EE795: Computer Vision and Intelligent Systems

Spring 2012
TTh 17:30-18:45 WRI C225

Lecture 02
130124

http://www.ee.unlv.edu/~b1morris/ecg795/

Outline

- Basics
- Image Formation
- Image Processing

Intelligent Systems

- Intelligence
  - The capacity to acquire knowledge
  - The faculty of thought and reason
- System
  - A group of interacting, interrelated or interdependent elements forming a complex whole

- This class uses computer vision to give a system intelligence
- The systems should perceive, reason, learn, and act intelligently

Vision

- Signal to symbol transformation

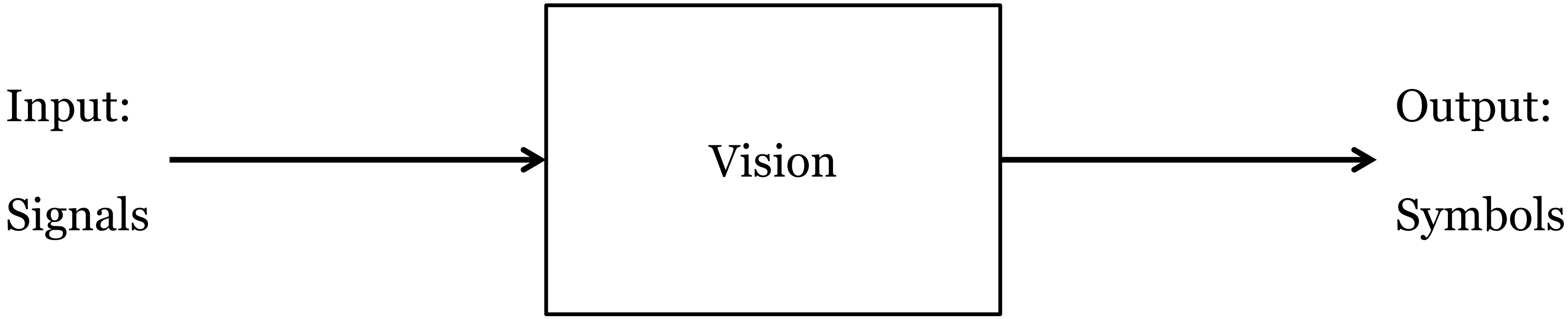


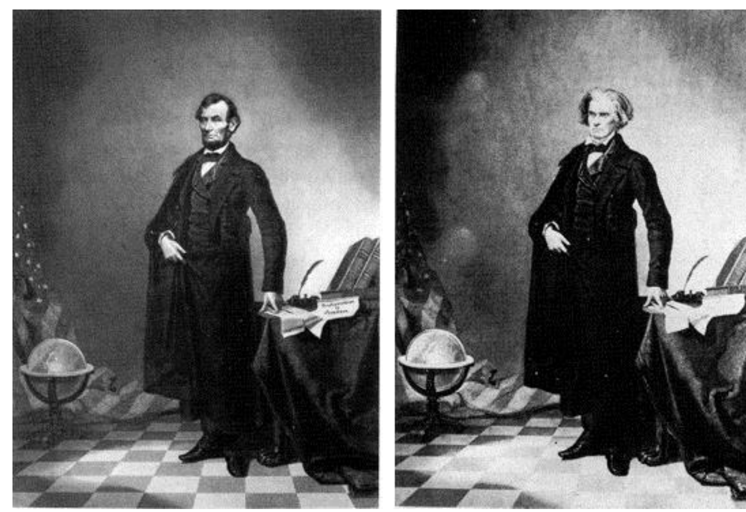
Image Processing

- Manipulation of images

Examples:

- “Photoshopping”
- Image enhancement
- Noise filtering
- Image compression

IP Examples

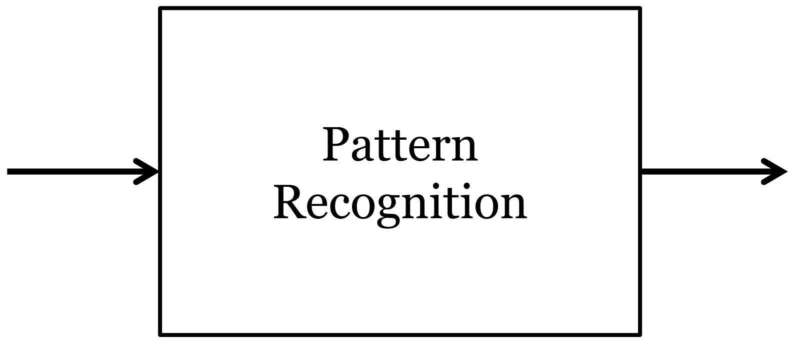


Pattern Recognition

- Assignment of a label to input value

Input:

