

---

# Digital Image Processing (CSE405)

---

## Introduction

Md. Ferdous  
Lecturer, Dept. of CSE, BSMRSTU

---

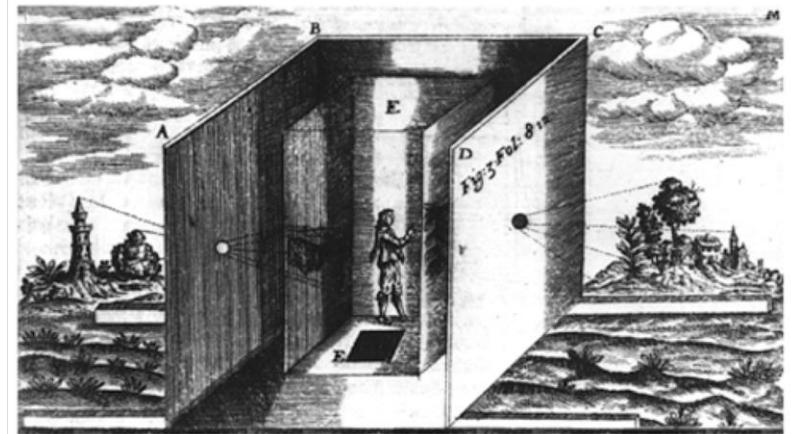
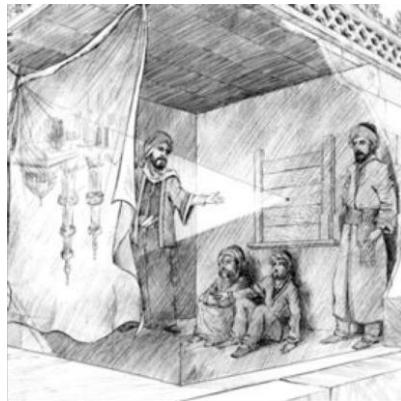
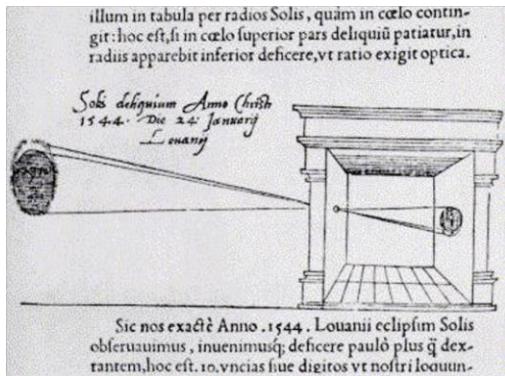
# Imaging



[Albrecht Dürer,  
1525]

- **Image:** a visual representation in form of a function  $f(x,y)$   
where  $f$  is related to the brightness (or color) at point  $(x,y)$
- Most images are defined over a rectangle
- Continuous in amplitude and space

# Imaging



Dark chamber with lenses [Kircher 1646]

- **Image:** a visual representation in form of a function  $f(x,y)$   
where  $f$  is related to the brightness (or color) at point  $(x,y)$
- Most images are defined over a rectangle
- Continuous in amplitude and space

# Analog Image

---

An image can be understood as a 2D light intensity function  $f(x,y)$  where:

- $x$  and  $y$  are spatial coordinates
- The value of  $f$  at any point  $(x, y)$  is proportional to the brightness or gray value of the image at that point

Cannot be stored as such on a digital computer.



# What is an image?

---

We can think of an **image** as a function,  $f$ , from  $\mathbb{R}^2$  to  $\mathbb{R}$ :

- $f(x, y)$  gives the **intensity** at position  $(x, y)$
- Realistically, we expect the image only to be defined over a rectangle, with a finite range:
  - $f: [a,b] \times [c,d] \rightarrow [0,1]$

A color image is just three functions pasted together.  
We can write this as a “vector-valued” function:

$$f(x, y) = \begin{bmatrix} r(x, y) \\ g(x, y) \\ b(x, y) \end{bmatrix}$$

---

# Digital Images and Pixels

- **Digital image:** discrete samples  $f[x,y]$  representing continuous image  $f(x,y)$
- Each element of the 2-d array  $f[x,y]$  is called a **pixel** or **pel** (from “picture element”)



200x200



100x100



50x50



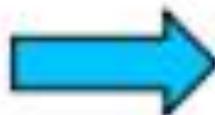
25x25

# Digital Image

---

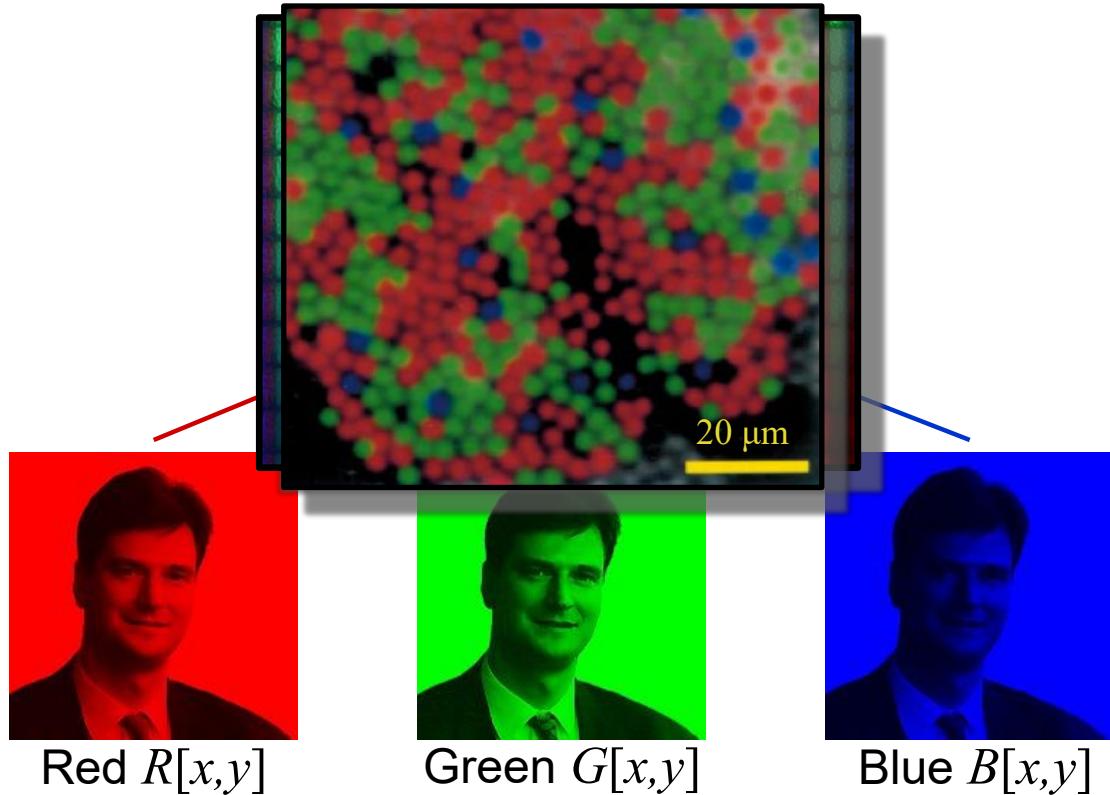
A digitized image is one in which:

