香港中文大學(深圳)
The Chinese University of Hong Kong, Shenzhen

EIE4512
Digital Image Processing

Zhen Li

lizhen@cuhk.edu.cn

School of Science and Engineering,

The Chinese University of Hong Kong, Shen Zhen

Basic Information

- Time & Venue
  - Tuesday: 10:30 am – 12:00 pm, TA207
  - Thursday: 10:30 am – 12:00 pm, TA 207

- Staff
  - Instructor: Zhen Li (lizhen@cuhk.edu.cn)
  - Tutors: Qin Wang (wangqin4377@qq.com)
    Bin Zeng (zeng_bin8888@163.com)

- One tutorial per week:
  Thursday: 8:00pm – 9:00pm, TA 207

Slides

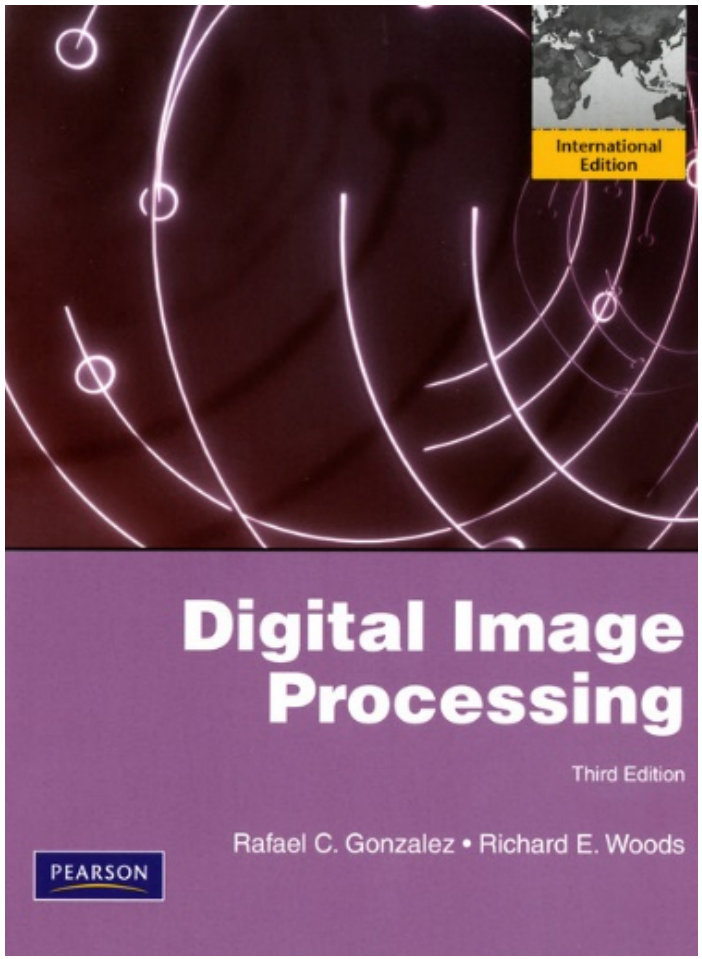
- Online address:
  - *https://github.com/icemansina/CUHKSZ_DIP*

- We use this repository for:
  - Make announcements
  - Release lecture slides
  - Release homework assignments and solutions
  - Discussions: wechat group

DIP course

该二维码7天内(1月14日前)有效，重新进入将更新

Reference Textbook



Digital Image Processing

By *Rafael C. Gonzalez & Richard E. Woods*

Grading

- Three assignments: 10% x 3
- Mid-term exam: 30%
- Final project: 40%
  - Topics
    - Applications of digital image processing
    - Implementation of digital image processing
    - Study digital image processing algorithms
  - You (at most two students as a group) should submit
    - One page proposal which briefly introduces the topic, idea, method, experiments.
    - A term paper of 4 pages (excluding figures) in maximum, double column, font size is equal or no larger than 10.
    - Code and sample data
    - Project presentation (5mins – 10mins)

What is this course about?

Face Beautification

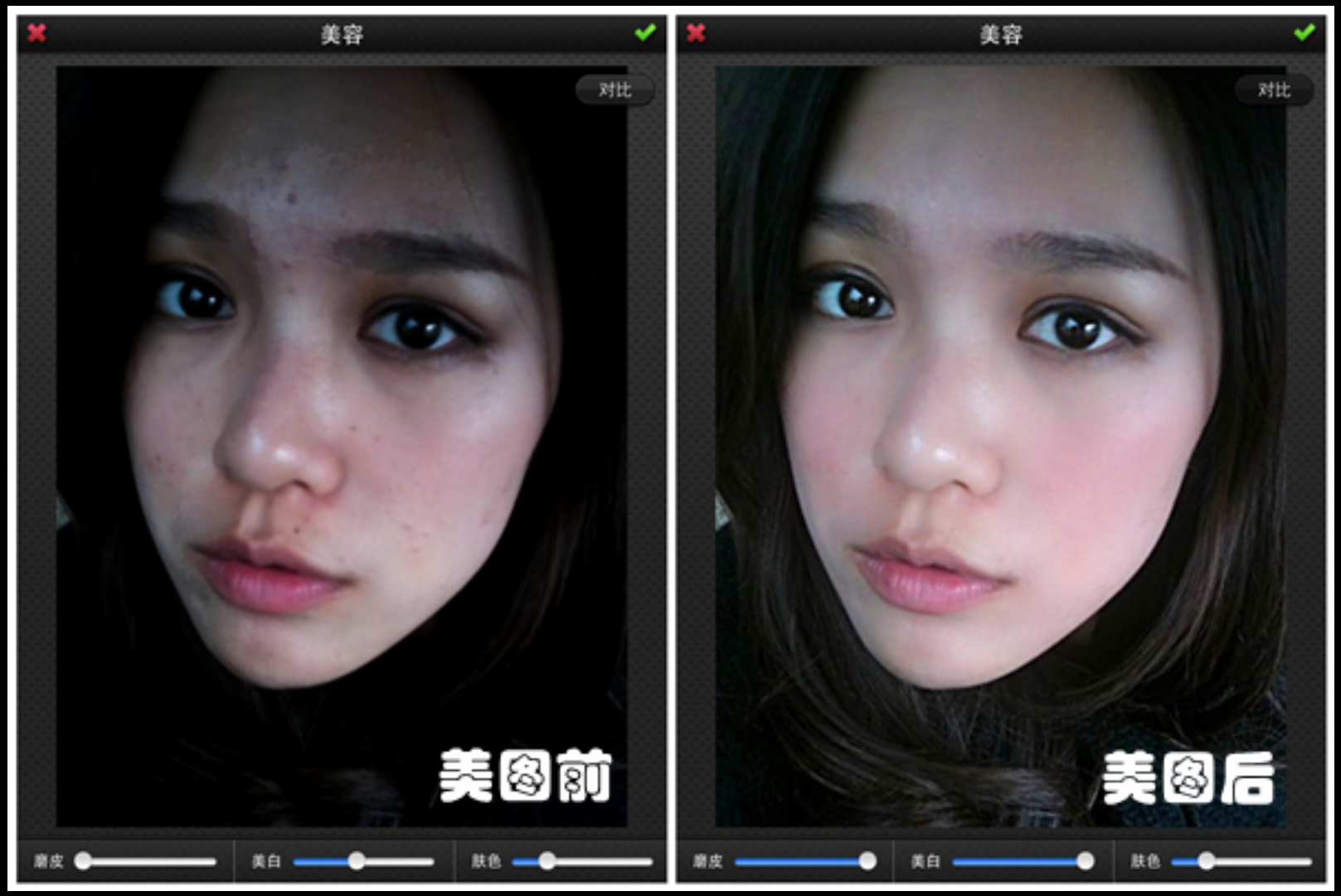


Image Filtering

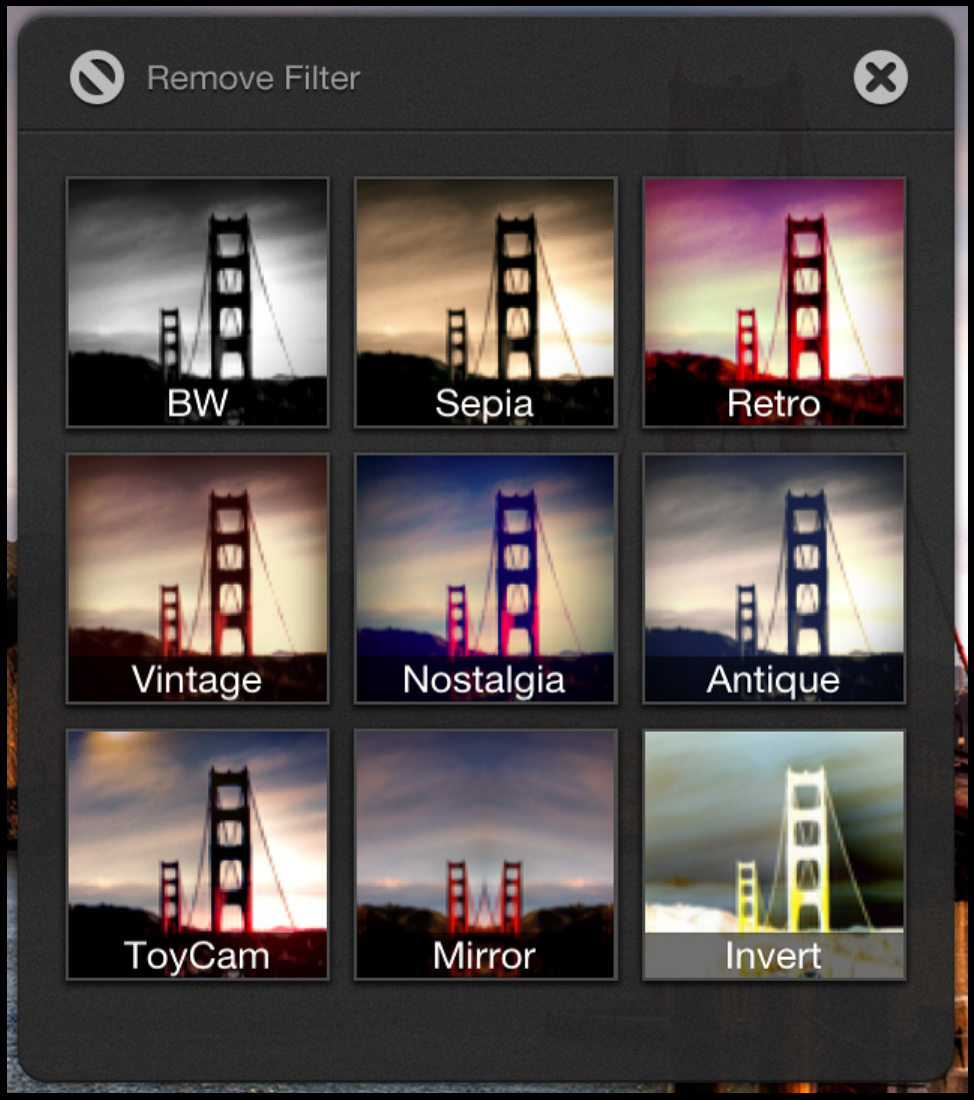
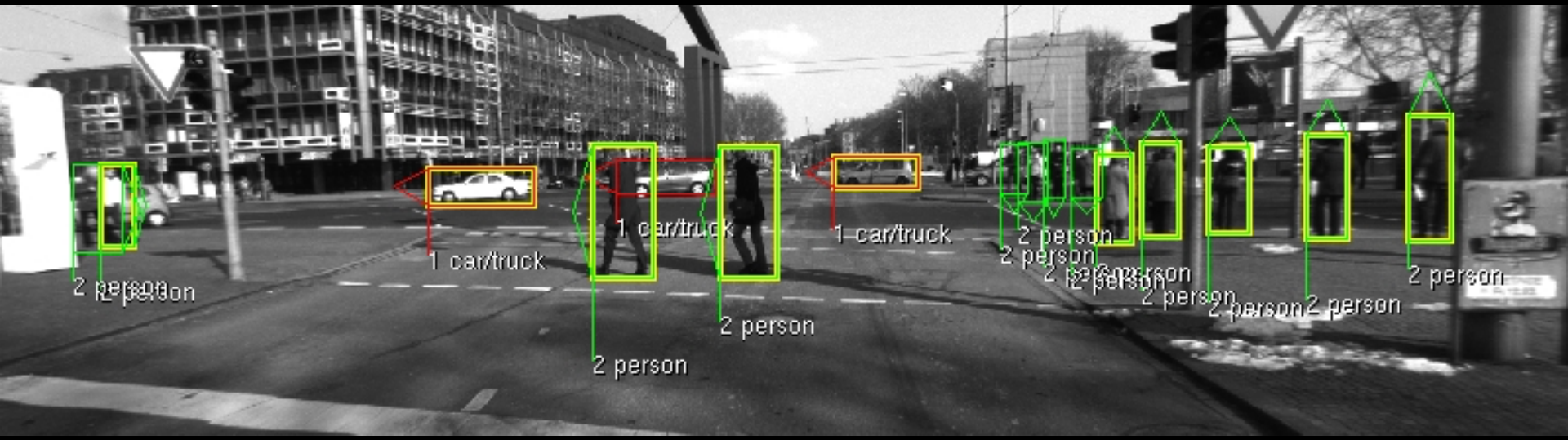


Image Superresolution

Face Recognition

Object Recognition

Pedestrian Detection



Scene Parsing

Topics

Image Representations

- Image Sensing and Acquisition
- Sampling and Quantization
- Basic Relationships between Pixels

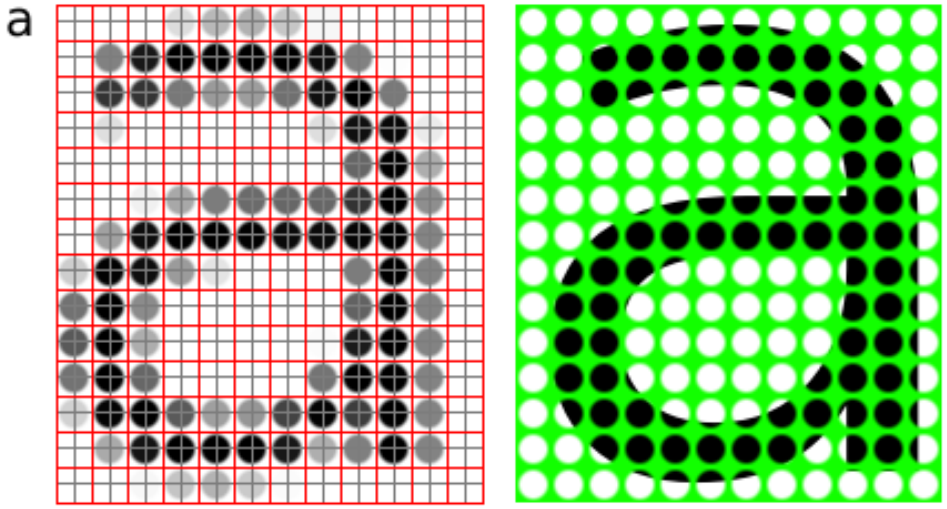
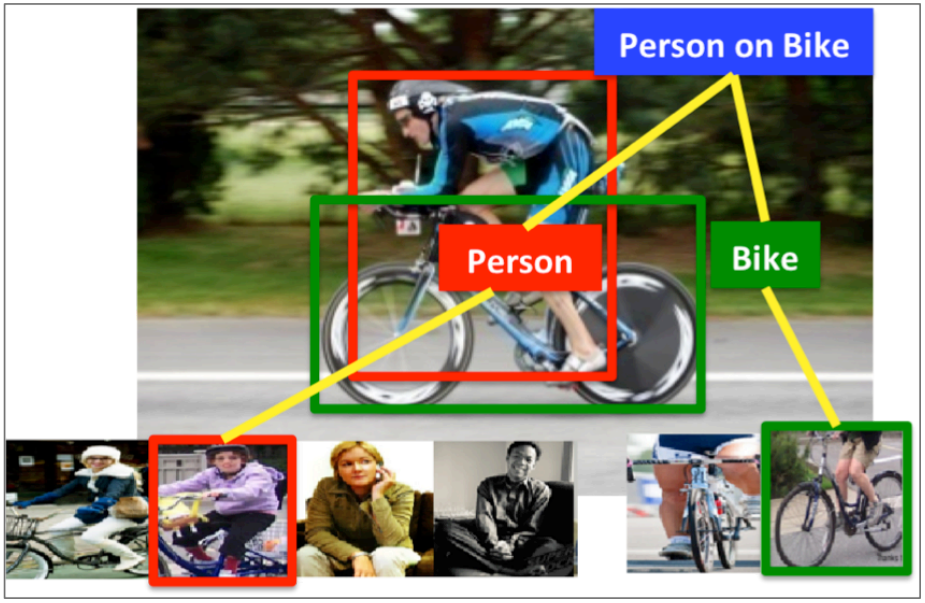


Image Processing

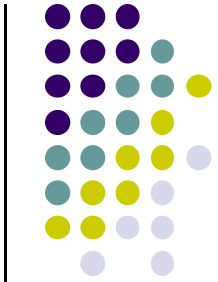
- Gray-level processing
- Image filtering
- Image restoration
- Morphological processing

Visual Understanding

- Features
- Image recognition
- Object detection
- Image segmentation



What is an Image?

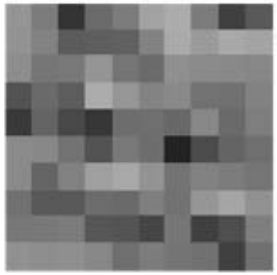


- 2-dimensional matrix of Intensity (gray or color) values

Set of Intensity values

Image coordinates are integers

$$I(u,v) \in \mathbb{P} \quad \text{and} \quad u,v \in \mathbb{N}.$$



$F(x,y)$

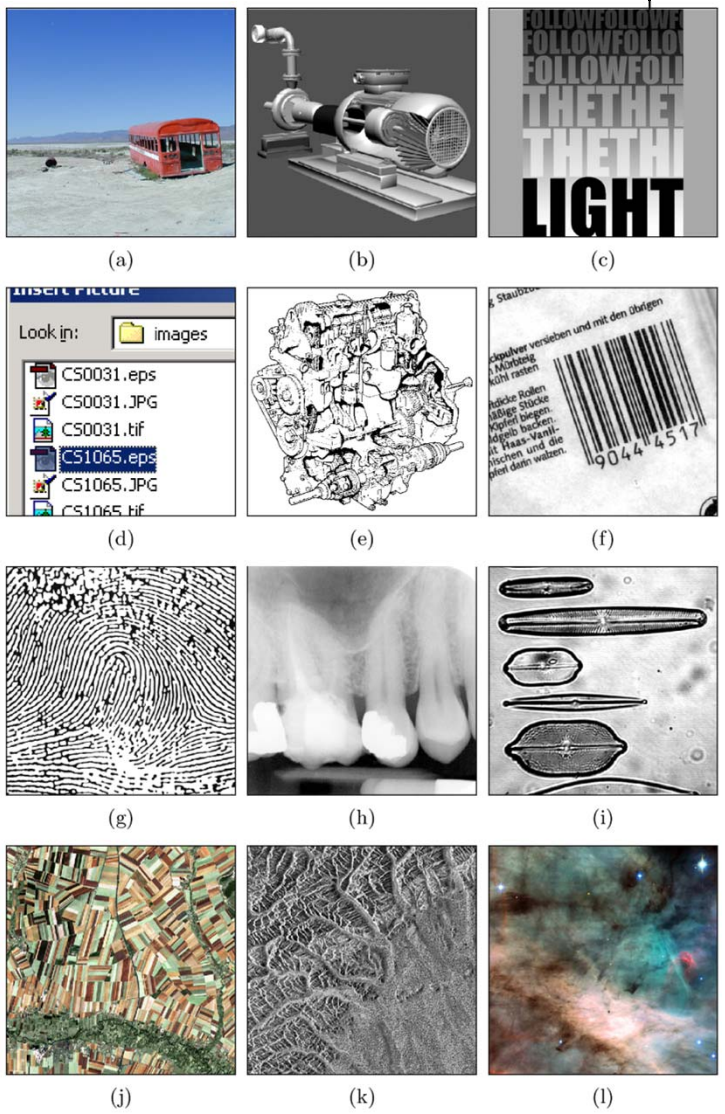
$\rightarrow$

148	123	52	107	123	162	172	123	64	89 ···
147	130	92	95	98	130	171	155	169	163 ···
141	118	121	148	117	107	144	137	136	134 ···
82	106	93	172	149	131	138	114	113	129 ···
57	101	72	54	109	111	104	135	106	125 ···
138	135	114	82	121	110	34	76	101	111 ···
138	102	128	159	168	147	116	129	124	117 ···
113	89	89	109	106	126	114	150	164	145 ···
120	121	123	87	85	70	119	64	79	127 ···
145	141	143	134	111	124	117	113	64	112 ···
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

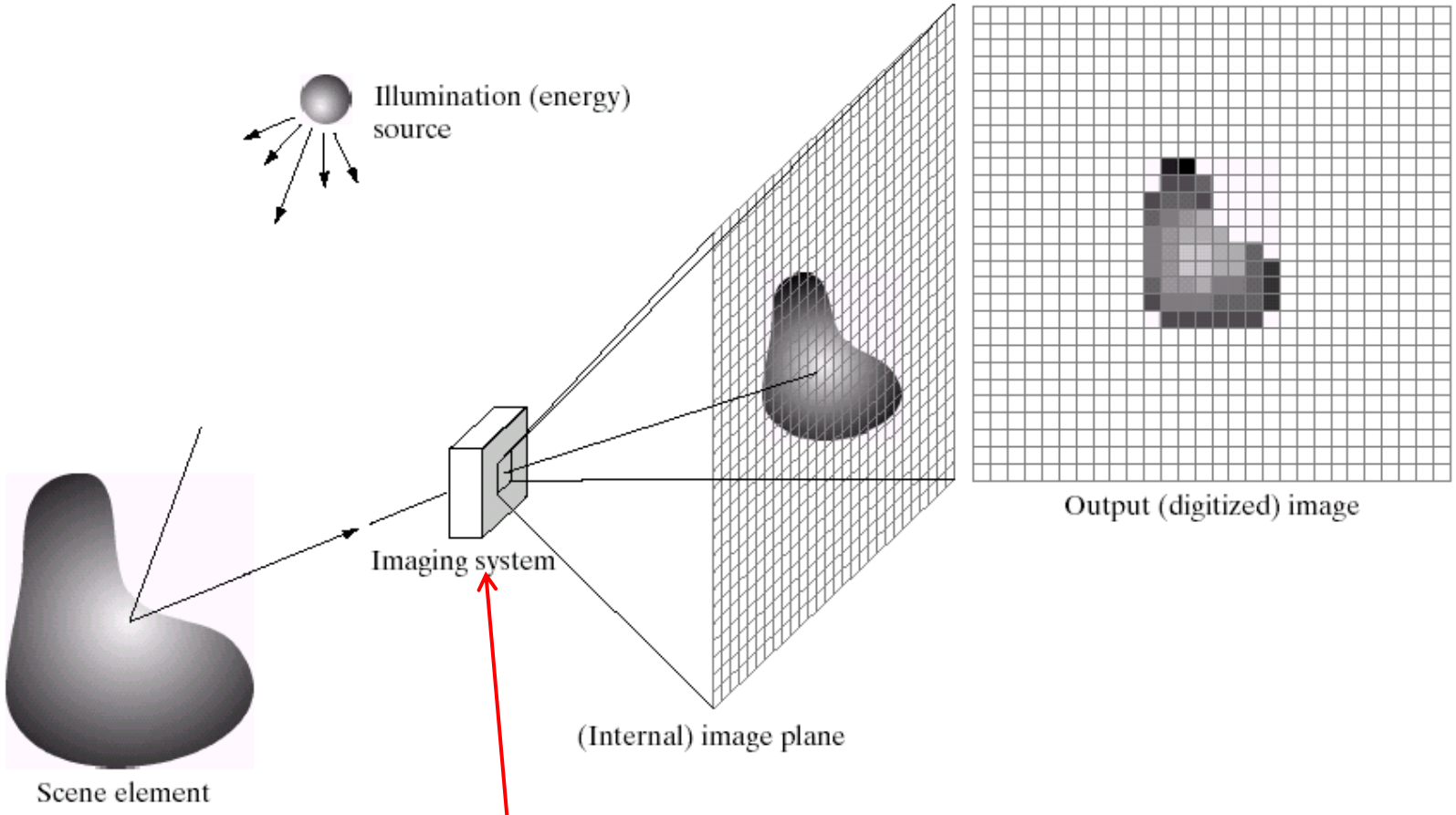
$I(u,v)$

Example of Digital Images

a) Natural landscape
b) Synthetically generated scene
c) Poster graphic
d) Computer screenshot
e) Black and white illustration
f) Barcode
g) Fingerprint
h) X-ray
i) Microscope slide
j) Satellite Image
k) Radar image
l) Astronomical object



Imaging System

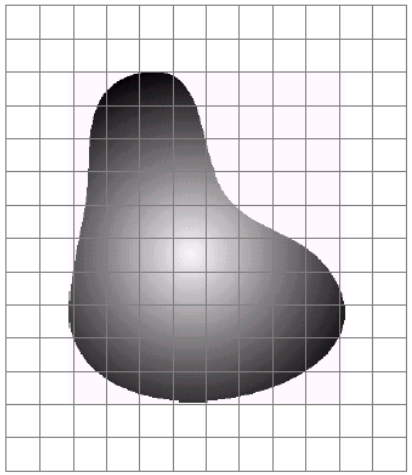


Example: a camera
Converts light to image

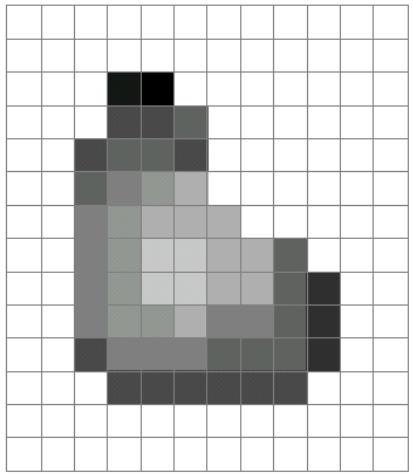
Credits: Gonzales and Woods

Digital Image?