Measurement
vector



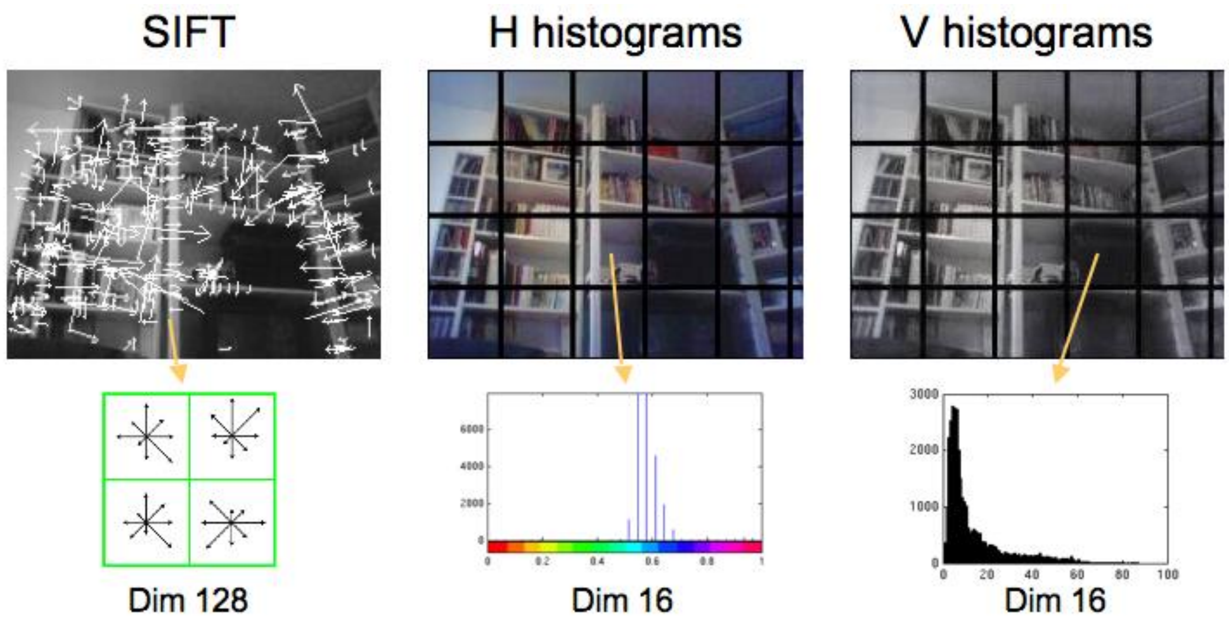
Output:

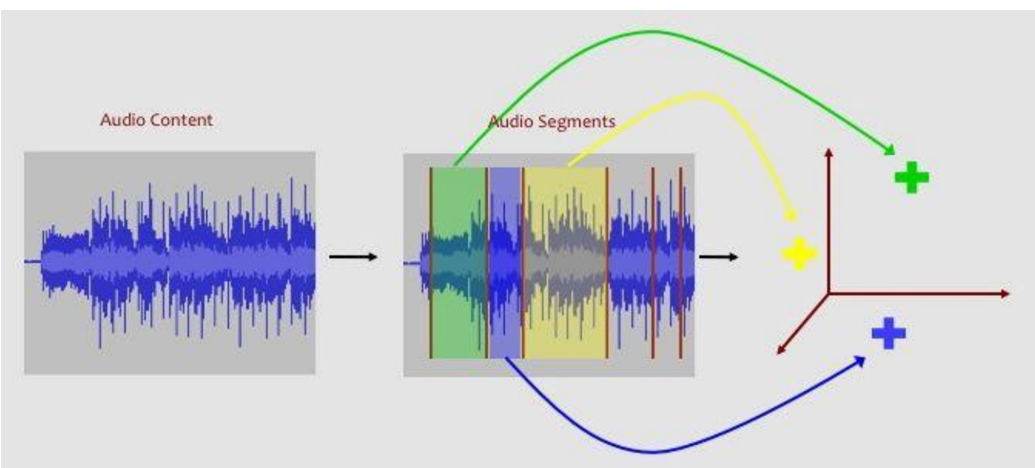
Label
("Classification")

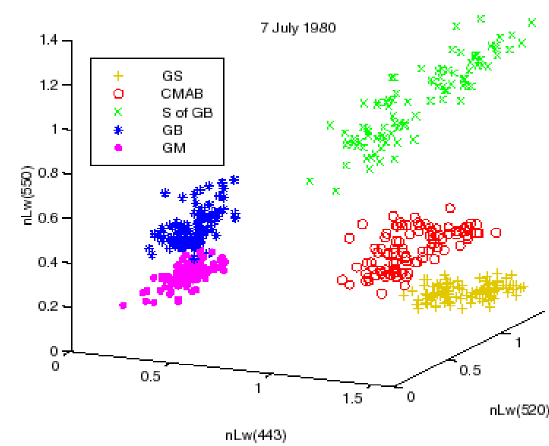
Examples:

- Classification (1/0)
- Regression (real valued)
- Labeling (multi label)

PR Examples


SIFT
H histograms
V histograms
Dim 128
Dim 16
Dim 16


Audio Content
Audio Segments

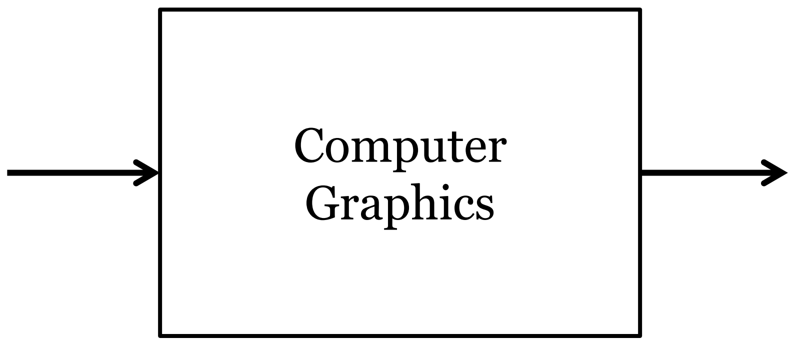

7 July 1980
GS
CMAB
S of GB
GB
GM
nLw(550)
nLw(443)
nLw(520)

Computer Graphics

- Create realistic images ("forward problem")

