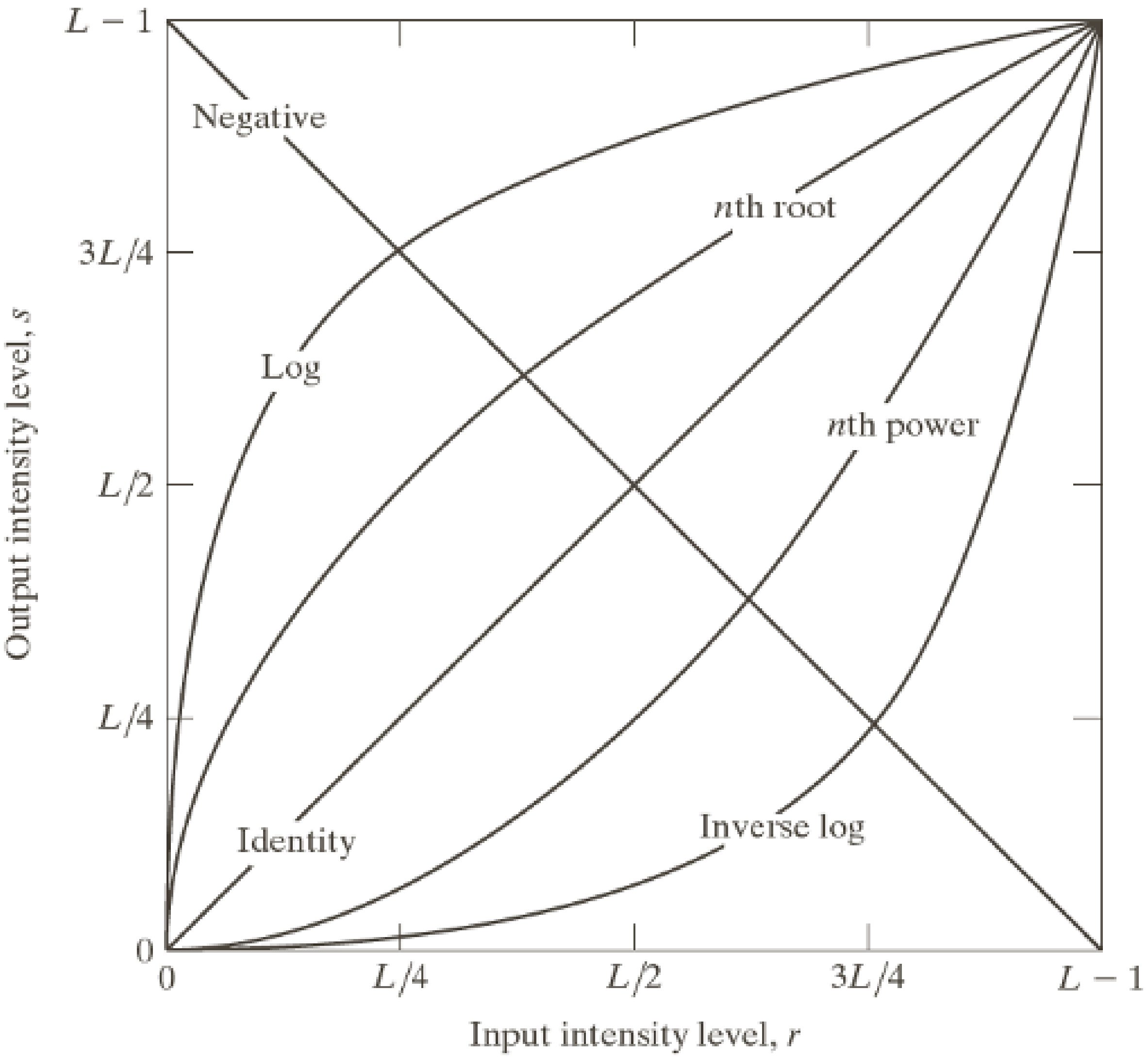


FIGURE 3.3 Some basic intensity transformation functions. All curves were scaled to fit in the range shown.

Intensity Transformations

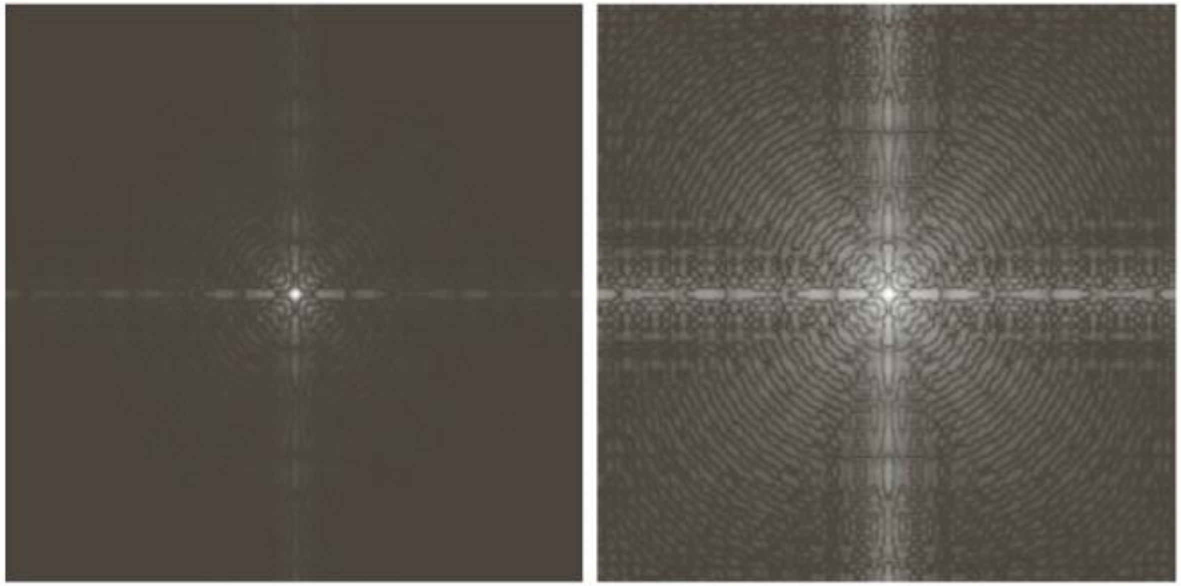
- Image Negatives

  s= L-1-r

- Log Transformations

  s= clog(1+r) , where r>=0;

- Maps narrow range of intensity values to wider range of low intensity values and the opposite is true for higher intensity.