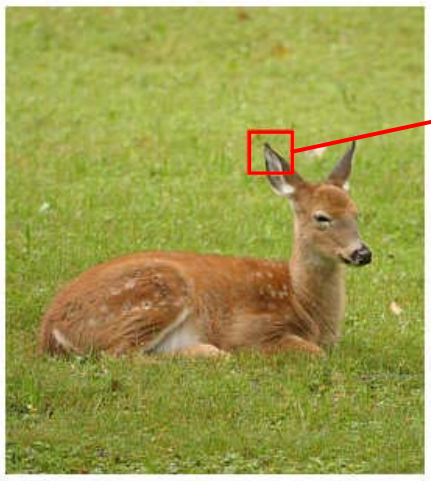
- **Remember:** *digitization* causes a digital image to become an *approximation* of a real scene



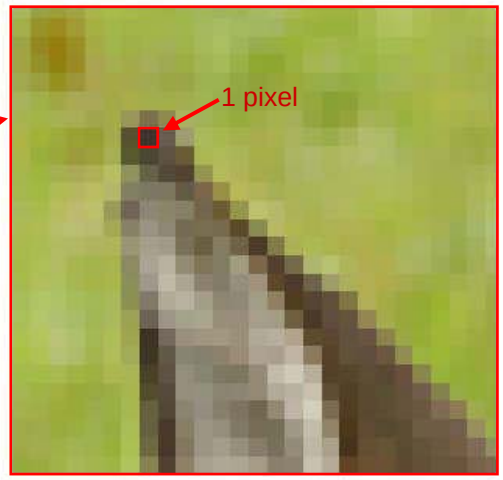
Real image



Digital Image
(an approximation)

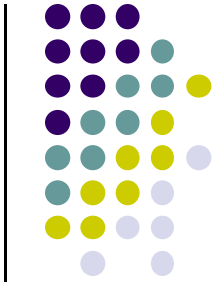


Real image



Digital Image
(an approximation)

Digital Image



- Common image formats include:
  - 1 values per point/pixel (B&W or Grayscale)
  - 3 values per point/pixel (Red, Green, and Blue)
  - 4 values per point/pixel (Red, Green, Blue, + “Alpha” or Opacity)

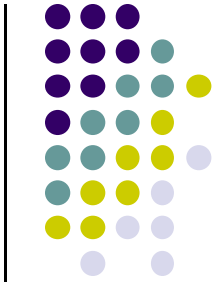
Grayscale

RGB

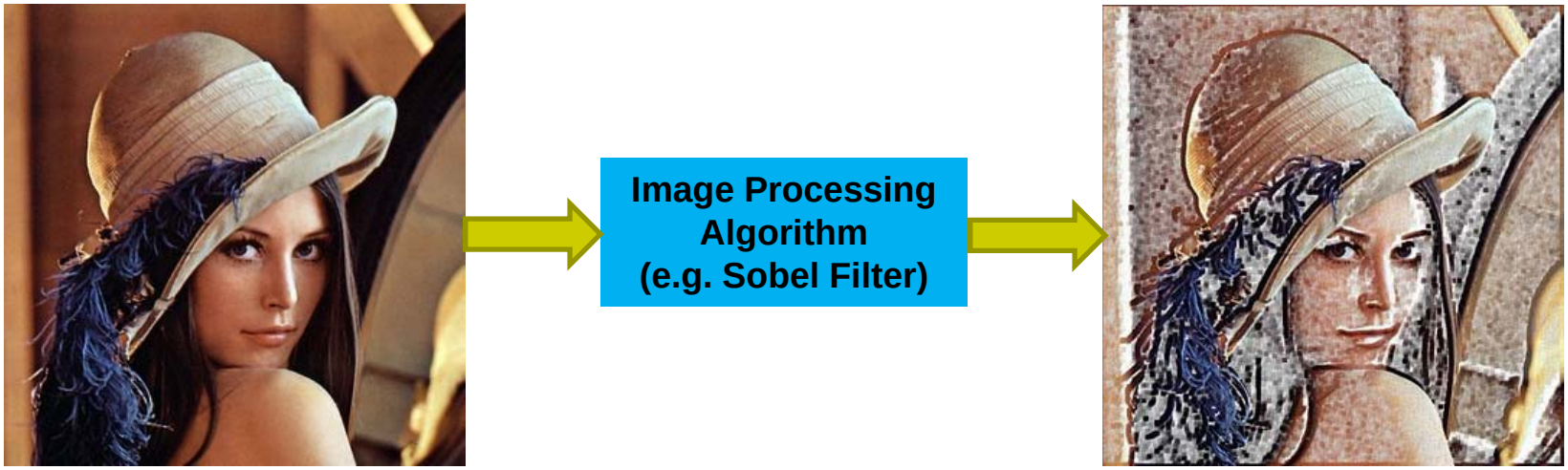
RGBA

- We will start with gray-scale images, extend to color later

What is image Processing?



- **Algorithms that alter an input image to create new image**
- **Input is image, output is image**



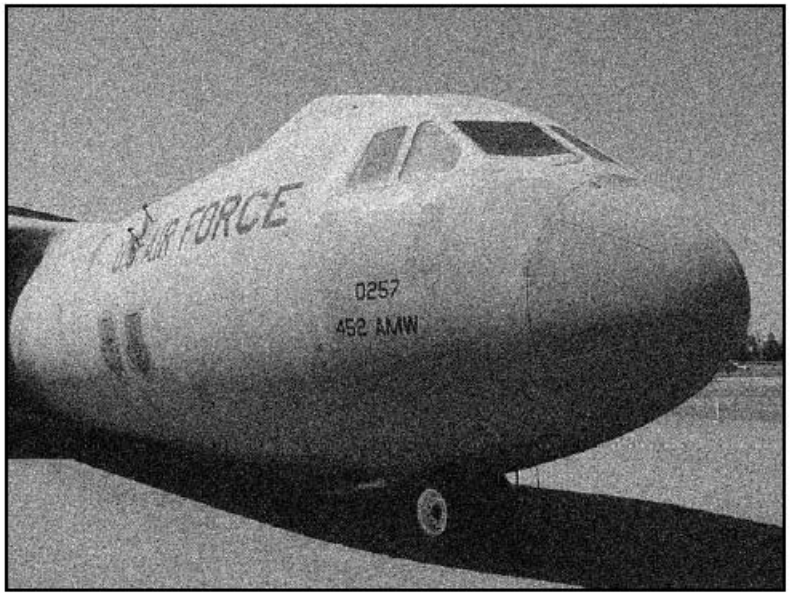
Original Image *Processed Image*

- **Improves an image for human interpretation in ways including:**
  - Image display and printing
  - Image editting
  - Image enhancement
  - Image compression

Example Operation: Noise Removal

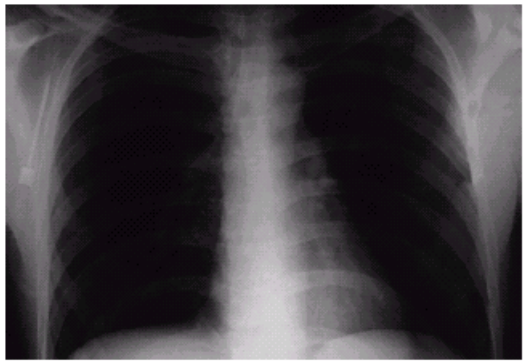
Noisy Image

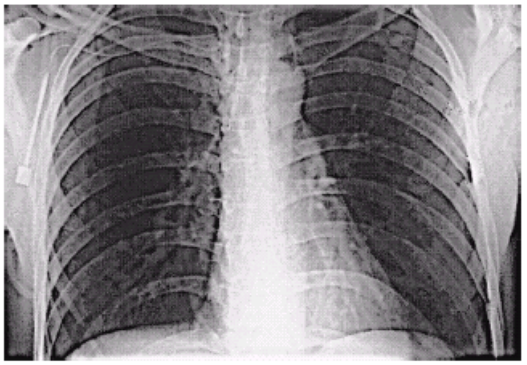
Denoised Image

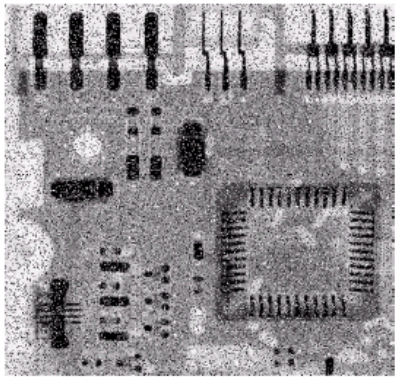


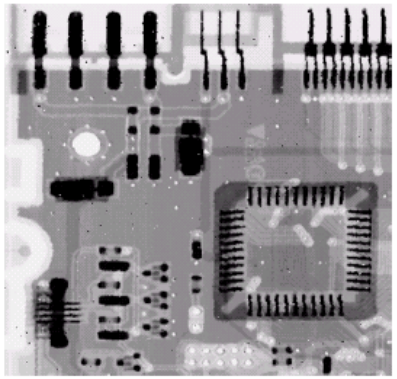
Think of noise as white specks on a picture (random or non-random)

Examples: Noise Removal









Images taken from Gonzalez & Woods, Digital Image Processing (2002)

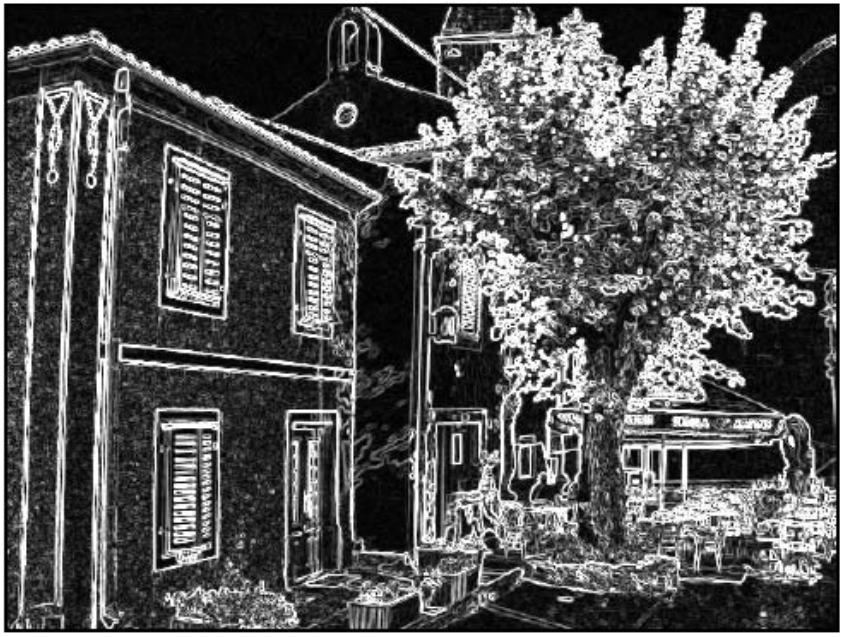
Example: Contrast Adjustment

Low Contrast

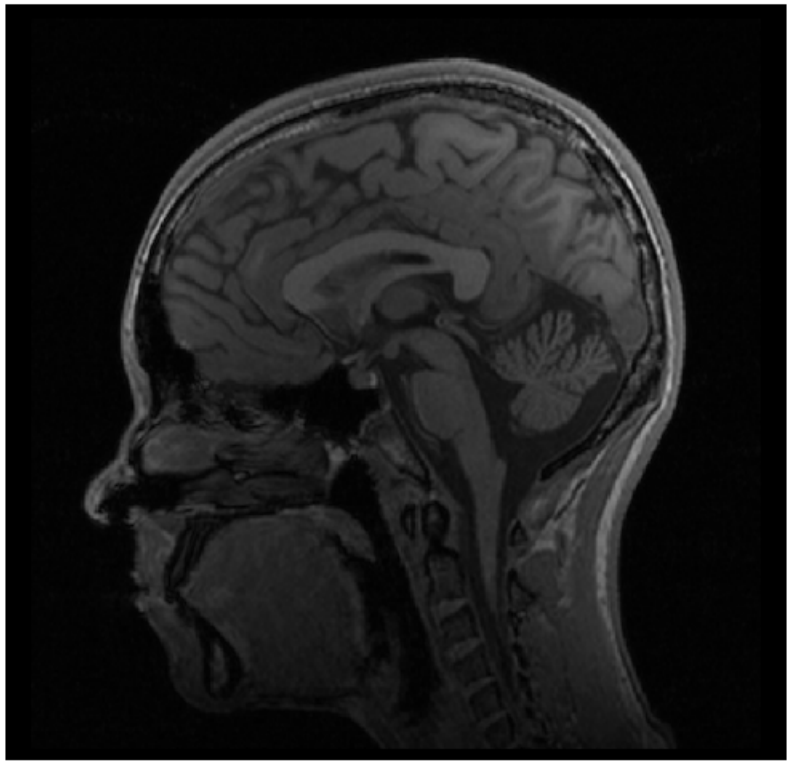
Original Contrast

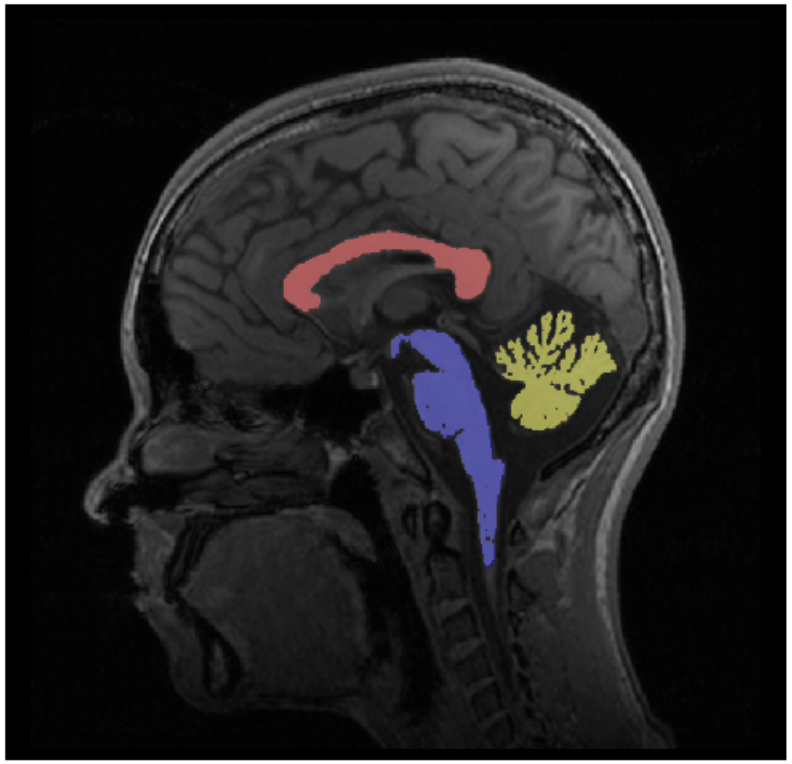
High Contrast

Example: Edge Detection



Example: Region Detection, Segmentation



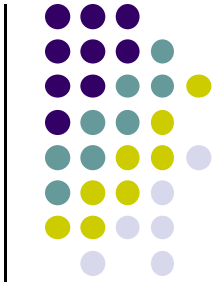


Example: Image Compression

Original, 2.1MB

JPEG Compression, 308KB (15%)

Example: Image Inpainting



Damaged Image

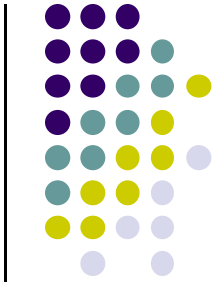
Restored Image

Credit: M. Bertalmio, G. Sapiro, V. Caselles, C. Ballester: Image Inpainting, SIGGRAPH 2000

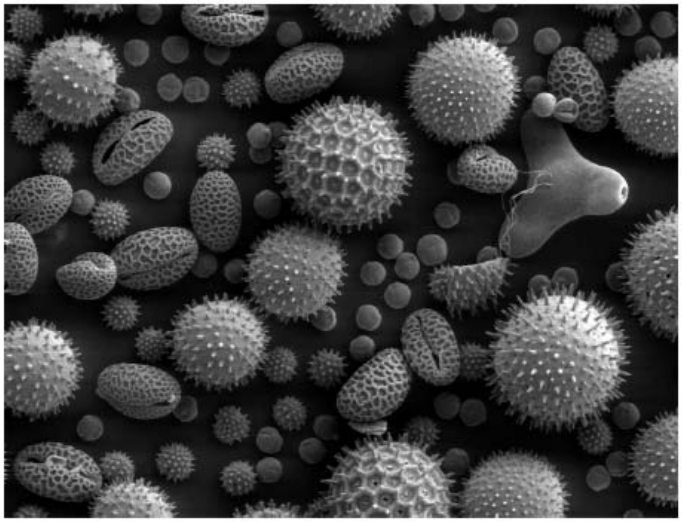
Inpainting? Reconstruct corrupted/destroyed parts of an image

Examples: Artistic (Movie Special)Effects

Applications of Image Processing

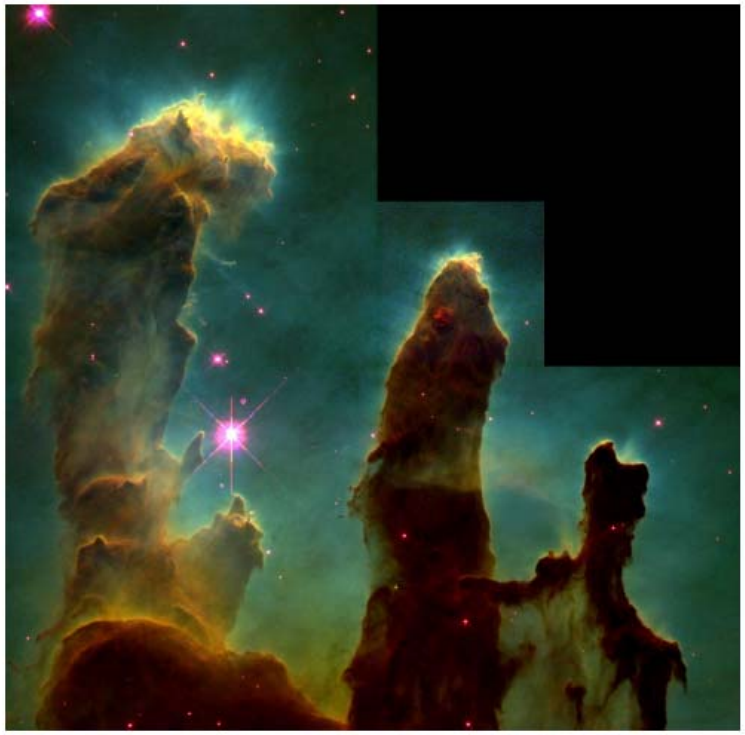


Biology



Credit: Dartmouth Electron Microscopy Facility

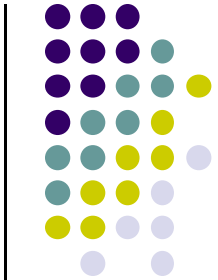
Astronomy



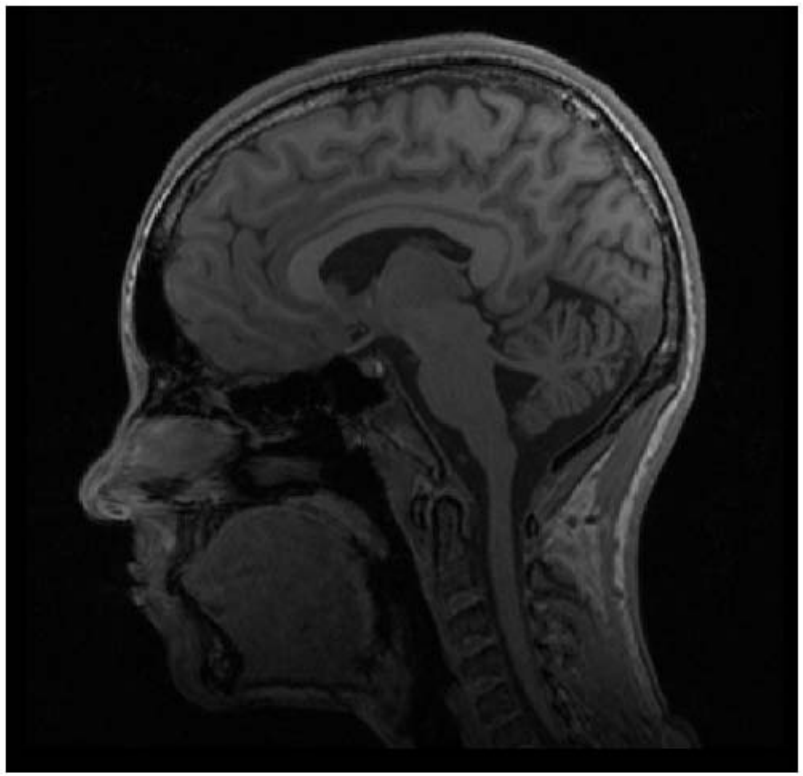
Credit: NASA, Jeff Hester, and Paul Scowen (Arizona State)

More info here

Applications of Image Processing



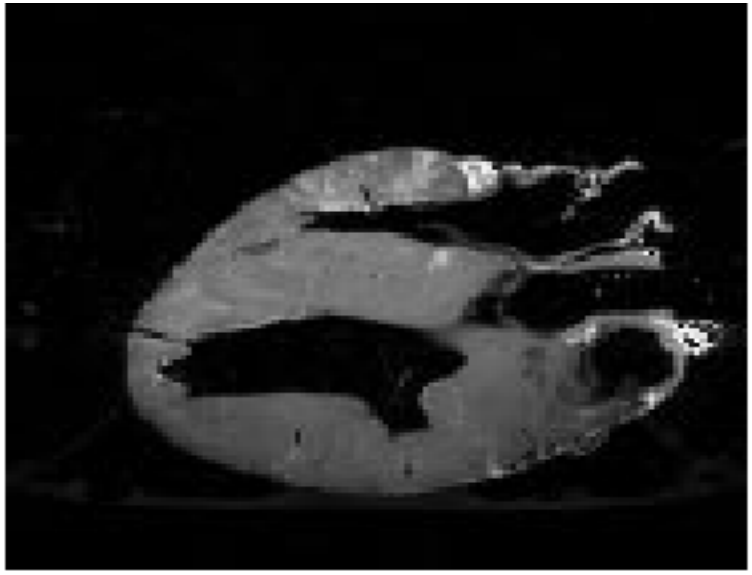
Medicine



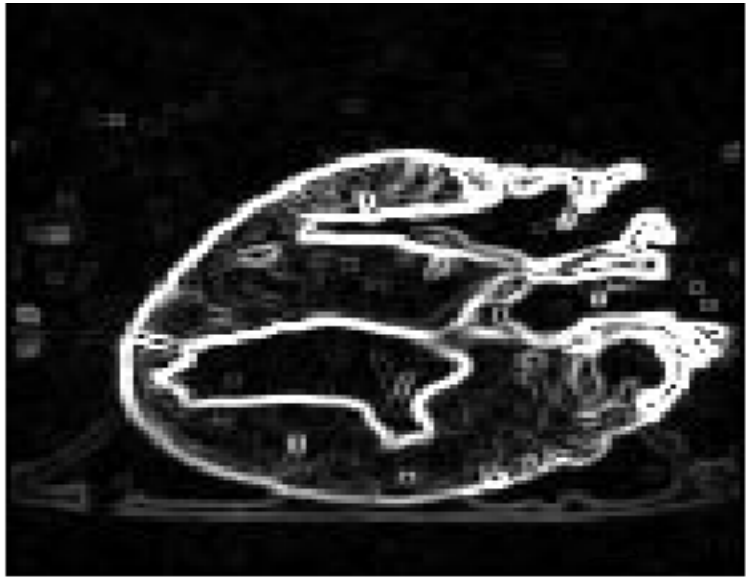
Credit: Dr. Janet Lainhart, UofU Psychiatry

Security, Biometrics

Applications of Image Processing: Medicine



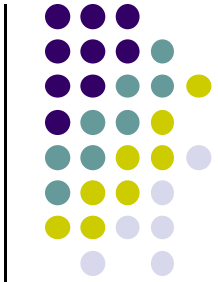
Original MRI Image of a Dog Heart



Edge Detection Image

Images taken from Gonzalez & Woods, Digital Image Processing (2002)