Input:

Mathematical model of objects and events



Output:

Images ("synthesized")

Examples:

- Simulation (flight, driving)
- Virtual tours
- Video games
- Movies

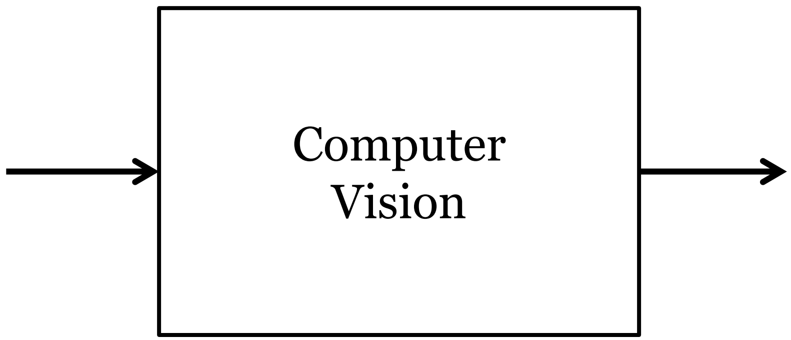
CG Examples

Computer Vision

- Interpretation and understanding of images

Input:

1. Image derived measurements
2. Models (prior knowledge)



Output:

Recognition of objects and events embedded in images and video

("Semantic" level classification)

Examples:

- Object recognition
- Face recognition
- Lane detection
- Activity analysis

Geometric Primitives

- Building blocks for description of 3D shapes

- Points
- Lines
- Planes
- Curves
- Surfaces
- Volumes

Points

- 2D – pixel coordinates in an image
  - $\boldsymbol{x} = \begin{bmatrix} x \\ y \end{bmatrix}$
- 3D – point in real world
  - $\boldsymbol{X} = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$

- Homogenous coordinates
  - Concatenate an extra term (usually 1) to point
  - $\bar{\boldsymbol{x}} = \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$ for 2D

2D Lines and 3D Planes

- 2D lines
  - $\tilde{l} = (a, b, c)$
  - Normalized line
  - $l = (\hat{n}_x, \hat{n}_y, d) = (\hat{\boldsymbol{n}}, d)$, $\|\hat{\boldsymbol{n}}\| = 1$
    - $\hat{\boldsymbol{n}}$ - normal vector to line
    - d – distance to origin
- Line equation
  - $\bar{x} \cdot \tilde{l} = ax + by + c = 0$
- 3D plane is extension of 2D line
  - $\tilde{m} = (a, b, c, d)$

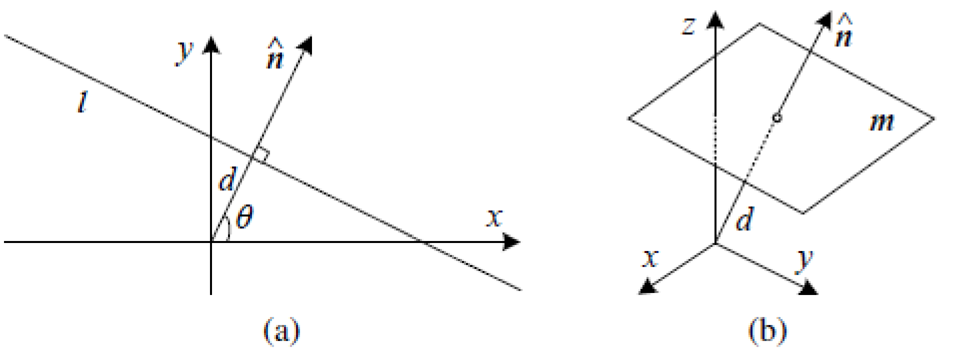


Figure 2.2 (a) 2D line equation and (b) 3D plane equation, expressed in terms of the normal $\hat{n}$ and distance to the origin d.

2D Transformations

- Translation
  - $x' = [I \quad t]\bar{x}$
  - I – identity matrix
- Rotation + translation
  - $x' = [R \quad t]\bar{x}$
  - $\begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix}$

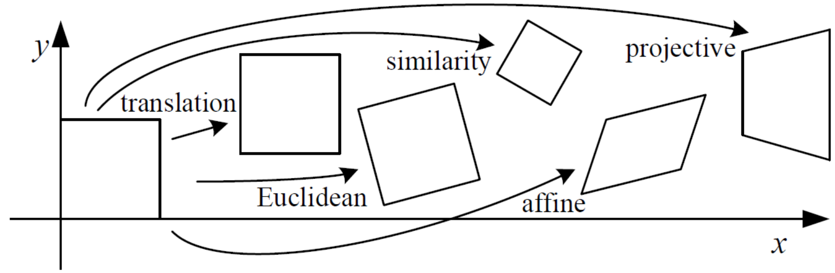


Figure 2.4 Basic set of 2D planar transformations.