- Spatial and grayscale values have been made discrete.
- Intensities measured across a regularly spaced grid in x and y directions are sampled to
  - 8 bits (256 values) per point for black and white,
  - 3x8 bits per point for color images.
- Stored as a 2D arrays of gray-level values. The array elements are called pixels and identified by their x, y coordinates.



192	149	136	126	113	149	151	154	156	163	169	159	151	
149	140	149	146	148	118	146	143	148	148	149	140	146	149
146	143	141	148	148	146	141	136	136	138	136	136	133	133
121	121	129	119	123	124	140	140	146	139	129	129	129	129
129	123	109	106	124	120	120	120	127	142	144	140	129	123
137	140	106	104	104	104	109	110	115	117	114	147	147	147
129	124	136	126	109	109	107	126	121	121	121	147	130	130
102	103	107	124	113	108	108	107	128	128	120	127	108	108
129	157	157	176	126	120	120	121	121	126	147	150	163	163
109	105	103	102	103	106	126	121	129	128	149	134	103	103
106	108	176	109	108	108	170	173	195	149	147	148	103	108
106	173	173	175	173	173	170	175	177	177	153	153	156	156
103	170	176	177	177	176	176	174	179	179	177	179	129	127
103	182	168	176	168	168	163	162	176	176	176	180	160	160

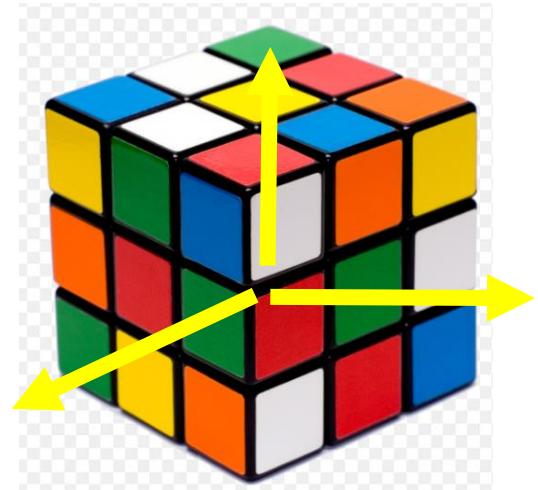
# Color Components



Monochrome image



$$R[x,y] = G[x,y] = B[x,y]$$

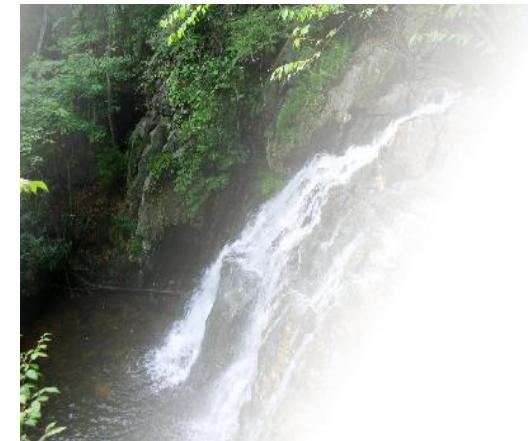


# What is a Digital Image? (cont...)

---

Common image formats include:

- 1 sample per point (B&W or Grayscale)
- 3 samples per point (Red, Green, and Blue)
- 4 samples per point (Red, Green, Blue, and “Alpha”, a.k.a. Opacity)

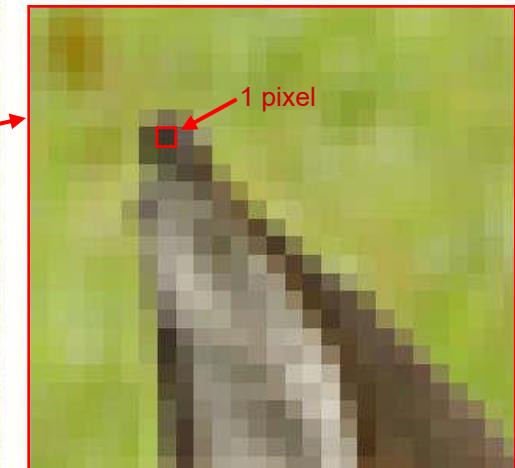
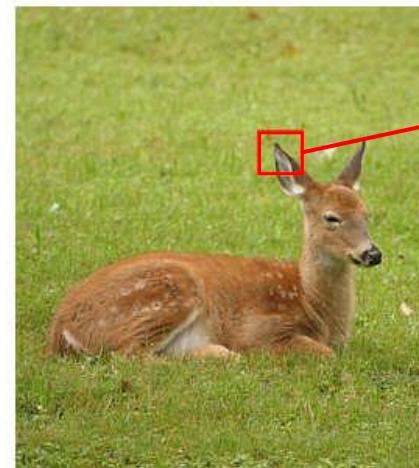
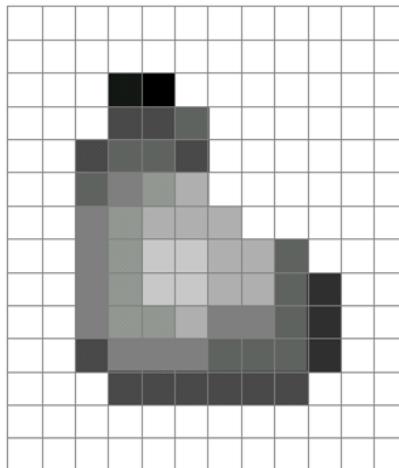
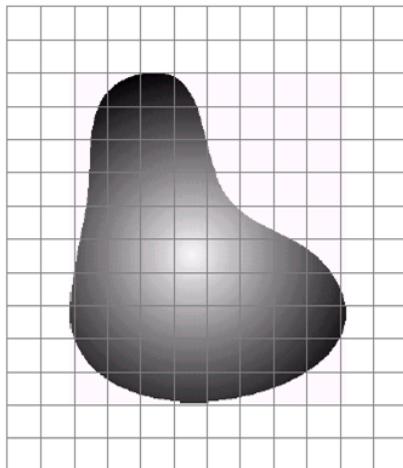


