INTRODUCTION
(DIGITAL IMAGE PROCESSING)

Faculty of Information Technology
Ton Duc Thang University

August 2023

DIGITAL IMAGE

- An image is defined as a two-dimensional function, $\boldsymbol{f(x,y)}$
  - **x** and **y** are spatial coordinates
  - the amplitude of $\boldsymbol{f}$ at any pair of coordinates $(\boldsymbol{x}, \boldsymbol{y})$ is called the **intensity** or **gray level** of the image at that point

  ⇒ when x, y, and the intensity values of f are all ***finite, discrete quantities***, we call the image a **digital image**

- A digital image can be represented by a two-dimensional array or matrix

The image size (height x width) is $M \times N$

Every element of this matrix is called image element , picture element , or pixel.

$$f(x,y) = \begin{bmatrix} f(0,0) & f(0,1) & f(0,2) & \ldots & f(0,N-1) \\ f(1,0) & f(1,1) & f(1,2) & \ldots & f(1,N-1) \\ \vdots & \vdots & \vdots & & \vdots \\ f(M-1,0) & f(M-1,1) & f(M-1,2) & \ldots & f(M-1,N-1) \end{bmatrix}$$

(https://www.geeksforgeeks.org/)

DIGITAL IMAGE (ct)

- A digital image is composed of a finite number of elements, each of which has a particular location and value
  - These elements are called picture elements, image elements, or **pixels**. Pixel is the term used most widely to denote the elements of a digital image.

DIGITAL IMAGE PROCESSING

- Digital image processing (DIP) refers to processing digital images by means of a digital computer
- DIP is the use of algorithms and mathematical models to process and analyze digital images
  - to enhance the quality of images
  - to improve pictorial information for human interpretation
  - to extract meaningful information from images
  - to process image data for tasks such as storage, transmission, and extraction of pictorial information

DIGITAL IMAGE PROCESSING (ct)

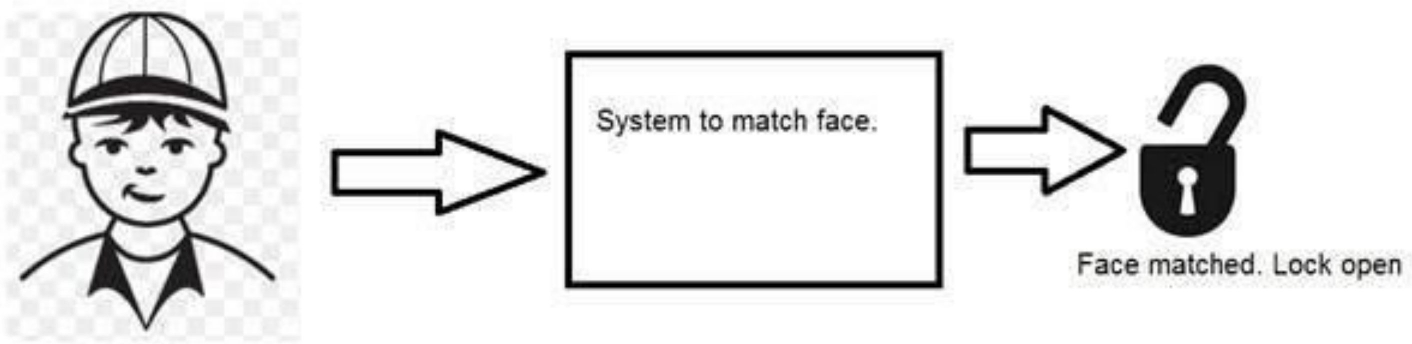
- In this course, DIP mainly include the following steps:
  - Importing the input images
  - Analysing and manipulating the images
  - Output in which result can be altered images and/or extracted information

DIP – Related fields

- Computer Vision (CV) is a field whose ultimate goal is to use computers to emulate human vision, including learning and being able to make inferences and take actions based on visual inputs
  - This area is a branch of artificial intelligence (AI) whose objective is to emulate human intelligence
- CV is the field of artificial intelligence which focused on enabling computers and systems to process and derive information from visual data such as images and videos just as a human would
- CV uses many techniques and DIP is just one of them

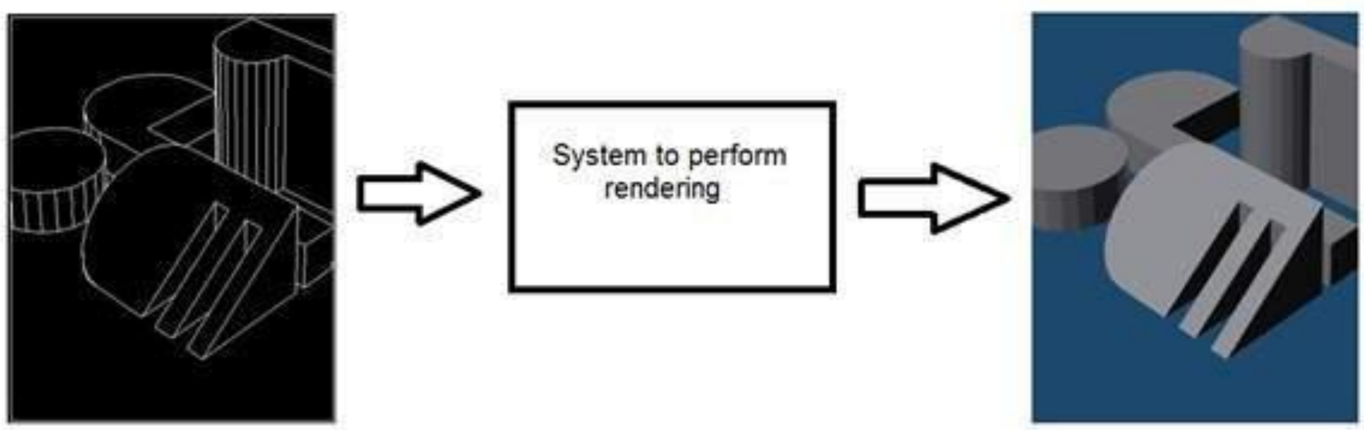
Computer Vision (ct)

- Examples: Self-Driving Cars, Augmented reality (AR) apps, Facial Recognition, Healthcare, …



DIP – Related fields (ct)

- Computer graphics is the field that deals with the creation and manipulation of images using a computer
  - It involves using algorithms to generate and render images, as well as techniques for animation, 3D modeling, and visualization
- For example: computer art, computer aided drawing, object rendering, creating motion pictures /music video/television shows/cartoon animation films, ...



DIP – Related fields (ct)

World Model → **Computer Graphics** →

geometry, physics
computer algorithms

→ **Computer Vision** → World Model

geometry, physics
computer algorithms

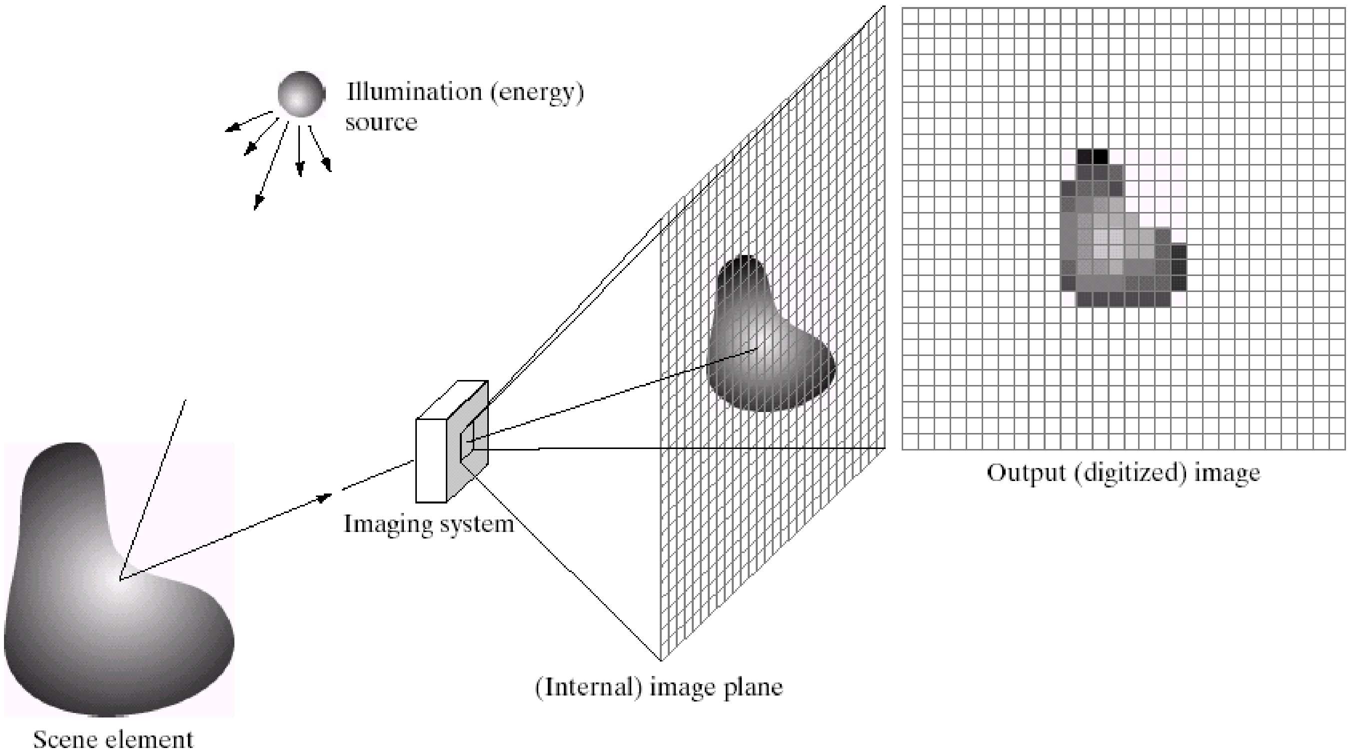
DIP – Related fields (ct)

- The continuum from Digital image processing to Computer vision can be broken up into low-, mid- and high-level processes

Low Level Process	Mid Level Process	High Level Process
Input: Image **Output:** Image	**Input:** Image **Output:** Attributes	**Input:** Attributes **Output:** Understanding
Examples: Noise removal, image sharpening, contrast	**Examples:** Object recognition, segmentation	**Examples:** Scene understanding, autonomous navigation

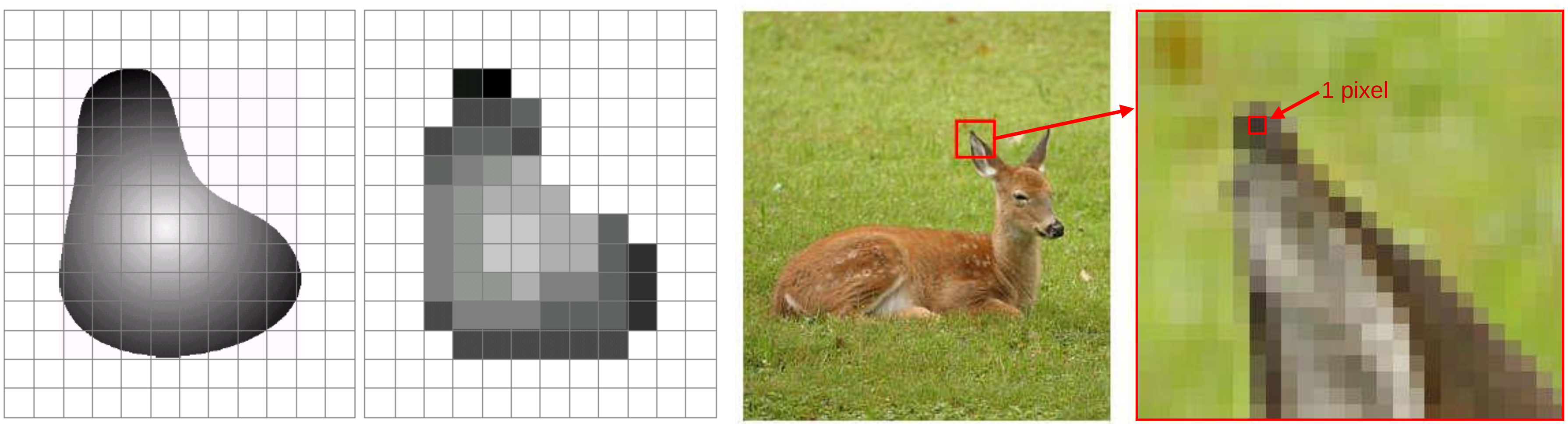
What is a Digital Image?