Transformation	Matrix	# DoF	Preserves	Icon
translation	$\begin{bmatrix} \boldsymbol{I} \mid \boldsymbol{t} \end{bmatrix}_{2\times 3}$	2	orientation	
rigid (Euclidean)	$\begin{bmatrix} \boldsymbol{R} \mid \boldsymbol{t} \end{bmatrix}_{2\times 3}$	3	lengths	
similarity	$\begin{bmatrix} s\boldsymbol{R} \mid \boldsymbol{t} \end{bmatrix}_{2\times 3}$	4	angles	
affine	$\begin{bmatrix} \boldsymbol{A} \end{bmatrix}_{2\times 3}$	6	parallelism	
projective	$\begin{bmatrix} \tilde{\boldsymbol{H}} \end{bmatrix}_{3\times 3}$	8	straight lines	

Image Formation

- Incoming light energy is focused and collected onto an image plane

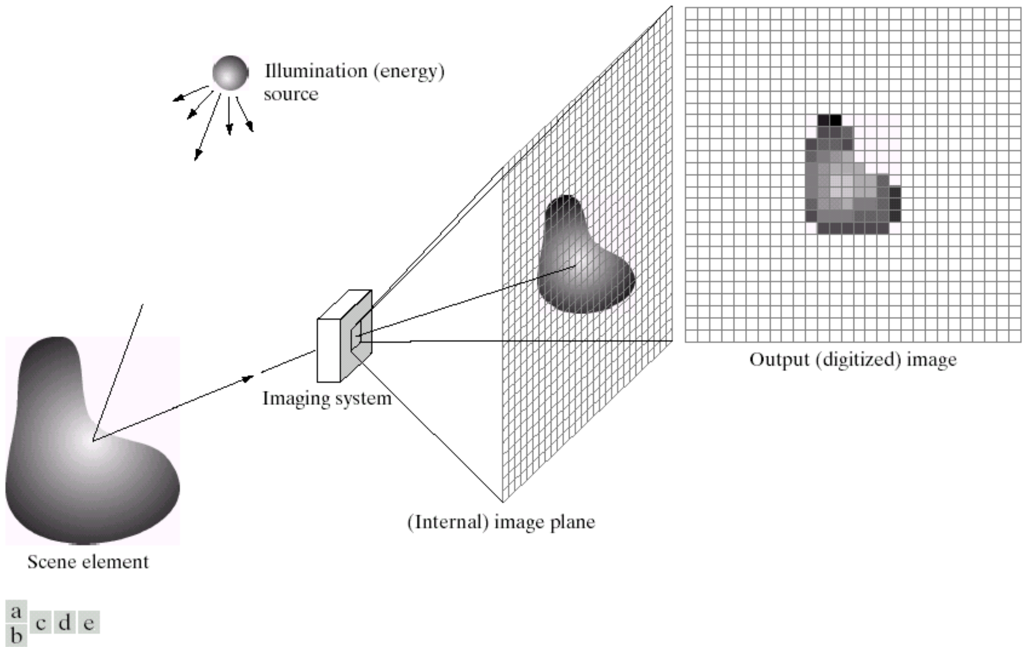
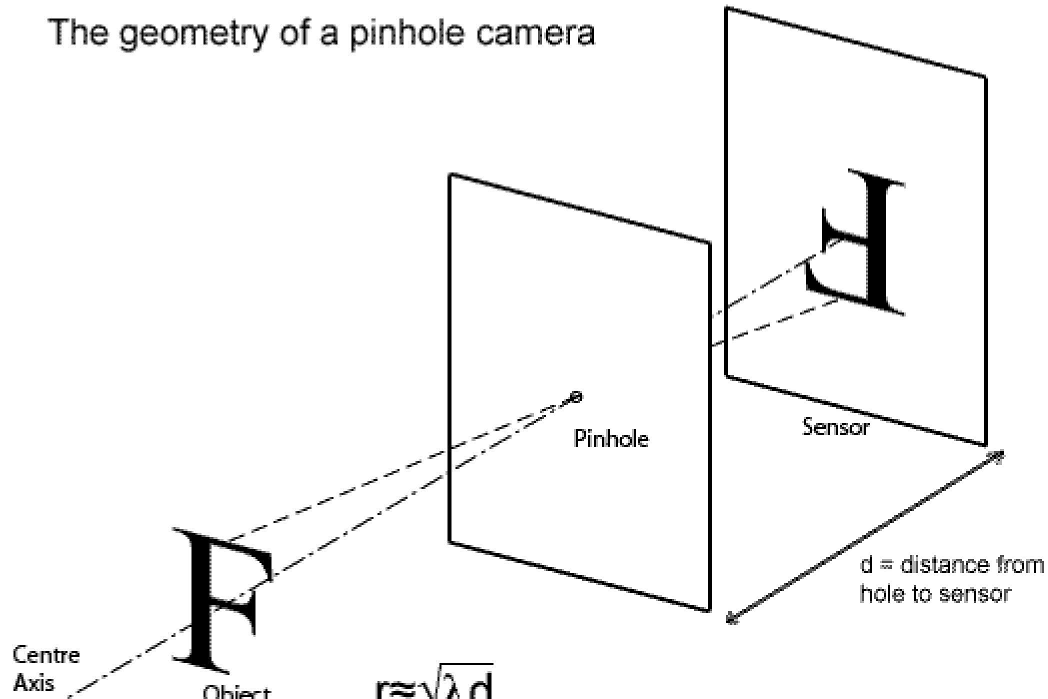


FIGURE 2.15 An example of the digital image acquisition process. (a) Energy ("illumination") source. (b) An element of a scene. (c) Imaging system. (d) Projection of the scene onto the image plane. (e) Digitized image.