For most of this course we will focus on grey-scale images

# What is a Digital Image? (cont...)

Pixel values typically represent gray levels, colours, heights, opacities etc

**Remember** *digitization* implies that a digital image is an *approximation* of a real scene

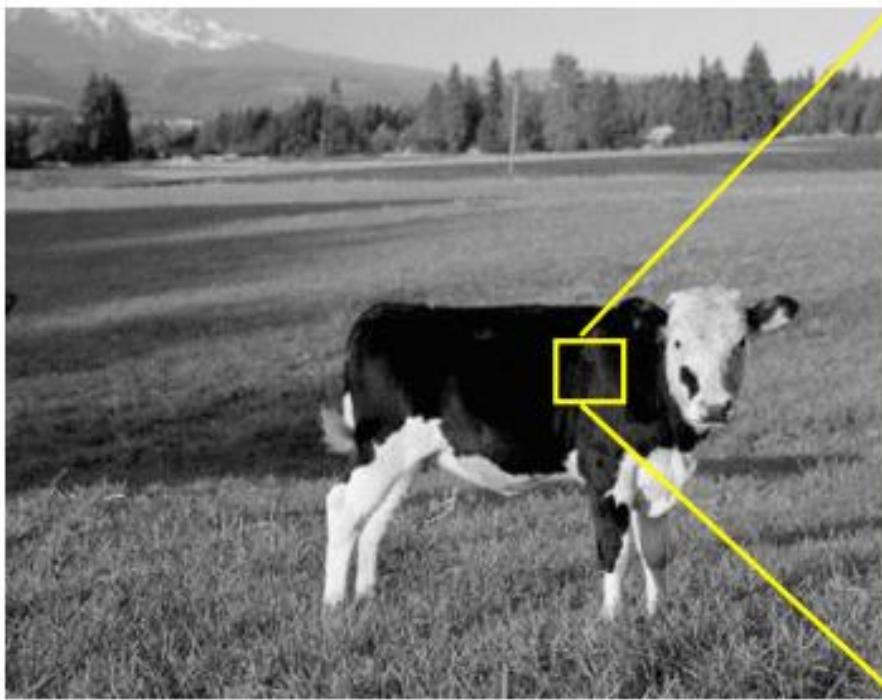


# Visualizing Image

---

Recall two ways of visualizing an image

Intensity pattern



2d array of numbers

Putdata: /home/camps/cowgray.jpg

File

146	161	165	159	165	177	166	142	143	141
149	154	152	149	158	171	164	147	144	141
147	146	145	148	157	160	151	139	140	138
147	149	157	167	167	155	139	129	133	132
148	154	167	176	169	150	135	131	131	131
139	144	152	155	149	139	133	133	133	134
131	132	132	131	132	133	131	127	130	132
133	132	129	127	134	141	134	122	125	127
129	127	126	128	131	132	130	127	129	127
129	127	126	128	131	132	130	128	130	129

We “see it” at this level

Computer works at this level

# What is Digital Image Processing?

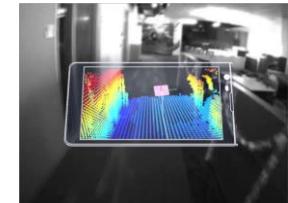
---

- Digital image processing focuses on two major tasks
  - Improvement of pictorial information for human interpretation
  - Processing of image data for storage, transmission and representation for autonomous machine perception
- Some argument about where image processing ends and fields such as image analysis and computer vision start