- A digital image is a representation of a two-dimensional image as a finite set of digital values, called picture elements or pixels



What is a Digital Image? (ct)

- Pixel values typically represent gray levels, colours, intensities, …
- Remember digitization implies that a digital image is an approximation of a real scene



What is a Digital Image? (ct)

pixel

j

i

62	79	23	119	120	05	4	0
10	10	9	62	12	78	34	0
10	58	197	46	46	0	0	48
176	135	5	188	191	68	0	49
2	1	1	29	26	37	0	77
0	89	144	147	187	102	62	208
255	252	0	166	123	62	0	31
166	63	127	17	1	0	99	30

IMAGE FORMATION

What the computer "sees" is
just a grid of numbers.

But the camera sees this:

194	210	201	212	199	213	215	195	178	158	182	209
180	189	190	221	209	205	191	167	147	115	129	163
114	126	140	188	176	165	152	140	170	106	78	88
87	103	115	154	143	142	149	153	173	101	57	57
102	112	106	131	122	138	152	147	128	84	58	66
94	95	79	104	105	124	129	113	107	87	69	67
68	71	69	98	89	92	98	95	89	88	76	67
41	56	68	99	63	45	60	82	58	76	74	65
20	41	69	75	56	41	51	73	55	70	63	44
50	50	57	69	75	75	73	74	53	68	59	37
72	59	53	66	84	92	84	74	57	72	63	42
67	61	58	65	75	78	76	73	59	75	69	50

Figure 1-1. To a computer, the car's side mirror is just a grid of numbers

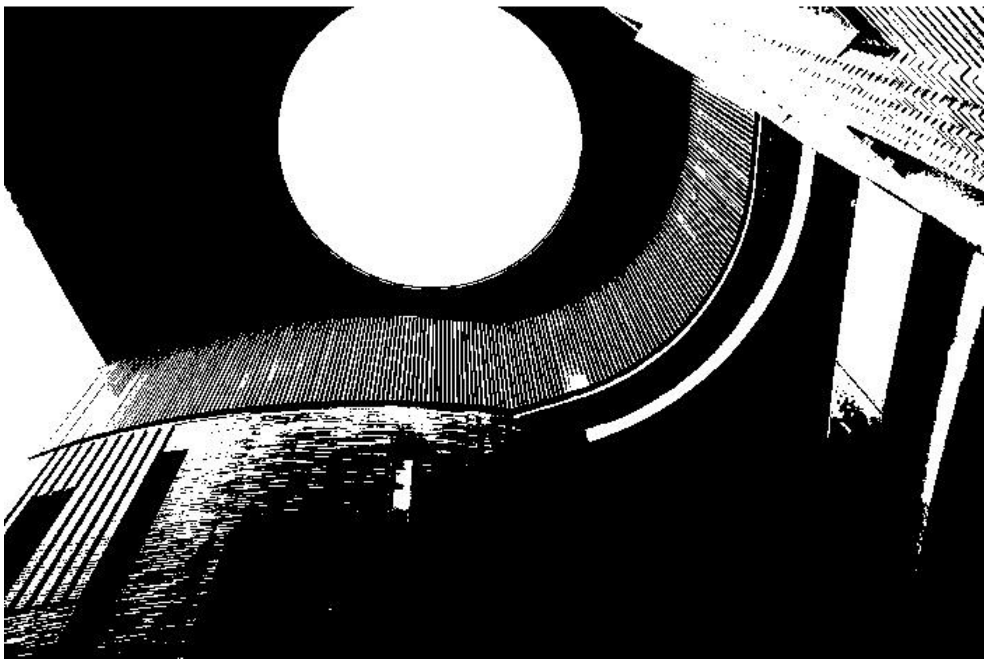
Common types of an image

Color model

- A color model is a specification of (1) a coordinate system, and (2) a subspace within that system, such that each color in the model is represented by a single point contained in that subspace
- Most color models in use today are oriented either toward hardware or toward applications
  - Grayscale (256 colors from black to white) model for reducing the required storage and computation
  - the RGB (red, green, blue) model for color monitors and a broad class of color video cameras
  - the CMY (cyan, magenta, yellow) and CMYK (cyan, magenta, yellow, black) models for color printing

Common types of an image

- Binary image: has only two intensity values or two colors, for ex., images with pixel values 0 and 255, representing black and white colours



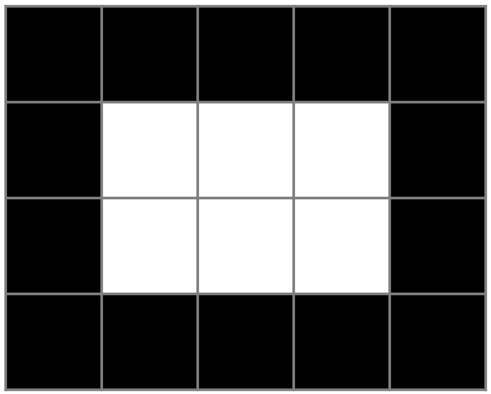
```
[[  0   0   0 ... 255 255   0]
 [  0   0   0 ... 255 255 255]
 [  0   0   0 ... 255 255 255]
 ...
 [  0   0   0 ...   0   0   0]
 [  0   0   0 ...   0   0   0]
 [  0   0   0 ...   0   0   0]]
```

Common types of an image (ct)

Binary image

- Calculate the average intensity of a binary image.

image size 4×5



```
[[  0   0   0   0   0]
 [  0 255 255 255   0]
 [  0 255 255 255   0]
 [  0   0   0   0   0]]
```

$\boldsymbol{avgI} = ???$

What does it mean?

Pseudo code

Input: a binary image $\boldsymbol{B}$ with size $M \times N$
Output: average intensity $\boldsymbol{avgI}$

$avgI = 0$

For $r = 1 \rightarrow M$:
  For $c = 1 \rightarrow N$:
      $avgI = avgI + \boldsymbol{B}[r, c]$

$$avgI = \frac{avgI}{M \times N}$$

Intensity of the pixel at
r-th row and **c**-th column of the image B

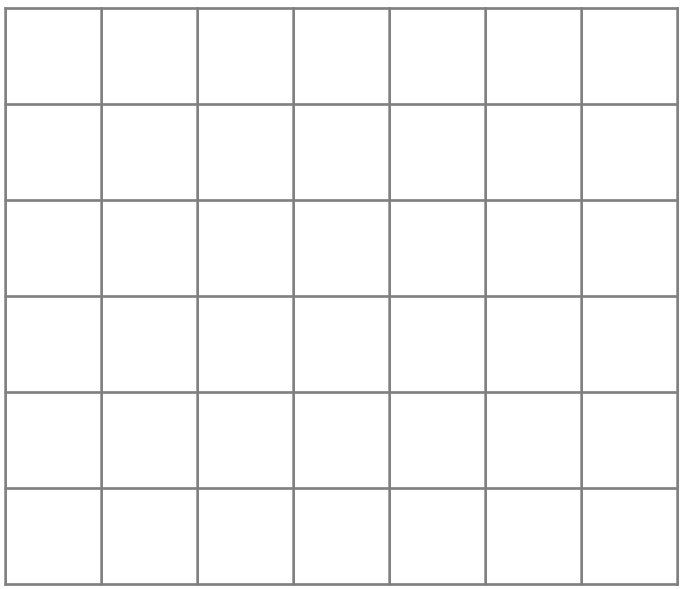
Common types of an image (ct)

Ex. 1

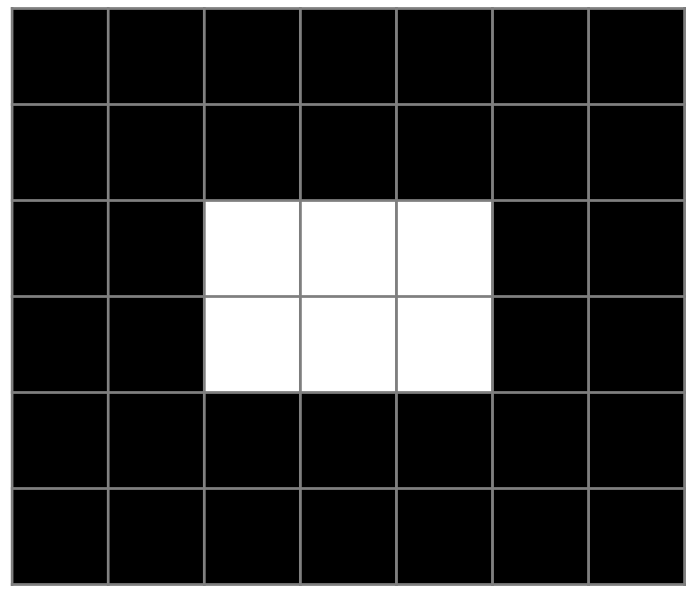
- Write pseudo code to fill the border (black color and 2 pixels thickness) of a binary image.

Input sample

image size 6×7



Output sample



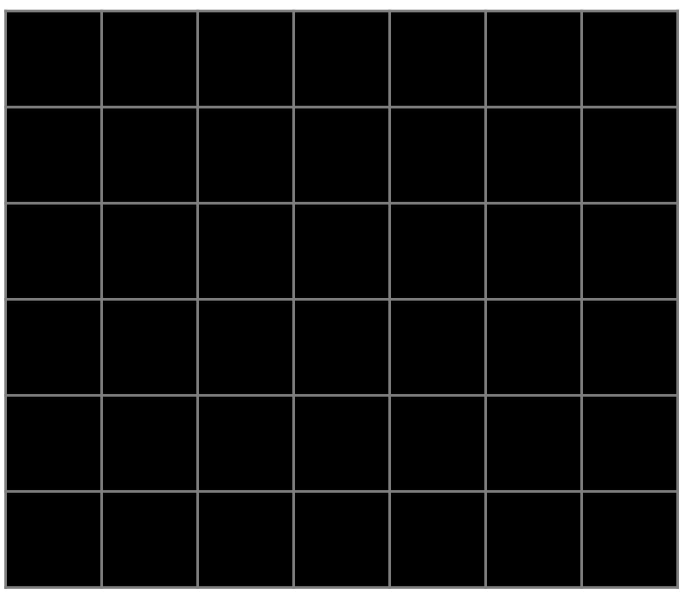
Common types of an image (ct)

Ex. 2

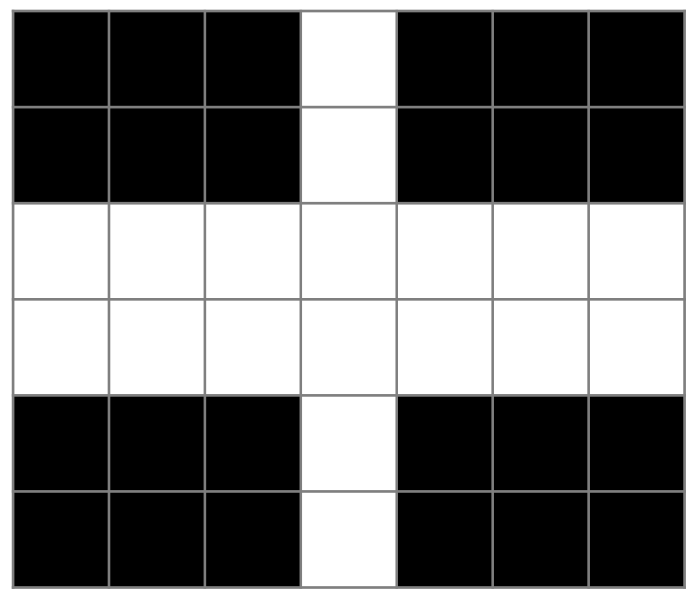
- Write pseudo code to fill the middle rows and columns (white color) of a binary image.