Pinhole Camera Model

- Simple idealized model for image formation
  - Gives mathematical relationship between 3D world coordinates and 2D image plane coordinates



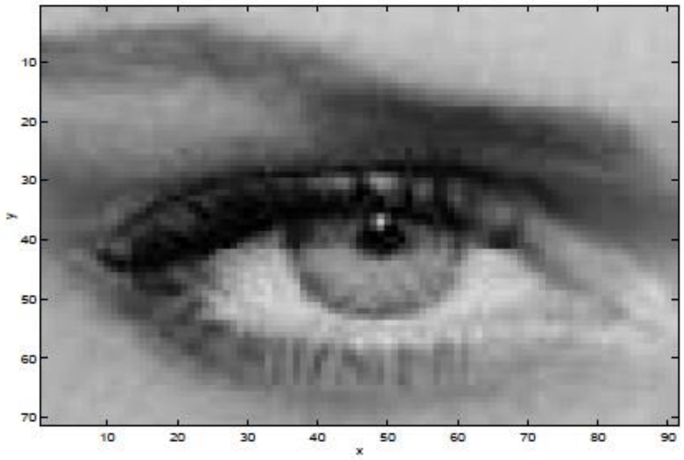
www.northlight-images.co.uk/article_pages/Canon_1ds_pinhole.html

Image as Function

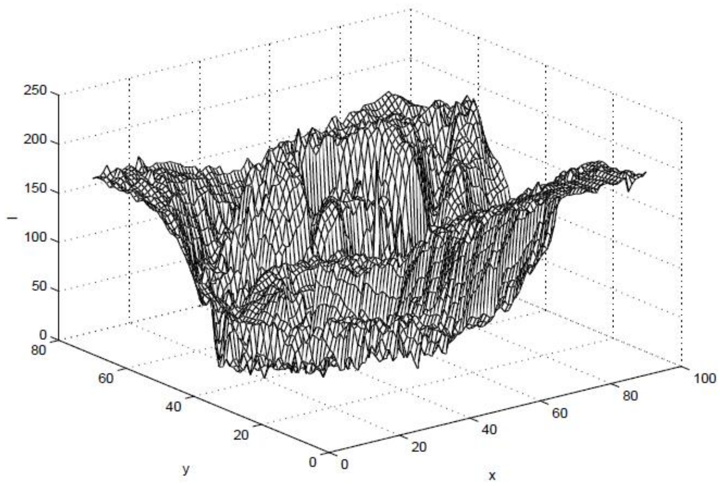
- Think of an image as a function, f, that maps from R^2 to R
  - $0 < f(x, y) < \infty$ is the intensity at a point (x, y)
- In reality, an image is defined over a rectangle with a finite range of values
  - $f: [a, b] \times [c, d] \rightarrow [0,1]$
- Computationally, [0,1] range is convenient but usually we have an 8-bit quantized representation
  - $0 < f(x, y) < 255$
- Color image is just three separate functions pasted together
  - $f(x, y) = [r(x, y); g(x, y); b(x, y)]$