---

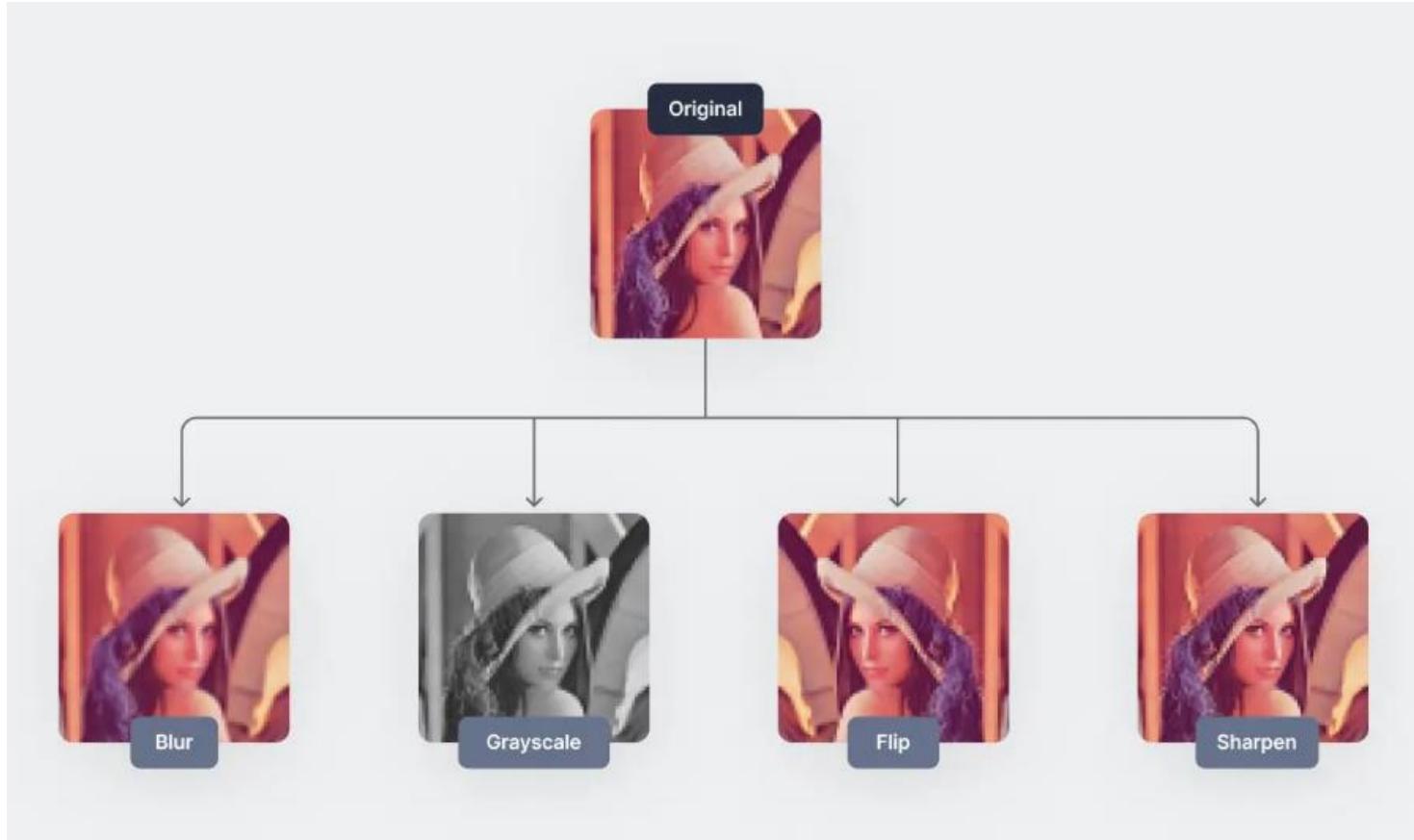
# Why do we process images?

- Acquire an image
  - *Correct aperture and color balance*
  - *Reconstruct image from projections*
- Prepare for display or printing
  - *Adjust image size*
  - *Color mapping, gamma-correction, halftoning*
- Facilitate picture storage and transmission
  - *Efficiently store an image in a digital camera*
  - *Send an image from space*
- Enhance and restore images
  - *Touch up personal photos*
  - *Color enhancement for security screening*
- Extract information from images
  - *Read 2-d bar codes*
  - *Character recognition*
  - *Depth estimation*
- Many more ... image processing is ubiquitous



# Have you done it?

---

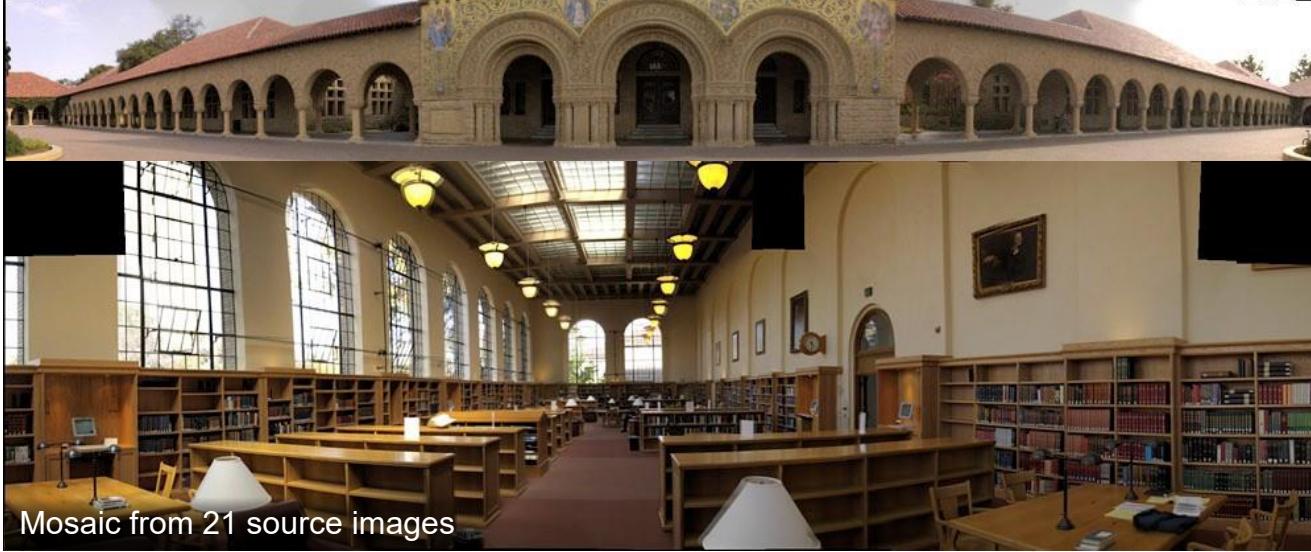


Examples of typical image processing operations

---

# Image Processing Examples

Mosaic from 33 source images



Mosaic from 21 source images

source: M. Borgmann, L. Meunier, EE368 class project, spring 2000.