Input sample

image size 6×7



Output sample



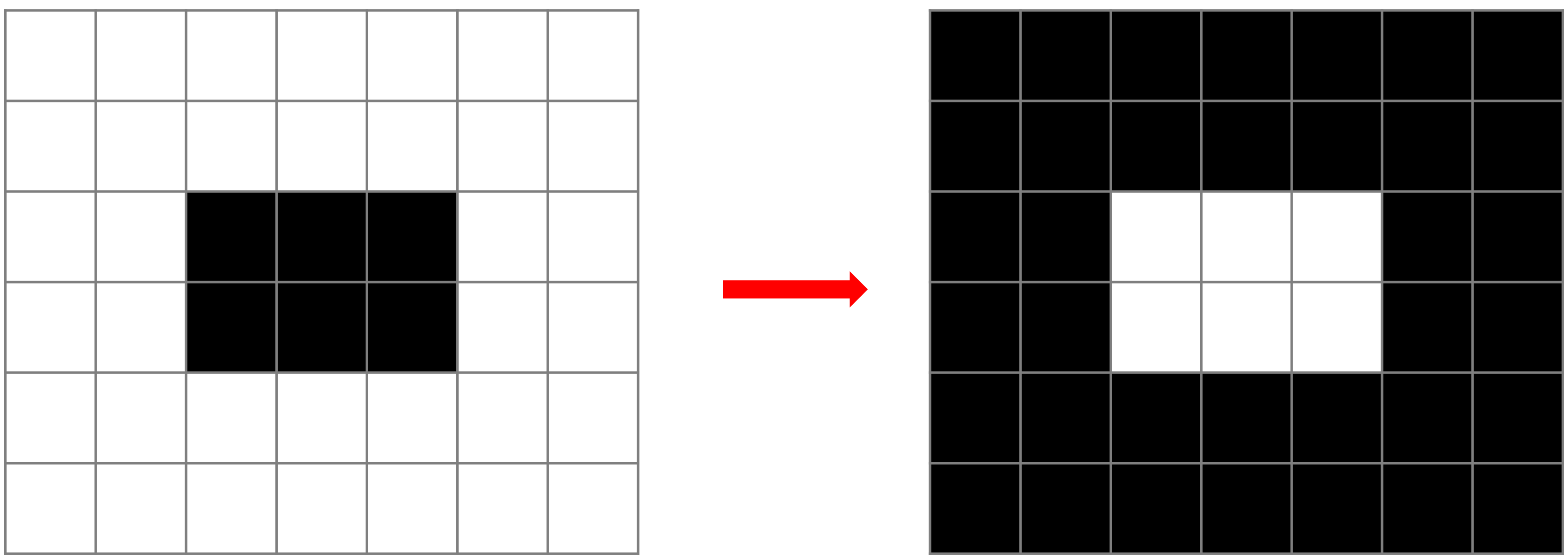
Common types of an image (ct)

Ex. 3

- Write pseudo code to invert color (black to white and vice versa) of a binary image.

Input sample **Output sample**

image size 6×7



Common types of an image (ct)

Ex. 4

- Write pseudo code to draw one of six following images, and then count the number of white pixels in the image.

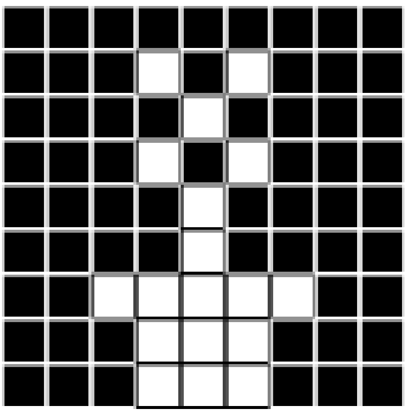


Image 1

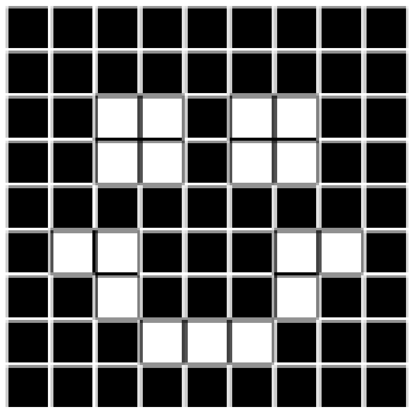


Image 2

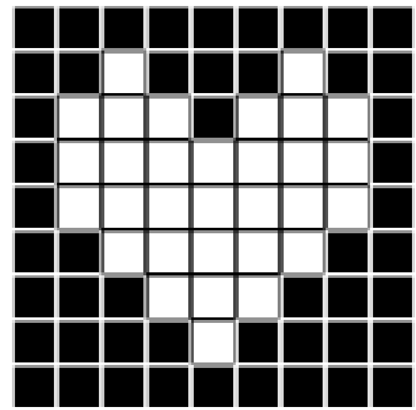


Image 3

Initialize an image $\boldsymbol{B}$ with all Black color and size $M \times N$

$$\boldsymbol{B}(M \times N) = 0$$

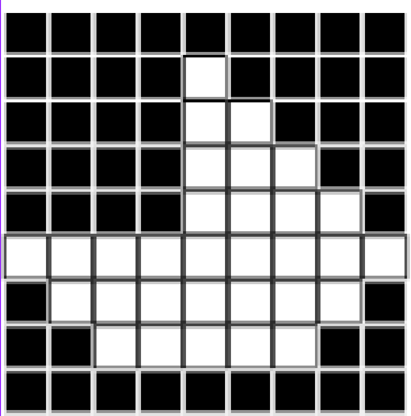


Image 4

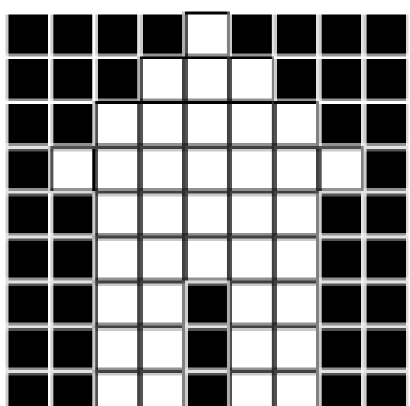


Image 5

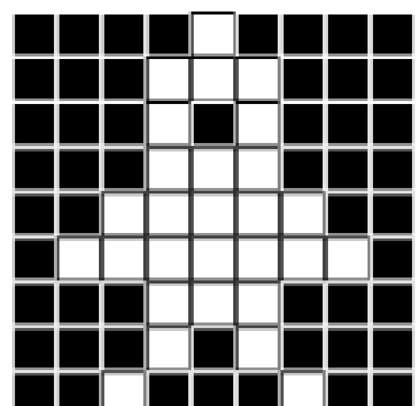
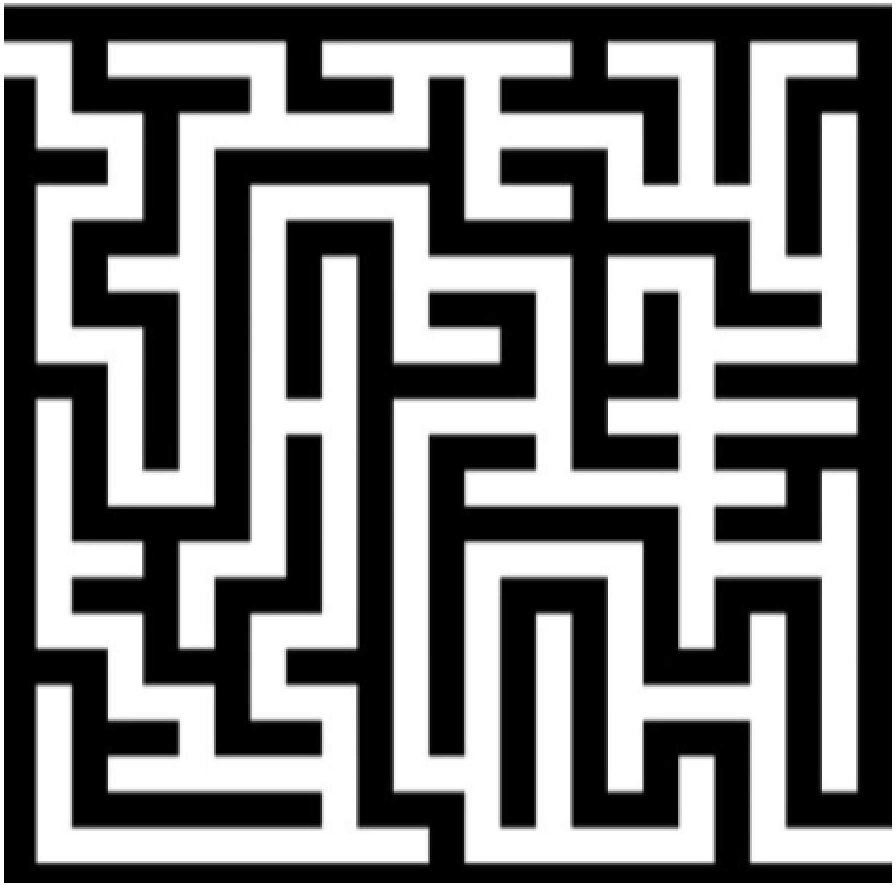
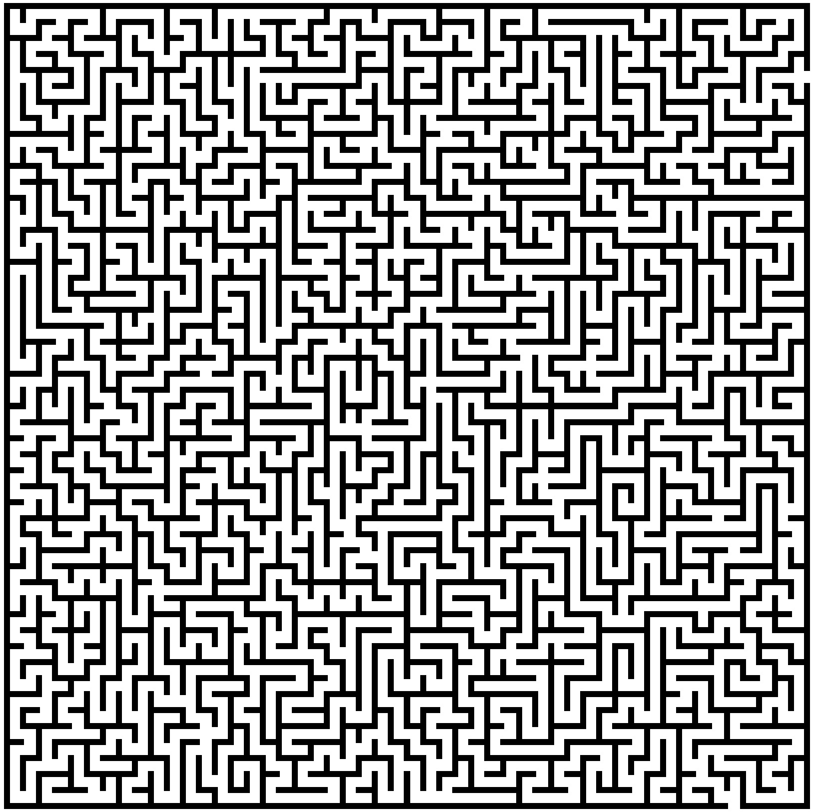


Image 6

Common types of an image (ct)

Binary image - examples





Common types of an image (ct)

- Grayscale image: has 8 bit COLOR FORMAT or 1 byte per pixel
  - It has 256 different colors, **0** stands for Black, and **255** stands for white, and 127 stands for gray
  - Each pixel is represented by an 8-bit integer number from 0 to 255

```
[[23 22 22 ... 79 78 68]
 [22 21 22 ... 94 94 85]
 [22 21 21 ... 92 93 87]
 ...
 [54 50 46 ... 10 10  9]
 [46 51 54 ... 10 10 10]
 [59 62 62 ... 10 10 10]]
```

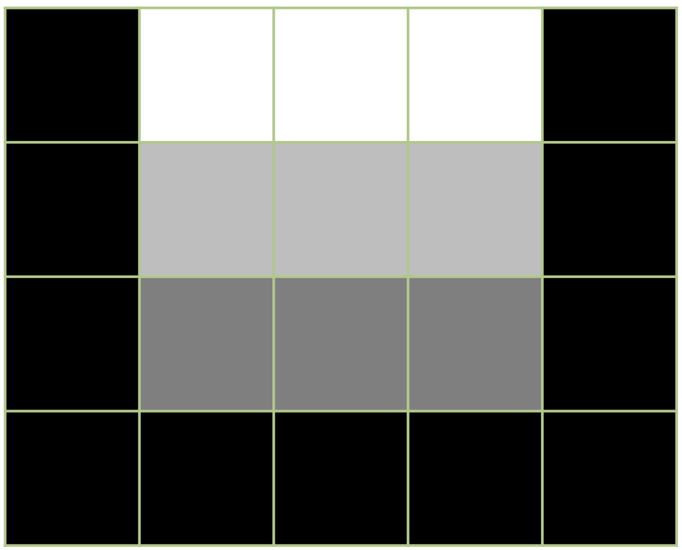
Common types of an image (ct)

Ex. 5

- Calculate the average intensity of the following grayscale image.