Image as Function

- Multiple equivalent representations
- Image



- Surface



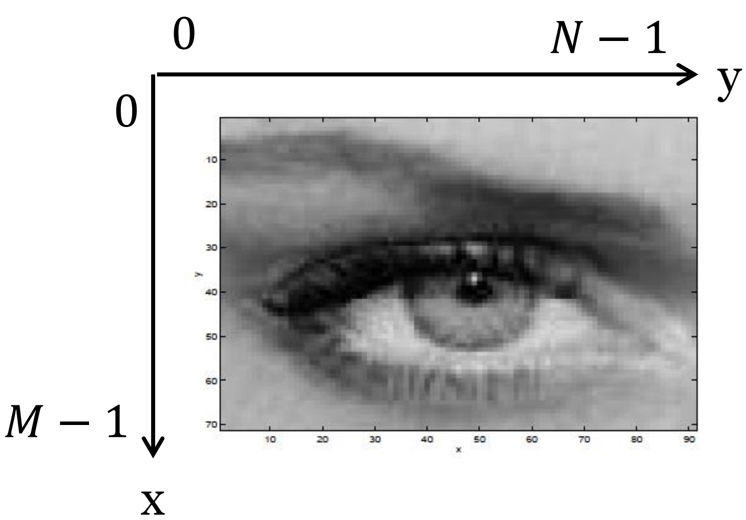
- Matrix

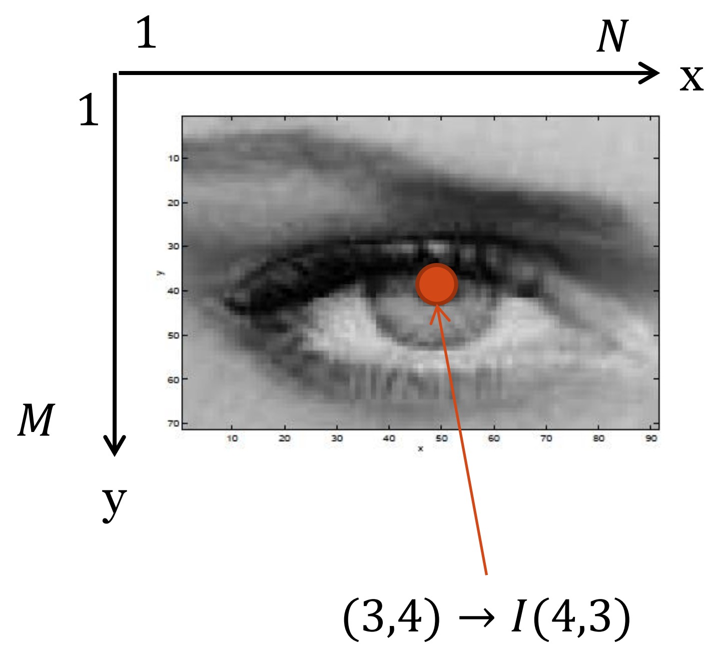
188	186	188	187	168	130	101	99	110	113	112	107	117	140	153	153	156	158	156	153
189	189	188	181	163	135	109	104	113	113	110	109	117	134	147	152	156	163	160	156
190	190	188	176	159	139	115	106	114	123	114	111	119	130	141	154	165	160	156	151
190	188	188	175	158	139	114	103	113	126	112	113	127	133	137	151	165	156	152	145
191	185	189	177	158	138	110	99	112	119	107	115	137	140	135	144	157	163	158	150
193	183	178	164	148	134	118	112	119	117	118	106	122	139	140	152	154	160	155	147
185	181	178	165	149	135	121	116	124	120	122	109	123	139	141	154	156	159	154	147
175	176	176	163	145	131	120	118	125	123	125	112	124	139	142	155	158	158	155	148
170	170	172	159	137	123	116	114	119	122	126	113	123	137	141	156	158	159	157	150
171	171	173	157	131	119	116	113	114	118	125	113	122	135	140	155	156	160	160	152
174	175	176	156	128	120	121	118	113	112	123	114	122	135	141	155	155	158	159	152
176	174	174	151	123	119	126	121	112	108	122	115	123	137	143	156	155	152	155	150
175	169	168	144	117	117	127	122	109	106	122	116	125	139	145	158	156	147	152	148
179	179	180	155	127	121	118	109	107	113	125	133	130	129	139	153	161	148	155	157
176	183	181	153	122	115	113	106	105	109	123	132	131	131	140	151	157	149	156	159
180	181	177	147	115	110	111	107	107	105	120	132	133	133	141	150	154	148	155	157
181	174	170	141	113	111	115	112	113	105	119	130	132	134	144	153	156	148	152	151
180	172	168	140	114	114	118	113	112	107	119	128	130	134	146	157	162	153	153	148
186	176	171	142	114	114	116	110	108	104	116	125	128	134	148	161	165	159	157	149
185	178	171	138	109	110	114	110	109	97	110	121	127	136	150	160	163	158	156	150

Matrix Notation

- Mathematical
- Notation starts with $f(0,0)$

- Matlab
- Notation starts with I(1,1)
  - No zero indexing





Sampling and Quantization

- Sampling gives fixed grid of image
- Quantization gives the number of output levels L

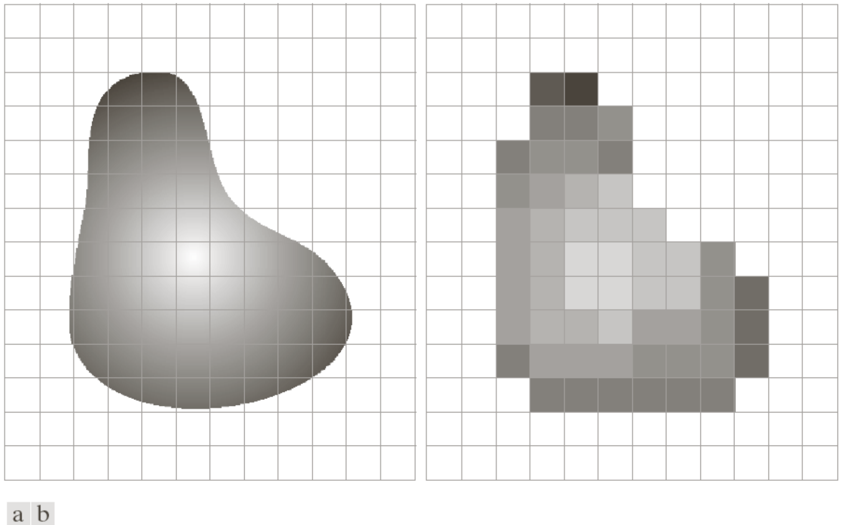
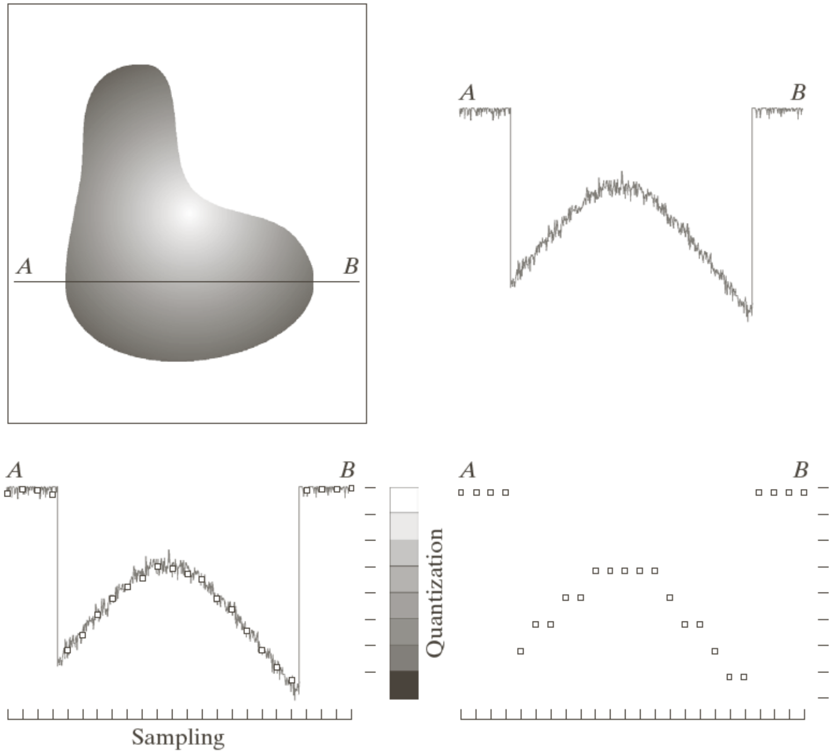


FIGURE 2.17 (a) Continuous image projected onto a sensor array. (b) Result of image sampling and quantization.