Google Jump



facebook 360

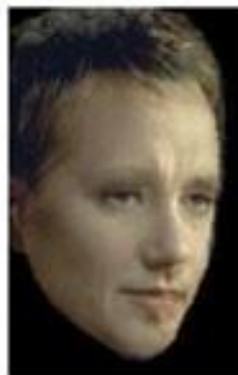
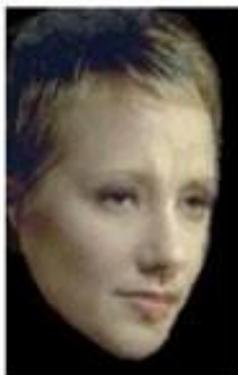


light.co

---

# Image Processing Examples

Face morphing

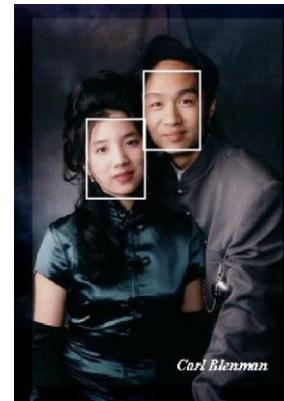
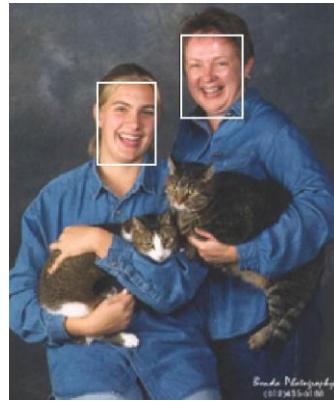
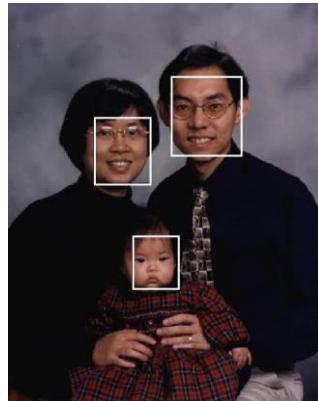


Source: Yi-Wen Liu and Yu-Li Hsueh, EE368 class project, spring 2000.

---

# Image Processing Examples

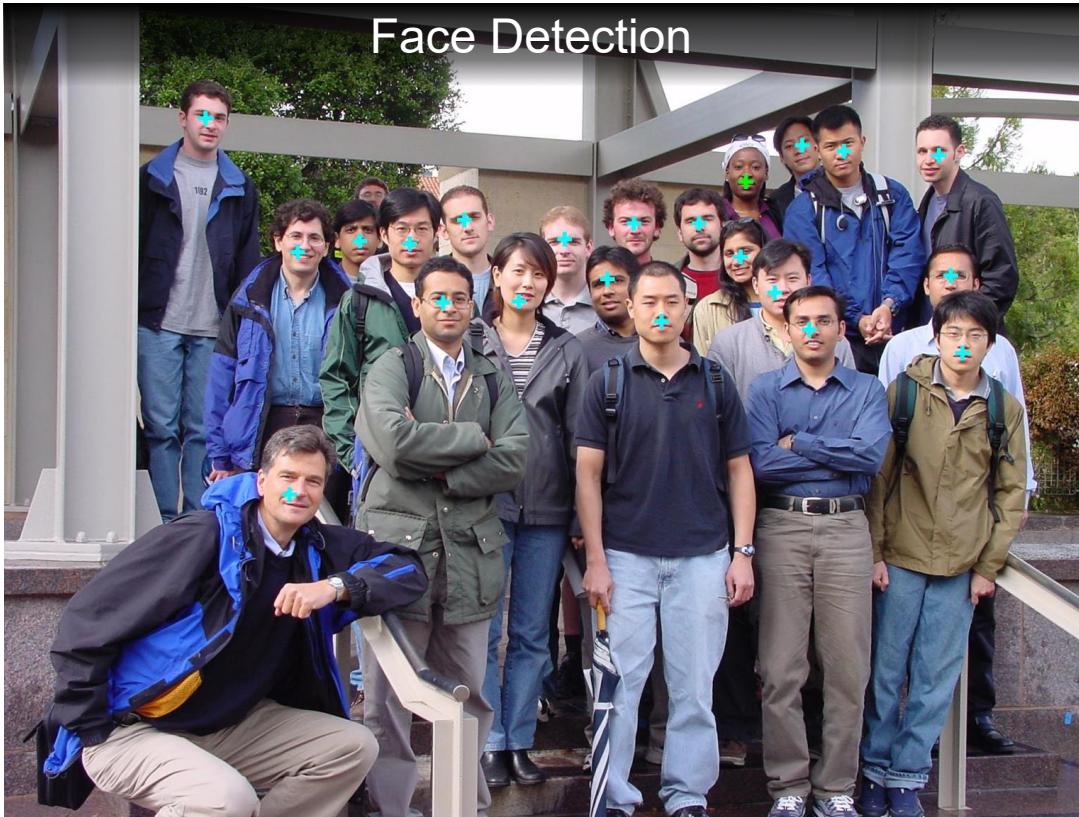
## Face Detection



source: Henry Chang, Ulises Robles, EE368 class project, spring 2000.

---

# Image Processing Examples

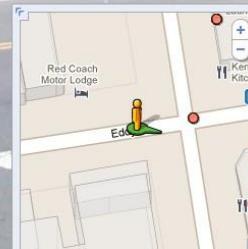
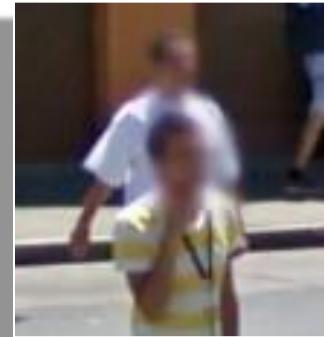


source: Michael Bax, Chunlei Liu, and Ping Li, EE368 class project, spring 2003.