Input sample

image size 4×5



```
[[  0 255 255 255   0]
 [  0 190 190 190   0]
 [  0 127 127 127   0]
 [  0   0   0   0   0]]
```

Output sample

$avgI = ???$

What does it mean?

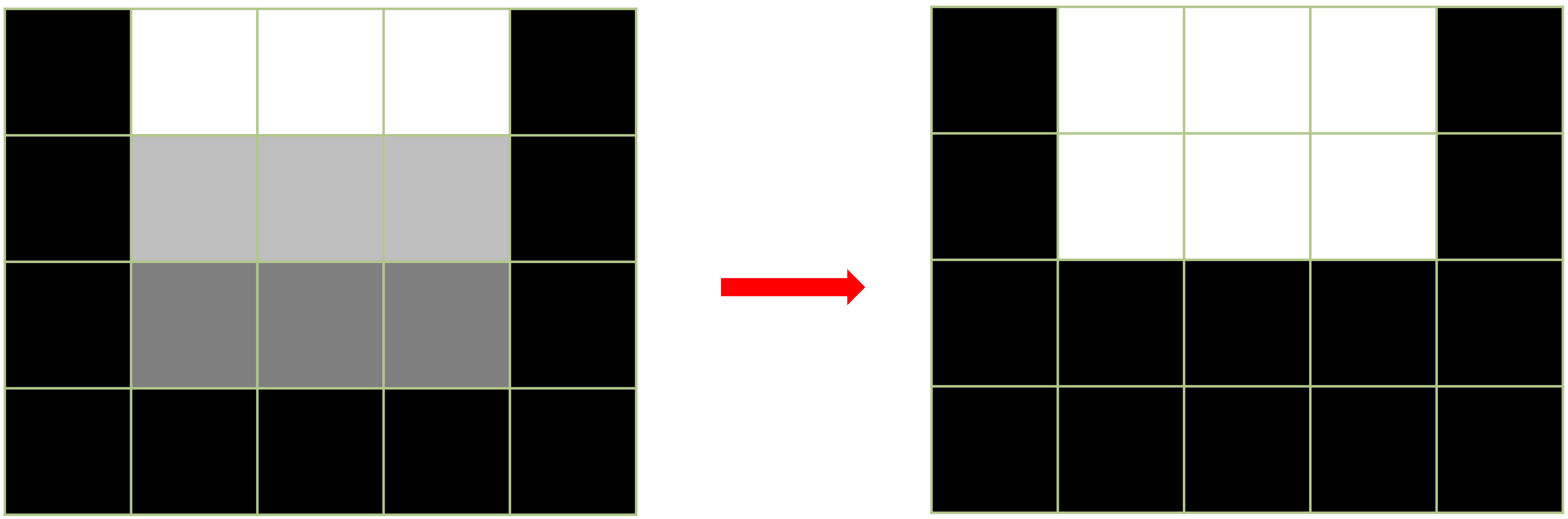
Common types of an image (ct)

Ex. 6

- Write pseudo code to change the pixels whose intensity values are smaller than 128 to black color and otherwise to white color.

Input sample **Output sample**

image size 4×5



```
[[  0 255 255 255   0]
 [  0 190 190 190   0]
 [  0 127 127 127   0]
 [  0   0   0   0   0]]
```

Common types of an image (ct)

Grayscale image examples

Common types of an image (ct)

Grayscale image examples

Common types of an image (ct)

- Color image: has many different colors compared to grayscale one
  - Each pixel is represented by **multiple 8-bit integer numbers** from 0 to 255
  - Some popular color models: RGB, RGBA, CMYK, HSV, HSI, …

Common types of an image (ct)

- 24-bit RGB image: is a color image format with 2^{24} colors
  - Each pixel is represented by **three 8-bit integer numbers** from 0 to 255

```
[[[ 27  18  31]
  [ 27  17  30]
  [ 29  16  32]
  ...
  [ 91  69  93]
  [ 90  71  88]
  [ 80  61  78]]
```

1st row

```
 [[ 27  17  30]
  [ 26  16  29]
  [ 29  16  32]
  ...
  [106  82 114]
  [106  82 112]
  [ 97  74 102]]
```

2nd row

```
 [[ 27  16  32]
  [ 26  15  31]
  [ 28  14  32]
  ...
  [103  74 123]
  [104  75 124]
  [ 98  70 116]]

 ...
```

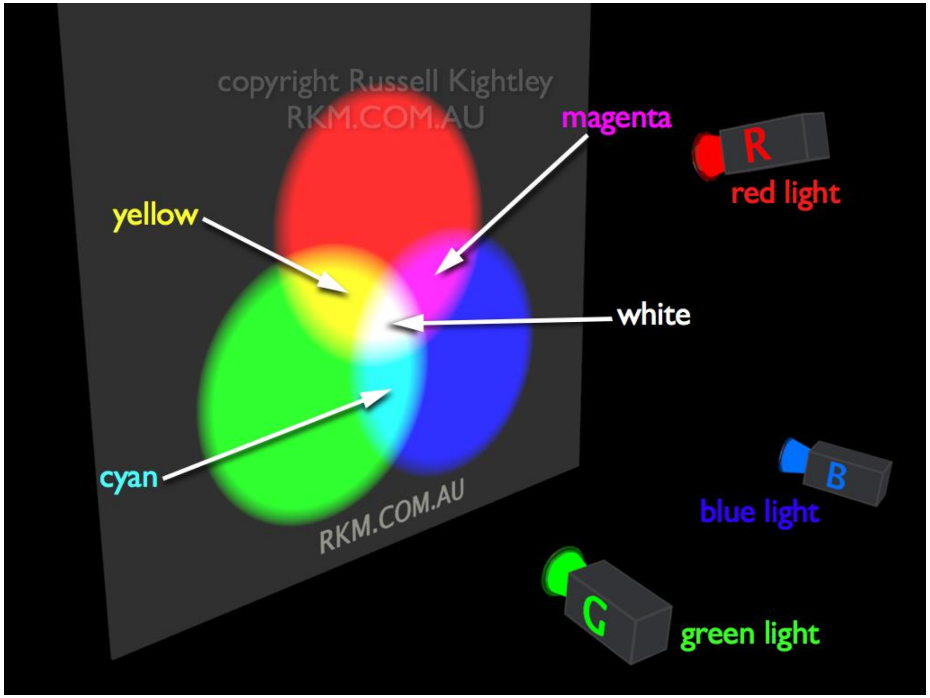
3rd row

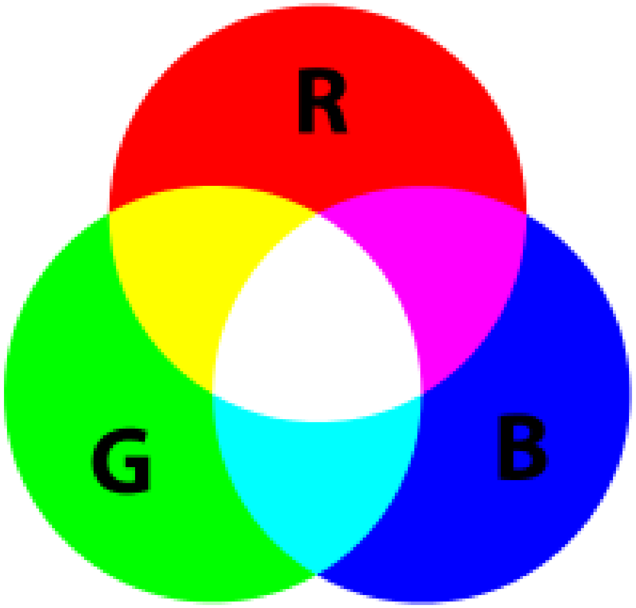
…

Common types of an image (ct)

RGB image

- RGB uses additive color mixing, because it describes what kind of light needs to be emitted to produce a given color
- An RGB pixel is specified with three values $(red, green, blue)$
  - Each parameter “red”, “green”, or “blue” defines the intensity of the corresponding color with a value between 0 and 255



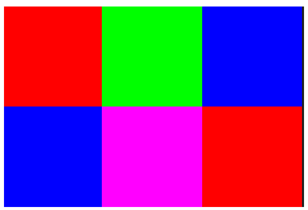


https://www.youtube.com/watch?v=KZ-mEddsYqo

Common types of an image (ct)

- 24-bit RGB image :

2 × 3 size
6 pixels



```
dip00_1 = np.array([
                    [ [0, 0, 255], [0, 255, 0], [255, 0, 0] ],
                    [ [255, 0, 0], [255, 0, 255], [0, 0, 255] ]
                   ]

                  , dtype=np.uint8)
```

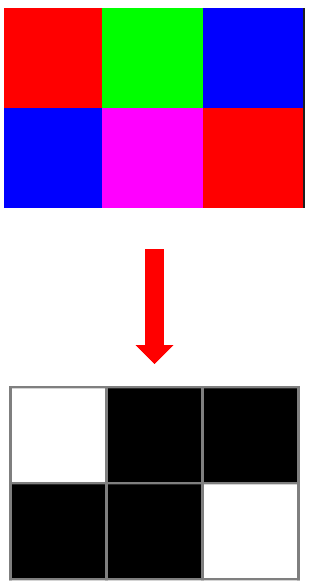
BGR order

Common types of an image (ct)

RGB image

- Write a pseudo code to change Red color to White color, otherwise to Black color of an RGB image.

2×3 size
6 pixels



Pseudo code

Input: an RGB image $\boldsymbol{B}$ with size $M \times N$
Output: a color-modified image $\boldsymbol{B}$

For $r = 1 \to M$:
  For $c = 1 \to N$:
    if $\boldsymbol{B}[r, c] == [255,0,0]$:
      $\boldsymbol{B}[r, c] = [255,255,255]$
    else :
      $\boldsymbol{B}[r, c] = [0,0,0]$

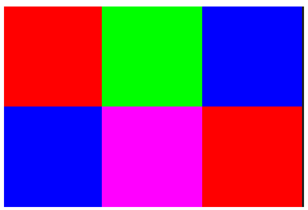
Common types of an image (ct)

Ex. 7

- Calculate the average intensity of the following RGB image.