Quantization

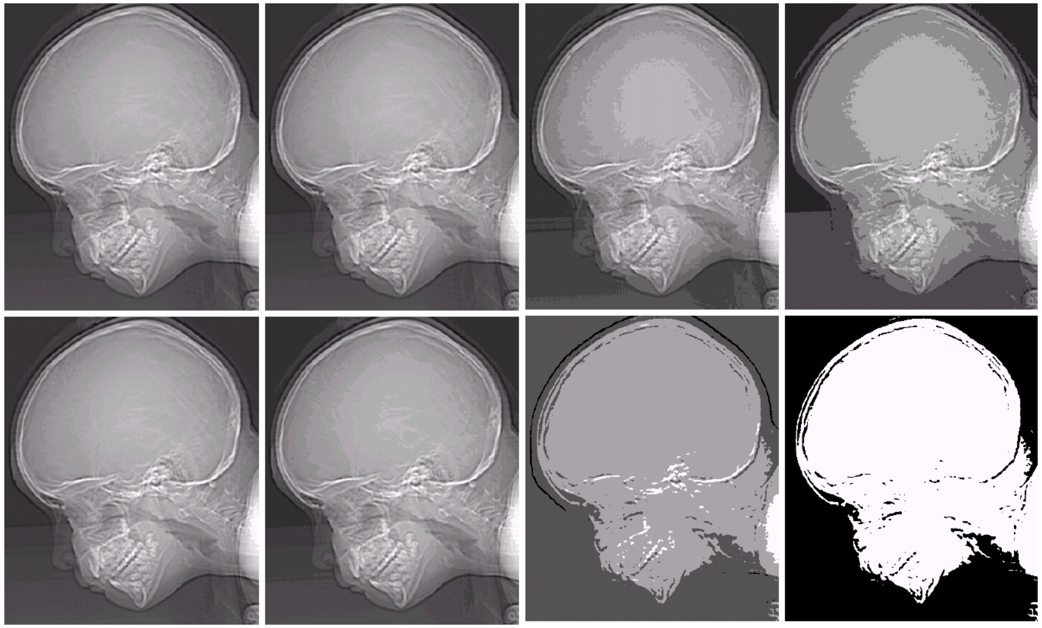
- L = number of output levels
- k = number of bits per pixel
- Output range of image
  - $[0, L-1] = [0, 2^k]$

- Image storage size
  - $b \ = \mathrm{M} \times N \times k$
  - Number of bits to store image with dimensions $\mathrm{M} \times N$

Resolution

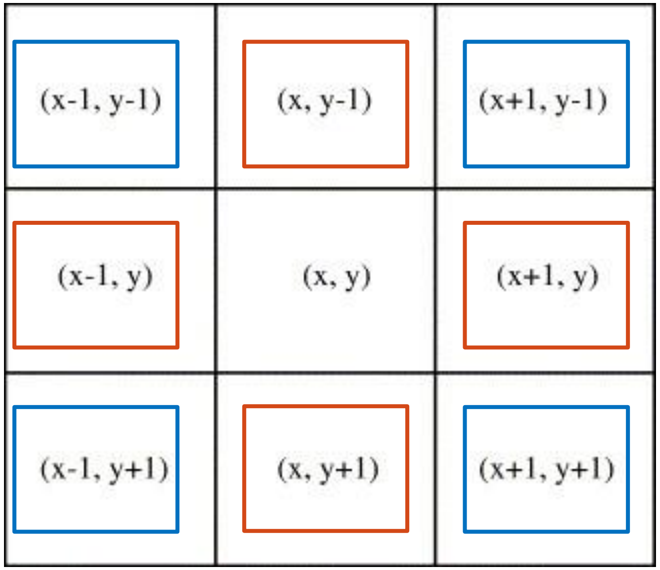
- Spatial resolution is the smallest discernible detail in an image
  - This is controlled by the sampling factor (the size $M \times N$ of the CMOS sensor)

- Gray-level resolution is the smallest discernible change in gray level
  - Based on number of bits for representation



Pixel Neighborhood

- The pixel neighborhood corresponds to nearby pixels

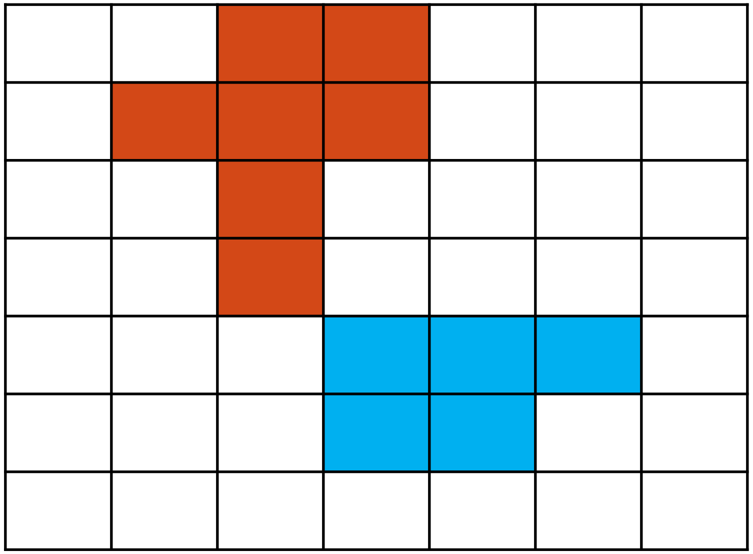


- 4-neighbors
  - Horizontal and vertical neighbors
- 8-neighbors
  - Include 4-neighbors and the diagonal pixels