---

# Image Processing Examples

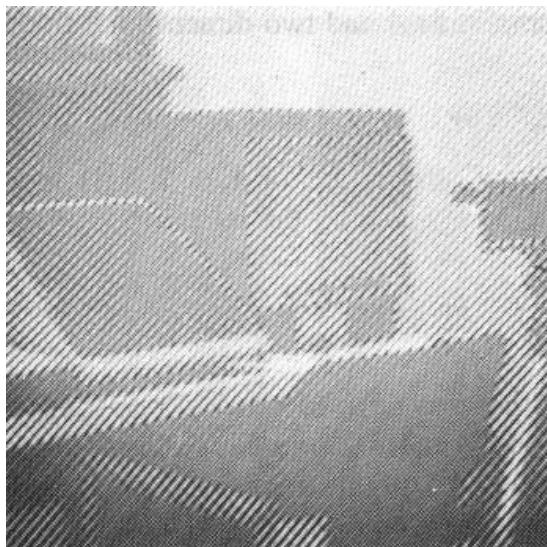
Face Blurring for Privacy Protection



# Operations in Frequency Domain

---

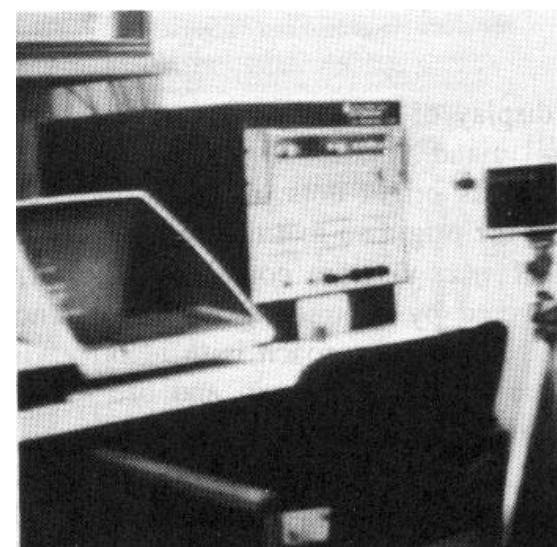
Original Noisy image



Fourier Spectrum



Filtered image



# Image Inpainting 1

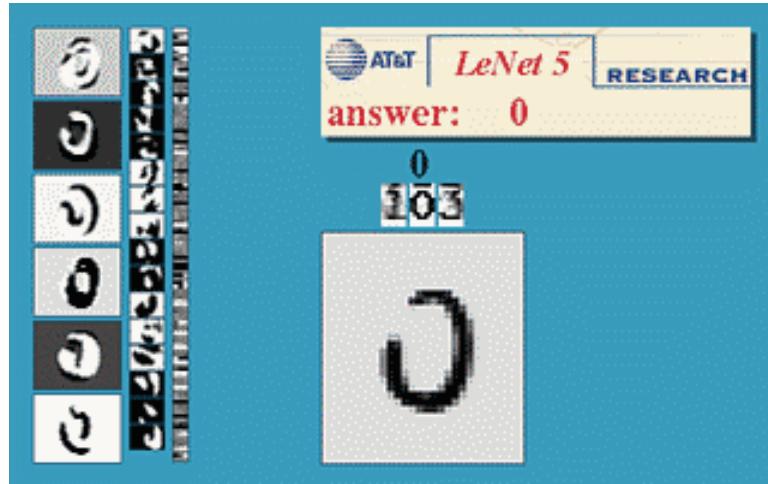
---



# Optical character recognition (OCR)

Technology to convert scanned docs to text

- If you have a scanner, it probably came with OCR software



Digit recognition, AT&T labs  
<http://www.research.att.com/~yann/>



License plate readers  
[http://en.wikipedia.org/wiki/Automatic\\_number\\_plate\\_recognition](http://en.wikipedia.org/wiki/Automatic_number_plate_recognition)

# Login without a password...

---



Fingerprint scanners on  
many new laptops,  
other devices



Face recognition systems now  
beginning to appear more widely  
<http://www.sensiblevision.com/>

# Object recognition (in mobile phones)



This is becoming real:

- **Lincoln** Microsoft Research
- [Point & Find](#), [Nokia](#)

# Sports

---



*Sportvision first down line*  
Nice [explanation](#) on [www.howstuffworks.com](http://www.howstuffworks.com)

# Smart cars

Slide content courtesy of Amnon Shashua

►► manufacturer products consumer products ◀◀

## Our Vision. Your Safety.

rear looking camera      forward looking camera      side looking camera

**> EyeQ** Vision on a Chip

> read more

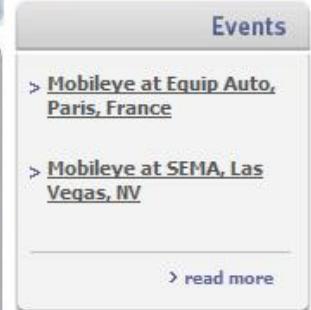
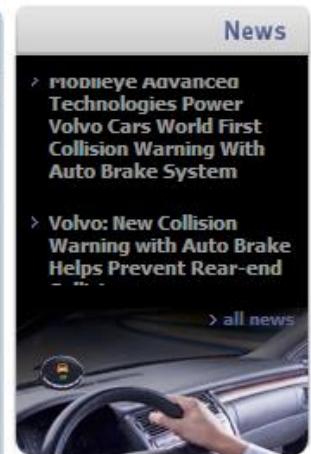
**> Vision Applications**

Road, Vehicle, Pedestrian Protection and more

> read more

**> AWS** Advance Warning System

> read more



## Mobileye

- Vision systems currently in high-end BMW, GM, Volvo models
- By 2010: 70% of car manufacturers.
- [Video demo](#)