2×3 size

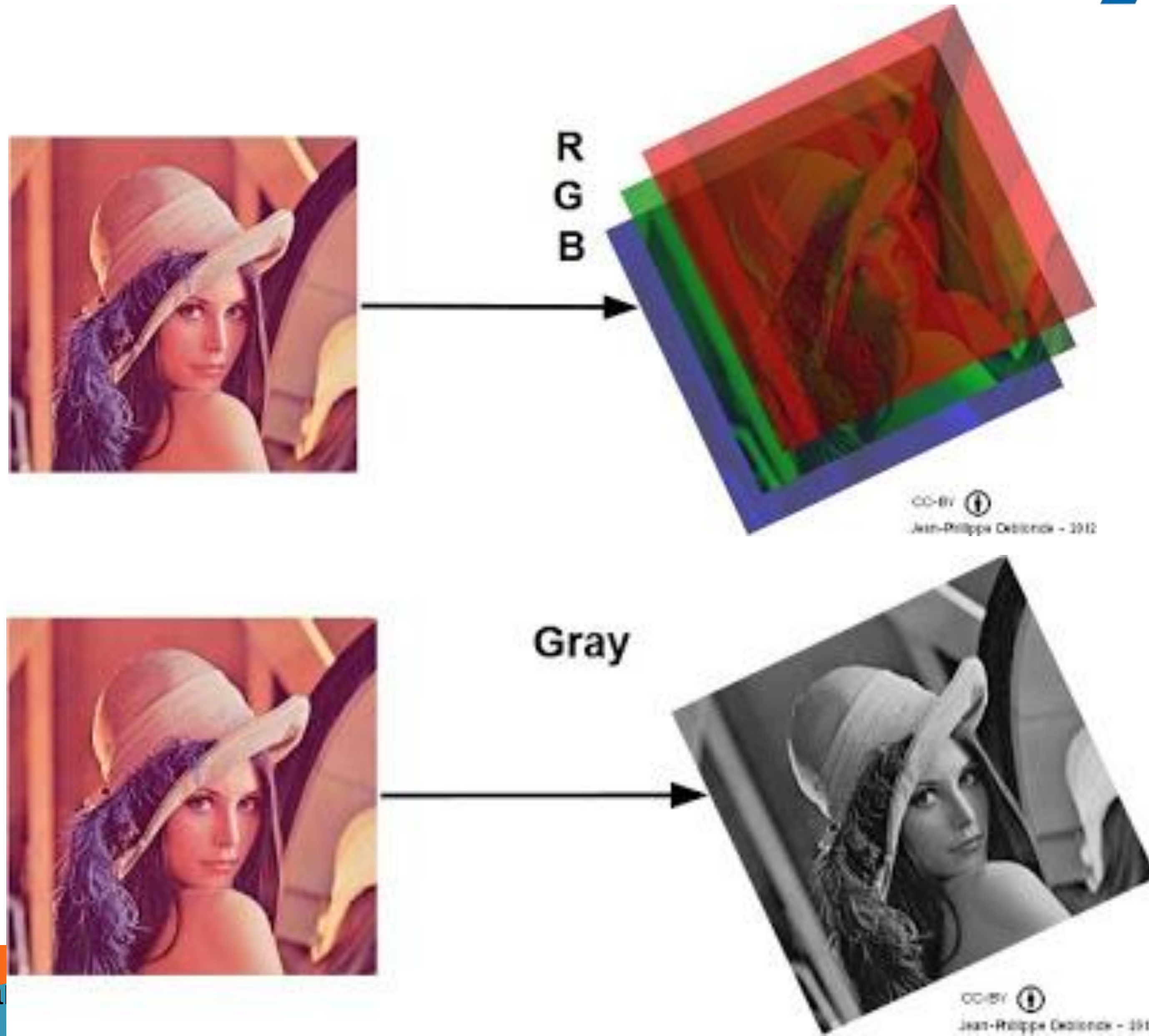
6 pixels



$avgI = ???$

What does it mean?

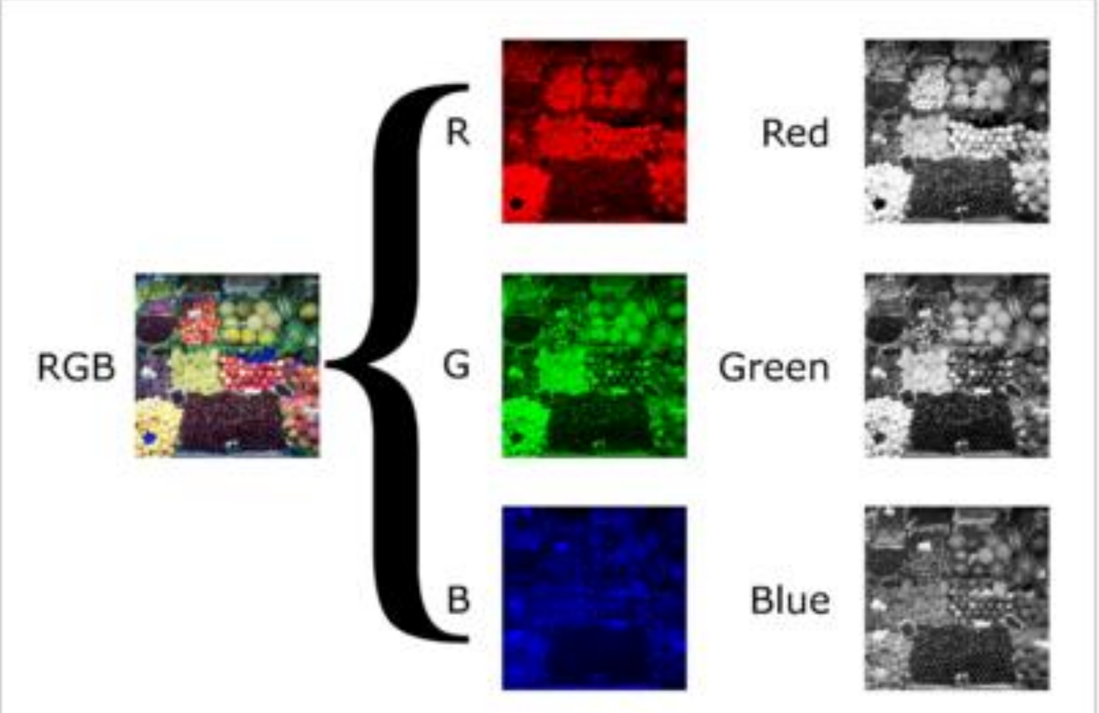
Grayscale & Color image



Links:
[1]

Grayscale as single channels of multichannel color images

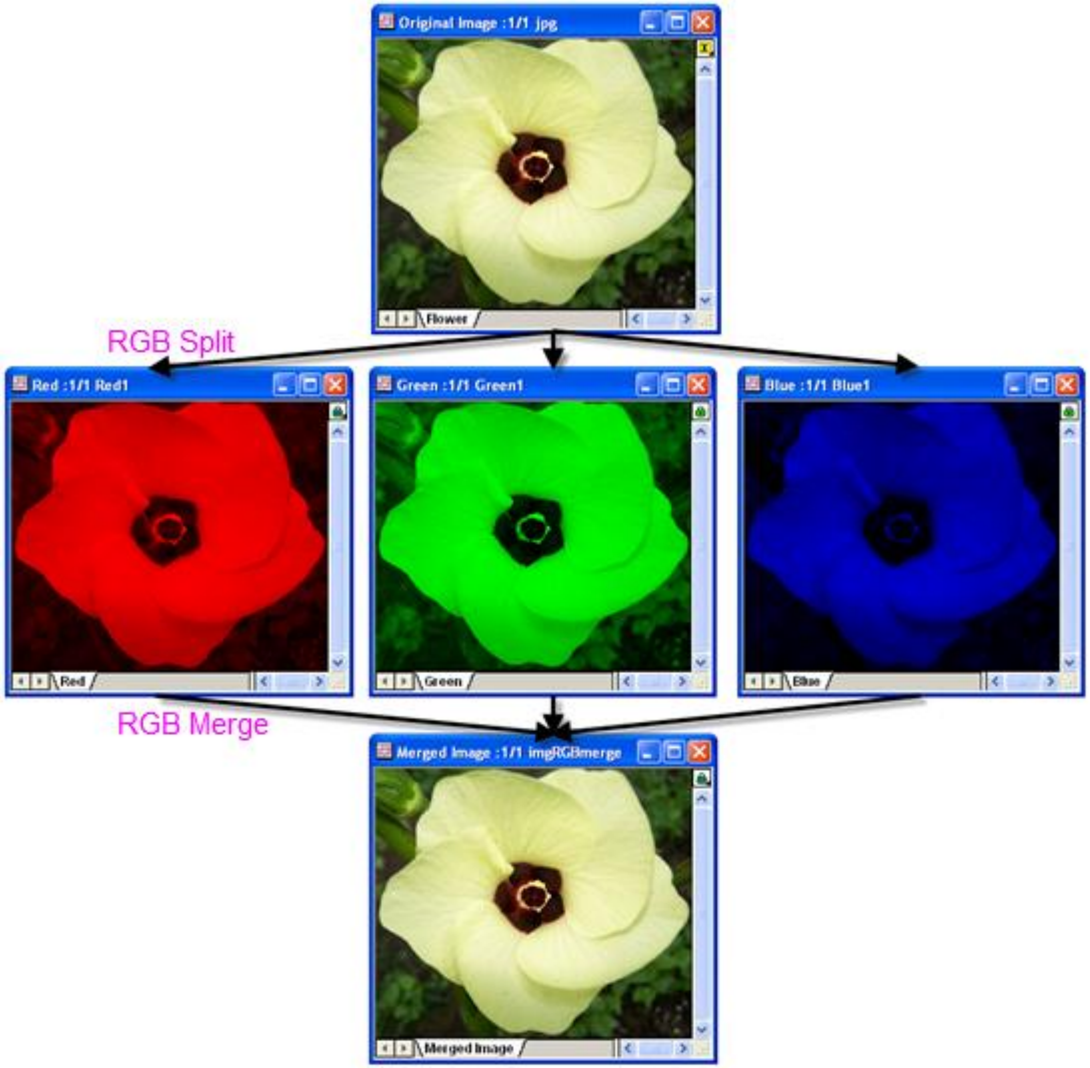
- Color images are often built of several stacked color channels
  - each of them representing value levels of the given channel
  - For example, RGB images are composed of three independent channels for red, green and blue primary color components



Composition of RGB from 3 Grayscale images

https://en.wikipedia.org/wiki/Grayscale

Grayscale as single channels of multichannel color images (ct)



RGB to Grayscale

- The lightness method

$$I = (\max(R, G, B) + \min(R, G, B)) / 2$$

- The average method

$$I = (R + G + B) / 3$$

- The luminosity method

$$I = 0.21\,R + 0.72\,G + 0.07\,B$$

https://www.johndcook.com/blog/2009/08/24/algorithms-convert-color-grayscale/

Common types of an image (ct)

Ex. 8

- Write three pseudo codes to convert an RGB image to a grayscale one based on three corresponding conversion methods at the previous slide.