Connectivity

- Path exists between pixels

- 4-connected



- 8-connected

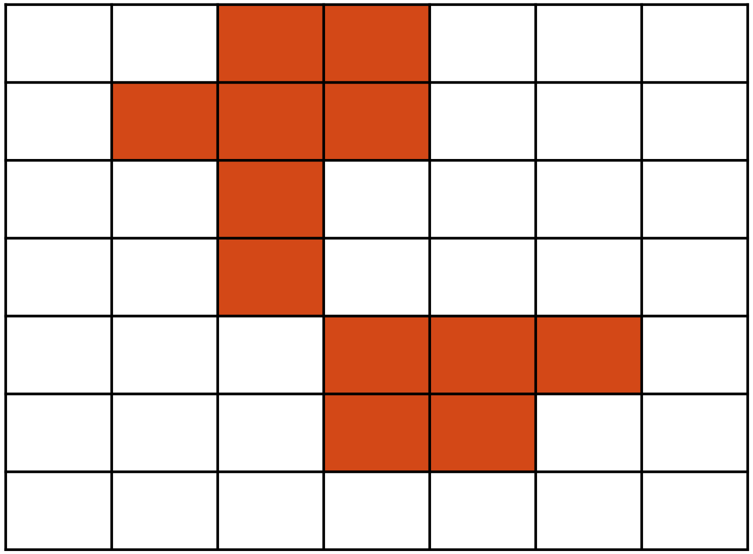


Image Processing

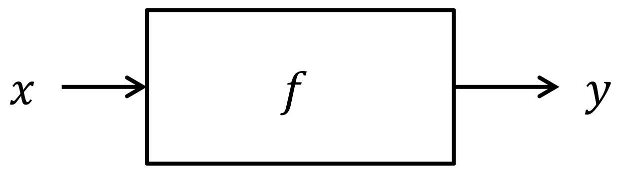
- Usually the first stage of computer vision applications
  - Pre-process an image to ensure it is in a suitable form for further analysis
- Typical operations include:
  - Exposure correction, color balancing, reduction in image noise, increasing sharpness, rotation of an image to straighten

- Digital Image Processing by Gonzalez and Woods is a great book to learn more

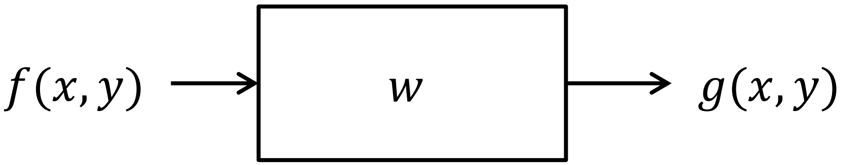
2D Signal Processing

- Image processing is an extension of signal processing to two independent variables
  - Input signal, output signal

- General system



- Image processing



Point Operators/Processes

- Output pixel value only depends on the corresponding input pixel value

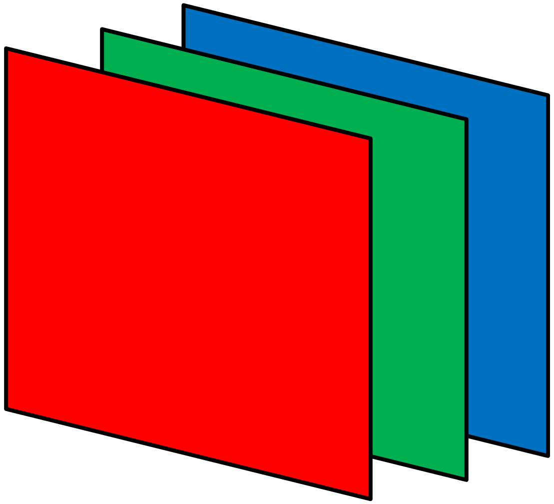
- Often times we will see operations like dividing one image by another
  - Matrix division is not defined
  - The operation is carried out between corresponding pixels in the two image
  - Element-by-element dot operation in Matlab
    - `>> I3 = I1./I2`
    - Where I1 and I2 are the same size

Pixel Transforms

- Gain and bias (Multiplication and addition of constant)
  - $g(x, y) = a(x, y)f(x, y) + b(x, y)$
  - a (gain) controls contrast
  - b (bias) controls brightness
    - Notice parameters can vary spatially (think gradients)
- Linear blend
  - $g(x) = (1 - \alpha)f_0(x) + \alpha f_1(x)$
  - We will see this used later for motion detection in video processing

Color Transforms

- Usually we think of a color image as three images concatenated together
  - Have a red, green, blue slice corresponding to the notion of primary colors



- Manipulations of these color channels may not correspond directly with desired perceptual response
  - Adding bias to all channels may actually change the apparent color instead of increasing brightness
- Need other representations of color for mathematical manipulation
- We will see more about color later

Compositing and Matting

- Techniques to remove an object and place it in a new scene
  - E.g. blue screen

- Matting – extracting an object from an original image
- Compositing – inserting object into another image (without visible artifacts)
- A fourth alpha channel is added to an RGB image
  - α describes the opacity (opposite of transparency) of a pixel
- Over operator
  - $C = (1 - \alpha)B + \alpha F$