Applications of Image Processing



Satellite Imagery

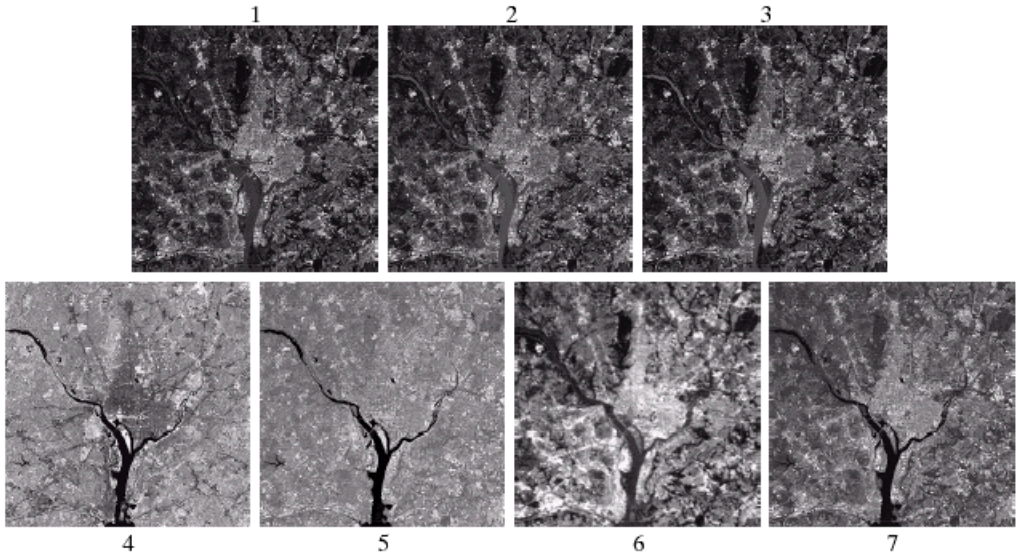
Credit: NASA

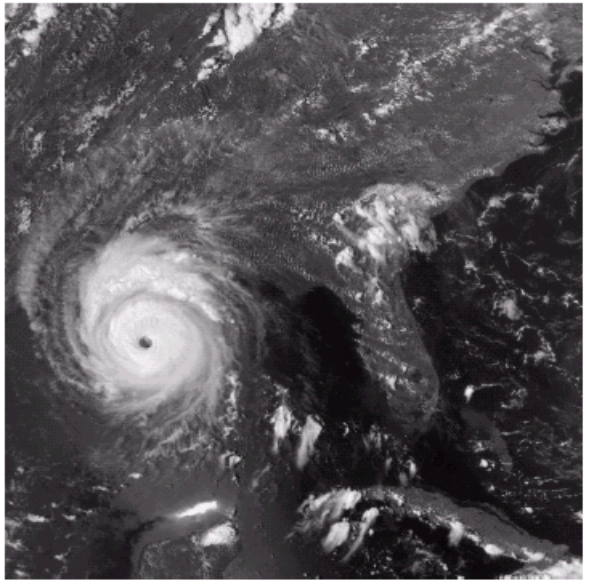
Personal Photos

Credit: Tom Fletcher

Applications of Image Processing: Geographic Information Systems (GIS)

- Terrain classification
- Meteorology (weather)

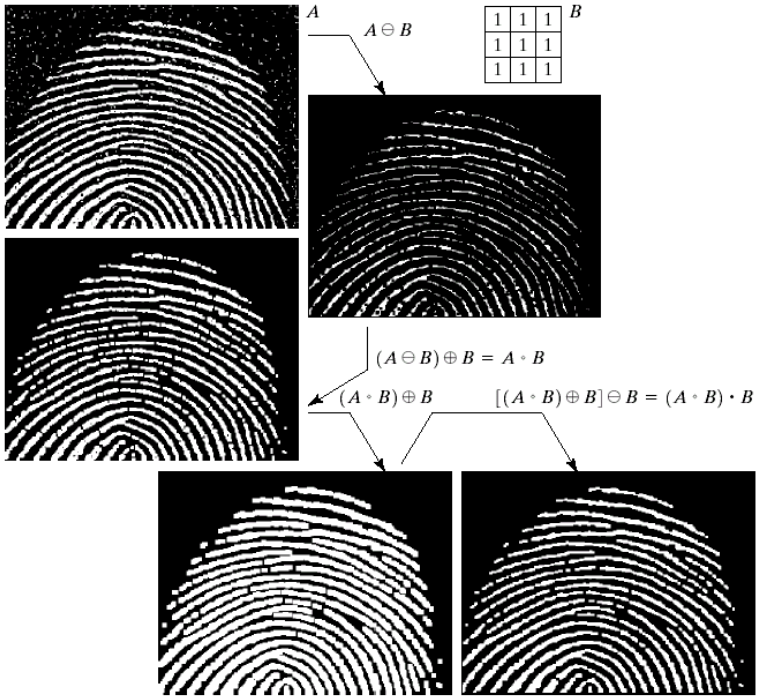




Images taken from Gonzalez & Woods, Digital Image Processing (2002)

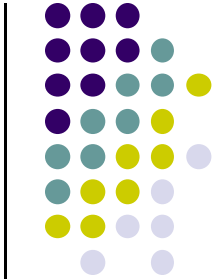
Applications of Image Processing: Law Enforcement

- Number plate recognition for speed cameras or automated toll systems
- Fingerprint recognition

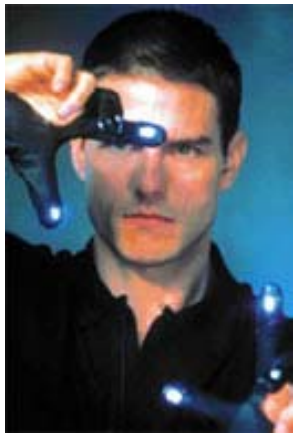


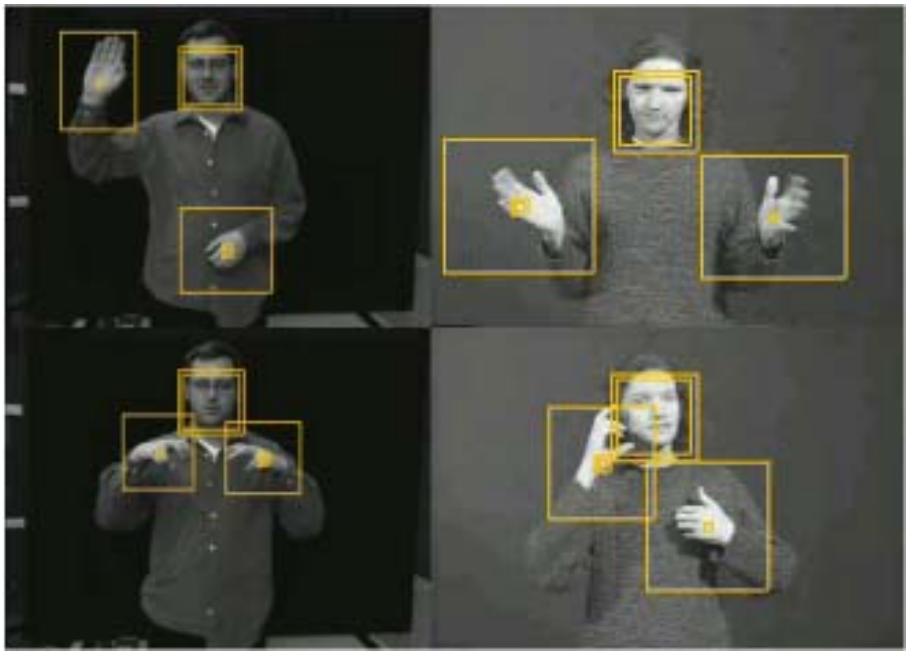
Images taken from Gonzalez & Woods, Digital Image Processing (2002)

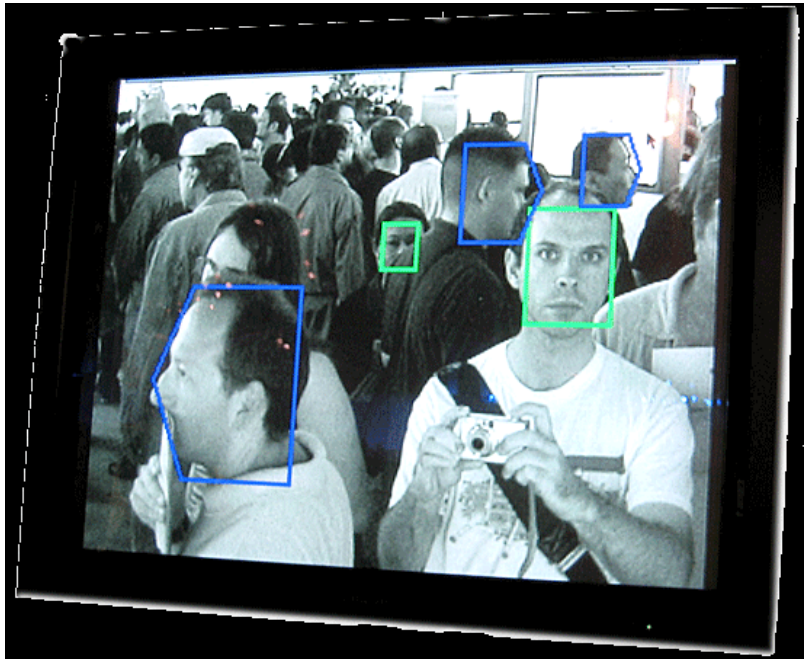
Applications of Image Processing: HCI



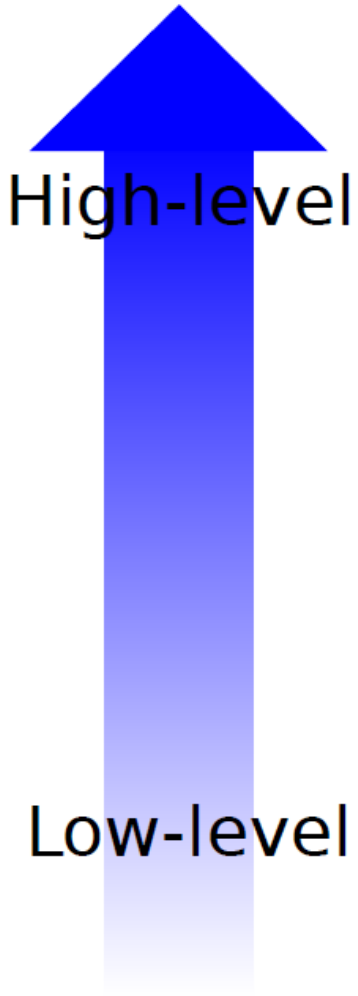
- Face recognition
- Gesture recognition







Relationship with other Fields



Computer Vision

Object detection, recognition, shape analysis, tracking
Use of Artificial Intelligence and Machine Learning

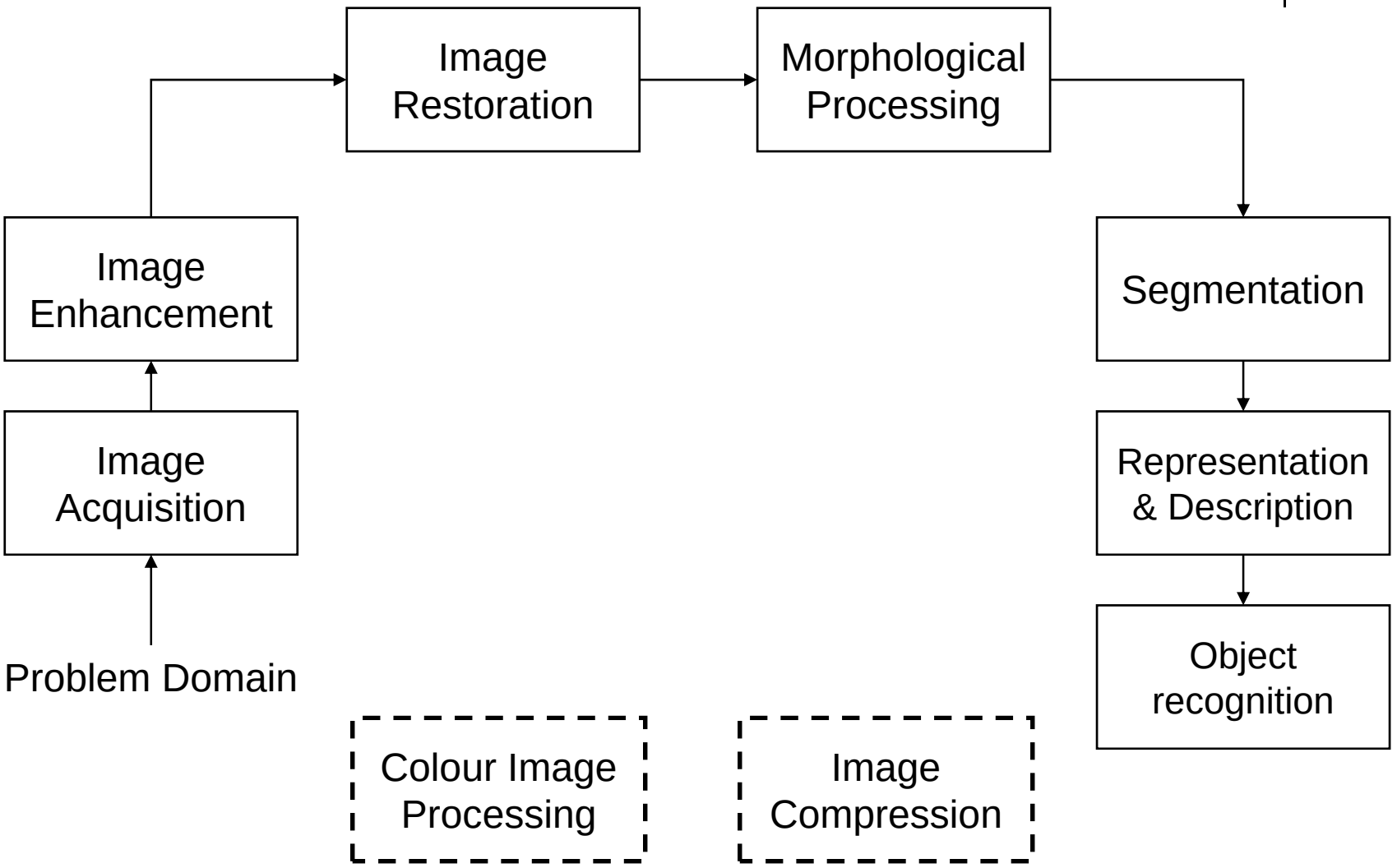
Image Analysis

Segmentation, image registration, matching

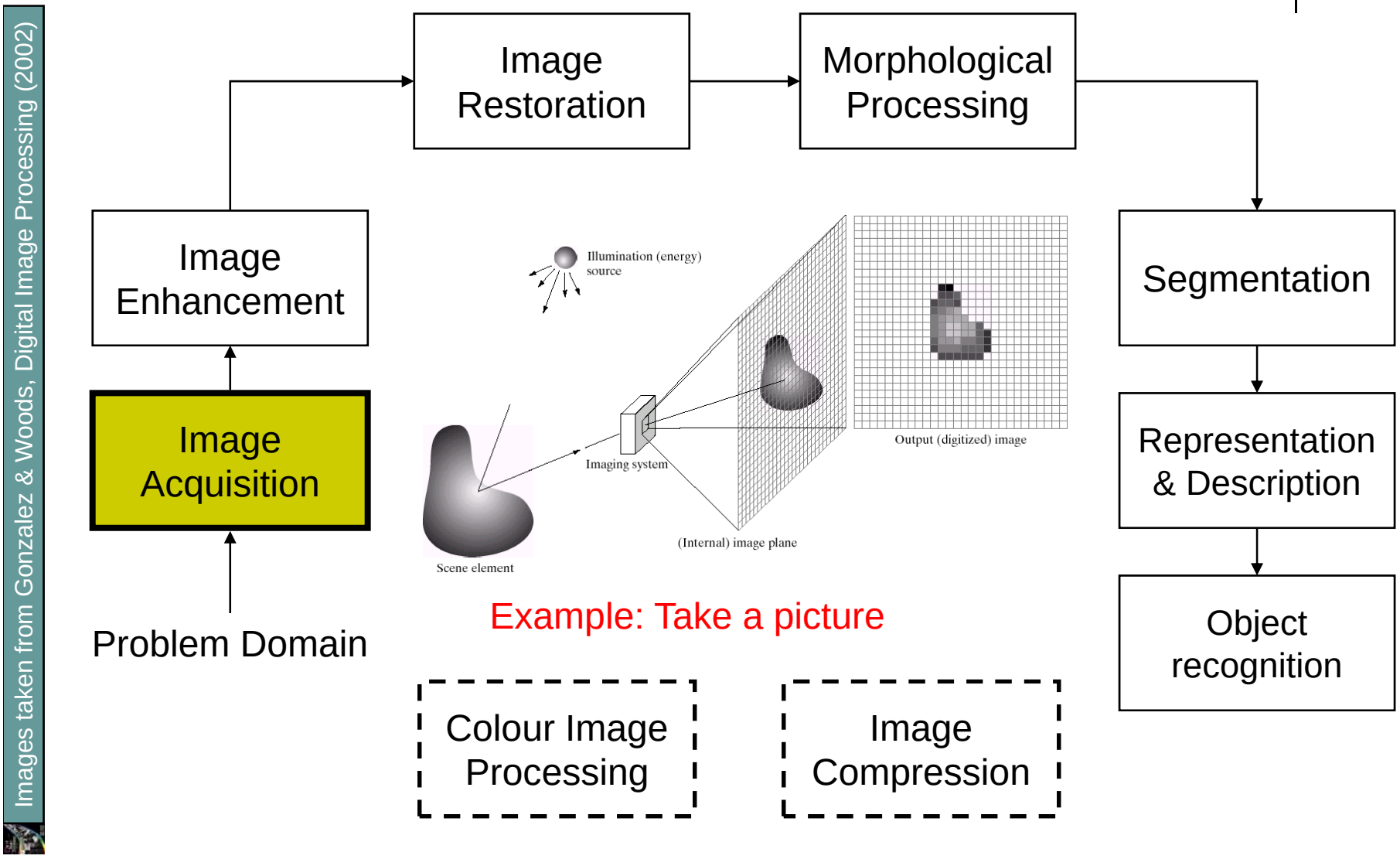
Image Processing

Image enhancement, noise removal, restoration,
feature detection, compression

Key Stages in Digital Image Processing

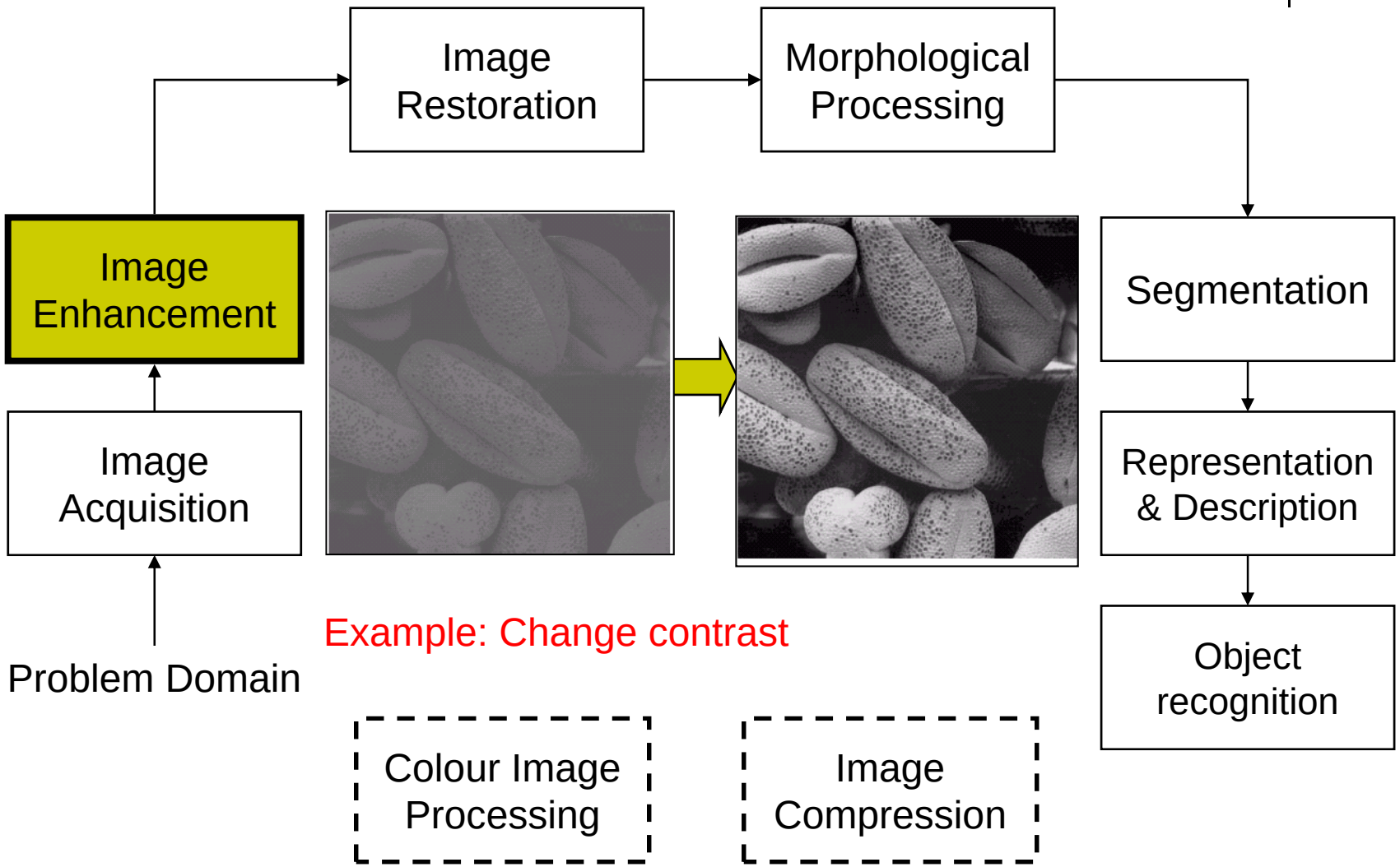


Key Stages in Digital Image Processing: Image Aquisition

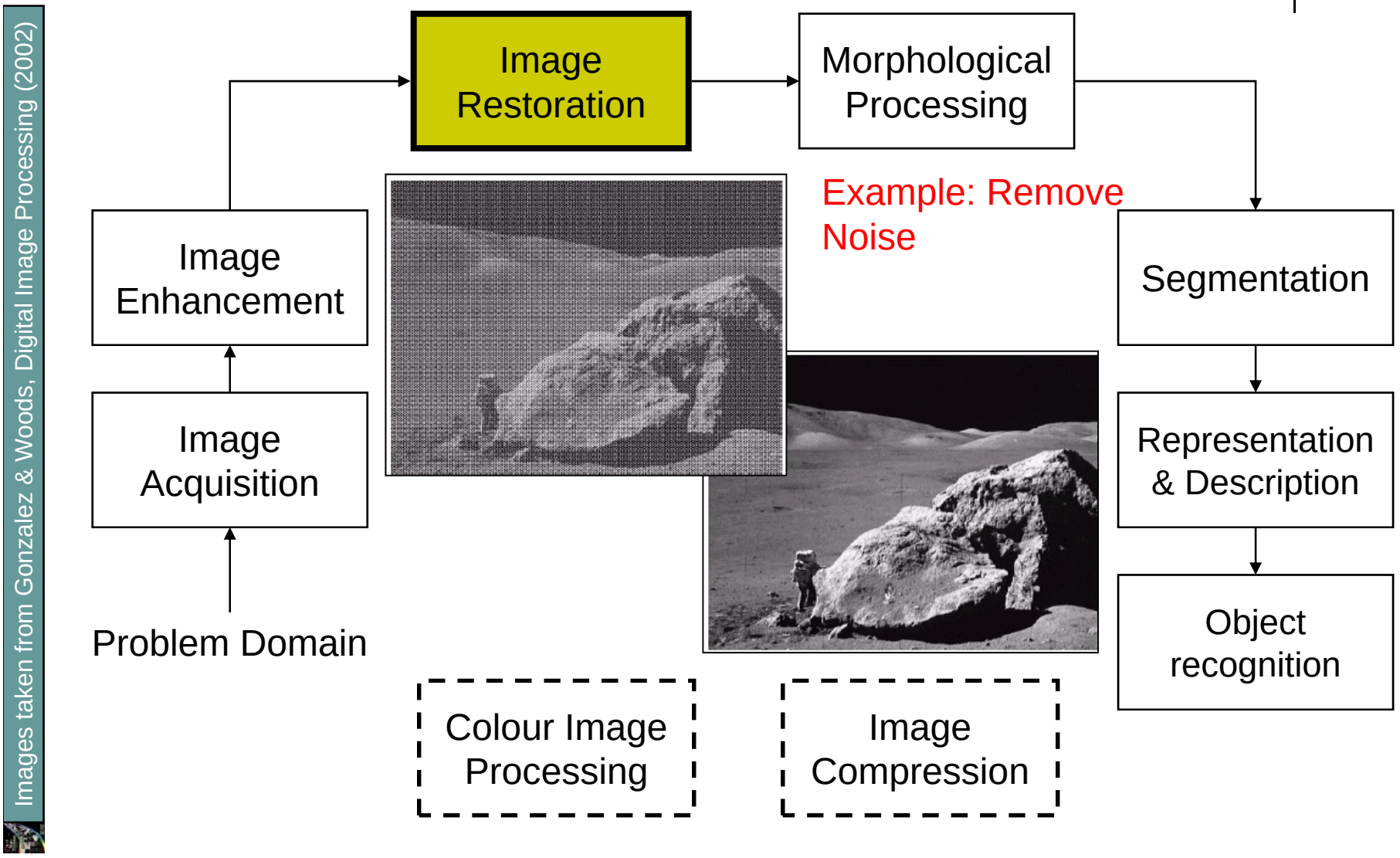


Images taken from Gonzalez & Woods, Digital Image Processing (2002)