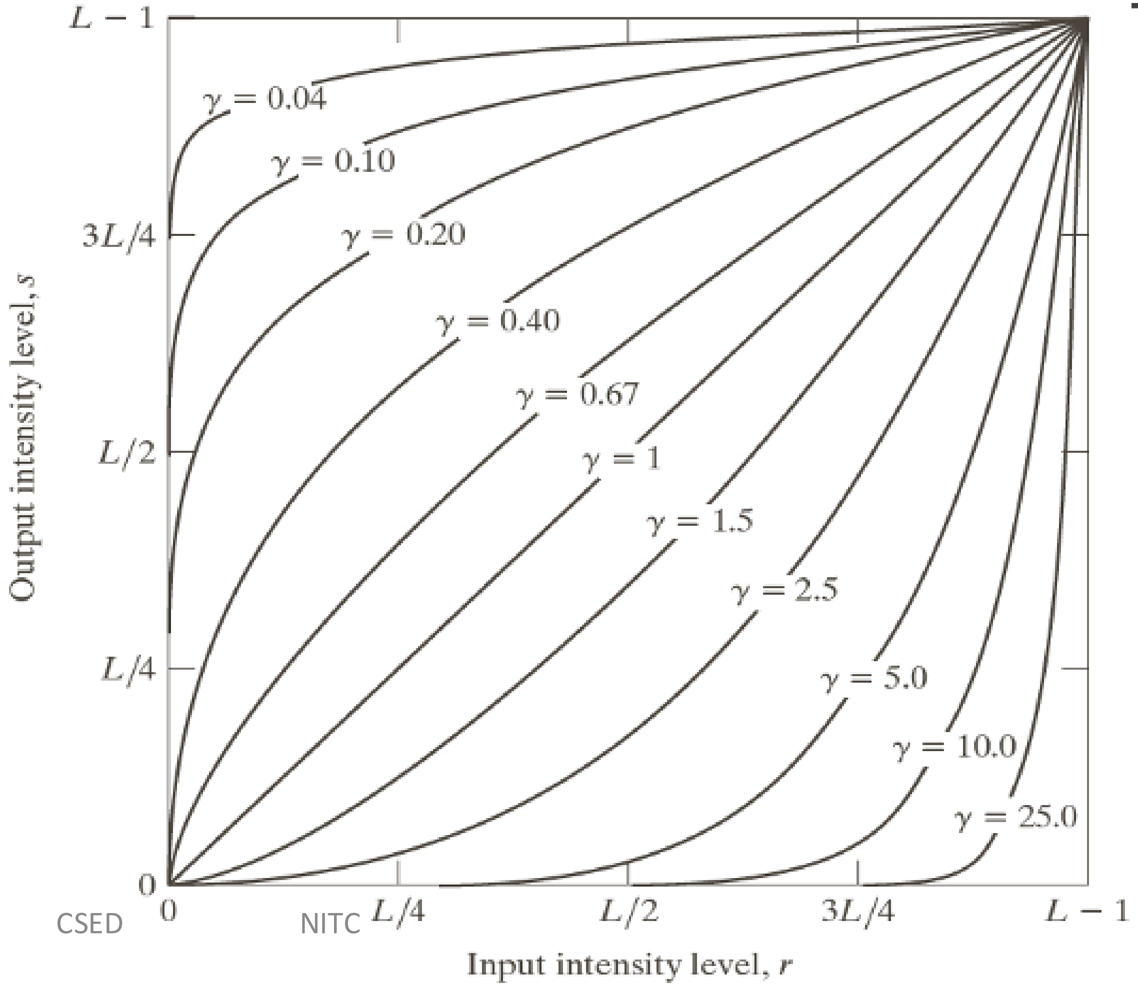


a b

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

Power-law Transformation

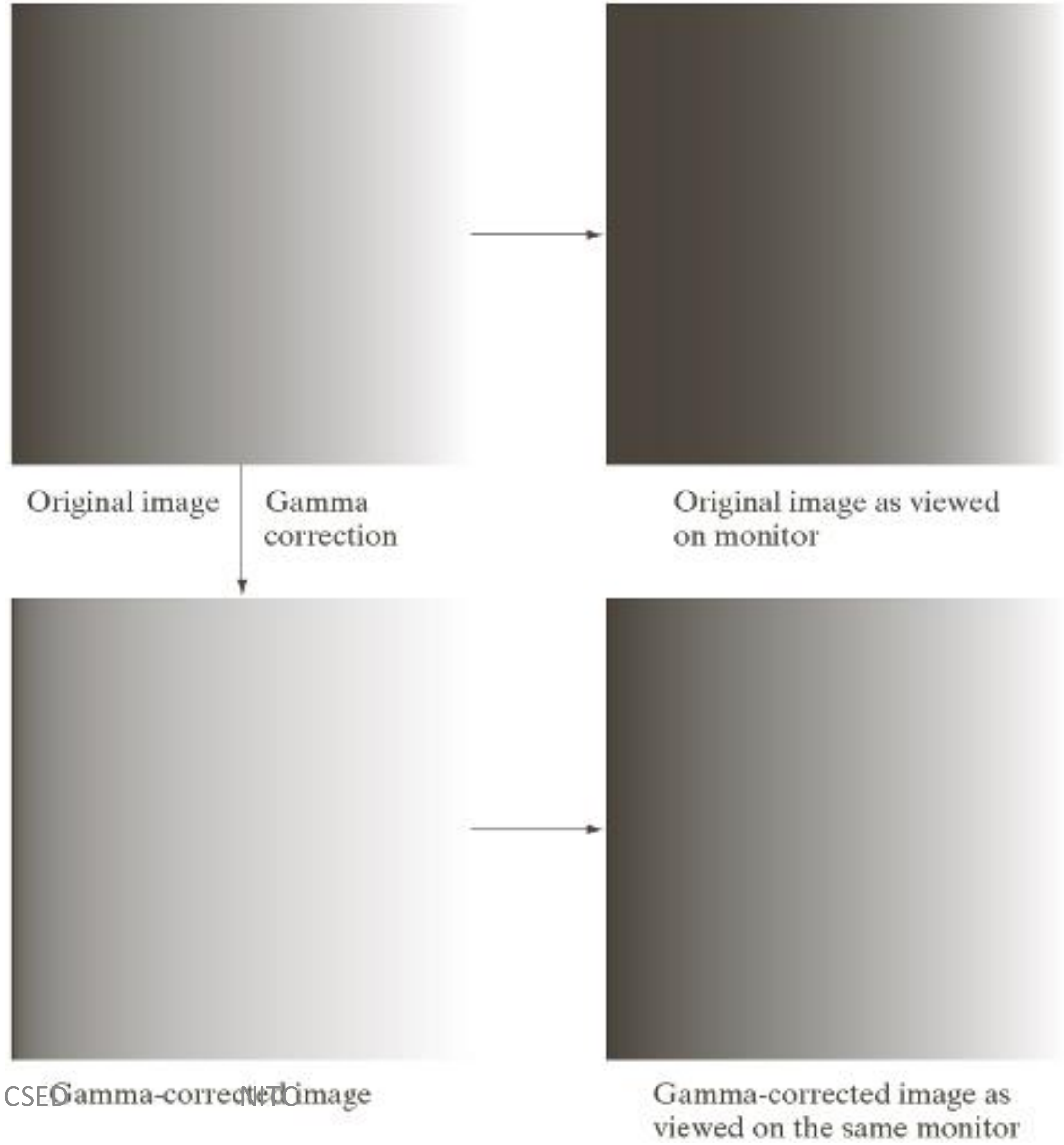
FIGURE 3.6 Plots of the equation $s = cr^{\gamma}$ for various values of γ ($c = 1$ in all cases). All curves were scaled to fit in the range shown.