Common types of an image (ct)

Gray Among The RGB

- As a color's three red/green/blue numbers are equal, the color is a shade of gray
  - red=50 green=50 blue=50 is gray, without any bias towards red, green, or blue hue
  - red=75 green=50 blue=50 it would be a bit reddish
- Examples of gray colors in RGB:

red	green	blue	color
50	50	50	dark gray
120	120	120	medium gray
200	200	200	light gray
0	0	0	black
255	255	255	white

Common types of an image (ct)

Gray Among The RGB (ct)

Grey color codes chart

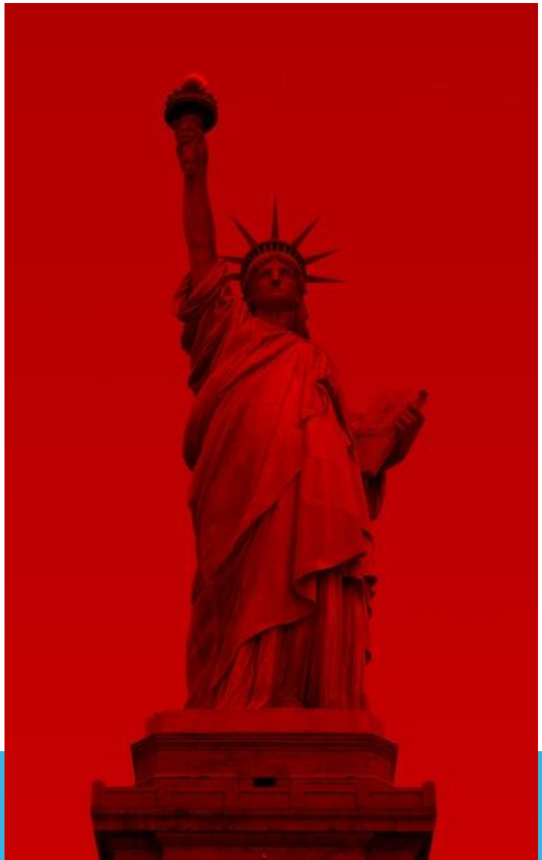
Color	HTML / CSS Color Name	Hex Code #RRGGBB	Decimal Code (R,G,B)
	gainsboro	#DCDCDC	rgb(220,220,220)
	lightgray / lightgrey	#D3D3D3	rgb(211,211,211)
	silver	#C0C0C0	rgb(192,192,192)
	darkgray / darkgrey	#A9A9A9	rgb(169,169,169)
	gray / grey	#808080	rgb(128,128,128)
	dimgray / dimgrey	#696969	rgb(105,105,105)
	lightslategray / lightslategrey	#778899	rgb(119,136,153)
	slategray / slategrey	#708090	rgb(112,128,144)
	darkslategray / darkslategrey	#2F4F4F	rgb(47,79,79)
	black	#000000	rgb(0,0,0)

Common types of an image (ct)

Red Liberty Example Problem

- All pixels is in the red values:
  - the whole image looks red. The green and blue values are all zero. This image looks quite wrong.
  - Write code to fix this image by copying the red value over to be used as the green and blue value. So for a pixel, if red is 27, set green and blue to also be 27.

red	green	blue
65	0	0
53	0	0
100	0	0
19	0	0
...	0	0



Common types of an image (ct)

RGB brightness

- Q: How to decide which pixel below is brightest? darkest?
  - The average combines and summarizes the three values into one number 0..255
  - The average shows how bright the pixel is, ignoring hue: 0 = totally dark, 255=totally bright

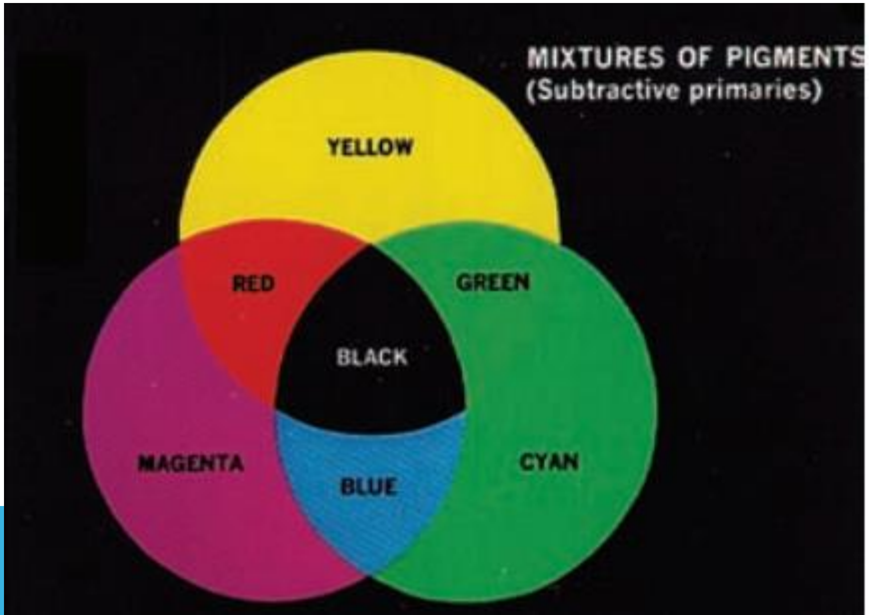
	red	green	blue	average
				average = (red + green + blue) / 3
pixel-1	200	50	50	100 (medium bright)
pixel-2	0	75	75	50 (darkest)
pixel-3	100	250	250	200 (brightest)

https://web.stanford.edu/class/cs101/image-6-grayscale.html

Common types of an image (ct)

Other color models

- CMY and CMYK (cyan, magenta, yellow, black) use subtractive color mixing, and are used in the printing process
  - When a surface coated with cyan pigment is illuminated with white light, no red light is reflected from the surface. That is, cyan subtracts red light from reflected white light.
  - When light falls on a pigment, that pigment absorbs part of the spectrum and reflects the rest, resulting in its color. An object that is *green* has only reflected *green* light and absorbed everything else.
  - This subtractive quality of pigments also means that mixing colors always produces a darker color.

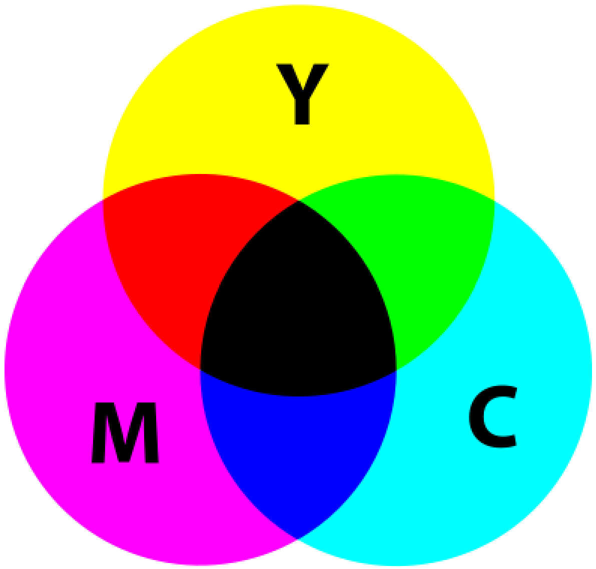


https://www.youtube.com/watch?v=r1djX9htjFU

Common types of an image (ct)

Other color models (ct)

- CMYK - four-color printing
  - Equal amounts of cyan, magenta, and yellow should produce black. However, in real life, combining these pigments produces a muddy-colored black. To produce pure black, which is quite commonly used while printing, we add a fourth color — black, to the pigment mixture.
  - In CMYK, higher values are associated with darker colors rather than lighter ones



https://www.geeksforgeeks.org/python-cmy-and-cmyk-color-models/

https://www.w3schools.com/colors/colors_cmyk.asp

Common types of an image (ct)

Other color models (ct)

- RGB & CMY conversions
  - White light minus red leaves cyan, green subtracted from white leaves magenta, and white minus blue returns yellow
  - All the values are represented on the scale of 0 to 100% or scale $[0,1]$

$$\begin{bmatrix} C \\ M \\ Y \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} - \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

  - where all RGB color values have been normalized to the range $[0,1]$
  - Light reflected from a surface coated with pure cyan does not contain red (that is, $C = 1 - R$ in the equation)
  - Pure magenta does not reflect green, and pure yellow does not reflect blue

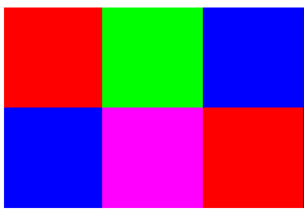
https://www.geeksforgeeks.org/python-cmy-and-cmyk-color-models/

Common types of an image (ct)

Ex. 9

- Write a pseudo code to covert an image from RGB to CMY color space and vice versa.

- Manually convert the following RGB image to a CMY image:



Pixel values in the CMY model ???

```
dip00_1 = np.array([
                    [ [0, 0, 255], [0, 255, 0], [255, 0, 0] ],
                    [ [255, 0, 0], [255, 0, 255], [0, 0, 255] ]
                   ]

                 , dtype=np.uint8)
```

BGR order

Common types of an image (ct)

Other color models (ct)

- CMY to CMYK
- Step 1: $K = min(C, M, Y)$
- Step 2:
  - $K = 1$: $C^* = M^* = Y^* = 0$
  - $K \neq 1$: $\begin{cases} C^* = (C - K)/(1 - K) \\ M^* = (M - K)/(1 - K) \\ Y^* = (Y - K)/(1 - K) \end{cases}$
    - C^*, M^*, Y^* are belong to CMYK model

- CMYK to CMY

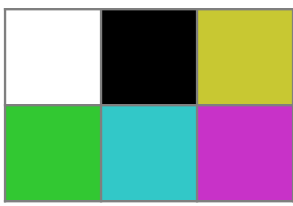
$$\begin{cases} C = C^* \times (1 - K) + K \\ M = M^* \times (1 - K) + K \\ Y = Y^* \times (1 - K) + K \end{cases}$$

https://www.w3schools.com/colors/colors_cmyk.asp

Common types of an image (ct)

Ex. 10

- Write a pseudo code to covert an image from RGB to CMYK color space.
- Manually convert the following RGB image to a CMYK image:



Pixel values in the CMYK model ???

255, 255, 255	0, 0, 0	200, 200, 50
50, 200, 50	50, 200, 200	200, 50, 200

RGB order

Common types of an image (ct)

Other color models (ct)

- RGB, CMY, and other similar color models are not well suited for describing colors in terms that are practical for human interpretation

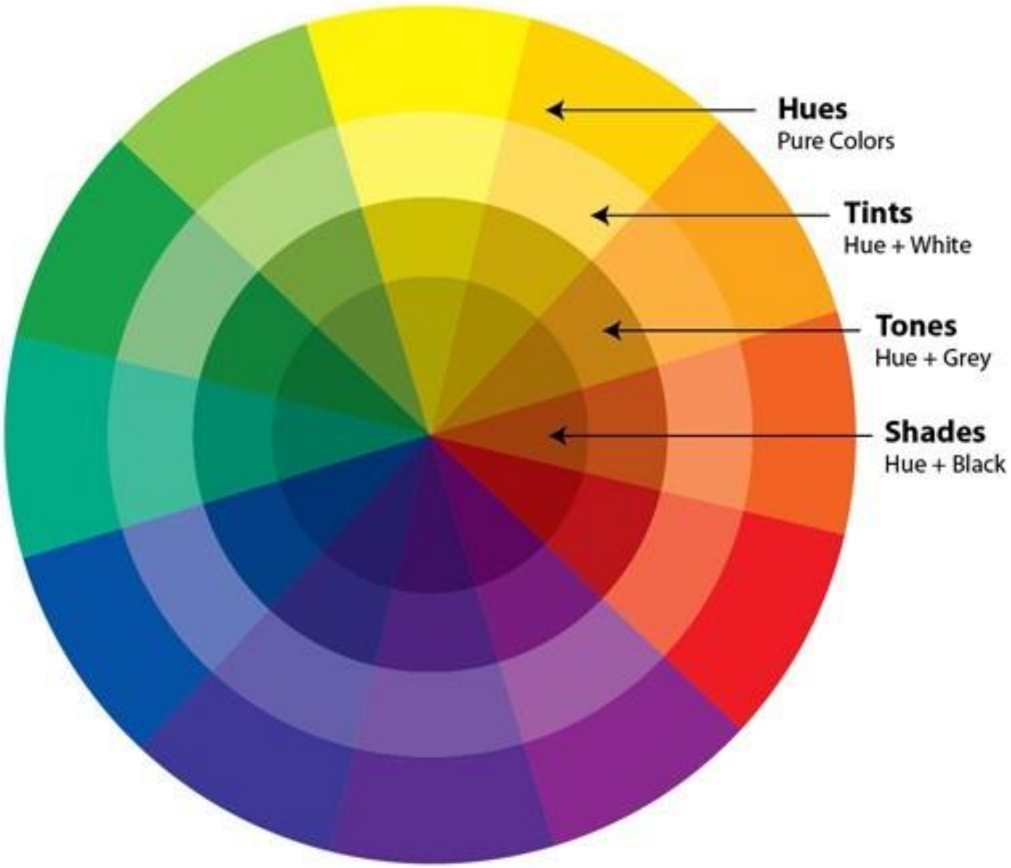
- When humans view a color object, we describe it by its hue, saturation, and brightness

Common types of an image (ct)

Other color models (ct)