Key Stages in Digital Image Processing: Image Enhancement

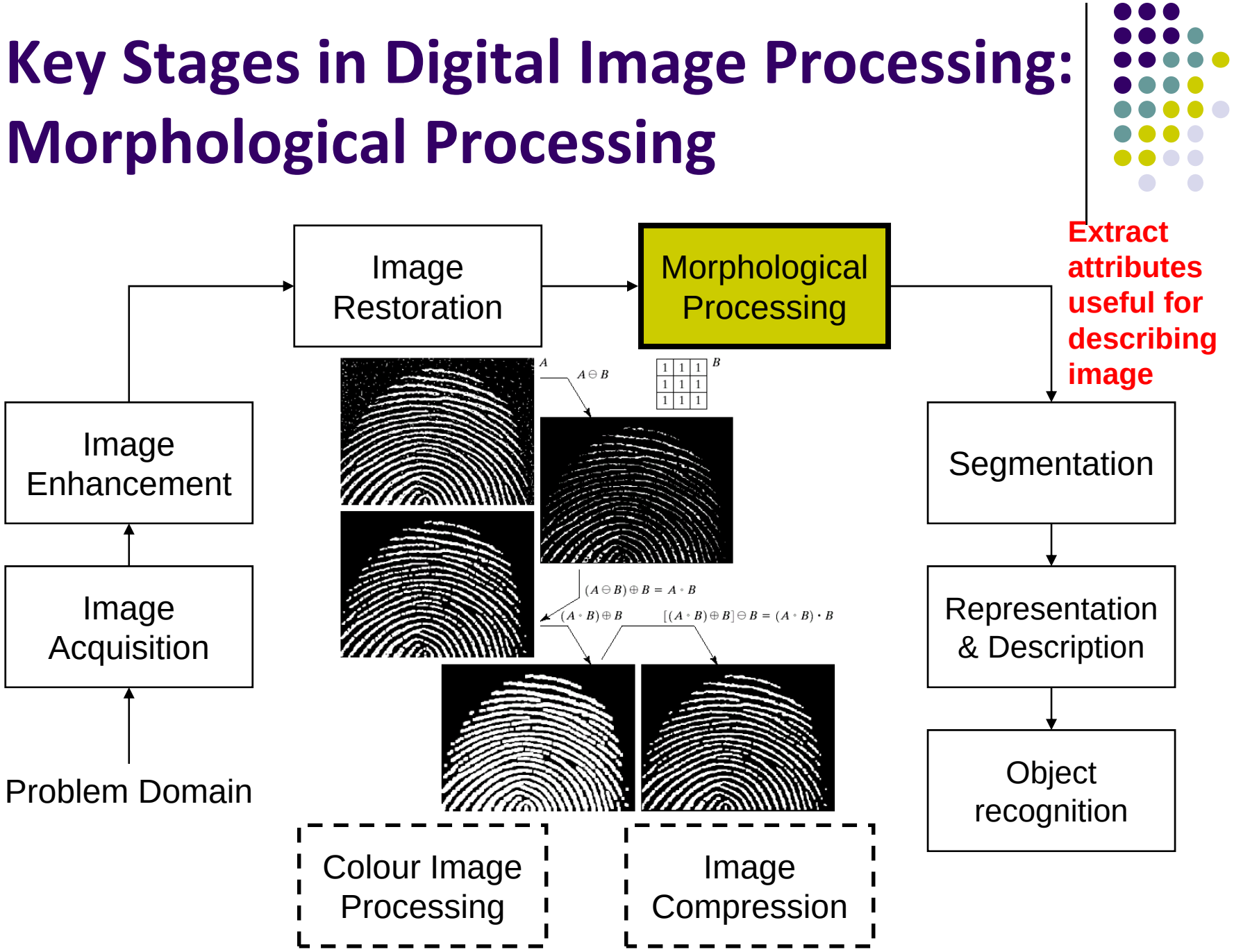


Key Stages in Digital Image Processing: Image Restoration



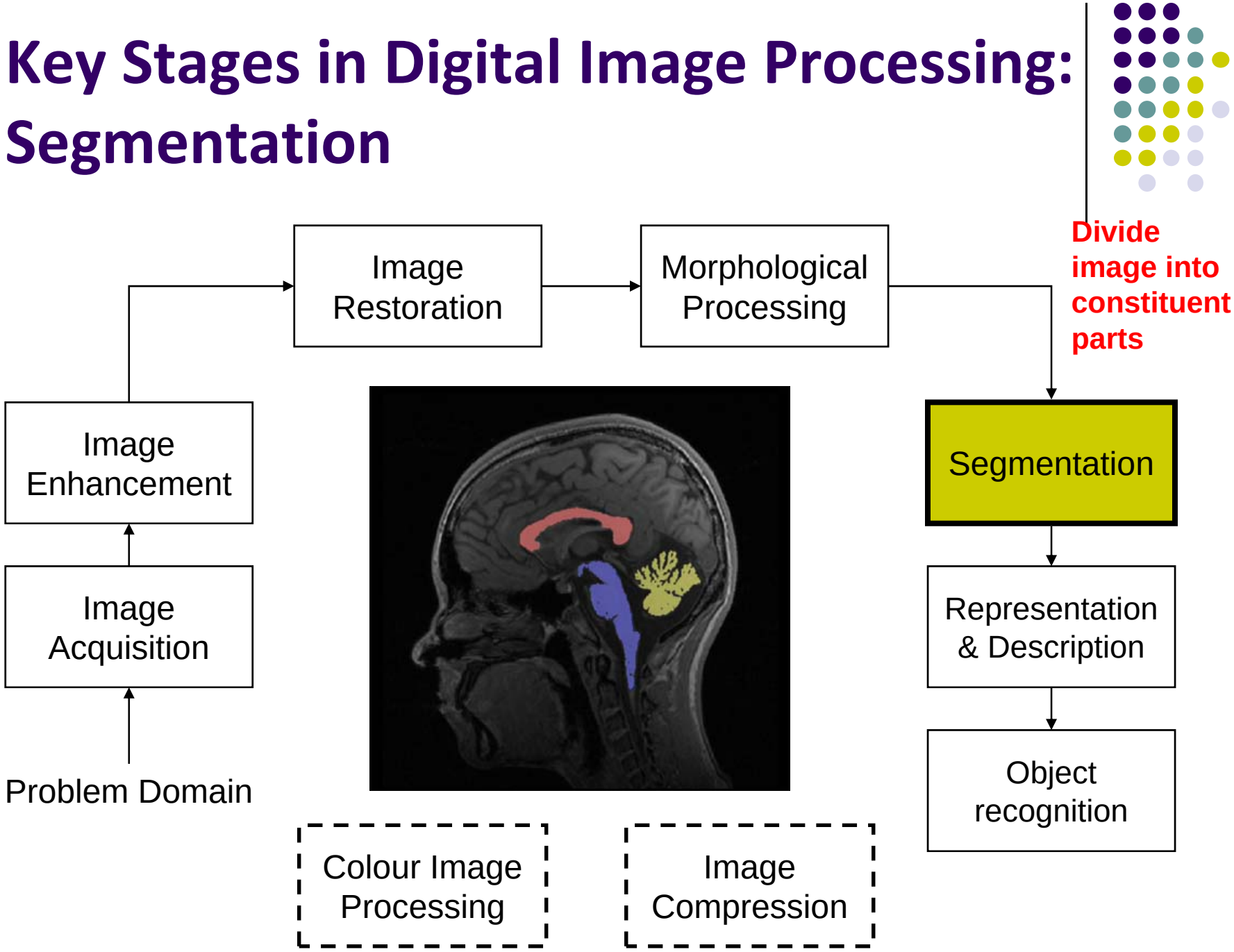
Images taken from Gonzalez & Woods, Digital Image Processing (2002)

Key Stages in Digital Image Processing: Morphological Processing



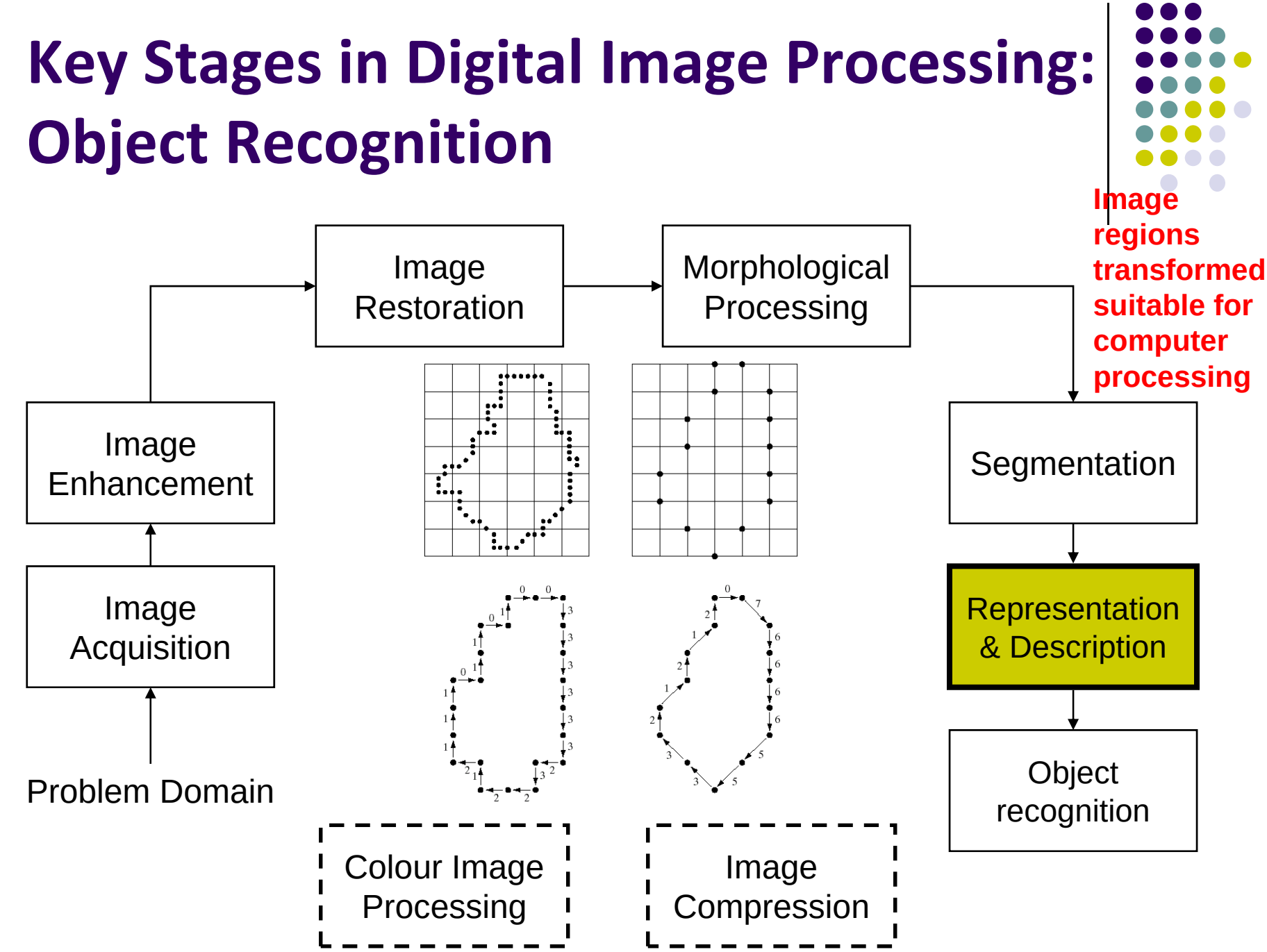
Images taken from Gonzalez & Woods, Digital Image Processing (2002)

Key Stages in Digital Image Processing: Segmentation



Images taken from Gonzalez & Woods, Digital Image Processing (2002)

Key Stages in Digital Image Processing: Object Recognition



Images taken from Gonzalez & Woods, Digital Image Processing (2002)

Key Stages in Digital Image Processing: Representation & Description

Finds & Labels objects in scene (e.g. motorbike)

Image Restoration

Morphological Processing

Image Enhancement

Segmentation

Image Acquisition

Representation & Description

Problem Domain

Object recognition

Colour Image Processing

Image Compression

Images taken from Gonzalez & Woods, Digital Image Processing (2002)

Key Stages in Digital Image Processing: Image Compression

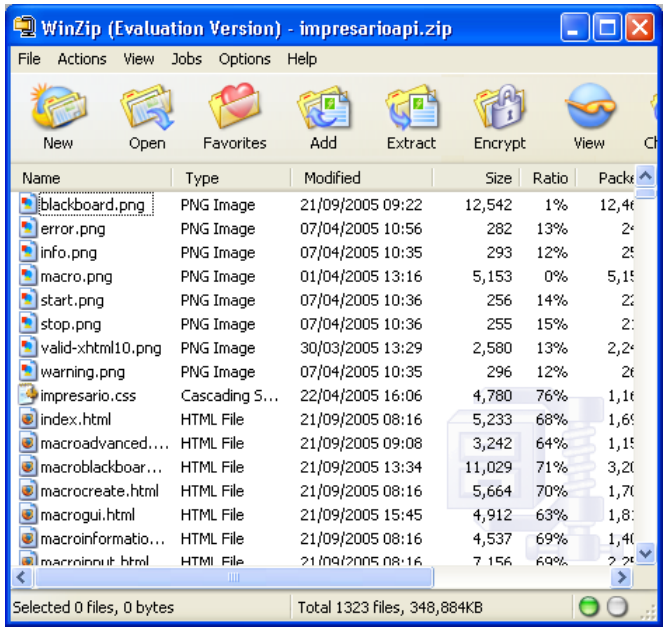
Reduce image size (e.g. JPEG)

Image Restoration

Morphological Processing

Image Enhancement

Segmentation



Representation & Description

Image Acquisition

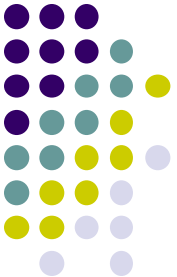
Object recognition

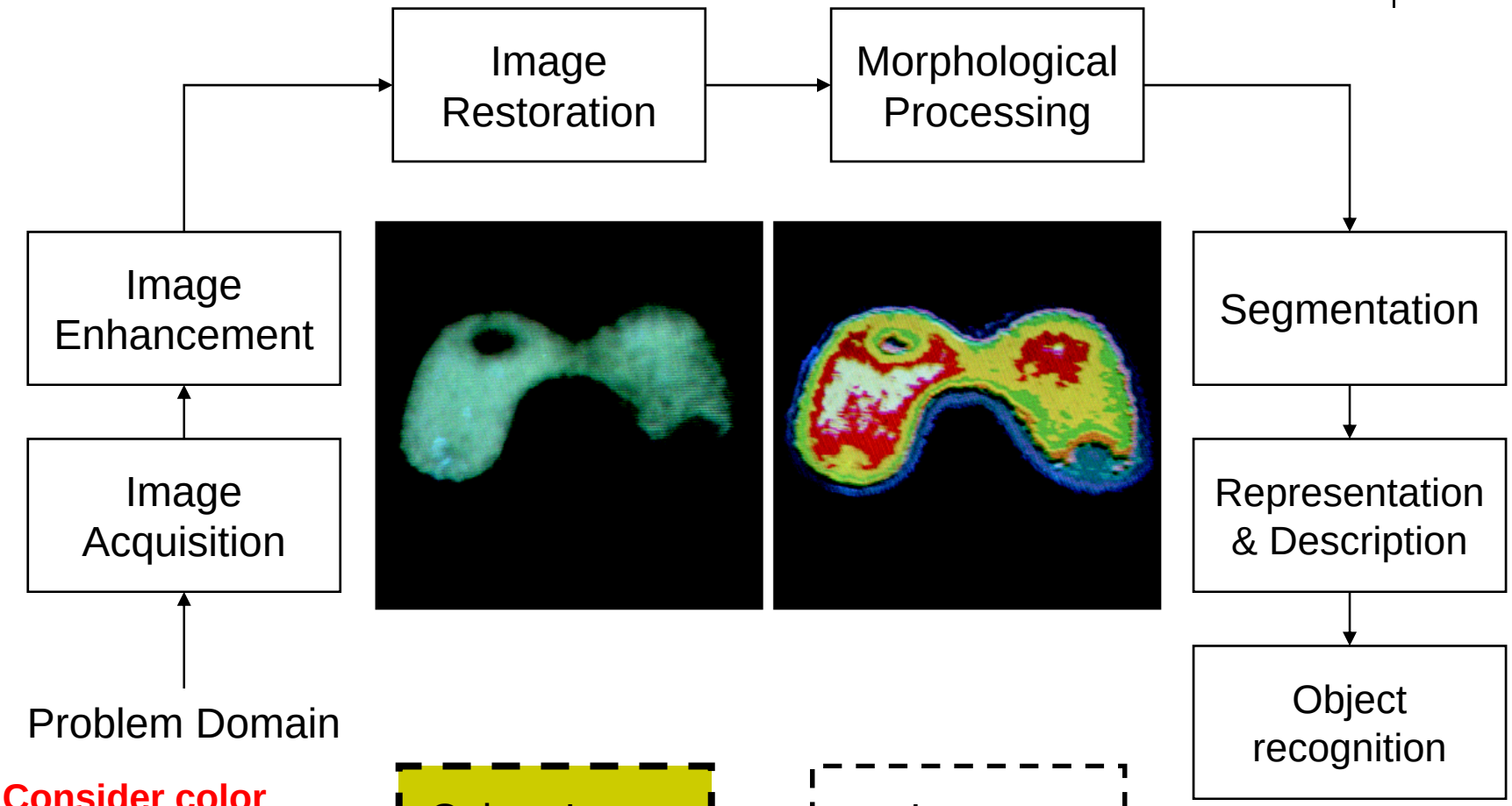
Problem Domain

Colour Image Processing

Image Compression

Key Stages in Digital Image Processing: Colour Image Processing

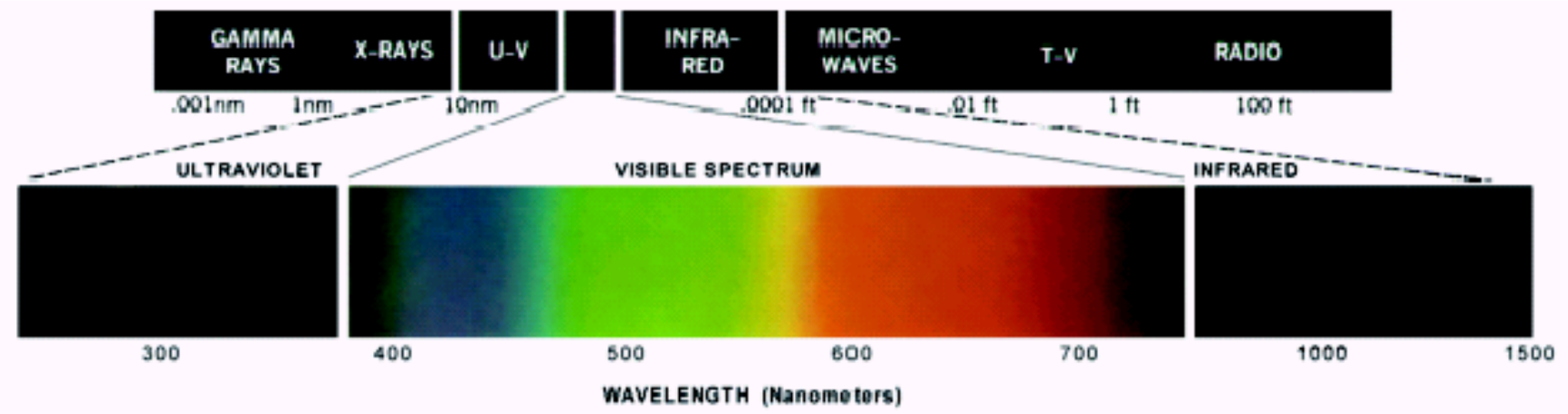




Light And The Electromagnetic Spectrum

- Light: just a particular part of electromagnetic spectrum that can be sensed by the human eye

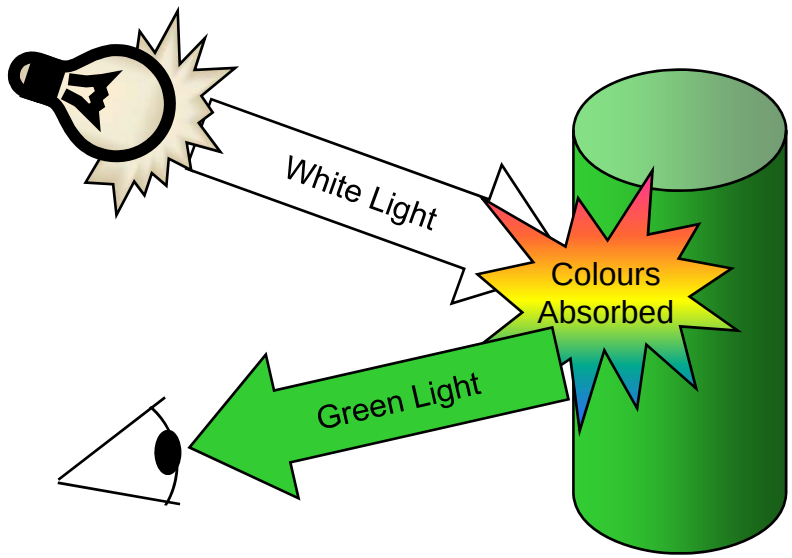
- The electromagnetic spectrum is split up according to the wavelengths of different forms of energy



Reflected Light

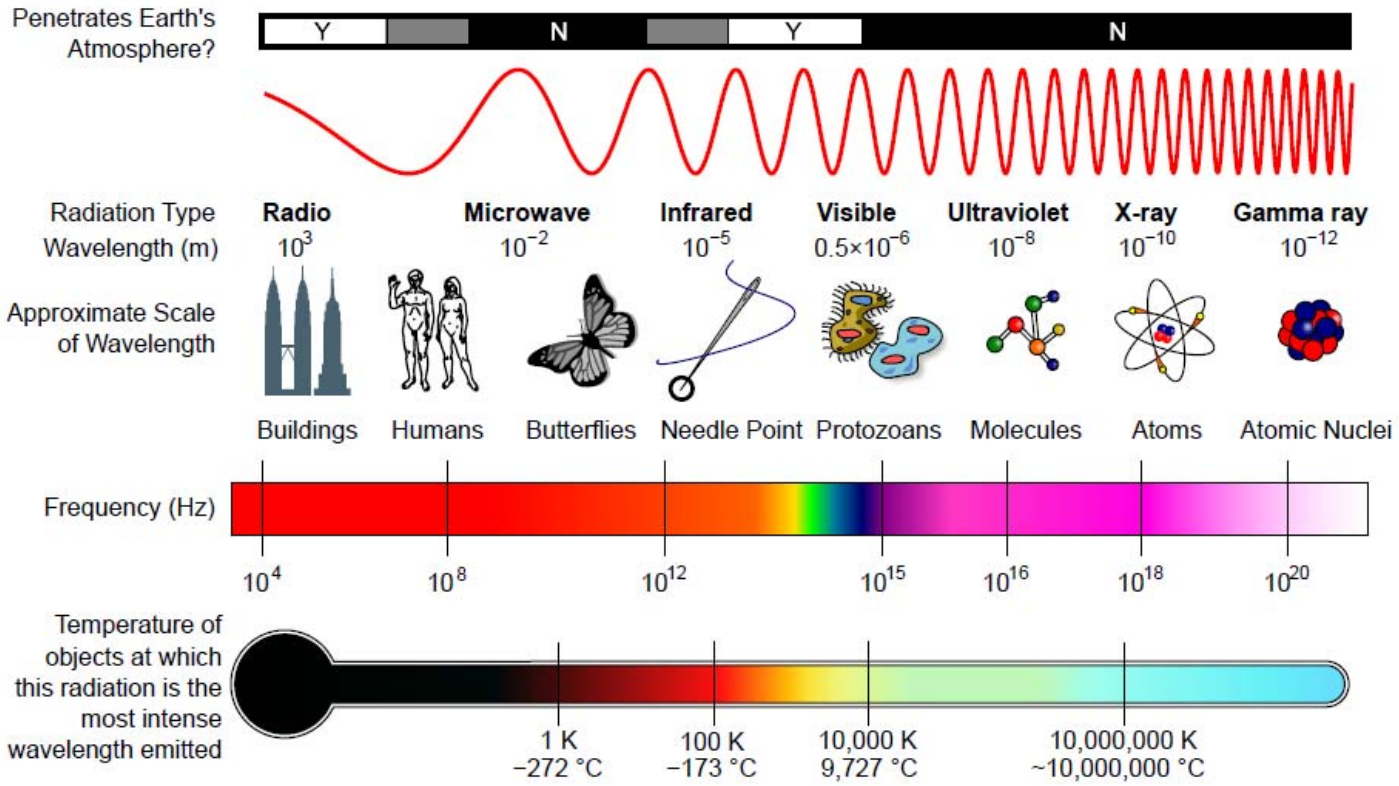
- The colours humans perceive are determined by nature of light reflected from an object