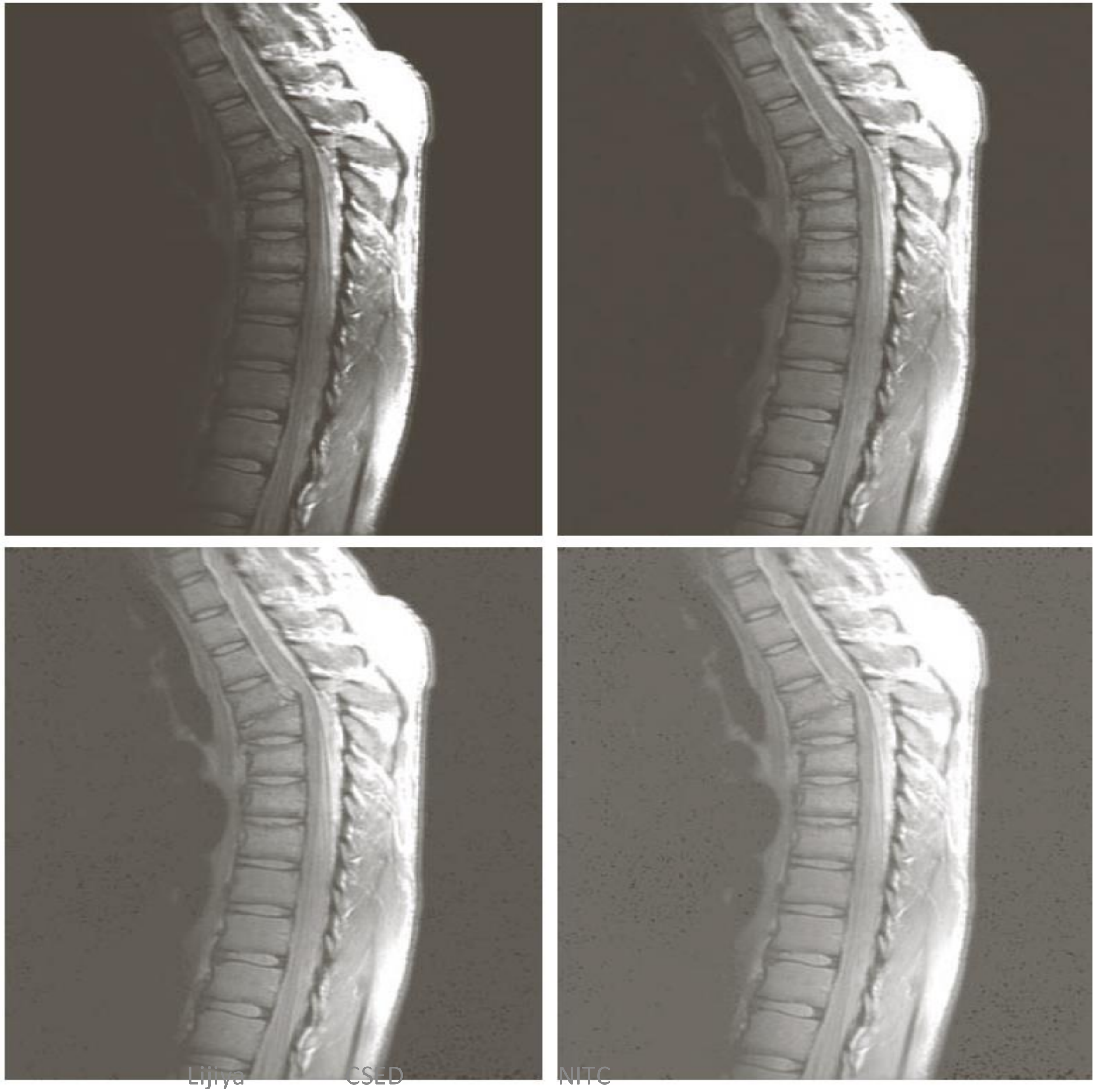
Gamma Correction

a b
c d

FIGURE 3.7
(a) Intensity ramp image. (b) Image as viewed on a simulated monitor with a gamma of 2.5. (c) Gamma-corrected image. (d) Corrected image as viewed on the same monitor. Compare (d) and (a).



Contrast Enhancement



a	b
c	d

FIGURE 3.8 (a) Magnetic resonance image (MRI) of a fractured human spine. (b)–(d) Results of applying the transformation in Eq. (3.2-3) with $c = 1$ and $\gamma = 0.6, 0.4$, and 0.3, respectively. (Original image courtesy of Dr. David R. Pickens, Department of Radiology and Radiological Sciences, Vanderbilt University Medical Center.)