# Challenges: viewpoint variation

---



Michelangelo

slide credit: Fei-

# Challenges: illumination

---

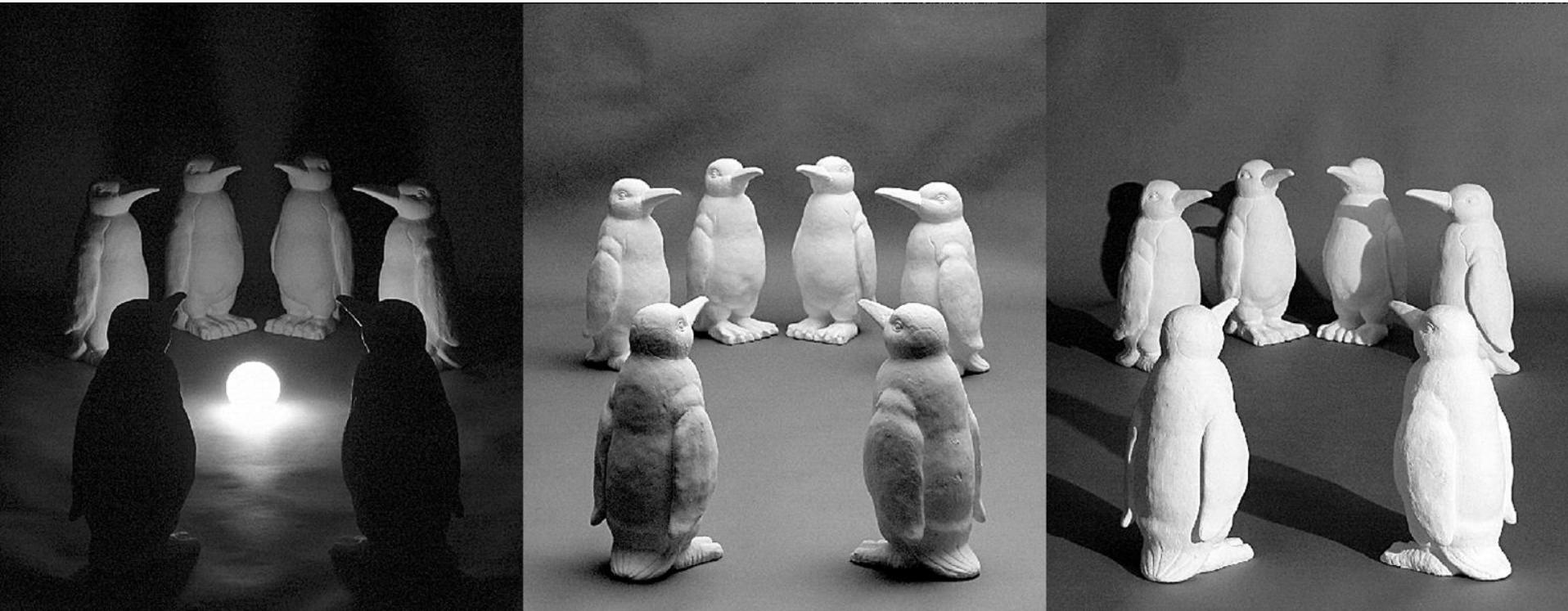


image credit:

# Challenges: scale

---

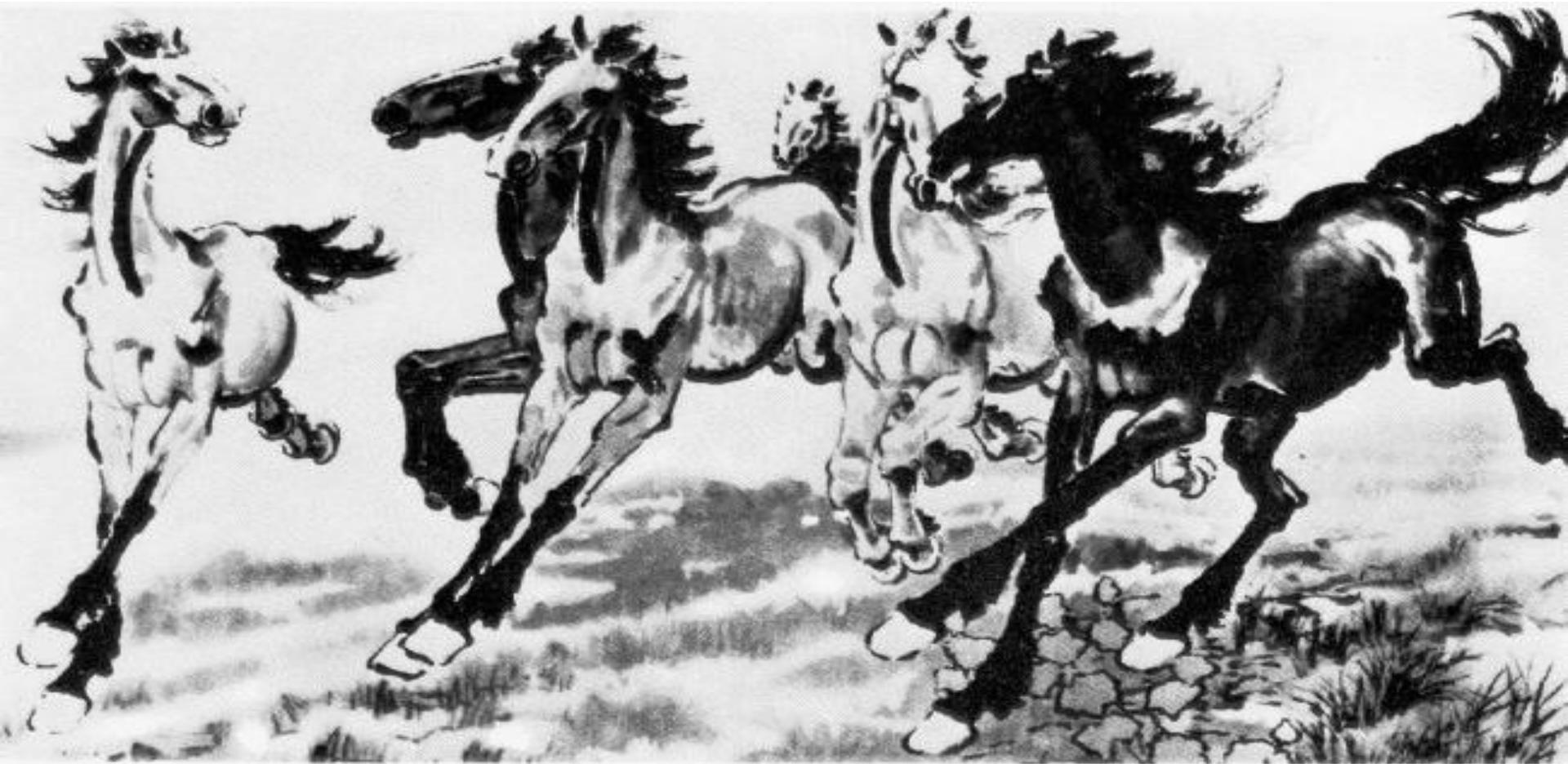
and small things  
from Apple.  
(Actual size)