- For example, if white light (contains all wavelengths) is shone onto green object it absorbs most wavelengths absorbed except green wavelength (color)



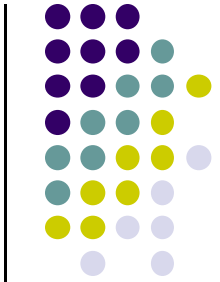
Electromagnetic Spectrum and IP

- Images can be made from any form of EM radiation



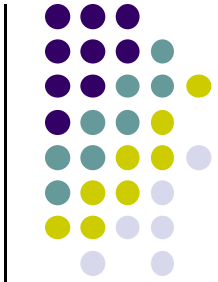
From Wikipedia

Images from Different EM Radiation



- Radar imaging (radio waves)
- Magnetic Resonance Imaging (MRI) (Radio waves)
- Microwave imaging
- Infrared imaging
- Photographs
- Ultraviolet imaging telescopes
- X-rays and Computed tomography
- Positron emission tomography (gamma rays)
- Ultrasound (not EM waves)

Human Visual System: Structure Of The Human Eye



- The lens focuses light from objects onto the retina

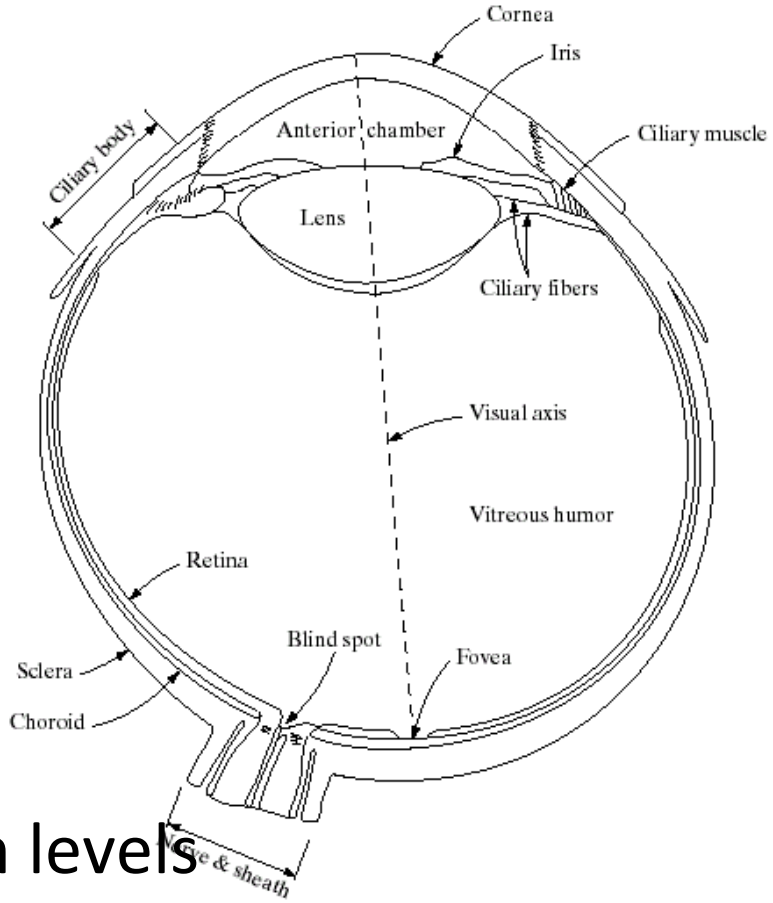
- Retina covered with light receptors called ***cones*** (6-7 million) and ***rods*** (75-150 million)

  视椎细胞

- Cones concentrated around fovea. Very sensitive to colour

  视杆细胞

- Rods more spread out and sensitive to low illumination levels



Images taken from Gonzalez & Woods, Digital Image Processing (2002)

Image Formation In The Eye

- Muscles in eye can change the shape of the lens allowing us focus on near or far objects
- An image is focused onto retina exciting the rods and cones and send signals to the brain

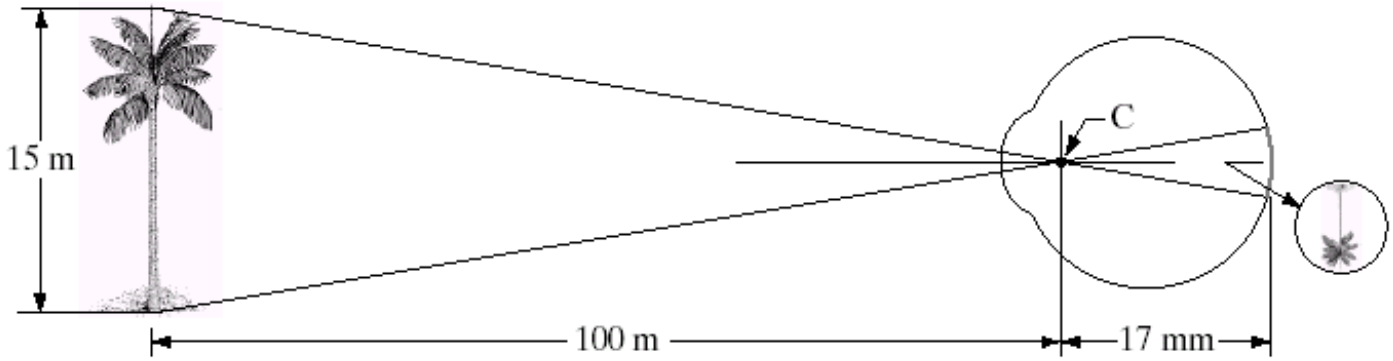
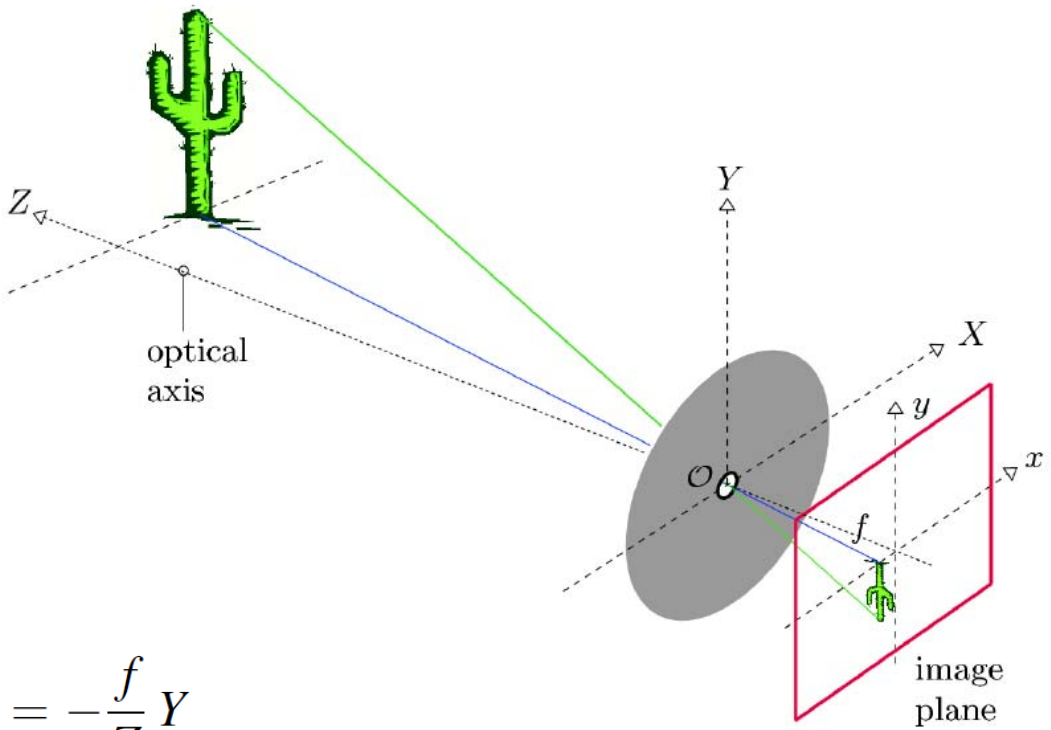


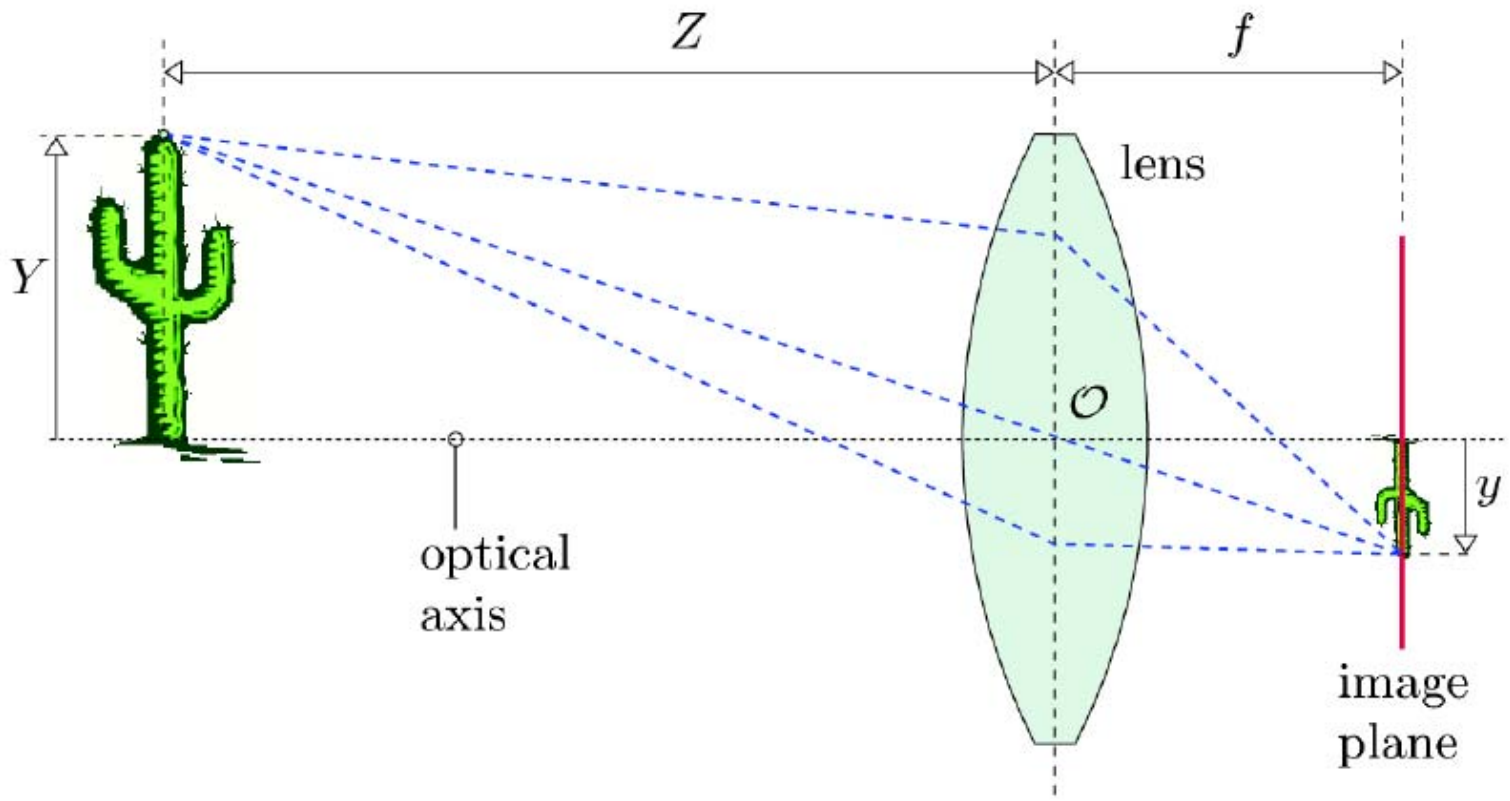
Image Formation

- The Pinhole Camera (abstraction)
  - First described by ancient Chinese and Greeks (300-400AD)

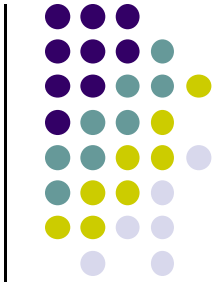


$$x = -\frac{f}{Z}X, \qquad y = -\frac{f}{Z}Y$$

Thin Lens

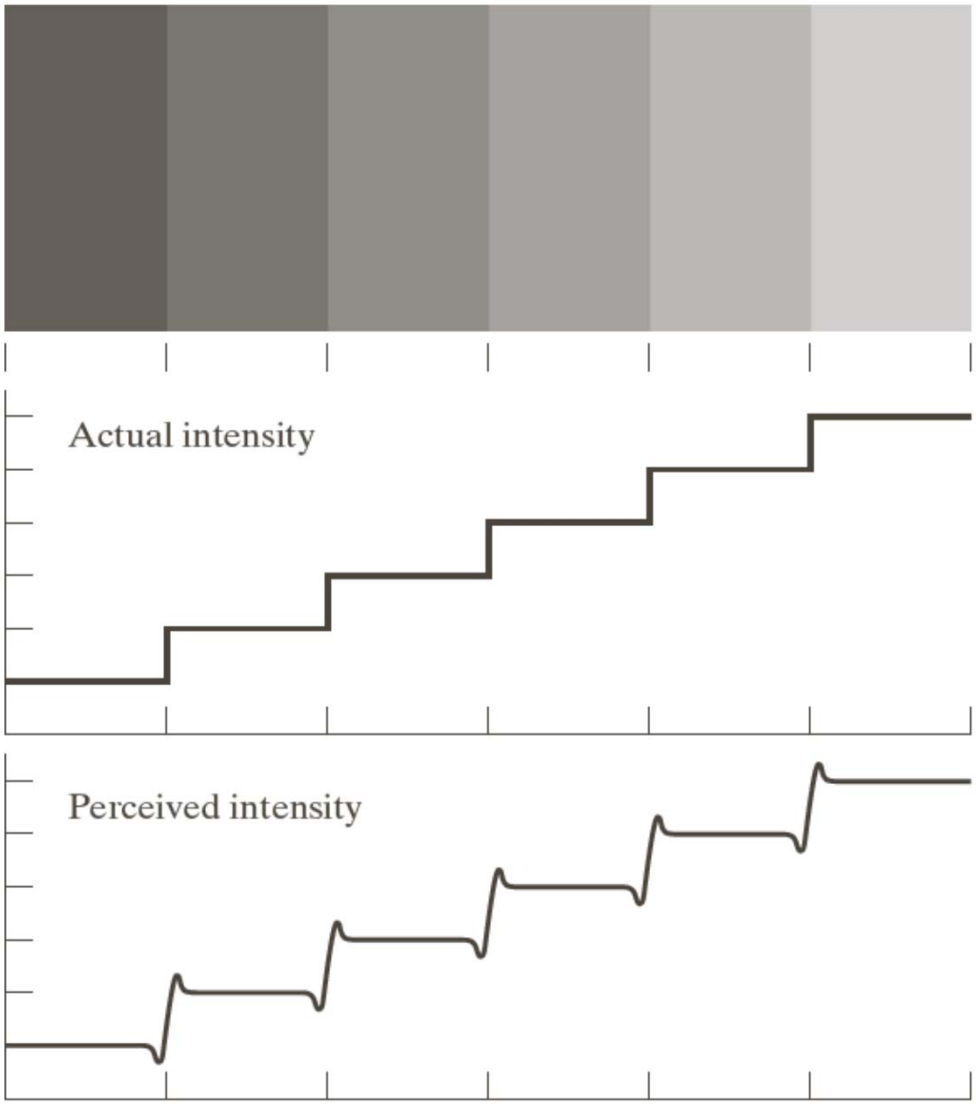


Brightness Adaptation & Discrimination



- The human visual system can perceive approximately 10^{10} different light intensity levels
- However, at any one time we can only discriminate between a much smaller number – ***brightness adaptation***
- Similarly, ***perceived intensity*** of a region is related to the light intensities of the regions surrounding it

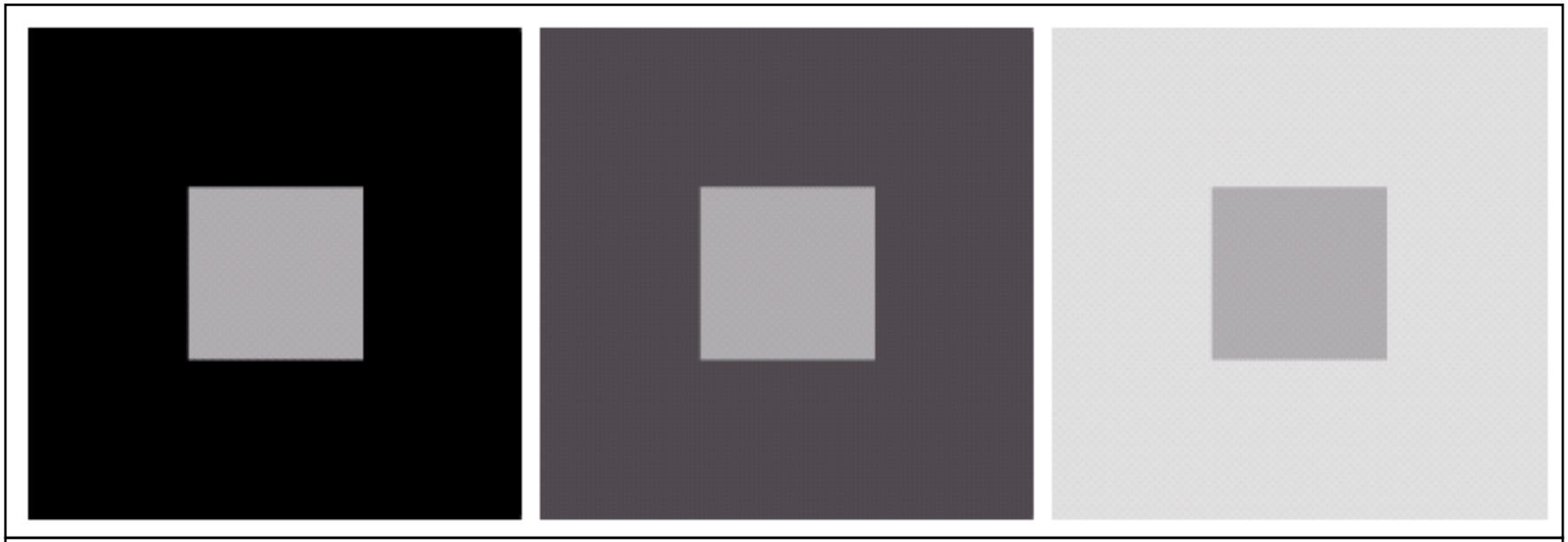
Brightness Adaptation & Discrimination: Mach Band Effect



Perceived intensity overshoots or undershoots at areas of intensity change

Images taken from Gonzalez & Woods, Digital Image Processing (2002)

Brightness Adaptation & Discrimination



An example of *simultaneous contrast*

All inner squares have same intensity but appear darker as outer square (surrounding area) gets lighter

Images taken from Gonzalez & Woods, Digital Image Processing (2002)

Image Acquisition

- Images typically generated by ***illuminating a scene*** and absorbing energy reflected by scene objects

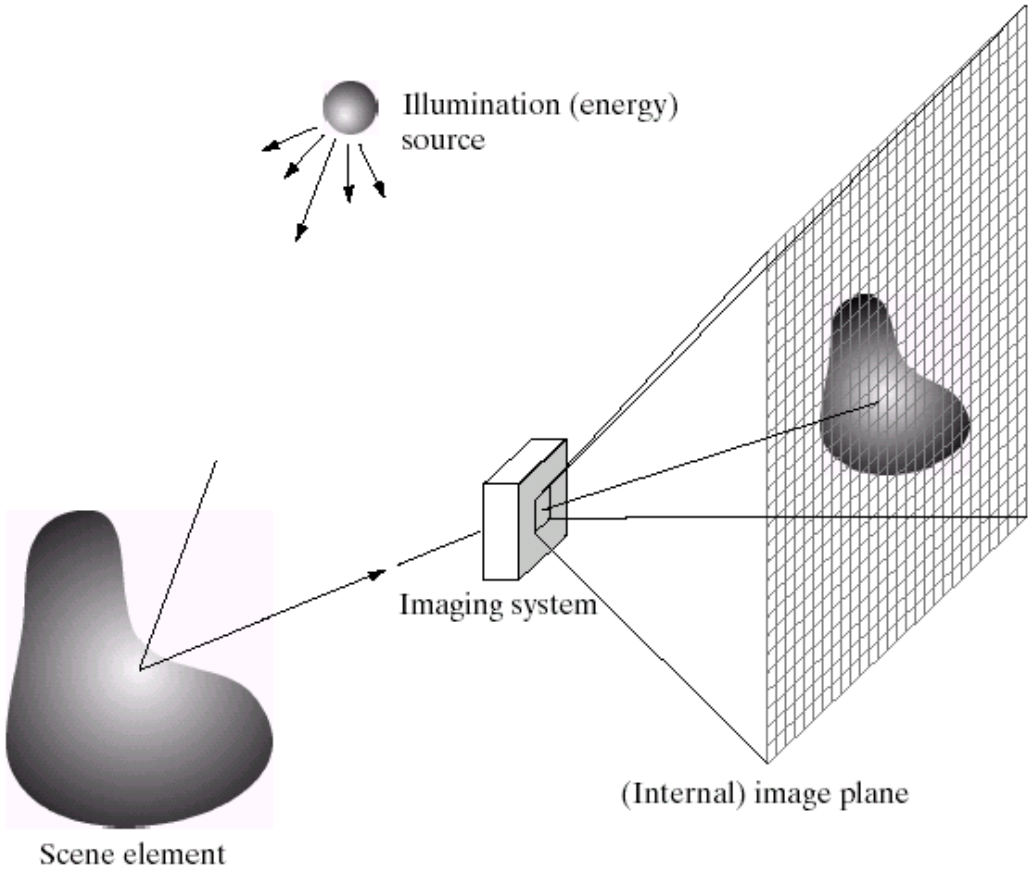
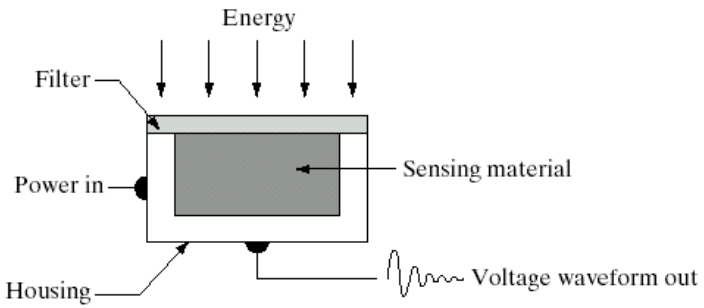


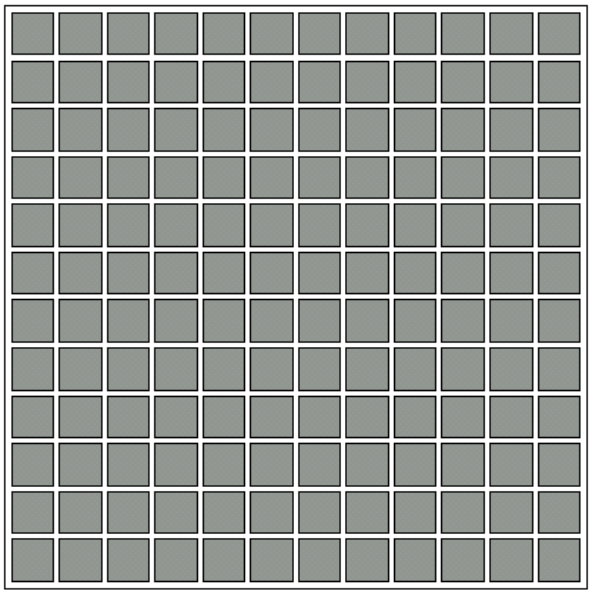
Image Sensing

- Incoming energy (e.g. light) lands on a sensor material responsive to that type of energy, generating a voltage
- Collections of sensors are arranged to capture images



Imaging Sensor

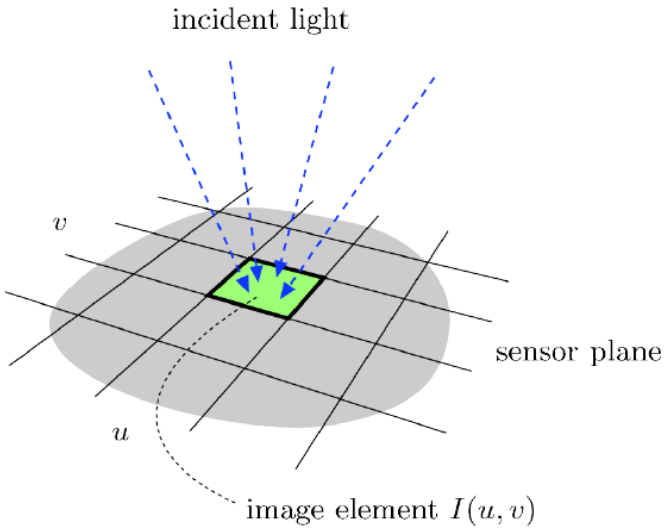
Line of Image Sensors



Array of Image Sensors

Spatial Sampling

- Cannot record image values for all (x,y)
- Sample/record image values at discrete (x,y)
- Sensors arranged in grid to sample image