a b
c d

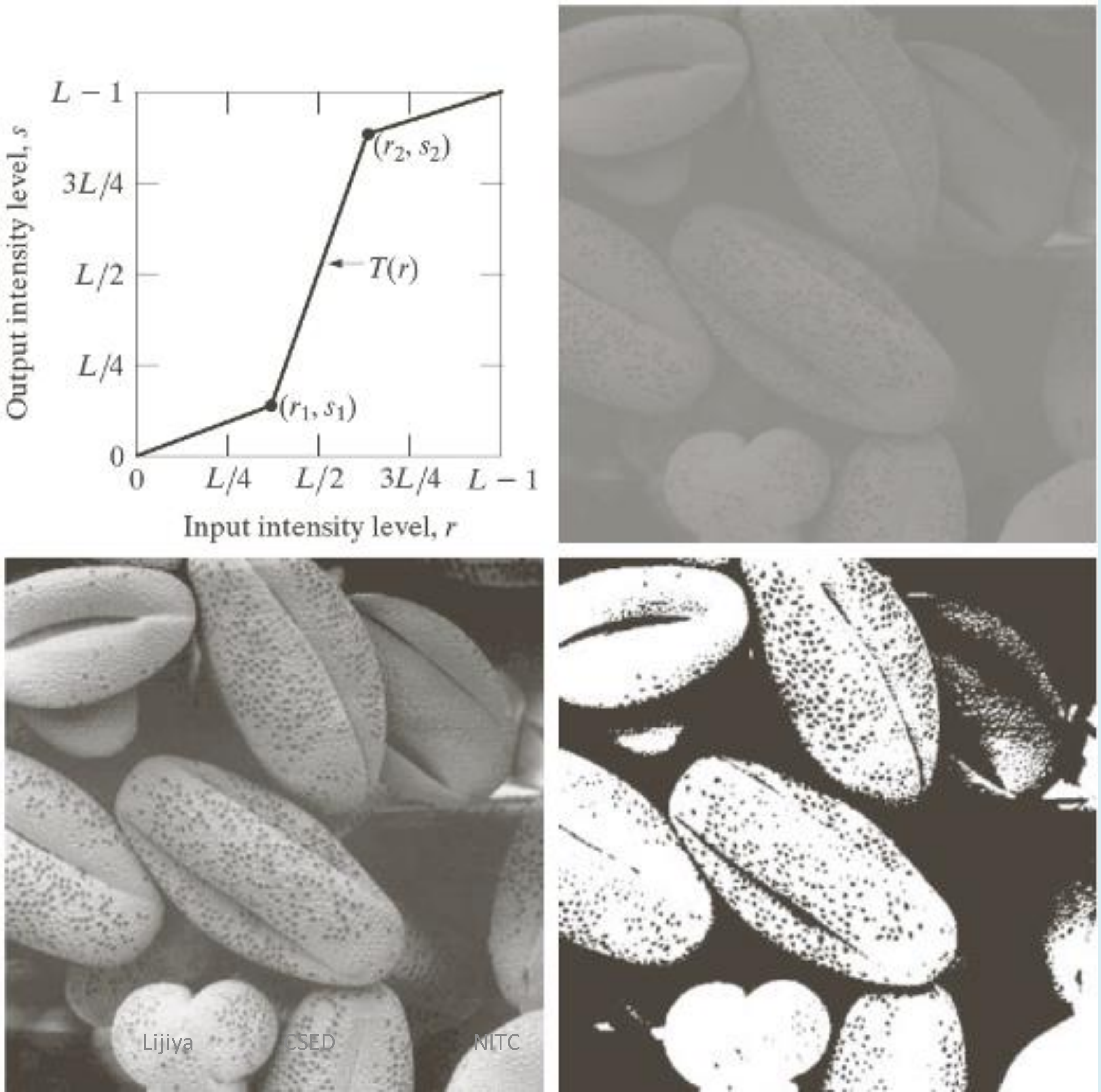
FIGURE 3.9
(a) Aerial image. (b)–(d) Results of applying the transformation in Eq. (3.2-3) with $c = 1$ and $\gamma = 3.0, 4.0,$ and 5.0, respectively. (Original image for this example courtesy of NASA.)

Contrast Streching

- ocation of points (r1,s1) and (r2,s2) controls the shape of the transformation function
- If r1=s1 & r2=s2, it is a linear function
- If r1=r2 & s1=0 & s2=L-1, it is a thresholding function.
- In general, r1<=r2 & s1<=s2 is assumed to get a single valued monotonically increasing function

a b
c d

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



a b

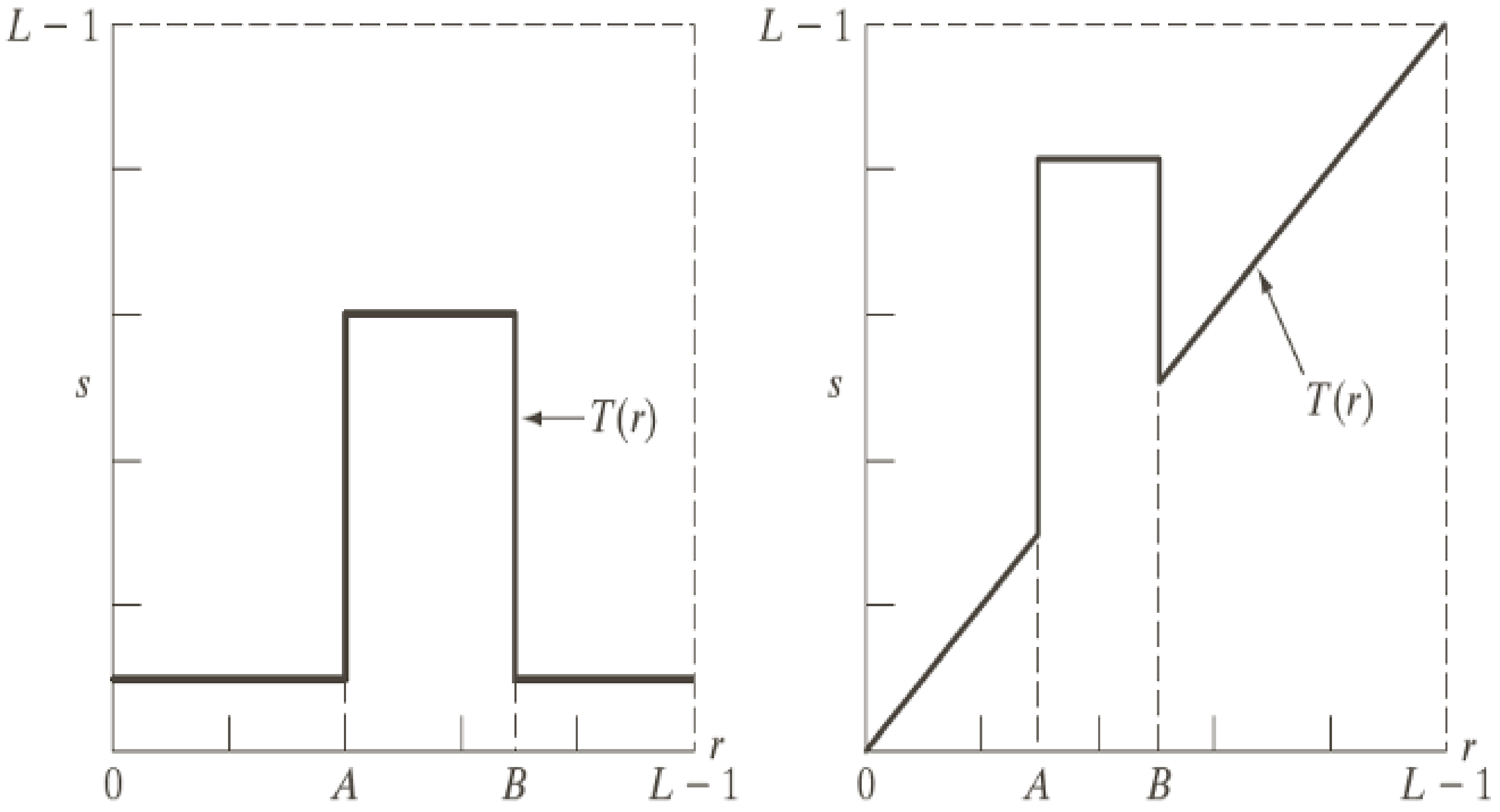


FIGURE 3.11 (a) This transformation highlights intensity range $[A, B]$ and reduces all other intensities to a lower level. (b) This transformation highlights range $[A, B]$ and preserves all other intensity levels.