slide credit: Fei-

# Challenges: deformation

---



Xu, Beihong  
1943

slide credit: Fei-

# Challenges: occlusion

---



Magritte,

slide credit: Fei-

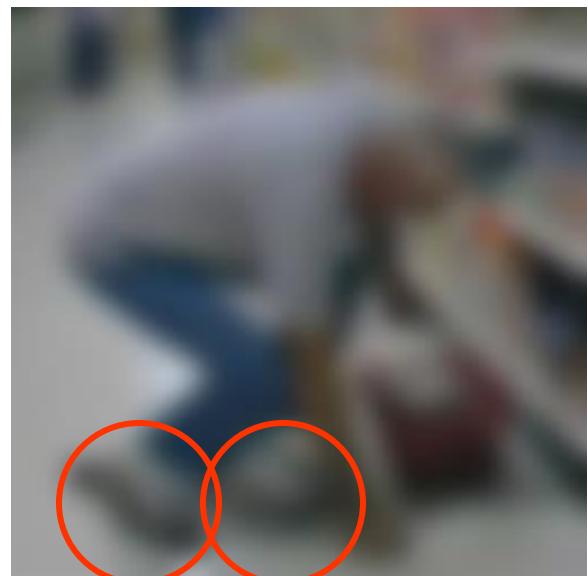
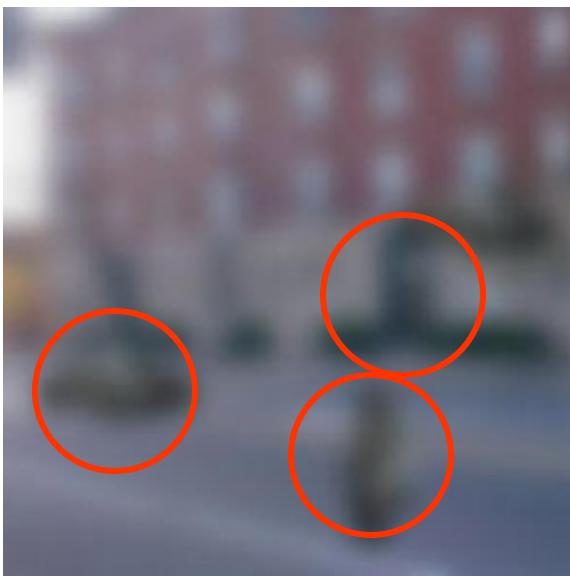
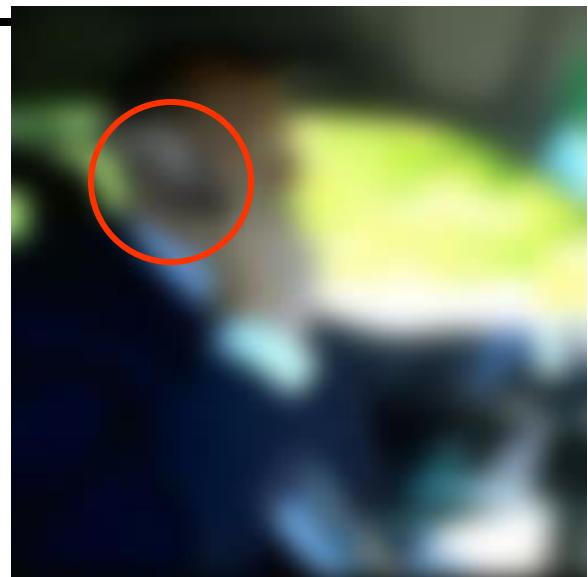
# Challenges: background clutter



Emperor shrimp and commensal crab on a sea cucumber in Fiji  
Photograph by Tim Laman

NATIONAL  
GEOGRAPHIC

# Challenges: local ambiguity



slide credit: Fei-

# Challenges or opportunities?

---

- Images are confusing, but they also reveal the structure of the world through numerous cues
- Our job is to interpret the cues!



# Bottom line

---

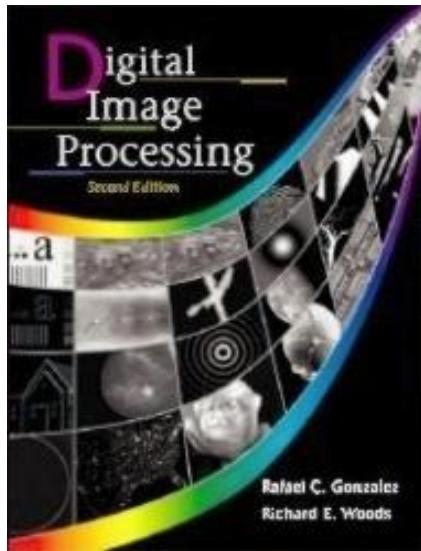
- Perception is an inherently ambiguous problem
  - Many different 3D scenes could have given rise to a particular 2D picture



- Possible solutions
  - Bring in more constraints (or more images)
  - Use prior knowledge about the structure of the world
- Need both exact measurements and statistical inference!

# Textbook

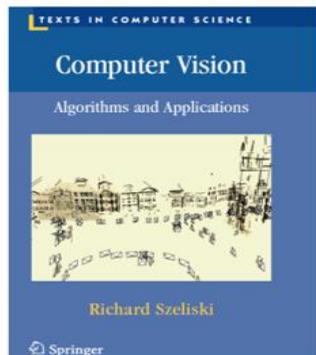
---



“Digital Image Processing”,  
Rafael C. Gonzalez & Richard E. Woods,

## Computer Vision: Algorithms and Applications

© 2010 [Richard Szeliski](#), Microsoft Research



<http://szeliski.org/Book/>

# Important links

---

1. [https://bioimagebook.github.io/chapters/1-concepts/1-images\\_and\\_pixels/images\\_and\\_pixels.html](https://bioimagebook.github.io/chapters/1-concepts/1-images_and_pixels/images_and_pixels.html)
2. <https://www.oreilly.com/library/view/practical-computer-vision/9781449337865/ch04.html>

Recommended books:

1. Digital Image Processing- Rafael C.Gonzalez and Richard Eugene Woods

# What are you looking for

---