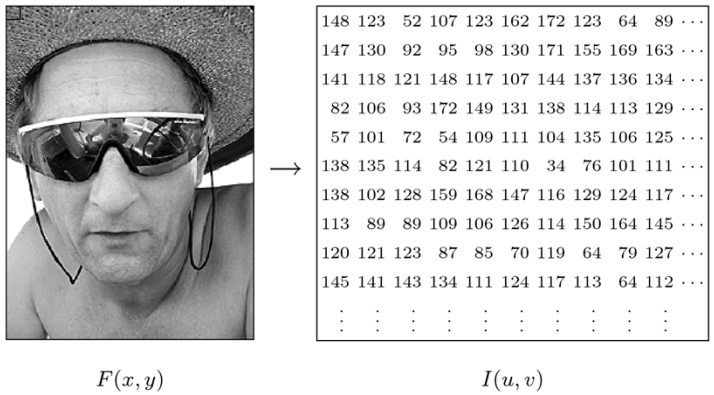
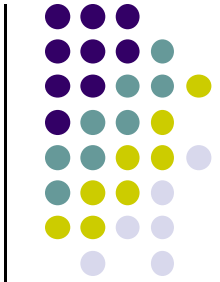


Image (Spatial) Sampling



- A digital sensor can only measure a limited number of **samples** at a **discrete** set of energy levels

- **Sampling** can be thought of as:

  Continuous signal x comb function

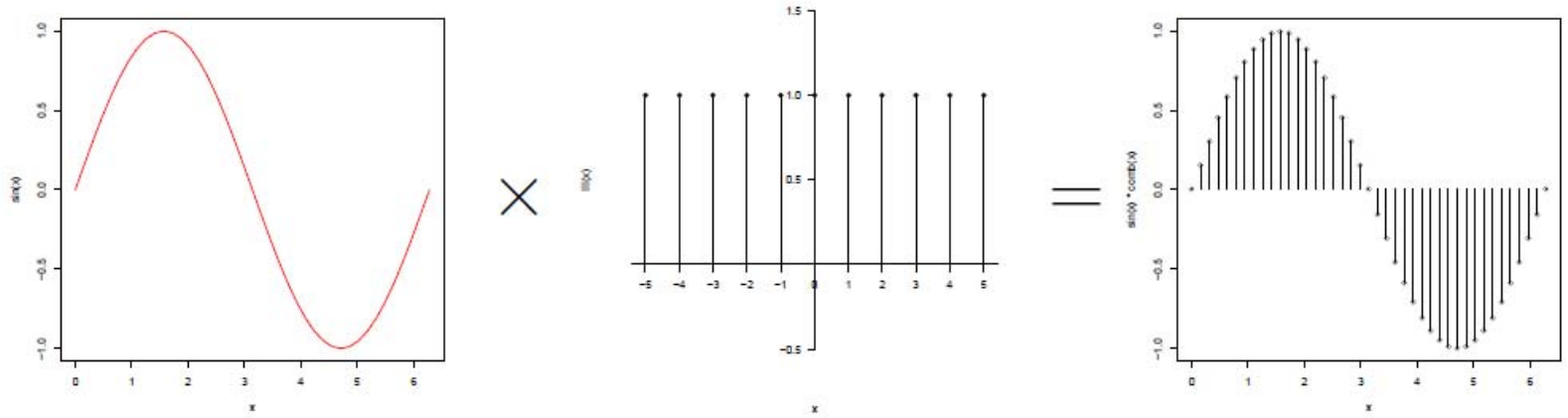


Image Quantization

- ***Quantization*: process of converting continuous **analog** signal into its digital representation**
- Discretize image *I(u,v)* values
- Limit values image can take

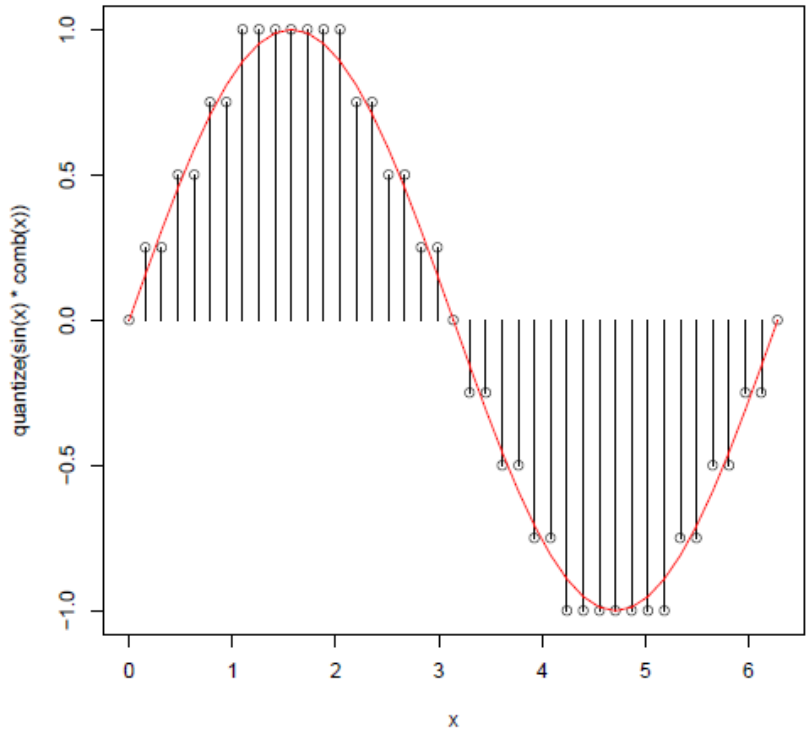


Image Sampling And Quantization

- Sampling and quantization generates **approximation** of a real world scene

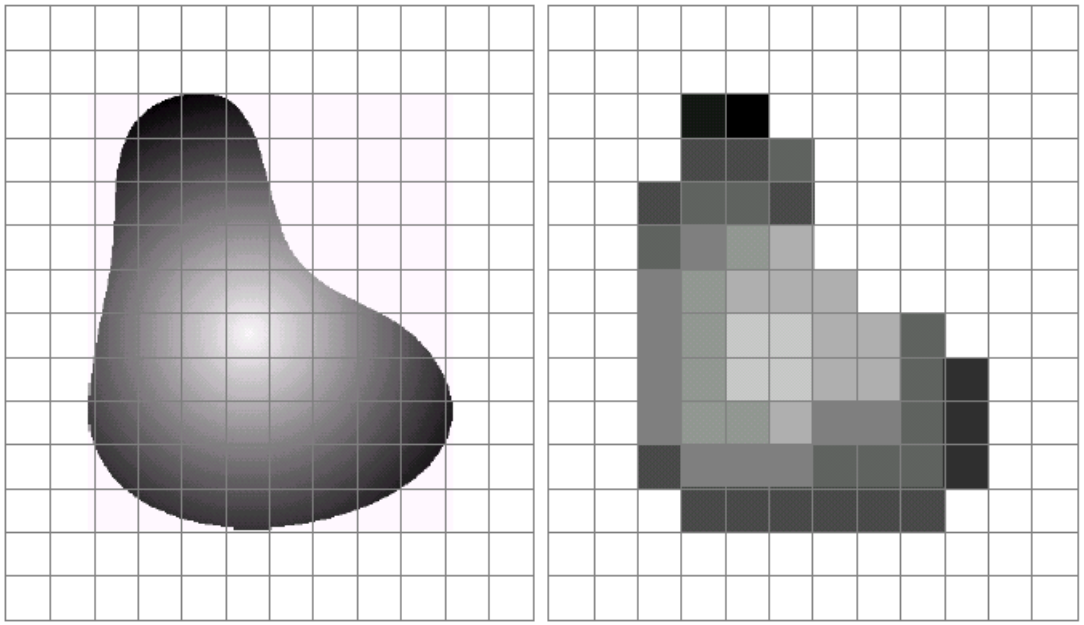
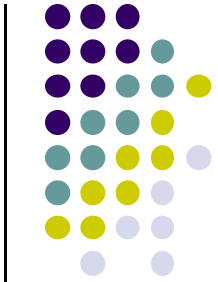


Image as Discrete Function



After spatial sampling and quantization, an image is a discrete function. The image domain Ω is now discrete:

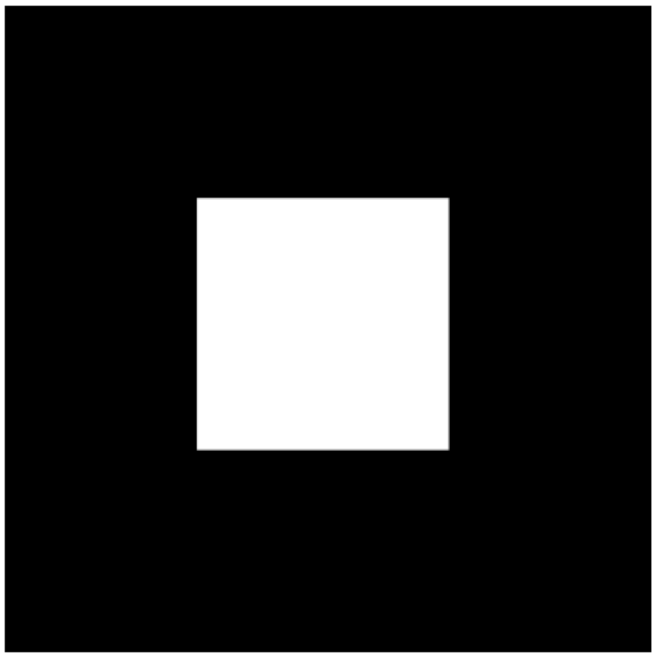
$$\Omega \subset \mathbb{N}^2,$$

and so is the image range:

$$I : \Omega \to \{1, \ldots, K\},$$

where $K \in \mathbb{N}$.

Image as a Function



A simple image

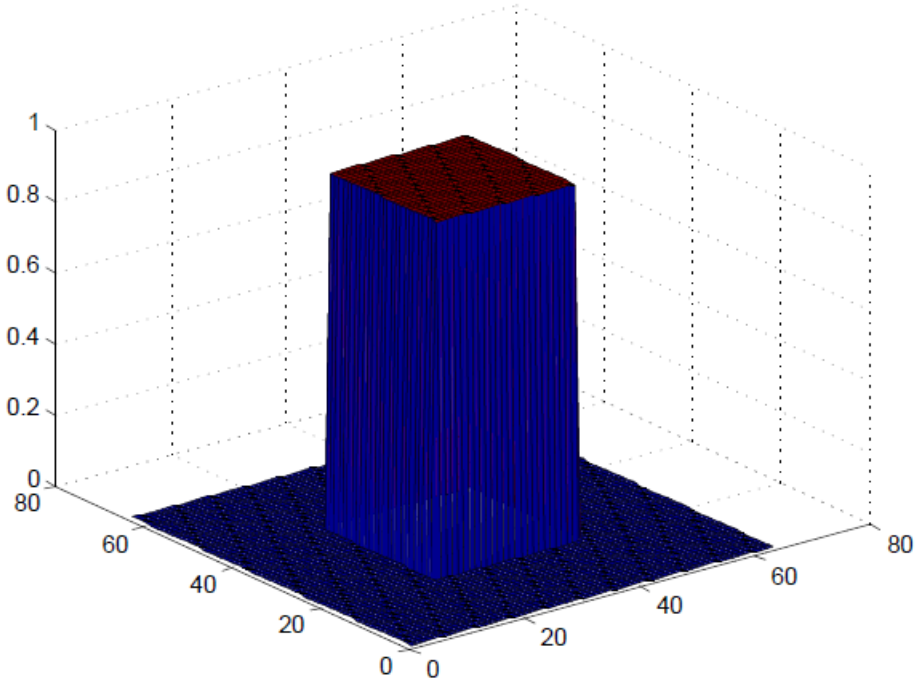
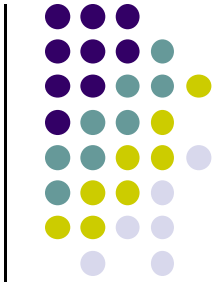
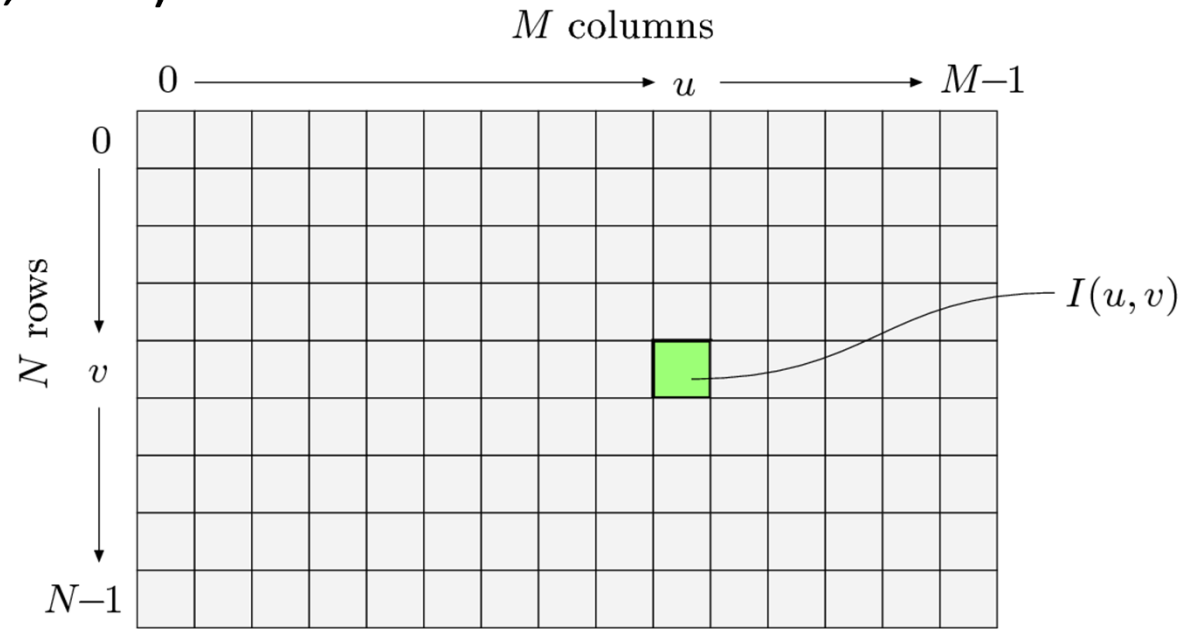


Image function as a height field

Representing Images



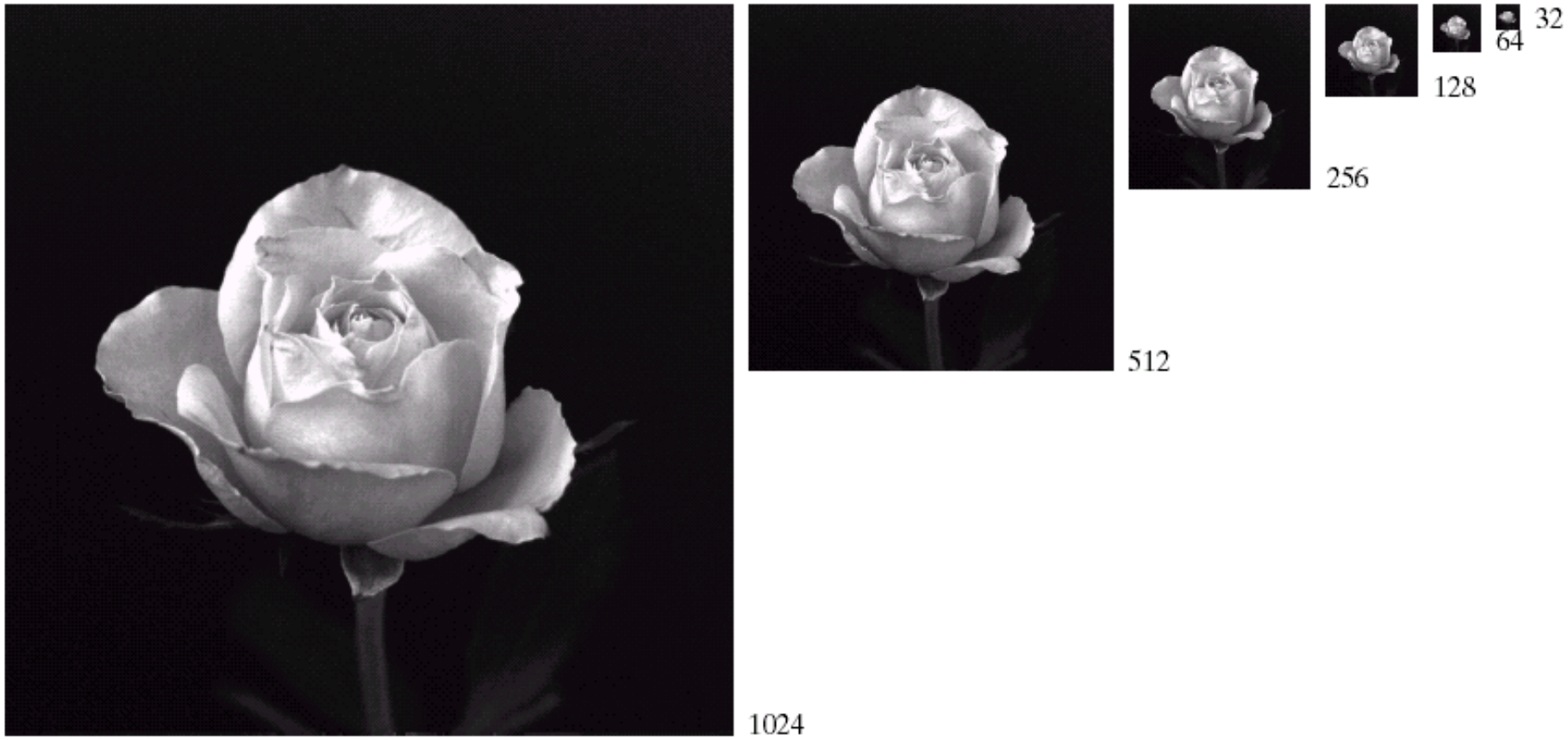
- Image data structure is 2D array of pixel values
- Pixel values are gray levels in range 0-255 or RGB colors
- Array values can be any data type (bit, byte, int, float, double, etc.)



Spatial Resolution

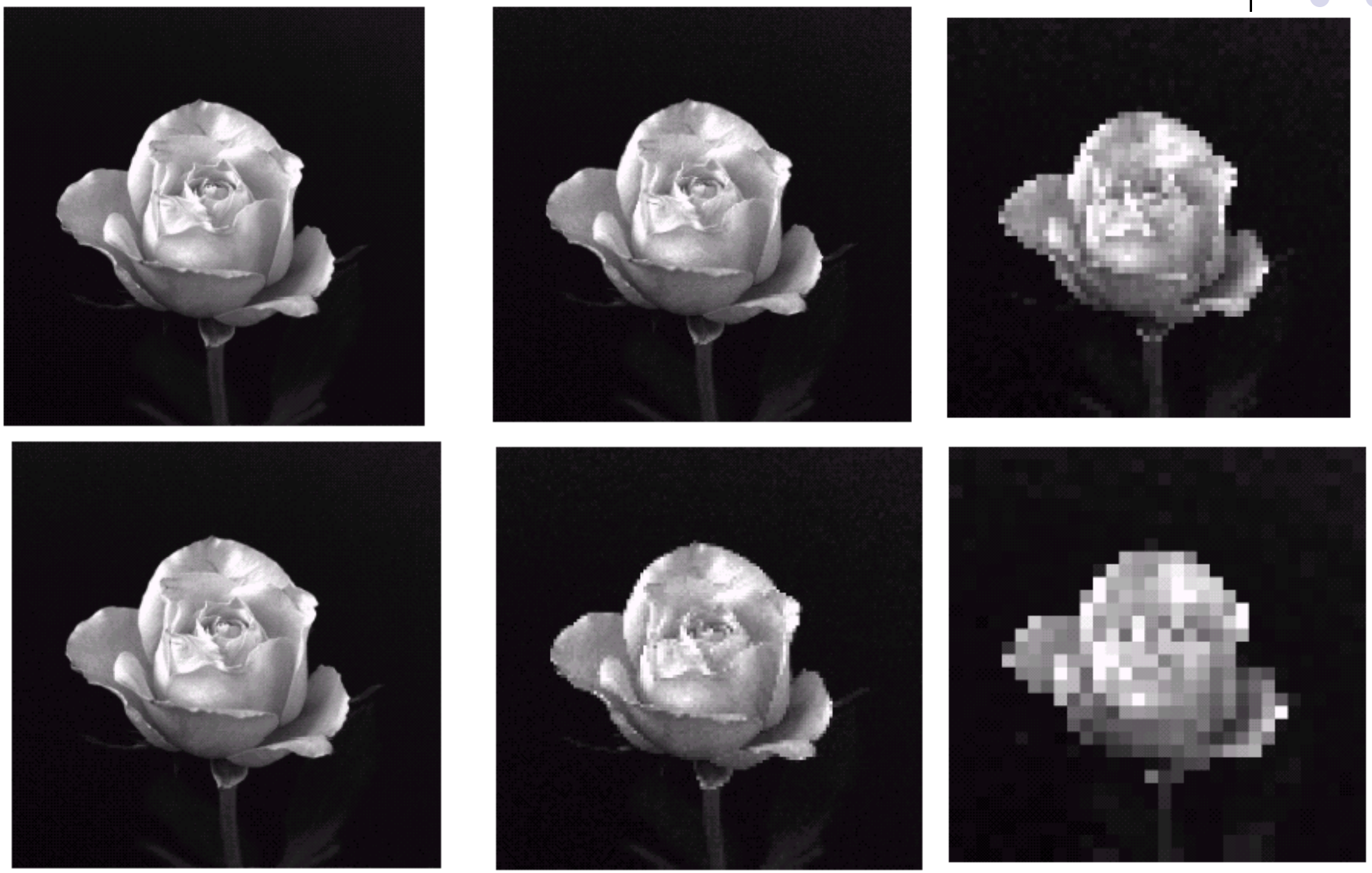
- *The **spatial resolution*** of an image is determined by how fine/coarse sampling was carried out
- **Spatial resolution:** smallest discernable image detail
  - Vision specialists talk about image resolution
  - Graphic designers talk about *dots per inch* (DPI)

Spatial Resolution



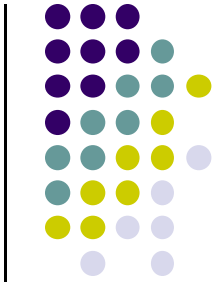
Images taken from Gonzalez & Woods, Digital Image Processing (2002)

Spatial Resolution: Stretched Images



Images taken from Gonzalez & Woods, Digital Image Processing (2002)

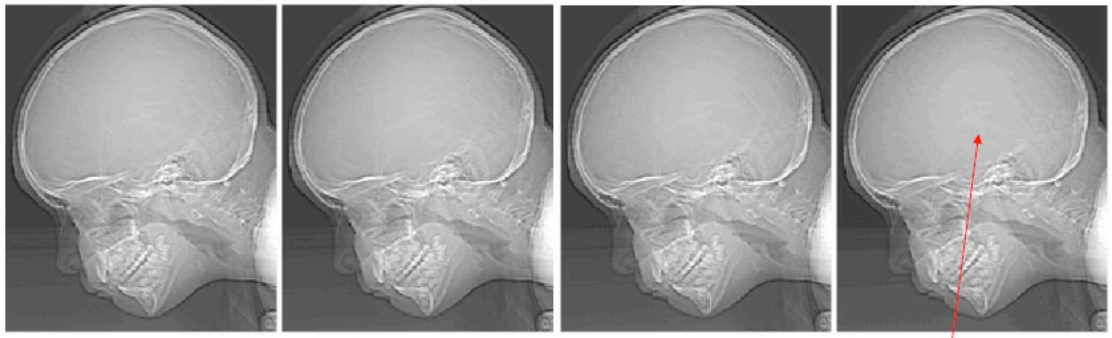
Intensity Level Resolution



- ***Intensity level resolution***: number of intensity levels used to represent the image
  - The more intensity levels used, the finer the level of detail discernable in an image
  - Intensity level resolution usually given in terms of 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