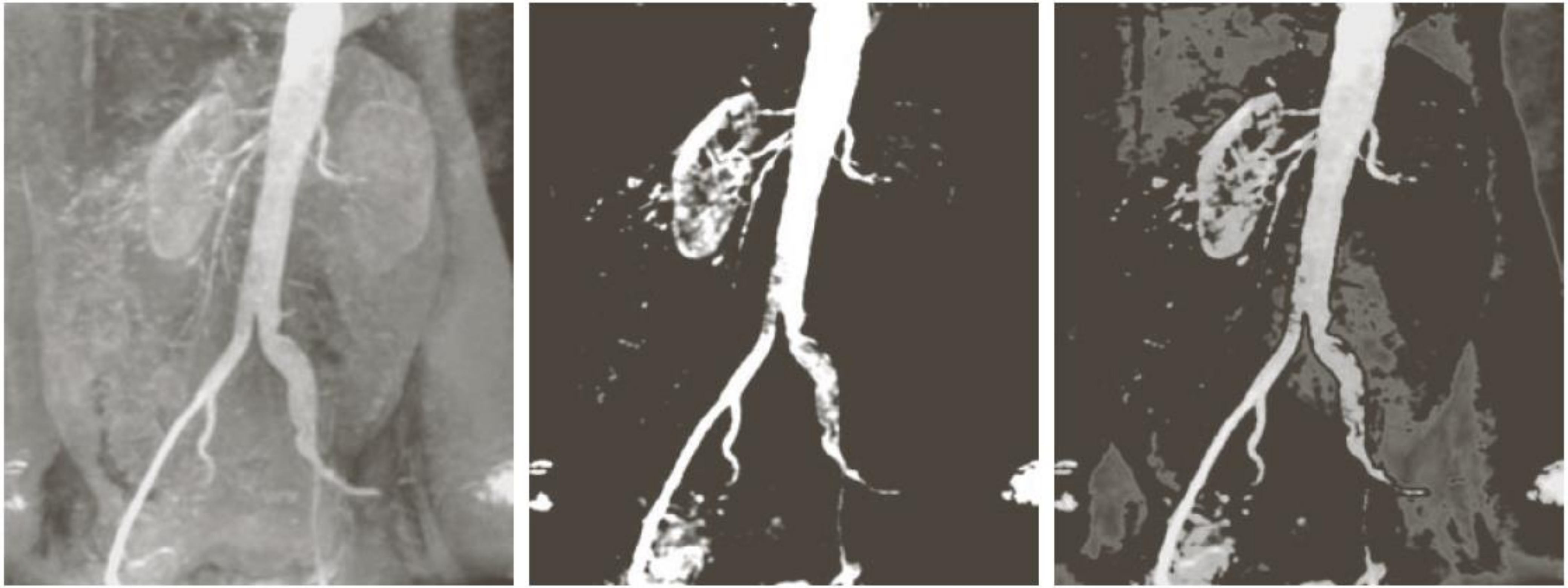
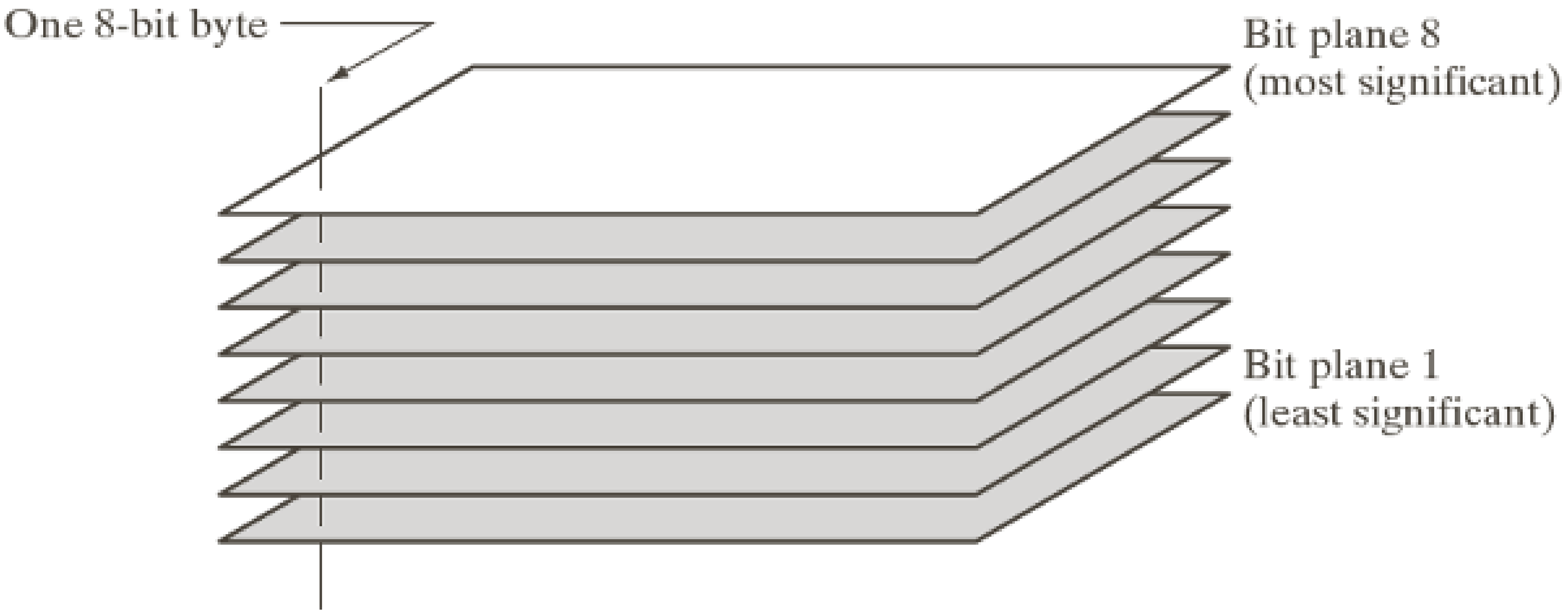


FIGURE 3.12 (a) Aortic angiogram. (b) Result of using a slicing transformation of the type illustrated in Fig. 3.11(a), with the range of intensities of interest selected in the upper end of the gray scale. (c) Result of using the transformation in Fig. 3.11(b), with the selected area set to black, so that grays in the area of the blood vessels and kidneys were preserved. (Original image courtesy of Dr. Thomas R. Gest, University of Michigan Medical School.)

Bit-Plane Slicing

- Bit-plane slicing : Pixels are digital numbers composed of bits. For example, the intensity of each pixel in an 256-level gray-scale image is composed of 8 bits (i.e., one byte).
- Instead of highlighting intensity-level ranges, we could highlight the contribution made to total image appearance by specific bits.



a b c
d e f
g h i

FIGURE 3.14 (a) An 8-bit gray-scale image of size 500 × 1192 pixels. (b) through (i) Bit planes 1 through 8, with bit plane 1 corresponding to the least significant bit. Each bit plane is a binary image.

a b c

FIGURE 3.15 Images reconstructed using (a) bit planes 8 and 7; (b) bit planes 8, 7, and 6; and (c) bit planes 8, 7, 6, and 5. Compare (c) with Fig. 3.14(a).