- Hue is a color attribute that describes a pure color (pure yellow, …)
  - hue represents dominant color as perceived by an observer. Thus, when we call an object red, orange, or yellow, we are referring to its hue.

https://www.beachpainting.com/blog/color-hue-tint-tone-and-shade/



Common types of an image (ct)

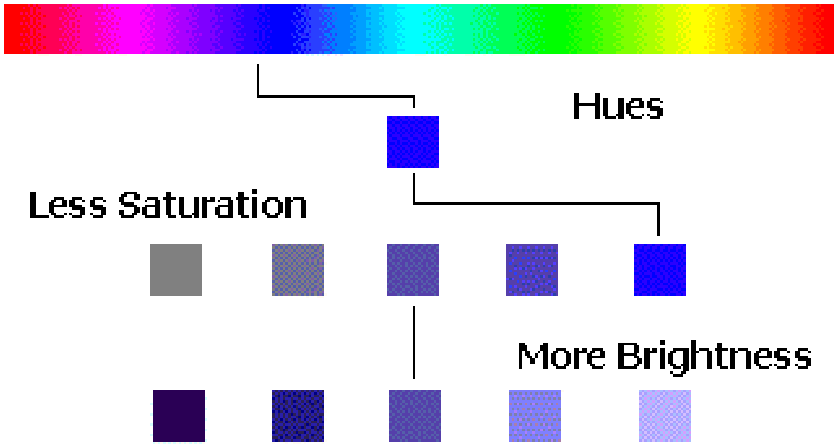
Other color models (ct)

- Saturation refers to the relative purity or the amount of white light mixed with a hue
- Saturation gives a measure of the degree to which a pure color is diluted by white light
- Saturation describes the intensity (purity) of the Hue
  - degree of saturation being inversely proportional to the amount of white light added
    - pure spectrum colors are fully saturated
    - colors such as pink (red and white) and lavender (violet and white) are less saturated
- A grayscale or black-and-white photo has no color saturation, while a full-color photo of a field of sunlit wildflowers might be extremely saturated

Common types of an image (ct)

Other color models (ct)

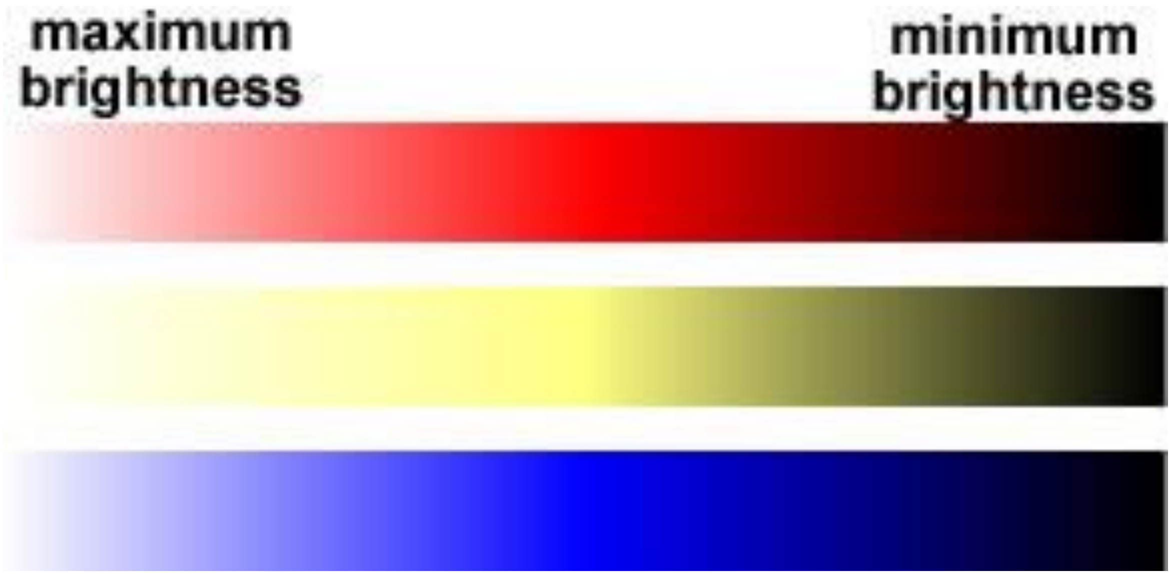
- How saturation changes the feel of a photo?
  - Highly saturated photos can look artificial, so use saturation with care — especially if you're going for a natural look
  - "It's rare to see pure colors in nature because ambient light makes colors less saturated," says photographer Heather Barnes. "That's why you have to be really careful in post to not overly saturate your color."



Common types of an image (ct)

Other color models (ct)

- Brightness embodies the achromatic notion of intensity
- Brightness is the amount of white or black mixed in with the color. It's also calculated as a percentage value between 0% and 100%.
- Brightness is the perceived intensity of light coming from a screen



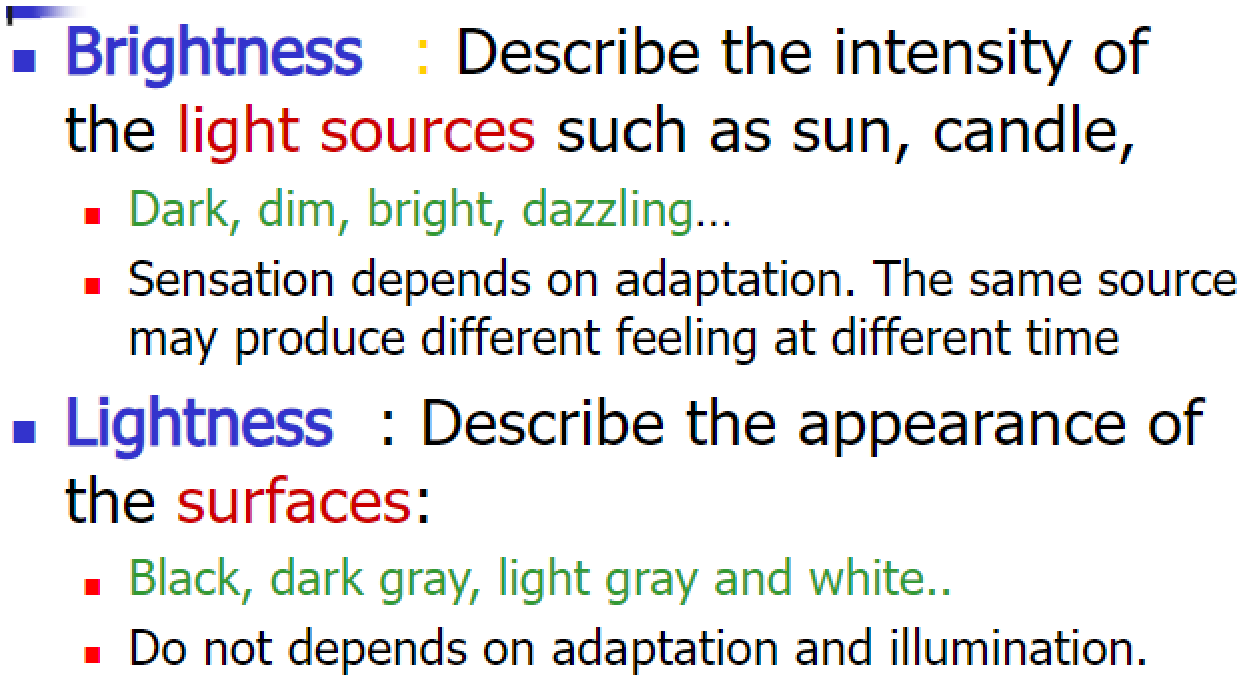
https://web.mst.edu/~rhall/web_design/color_mixing.html

Common types of an image (ct)

Other color models (ct)

- Brightness: Perceived amount of light coming from a source
- Lightness: Perceived reflectance of a surface, for ex., white surface is light, black surface is dark
- In simplest terms, brightness is the appearance of luminance and lightness is the appearance of objects

https://slideplayer.com/slide/5102224/



Common types of an image (ct)

Other color models (ct)

- HSV (hue, saturation, value), also known as HSB (hue, saturation, brightness) is often used by artists
  - it is often more natural to think about a color in terms of hue and saturation than in terms of additive or subtractive color components
  - It is the most accurate color model as long as the way humans perceive colors
  - HSV is a transformation of an RGB colorspace, and its components and colorimetry are relative to the RGB colorspace from which it was derived
  - HSV color space separates the luma and hue from the color information, thus HSV model is used in histogram equalization and color identification

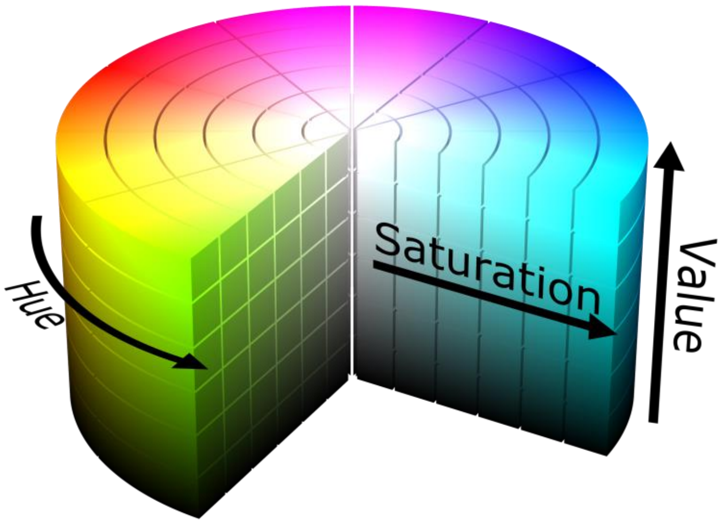
https://www.geeksforgeeks.org/hsv-color-model-in-computer-graphics/

Common types of an image (ct)

Other color models (ct)

- Hue represents the pure color, and ranges from 0 to 360 degrees
- Saturation: ranges from 0% to 100%
  - 100% saturation means that complete pure color is added
  - 0% saturation means no color is added, resulting in grayscale
- Value: represents the brightness, ranges from 0% to 100%

Hue (in degree)	Color
0-60	Red
60-120	Yellow
120-180	Green
180-240	Cyan
240-300	Blue
300-360	Magenta



https://colorpicker.me/#ff00aa

Common types of an image (ct)

Other color models (ct)

- RGB to HSV

1. Divide r, g, b by 255

2. Compute cmax, cmin, difference

3. Hue calculation :

- *if cmax and cmin are equal, then h = 0*
- *if cmax equal r then compute h = (60 * ((g – b) / diff) + 360) % 360*
- *if cmax equal g then compute h = (60 * ((b – r) / diff) + 120) % 360*
- *if cmax equal b then compute h = (60 * ((r – g) / diff) + 240) % 360*

4. Saturation computation :

- *if cmax = 0, then s = 0*
- *if cmax does not equal 0 then compute s = (diff/cmax)*100*

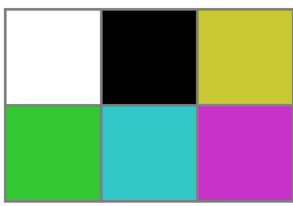
5. Value computation :

- *v = cmax*100*

https://www.geeksforgeeks.org/program-change-rgb-color-model-hsv-color-model/

Common types of an image (ct)
Ex. 11

- Manually convert the following RGB image to a HSV image:



Pixel values in the HSV model ???