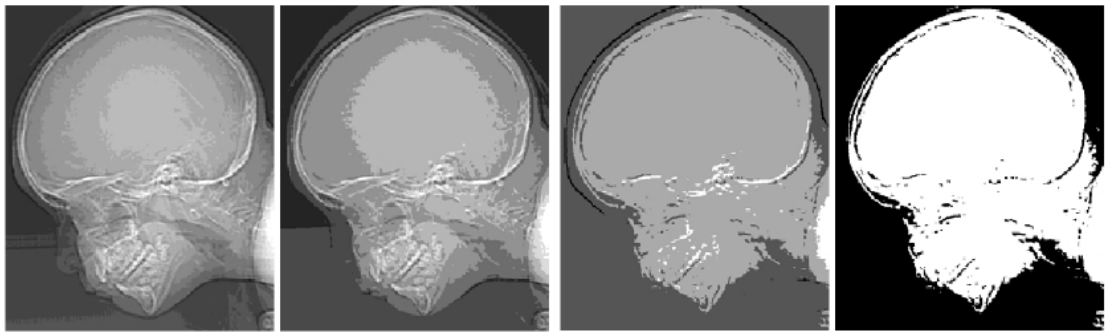
Intensity Level Resolution



8 bits, 256 levels | 7 bits, 128 levels | 6 bits, 64 levels | 5 bits, 32 levels

Ridgelike structures false contours

All with a spatial resolution of 452 x 374

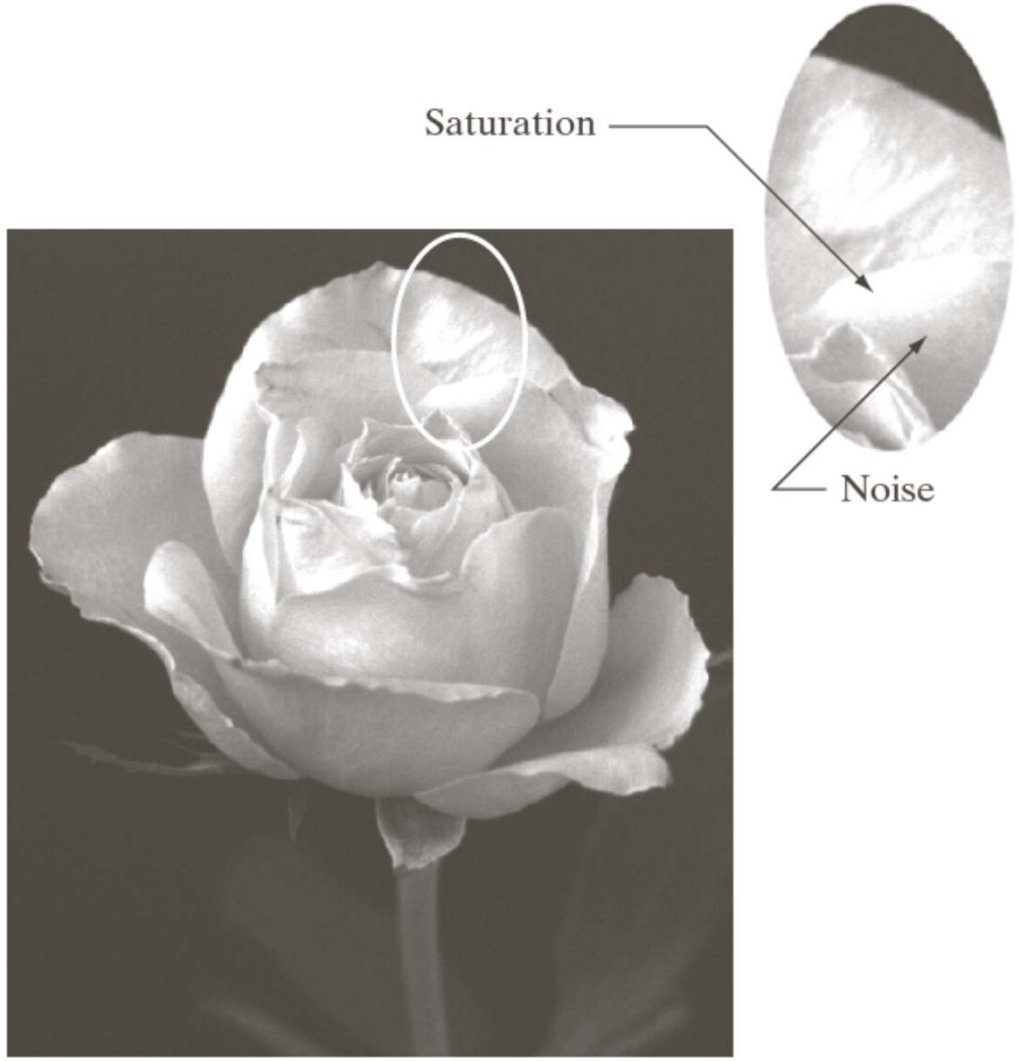


4 bits, 16 levels | 3 bits, 8 levels | 2 bits, 4 levels | 1 bits, 2 levels

All with a spatial resolution of 452 x 374

Images taken from Gonzalez & Woods, Digital Image Processing (2002)

Saturation & Noise

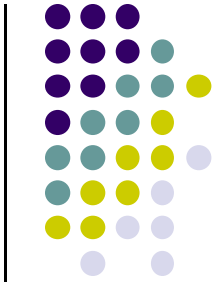


Saturation: highest intensity value above which color is washed out

Noise: grainy texture pattern

Images taken from Gonzalez & Woods, Digital Image Processing (2002)

Resolution: How Much Is Enough?



- The big question with resolution is always *how much is enough*?
  - Depends on what is in the image *(details)* and what you would like to do with it *(applications)*
  - Key questions:
    - Does image look aesthetically pleasing?
    - Can you see what you need to see in image?

Resolution: How Much Is Enough?

- **Example:** Picture on right okay for counting number of cars, but not for reading the number plate

Intensity Level Resolution

Low Detail

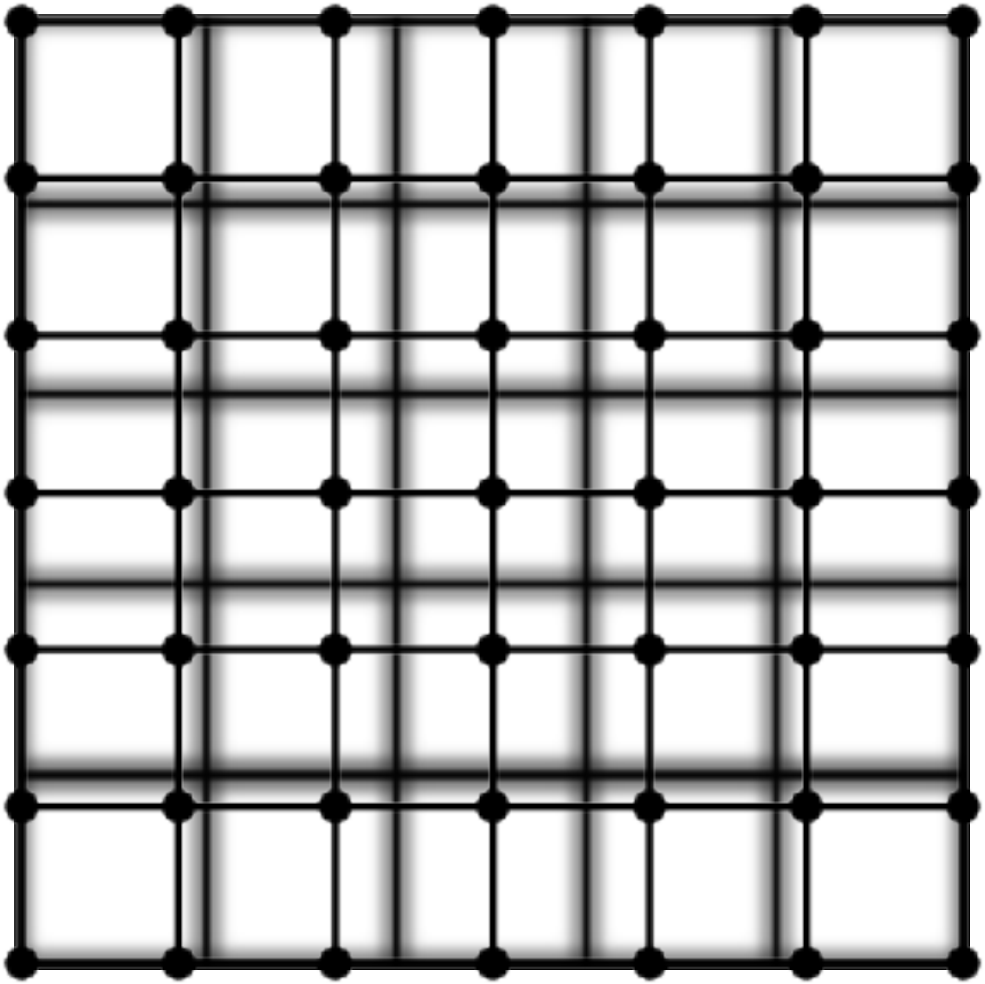
Medium Detail

High Detail

Images taken from Gonzalez & Woods, Digital Image Processing (2002)

Resampling & Interpolation

First step: align the grid.



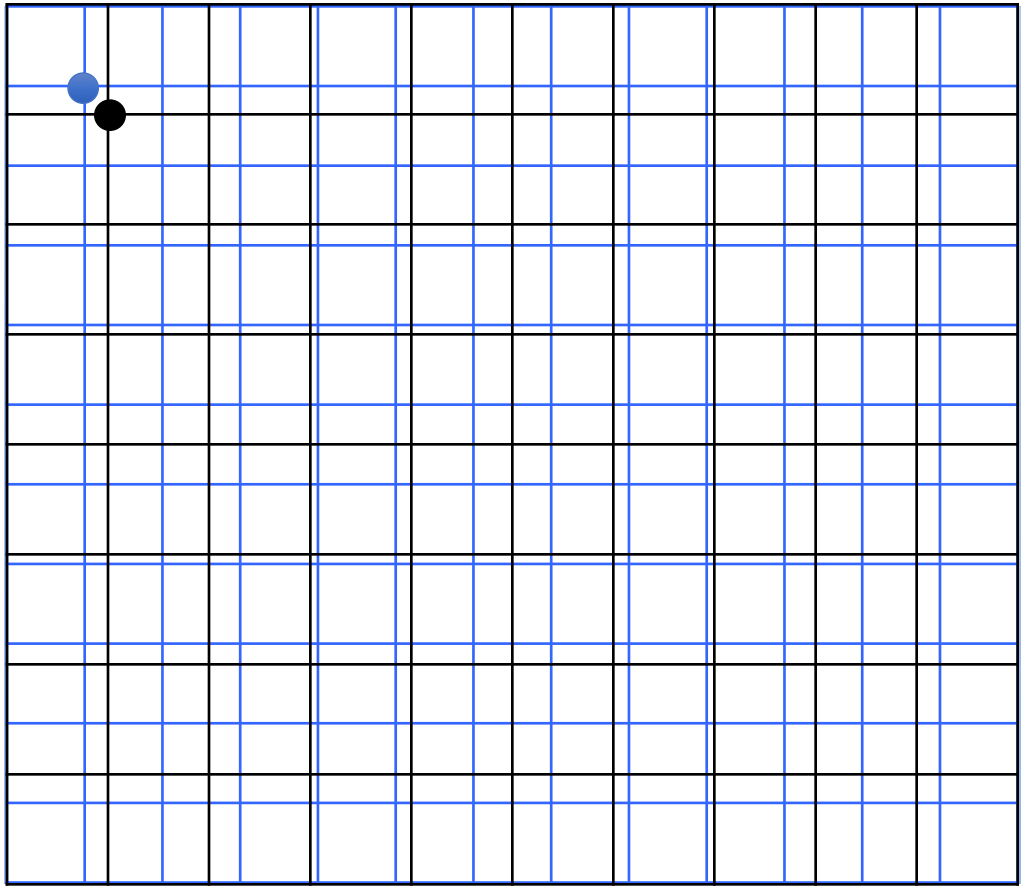
Second step:
Set the values at the new grid.

How can we get the values at **sub-pixel** locations?

Nearest Neighbor Interpolation

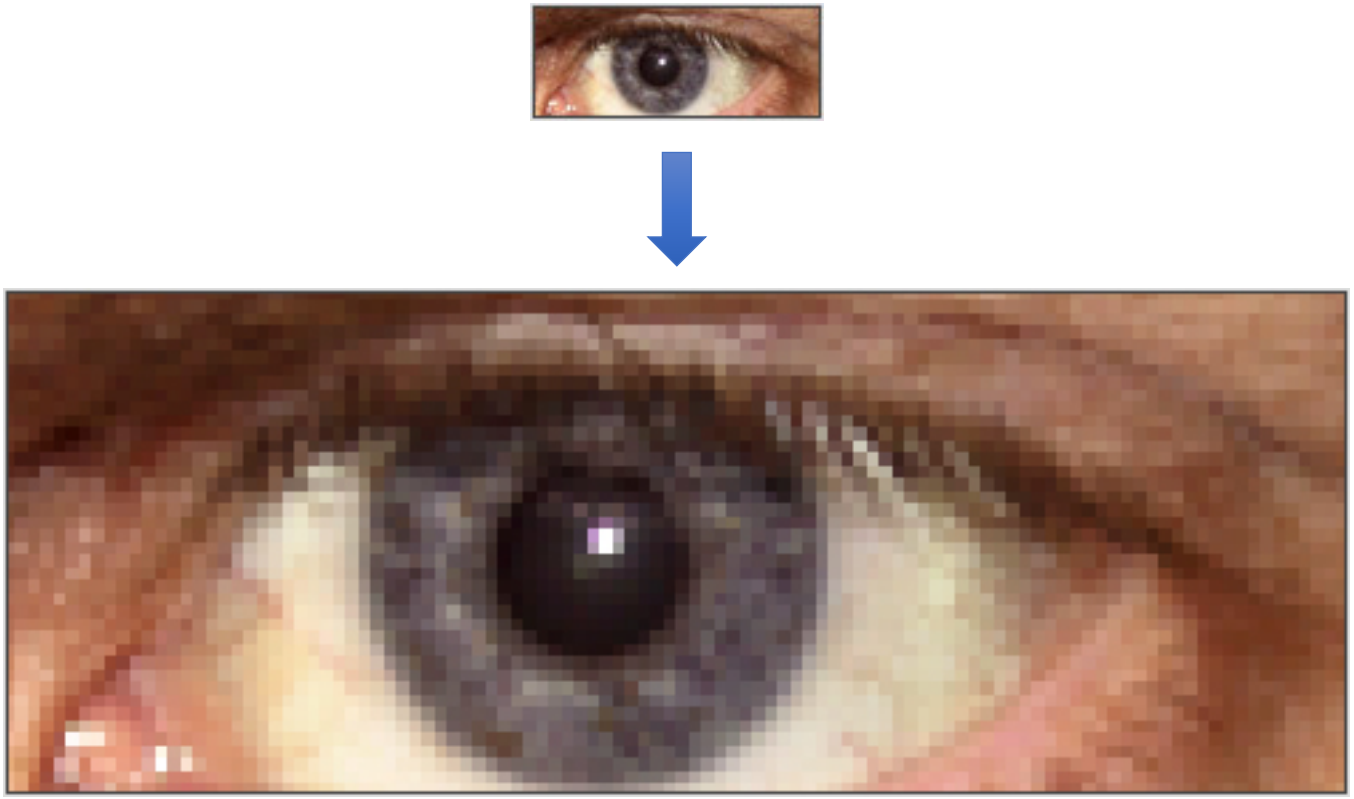
For each new pixel, look for the closest pixel in the original image and assign the value of that pixel to the new pixel.

will take value of

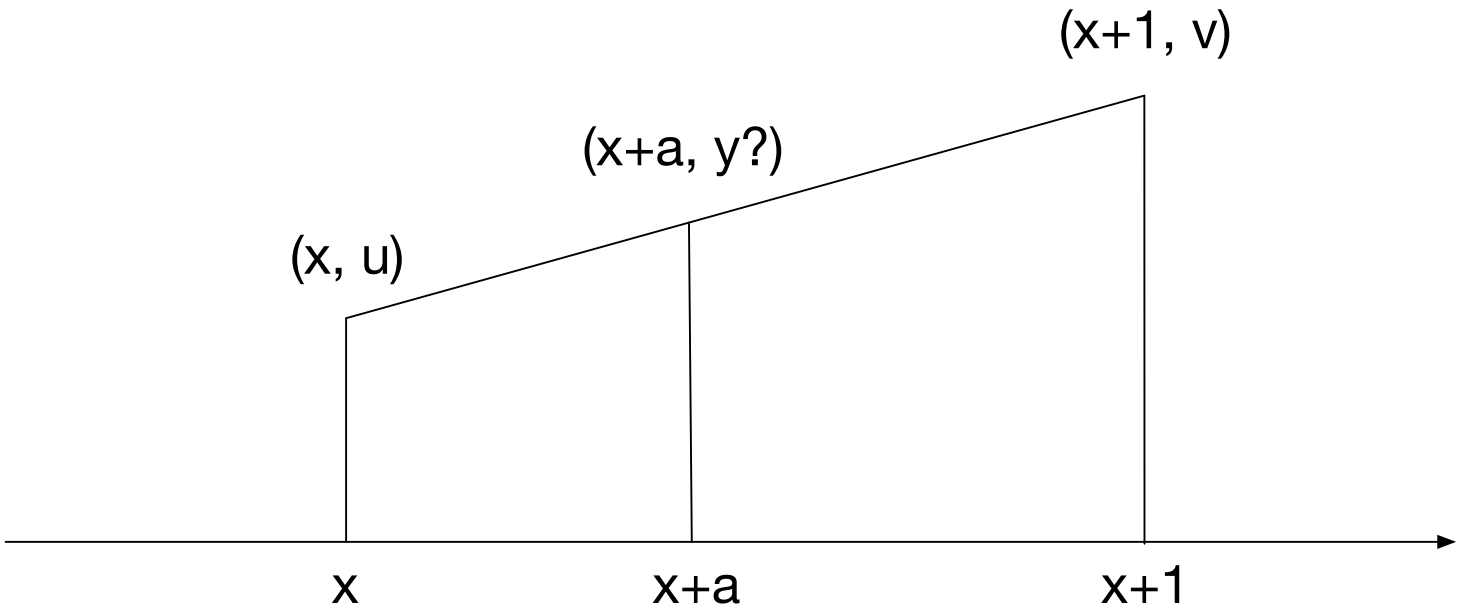


Issues of Nearest Neighbor Interpolation

Checkerboard artifact

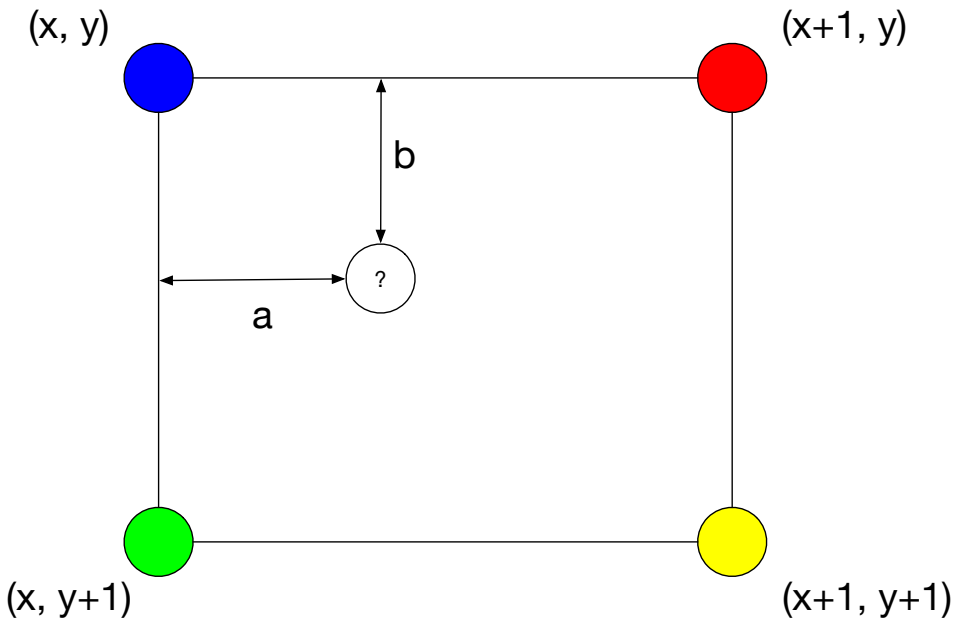


Linear Interpolation



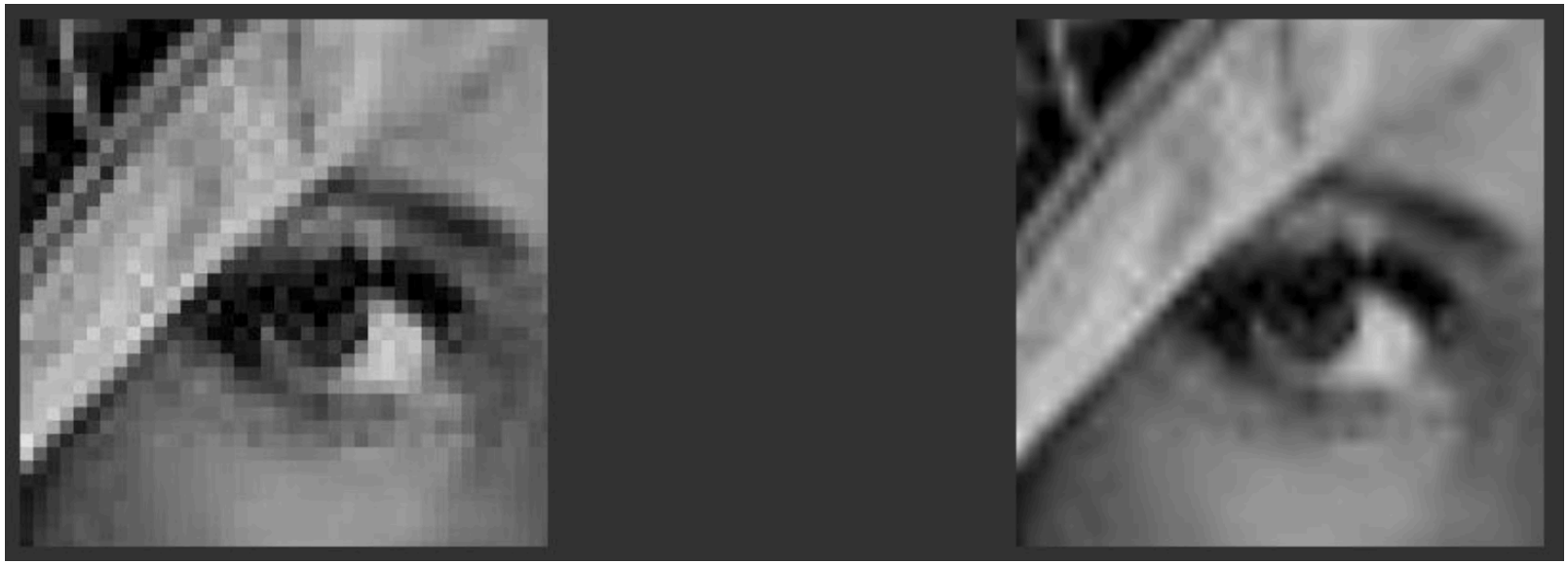
$$\frac{y-u}{a} = \frac{v-u}{1} \implies y = u + a(v-u) = (1-a)u + av$$

Bilinear Interpolation



$$\begin{aligned}\tilde{f}(x', y') = &(1-a)(1-b) \cdot f(x, y) \\ &+ a(1-b) \cdot f(x+1, y) \\ &+ (1-a)b \cdot f(x, y+1) \\ &+ ab \cdot f(x+1, y+1)\end{aligned}$$

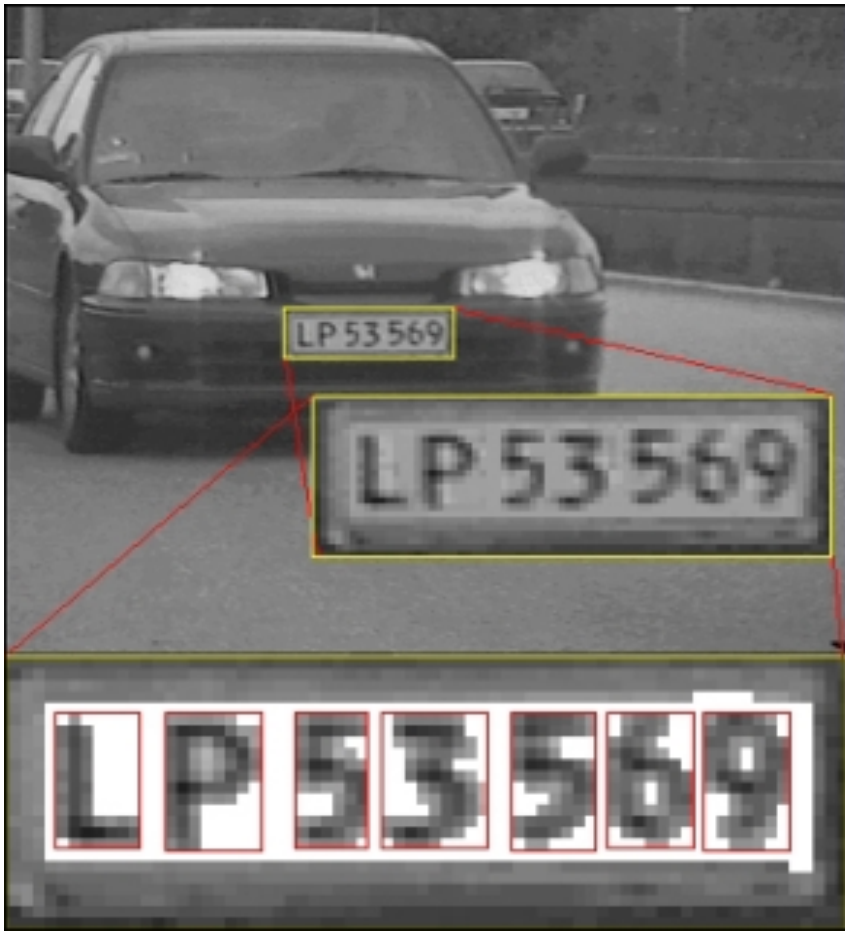
Comparison of Interpolation Methods



Nearest Neighbor

Bilinear

Adjacency

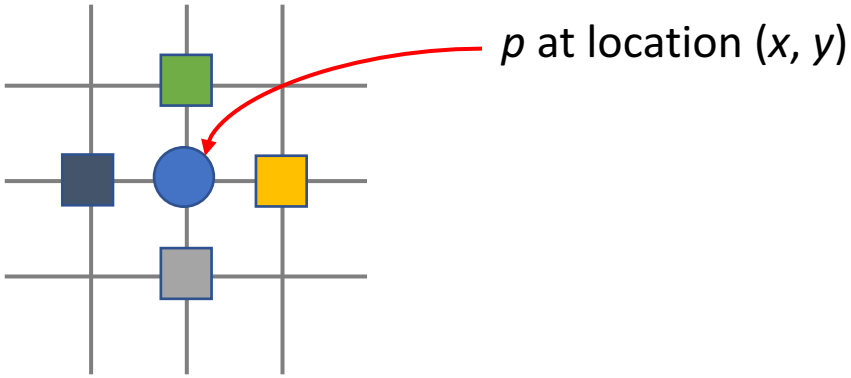


Adjacency between pixel is an important concept.