COLOUR

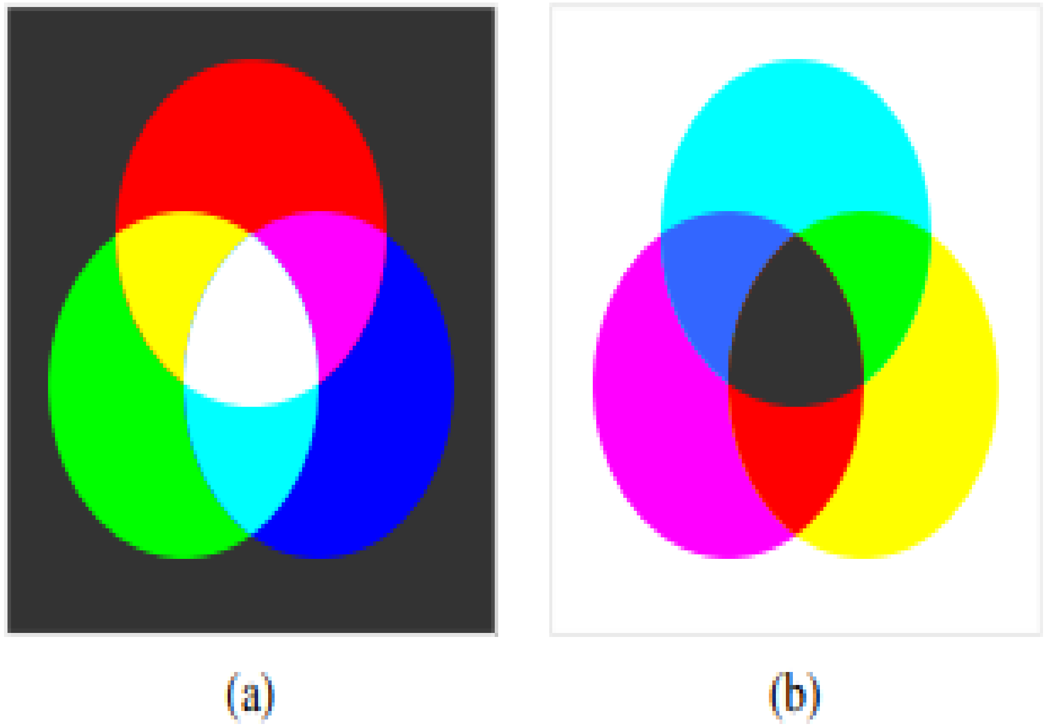
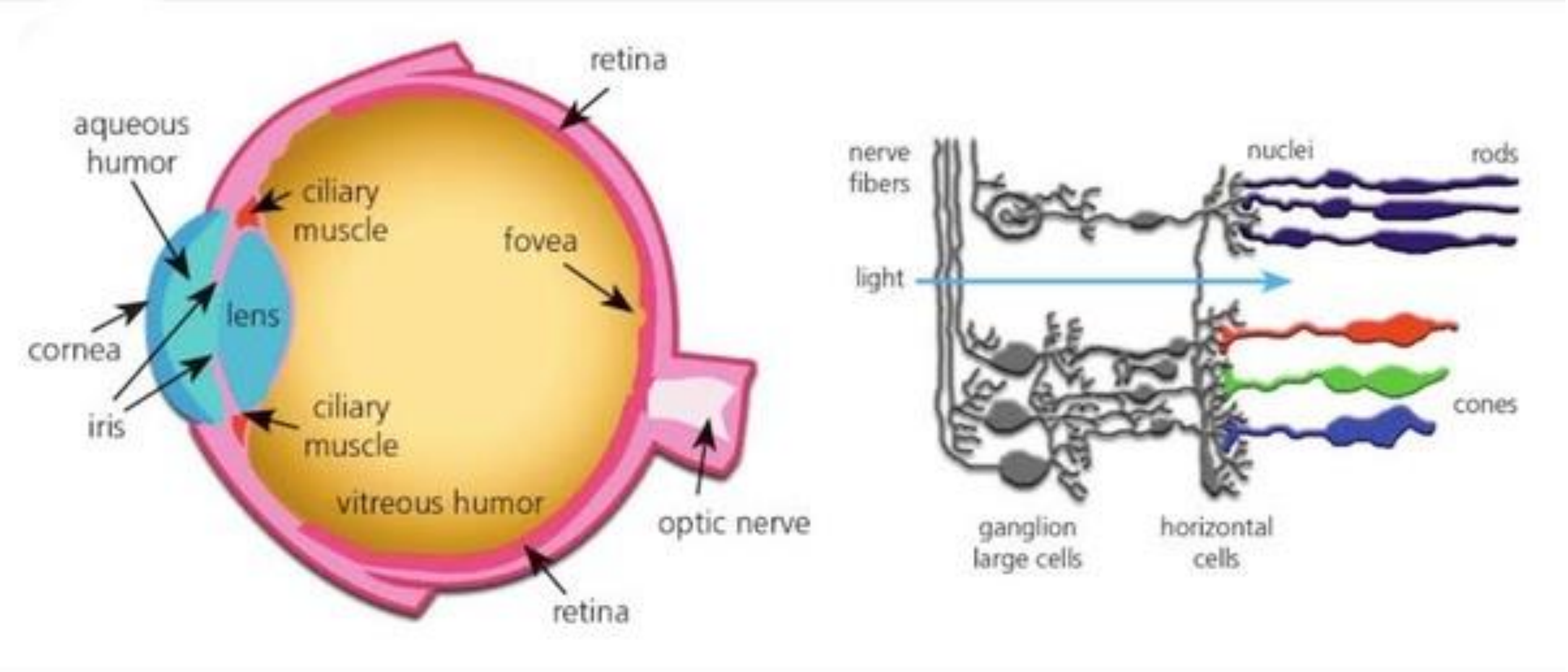


Figure 2.27 Primary and secondary colors: (a) additive colors red, green, and blue can be mixed to produce cyan, magenta, yellow, and white; (b) subtractive colors cyan, magenta, and yellow can be mixed to produce red, green, blue, and black.

Human Visual System



Cone cells, or **cones**, are photoreceptor cells in the retinas of vertebrate eyes including the human eye. They respond differently to light of different wavelengths, and are thus responsible for color vision, and function best in relatively bright light, as opposed to rod cells, which work better in dim light.

CIE RGB & XYZ

In the 1930s, the Commission Internationale d'Eclairage (CIE) standardized the RGB representation by performing such color matching experiments using the primary colors of red (700.0nm wavelength), green (546.1nm), and blue (435.8nm).

Figure 2.28 shows the results of performing these experiments with a standard observer, i.e., averaging perceptual results over a large number of subjects.

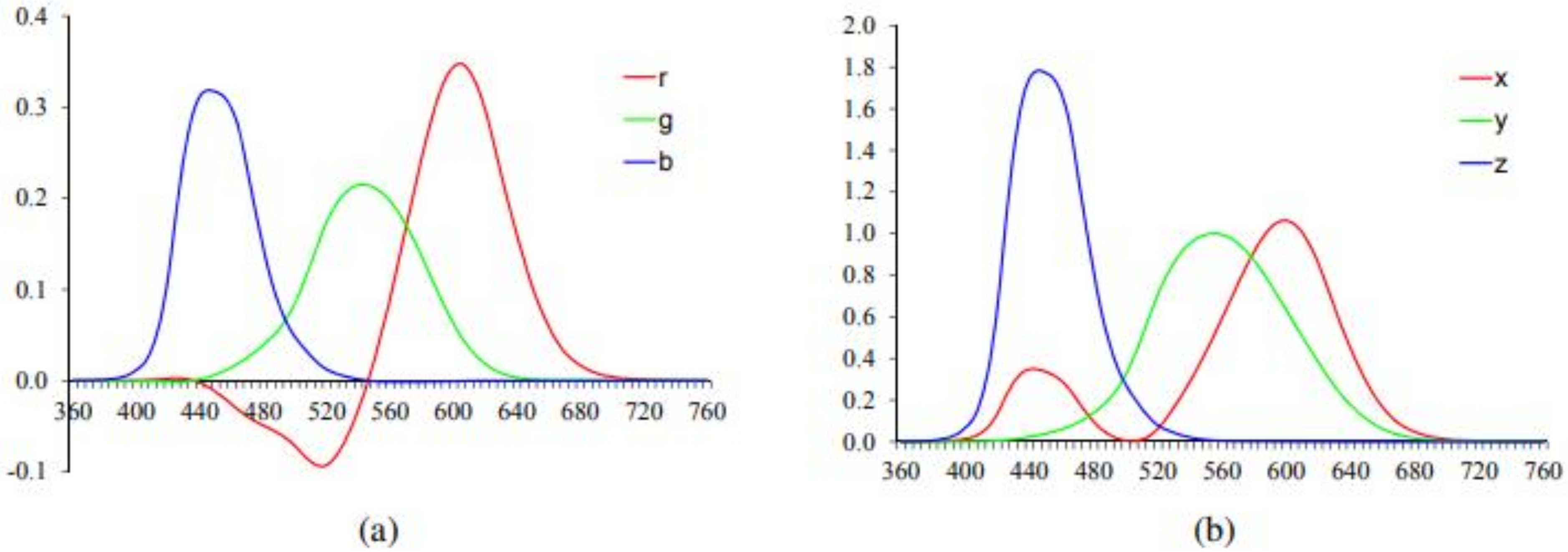


Figure 2.28 Standard CIE color matching functions: (a) $\bar{r}(\lambda)$, $\bar{g}(\lambda)$, $\bar{b}(\lambda)$ color spectra obtained from matching pure colors to the R=700.0nm, G=546.1nm, and B=435.8nm primaries; (b) $\bar{x}(\lambda)$, $\bar{y}(\lambda)$, $\bar{z}(\lambda)$ color matching functions, which are linear combinations of the $(\bar{r}(\lambda), \bar{g}(\lambda), \bar{b}(\lambda))$ spectra.

Because of the problem associated with mixing negative light, the CIE also developed a new color space called XYZ, which contains all of the pure spectral colors within its positive octant. (It also maps the Y axis to the *luminance*, i.e., perceived relative brightness, and maps pure white to a diagonal (equal-valued) vector.) The transformation from RGB to XYZ is given by

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \frac{1}{0.17697} \begin{bmatrix} 0.49 & 0.31 & 0.20 \\ 0.17697 & 0.81240 & 0.01063 \\ 0.00 & 0.01 & 0.99 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}. \tag{2.103}$$

[redacted]

While the official definition of the CIE XYZ standard has the matrix normalized so that the Y value corresponding to pure red is 1, a more commonly used form is to omit the leading fraction, so that the second row adds up to one, i.e., the RGB triplet $(1, 1, 1)$ maps to a Y value of 1. Linearly blending the $(\bar{r}(\lambda), \bar{g}(\lambda), \bar{b}(\lambda))$ curves in Figure 2.28a according to (2.103), we obtain the resulting $(\bar{x}(\lambda), \bar{y}(\lambda), \bar{z}(\lambda))$ curves shown in Figure 2.28b. Notice how all three spectra (color matching functions) now have only positive values and how the $\bar{y}(\lambda)$ curve matches that of the luminance perceived by humans.

If we divide the XYZ values by the sum of X+Y+Z, we obtain the *chromaticity coordinates*

$$x = \frac{X}{X+Y+Z}, \quad y = \frac{Y}{X+Y+Z}, \quad z = \frac{Z}{X+Y+Z}, \tag{2.104}$$