It is useful for

- Establishing object boundaries
- Defining image components/regions

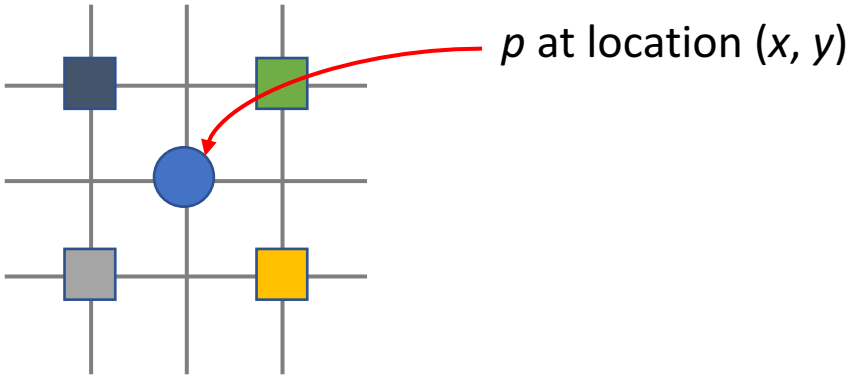
Neighbourhoods of a pixel



$N_4(p) = \{(x+1,y), (x-1,y), (x,y+1), (x,y-1)\}$

- A pixel p at location (x, y) has two horizontal and two vertical neighbours
- This set of four pixels is called 4-neighbours of p, $N_4(p)$
- Each of these neighbours is at a unit distance from p
- Fewer neighbours for a boundary pixel

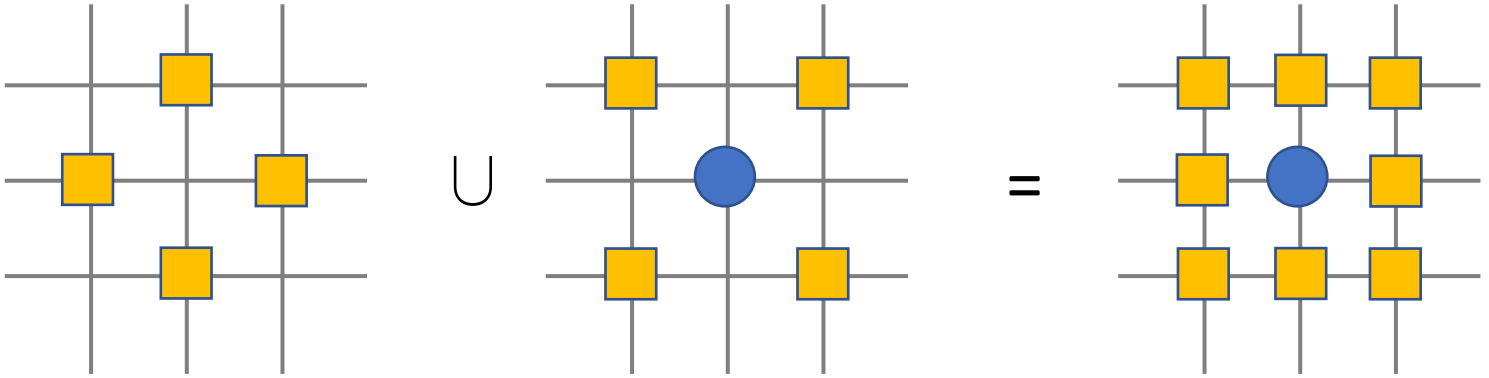
Neighbourhoods of a pixel



$N_D(p) = \{(x+1,y-1), (x-1,y+1), (x+1,y+1), (x-1,y-1)\}$

- A pixel p at location (x, y) also has four diagonal neighbours, $N_D(p)$
- Fewer neighbours for a boundary pixel

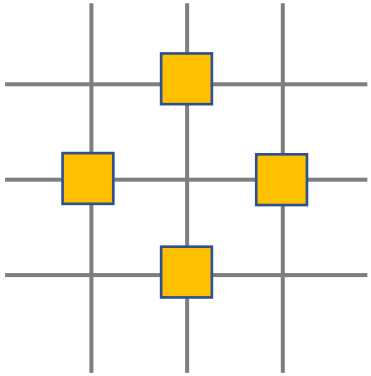
Neighbourhoods of a pixel



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

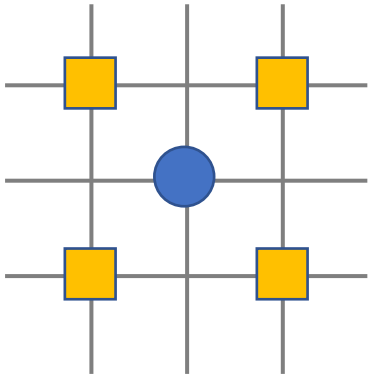
- The points of $N_4(p)$ and $N_D(p)$ together form the 8-neighbours of p

Neighbourhoods of a pixel



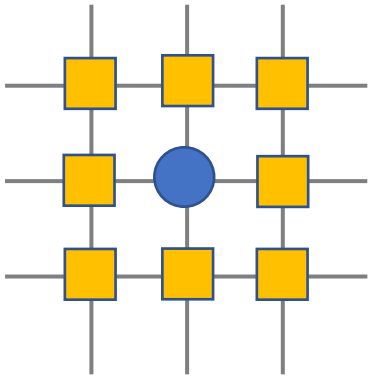
4-neighbours of p :

$N_4(p) = \{(x+1,y), (x-1,y), (x,y+1), (x,y-1)\}$



4 diagonal neighbours of p :

$N_D(p) = \{(x+1,y-1), (x-1,y+1), (x+1,y+1), (x-1,y-1)\}$



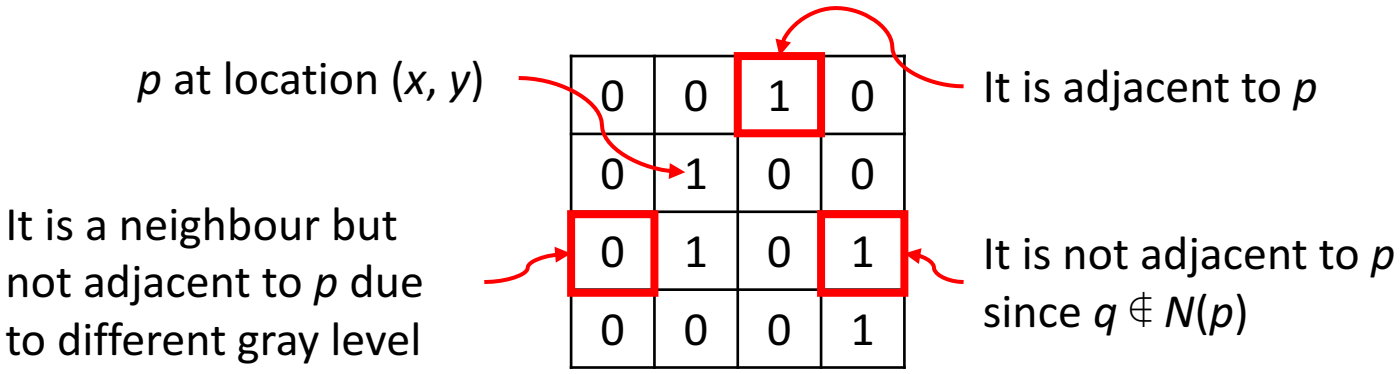
8-neighbours of p :

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

What is adjacency?

- Two pixels are said to be **adjacent** if
  - They are neighbours (N_4, N_D or N_8), and
  - Their intensity values are both in a set V

Let V = {1} and use N_8 neighbor



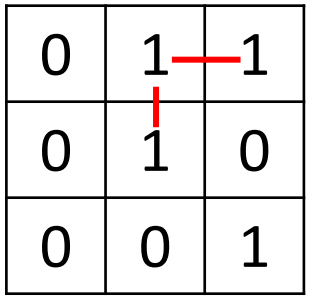
Three types of Adjacency

- **4-adjacency**
  - $p, q \in V$
  - $q \in N_4(p)$
- **8-adjacency**
  - $p, q \in V$
  - $q \in N_8(p)$
- **m-adjacency (mixed adjacency)**
  - $p, q \in V$
  - $q \in N_4(p)$ **or**
  - $q \in N_D(p)$ **and** $N_4(p) \cap N_4(q) = \varnothing$

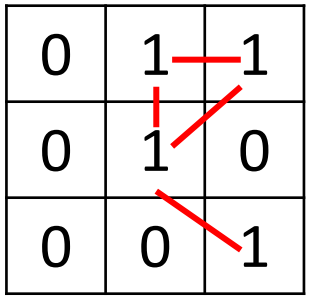
No pixels in the set of 4 neighbours of both p and q have values from V

What is adjacency?

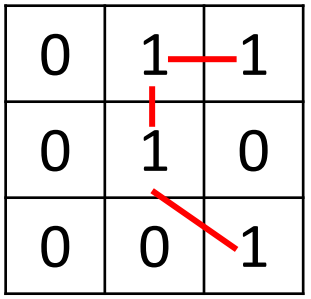
- Mixed adjacency is introduced to eliminate multiple path connections that often arise with 8-adjacency



4-adjacency



8-adjacency



***m*-adjacency**

Path

- A path from $p(x, y)$ to $q(s, t)$ is a sequence of distinct pixels

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

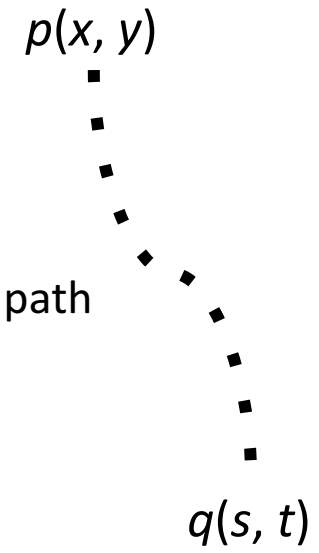
where

$(x_0, y_0) = (x, y)$, $(x_n, y_n) = (s, t)$

(x_i, y_i) is adjacent to (x_{i-1}, y_{i-1})

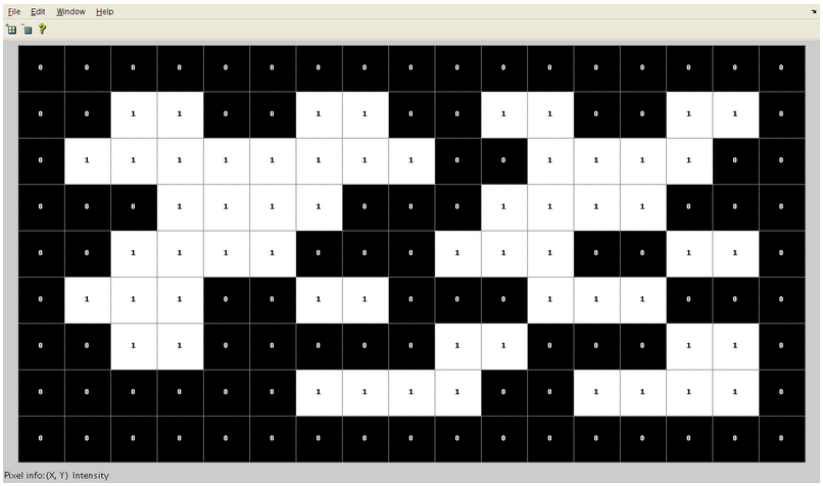
for $1 \leq i \leq n$

n = length of path



Connected Component

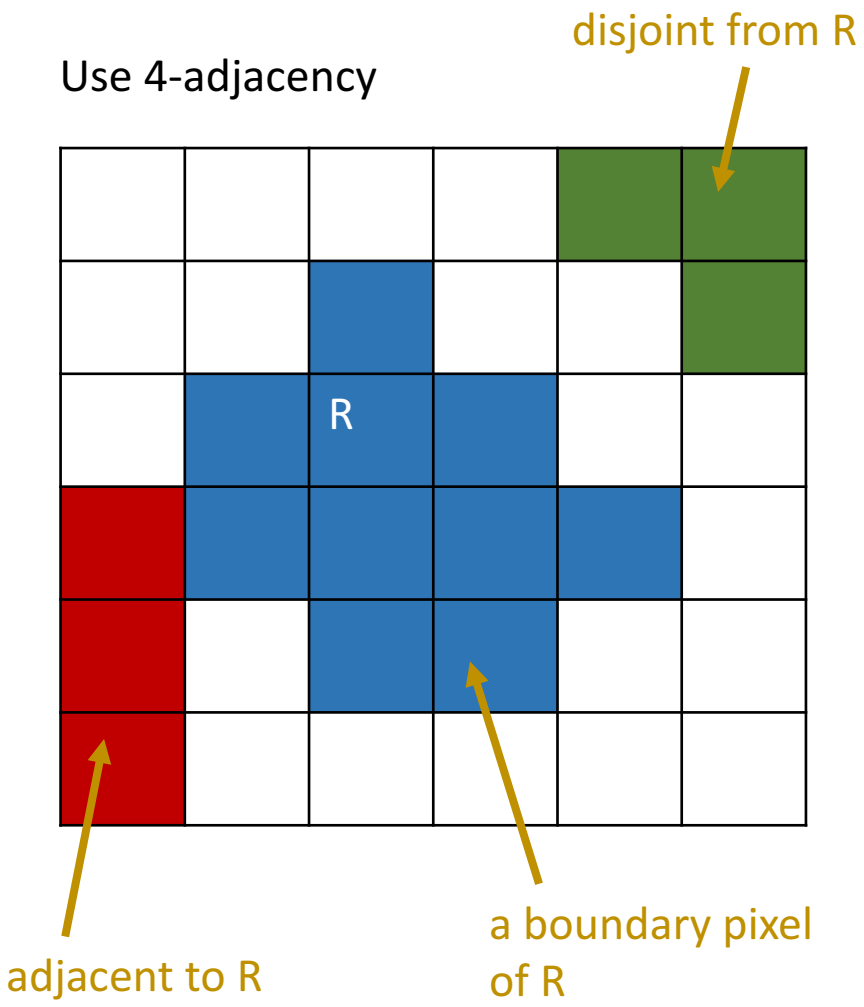
- Let S be a subset of pixels, p and q are said to be **connected in S** if there exists a path between them that are completely in S.
- For any pixel p, the set of pixels that are connected to it in S is called a **connected component**.
- If S contains only one connected component, it is called a **connected set**.



How many connected components?
Using N_4 and N_8 respectively.

Regions

- R is called a region if it is a connected set.
- Two regions are said to be adjacent if their union form a connected set, otherwise they are said to be disjoint.
- The boundary of a region R is a subset of R in which all the points are adjacent to the complement of R.



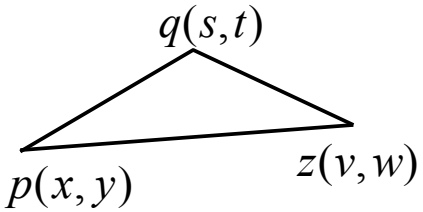
Distances

D is a *distance function* or *metric* if

$D(p,q) \geq 0 \quad (D = 0 \text{ iff } p = q)$ (positiveness)

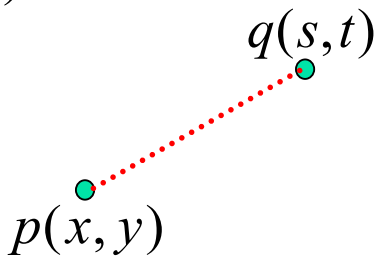
$D(p,q) = D(q,p)$ (symmetry)

$D(p,z) \leq D(q,p) + D(q,z)$ (triangular inequality)



Euclidean distance

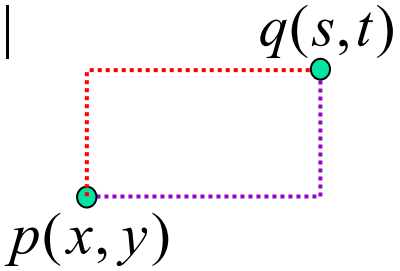
$$D_e(p,q) = \sqrt{(x-s)^2 + (y-t)^2}$$

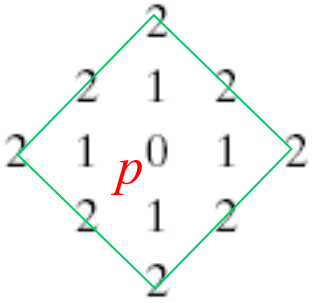


Distances

City-block distance

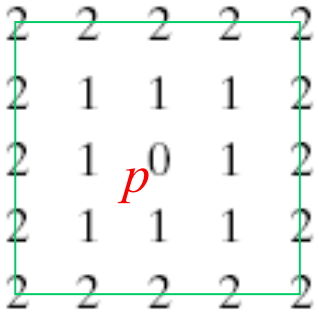
$$D_4(p,q) = |x-s| + |y-t|$$





Chessboard distance

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



Distance w.r.t. adjacency

In the simple binary image below, draw a shortest 4-path and a shortest 8-path from pixel *p* to pixel *q* (if they exist). The paths must pass through pixels of value 1.

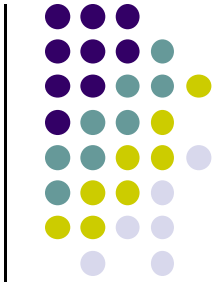
	1	0	0	1	1	1 *p*
	0	0	0	1	0	0
	0	1	1	1	1	0
	0	1	0	0	1	0
q	1	1	1	1	1	0

Distance w.r.t. adjacency

Shortest 4-path

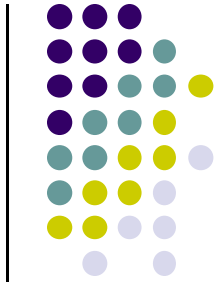
Shortest 8-path

Image File Formats



- Hundreds of image file formats. Examples
  - Tagged Image File Format (TIFF)
  - Graphics Interchange Format (GIF)
  - Portable Network Graphics (PNG)
  - JPEG, BMP, Portable Bitmap Format (PBM), etc
- Image pixel values can be
  - **Grayscale:** 0 – 255 range
  - **Binary:** 0 or 1
  - **Color:** RGB colors in 0-255 range (or other color model)
  - **Application specific** (e.g. floating point values in astronomy)

How many Bits Per Image Element?



Grayscale (Intensity Images):

Chan.	*Bits/Pix.*	*Range*	*Use*
1	1	0...1	Binary image: document, illustration, fax
1	8	0...255	Universal: photo, scan, print
1	12	0...4095	High quality: photo, scan, print
1	14	0...16383	Professional: photo, scan, print
1	16	0...65535	Highest quality: medicine, astronomy

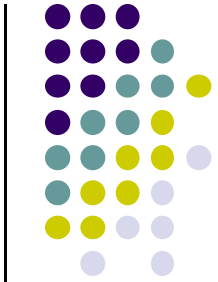
Color Images:

Chan.	*Bits/Pix.*	*Range*	*Use*
3	24	$[0...255]^3$	RGB, universal: photo, scan, print
3	36	$[0...4095]^3$	RGB, high quality: photo, scan, print
3	42	$[0...16383]^3$	RGB, professional: photo, scan, print
4	32	$[0...255]^4$	CMYK, digital prepress

Special Images:

Chan.	*Bits/Pix.*	*Range*	*Use*
1	16	$-32768...32767$	Whole numbers pos./neg., increased range
1	32	$\pm 3.4 \cdot 10^{38}$	Floating point: medicine, astronomy
1	64	$\pm 1.8 \cdot 10^{308}$	Floating point: internal processing

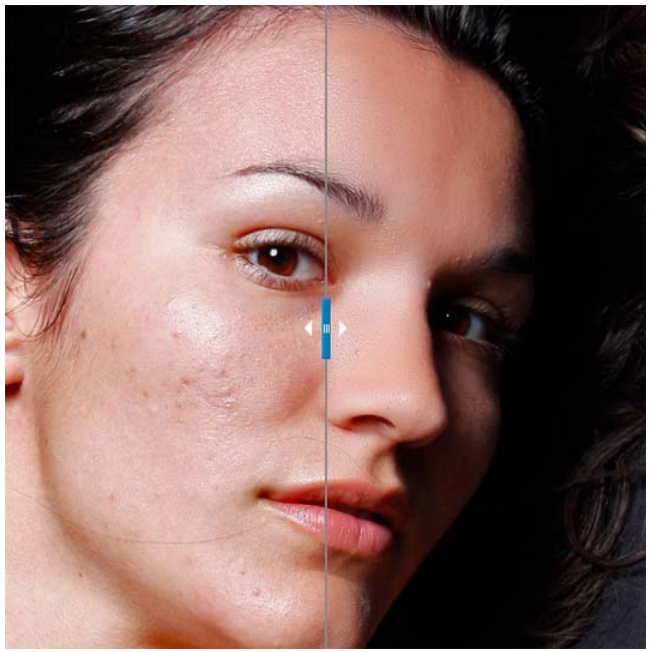
Mathematics for Image Processing



- Calculus
- Linear algebra
- Probability and statistics
- Differential Equations (PDEs and ODEs)
- Differential Geometry
- Harmonic Analysis (Fourier, wavelet, etc)

About This Course

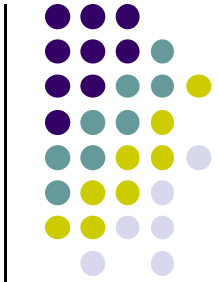
- Image Processing has many aspects
  - **Computer Scientists/Engineers** develop tools (e.g. photoshop)
    - **Requires** knowledge of maths, algorithms, programming
  - **Artists** use image processing tools to modify pictures
    - **DOES NOT** require knowledge of maths, algorithms, programming



Example: Portraiture photoshop plugin

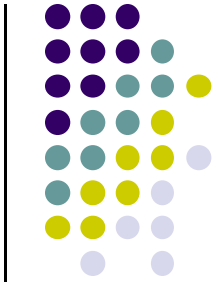
Example: Knoll Light Factory photoshop plugin

Example: ToonIt photoshop plugin



About This Course

- Most hobbyists follow artist path. Not much math!
- This Course: Image Processing for computer scientists and Engineers!!!
- Teaches concepts, uses Matlab/Python as concrete example
- Matlab Image Processing Toolbox
  - Includes lots of already working algorithms,
  - Can be extended by programming new image processing techniques
- Course is **NOT**
  - just about programming Matlab image processing toolbox
  - a comprehensive course in matlab image processing toolbox and python image processing library (PIL, scipy, opencv-python and so on)
  - about using packages like Photoshop



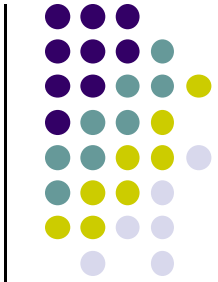
About This Course

- Class is concerned with:
  - How to implement image processing algorithms
  - Underlying mathematics
  - Underlying algorithms

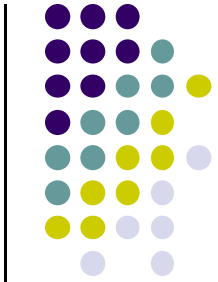
- This course is a lot of work. **Especially heavy coding!!!**
  Requires:

  - Lots of programming (maybe in MATLAB/Python)
  - Lots of math, linear systems, fourier analysis

Administrivia: Syllabus Summary



- 3 Assignments (30%), one Mid-Exam (30%), 1 final Projects (40%)
- Projects:
  - Develop image processing algorithm on any platform using MATLAB/Python
  - May discuss projects but turn in individual projects
- Text:
  - *Digital Image Processing: An Algorithmic Introduction using Java* by Wilhelm Burger and Mark J. Burge, Springer Verlag, 2008
- Cheating: Immediate 'F' in the course
- My advice:
  - Come to class
  - Read the text
  - Understand concepts before coding



References

- Wilhelm Burger and Mark J. Burge, Digital Image Processing, Springer, 2008
- University of Utah, CS 4640: Image Processing Basics, Spring 2012
- Gonzales and Woods, Digital Image Processing (3rd edition), Prentice Hall
- Digital Image Processing slides by Brian Mac Namee

Thank You