Luminance & Chrominance

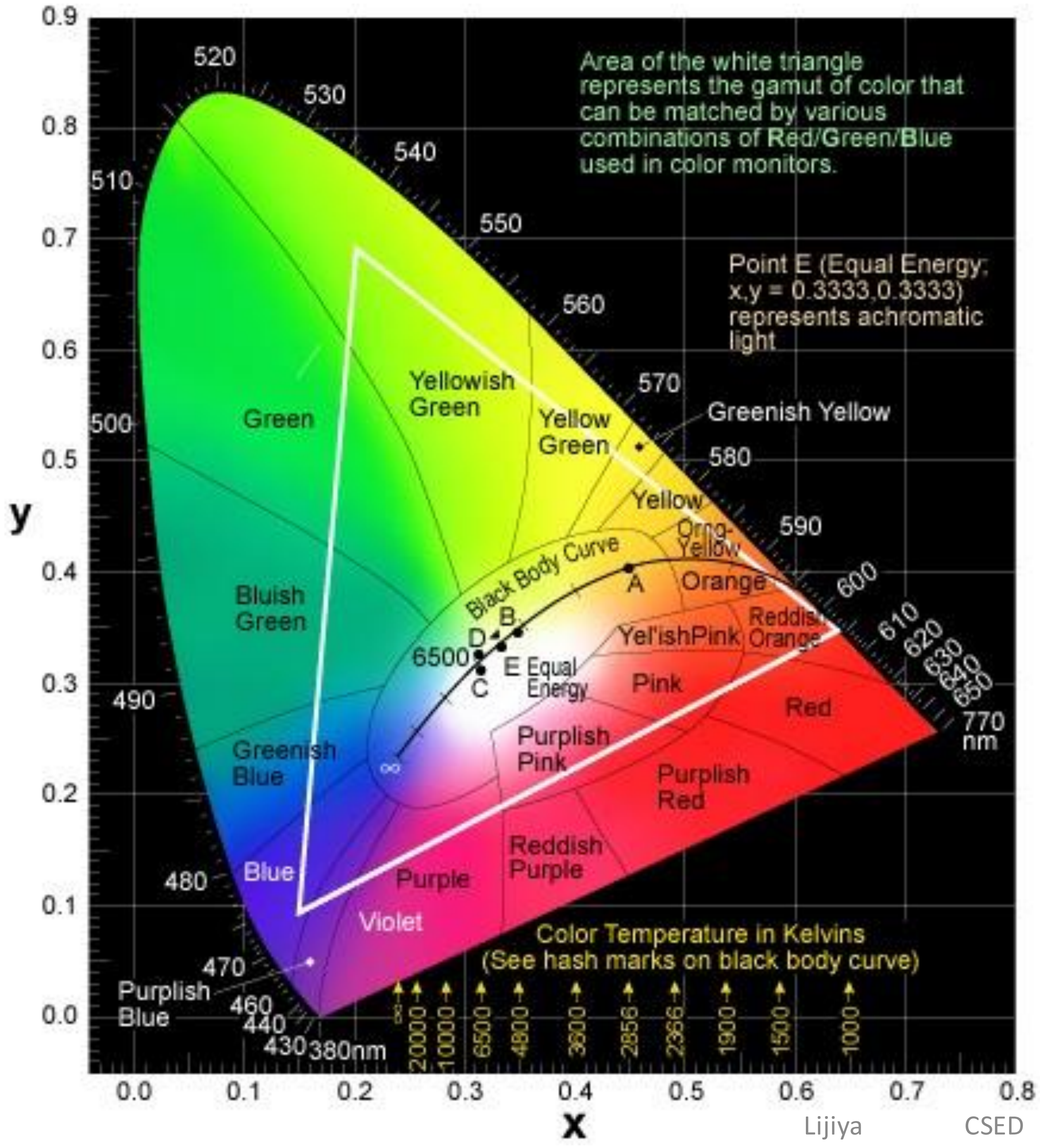
- Luminance:

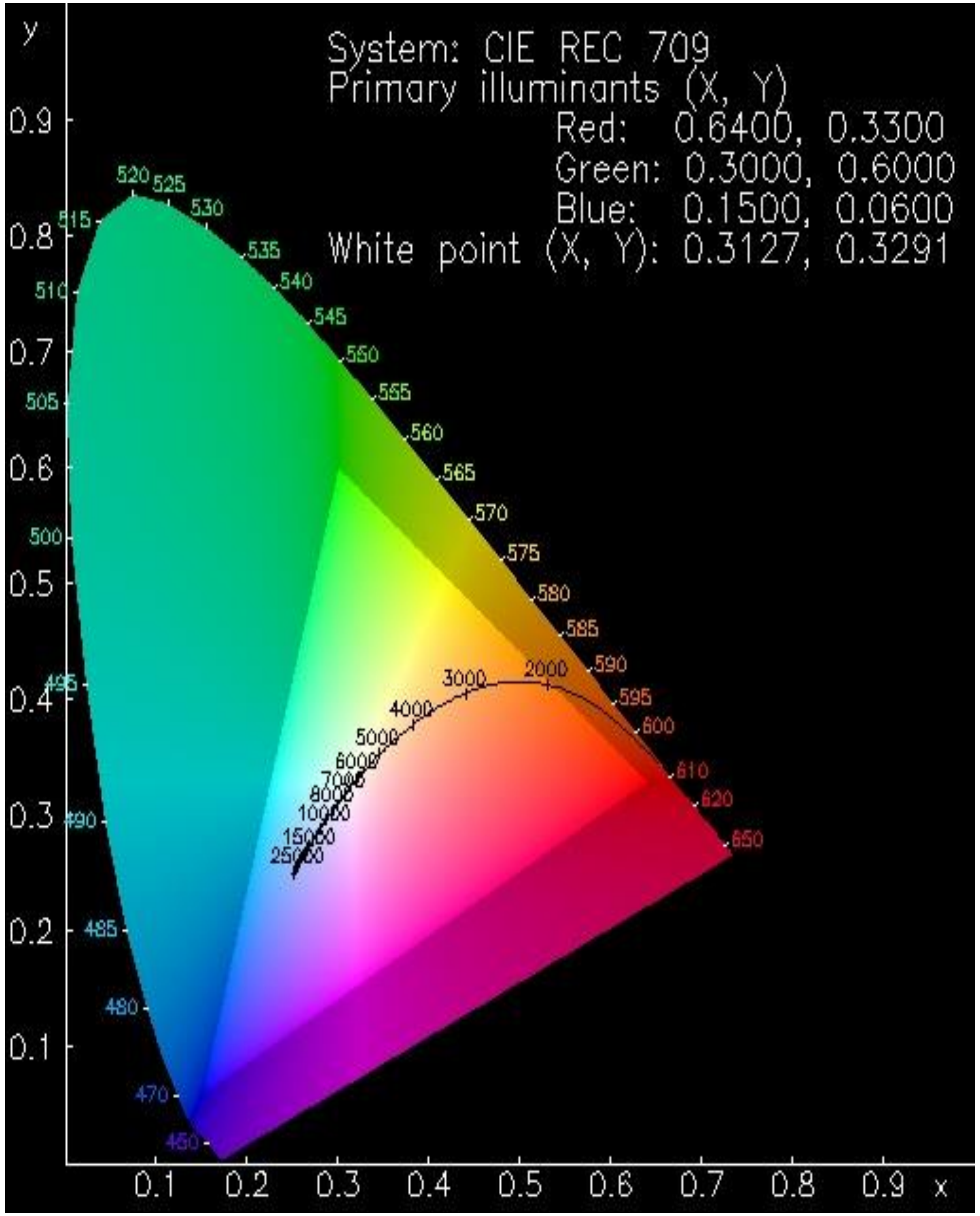
  Perceived Relative Brightness

- Chrominance:

  Colour Information

- **Chromaticity** is an objective specification of the quality of a color regardless of its luminance. Chromaticity consists of two parameter, hue(h) and saturation(or)colourfulness


Area of the white triangle represents the gamut of color that can be matched by various combinations of Red/Green/Blue used in color monitors.
Point E (Equal Energy; x,y = 0.3333,0.3333) represents achromatic light
Color Temperature in Kelvins
(See hash marks on black body curve)
Black Body Curve
Green
Yellowish Green
Yellow Green
Greenish Yellow
Yellow
Orng-Yellow
Orange
Reddish Orange
Red
Yel'ishPink
Pink
Purplish Pink
Purplish Red
Reddish Purple
Purple
Violet
Blue
Greenish Blue
Bluish Green
Purplish Blue
x
y


System: CIE REC 709
Primary illuminants (X, Y)
Red: 0.6400, 0.3300
Green: 0.3000, 0.6000
Blue: 0.1500, 0.0600
White point (X, Y): 0.3127, 0.3291