255, 255, 255	0, 0, 0	200, 200, 50
50, 200, 50	50, 200, 200	200, 50, 200

RGB order

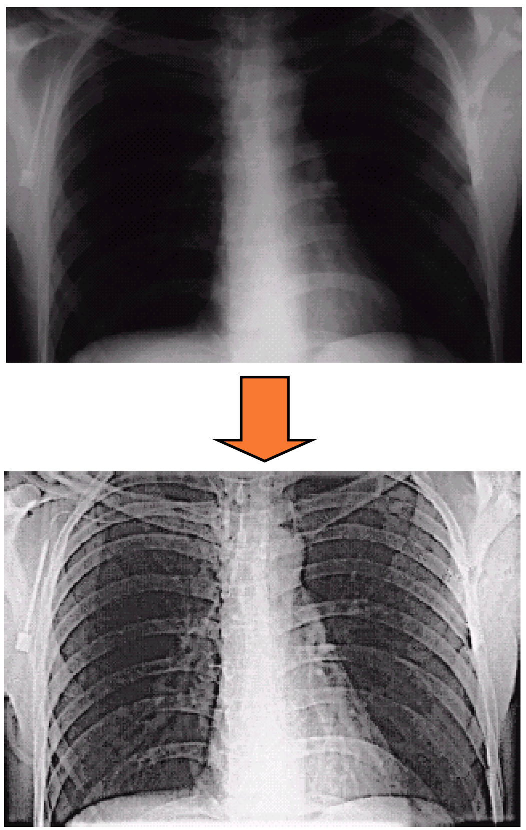
Common types of an image (ct)

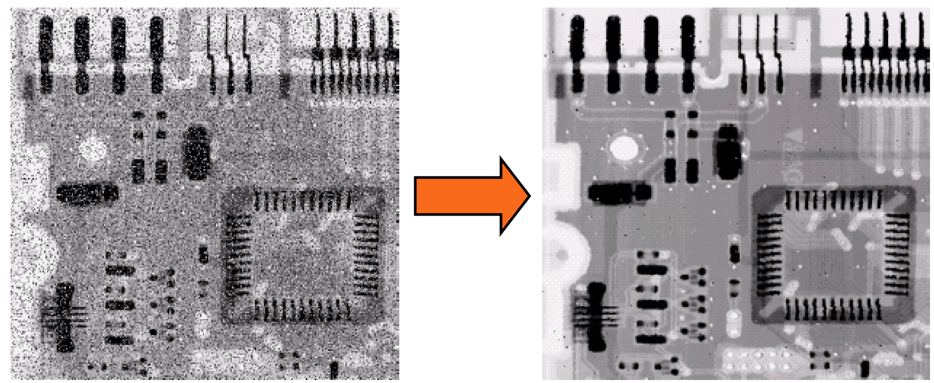
Other color models (ct)

- HSL (hue, saturation, lightness/luminance), also known as HLS or HSI (hue, saturation, intensity) is quite similar to HSV, with "lightness" replacing "brightness"

Examples: Image Enhancement

- One of the most common uses of DIP techniques: improve quality, remove noise etc





Examples: The Hubble Telescope

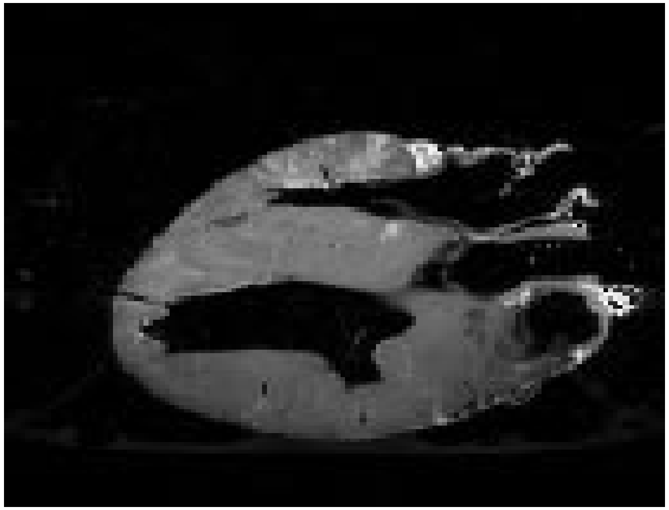
- Launched in 1990 the Hubble telescope can take images of very distant objects
- However, an incorrect mirror made many of Hubble's images useless
- Image processing techniques were used to fix this

Examples: Artistic Effects

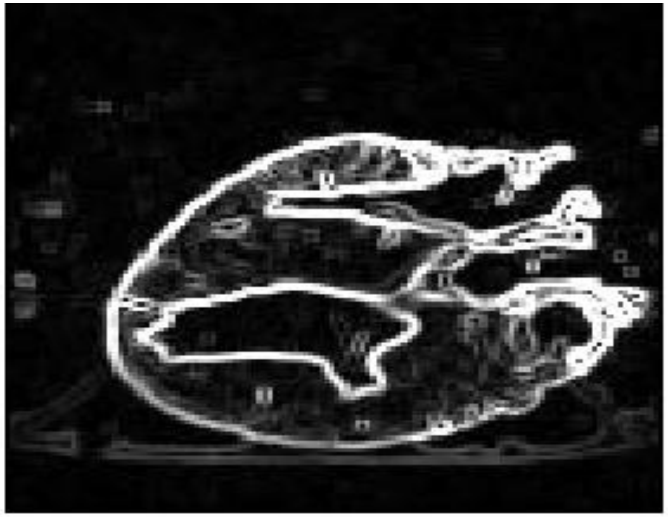
- Artistic effects are used to make images more visually appealing, to add special effects and to make composite images

Examples: Medicine

- Take slice from MRI scan of canine heart, and find boundaries between types of tissue
  - Image with gray levels representing tissue density
  - Use a suitable filter to highlight edges



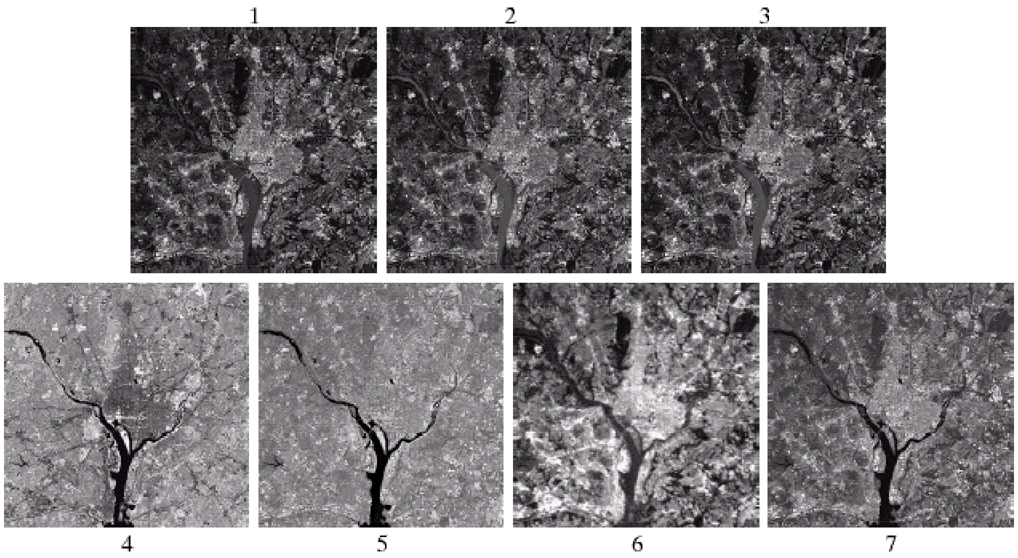
Original MRI Image of a Dog Heart

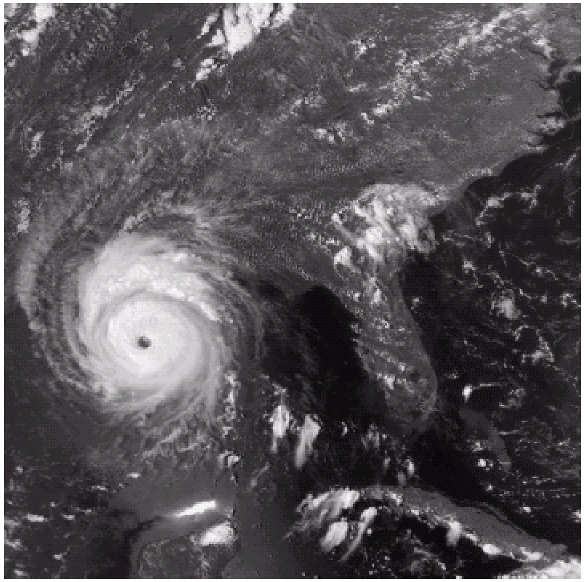


Edge Detection Image

Examples: GIS

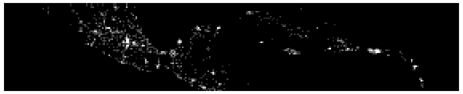
- Geographic Information Systems
  - Digital image processing techniques are used extensively to manipulate satellite imagery
  - Terrain classification
  - Meteorology





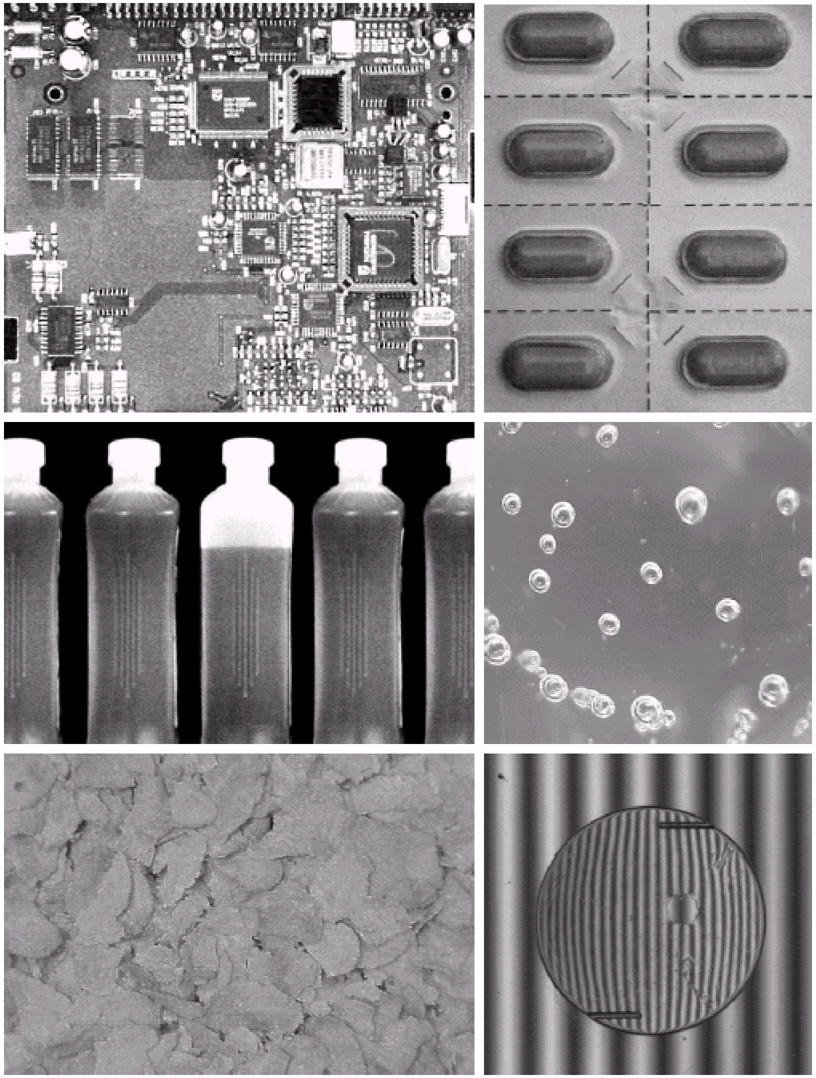
Examples: GIS (ct)

- Night-Time Lights of the World data set
  - Global inventory of human settlement
  - Not hard to imagine the kind of analysis that might be done using this data



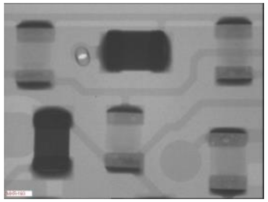
Examples: Industrial Inspection

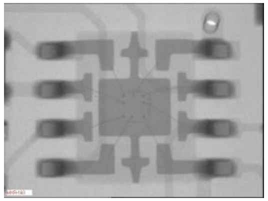
- Human operators are expensive, slow and unreliable
- Make machines do the job instead
- Industrial vision systems are used in all kinds of industries
- Can we trust them?

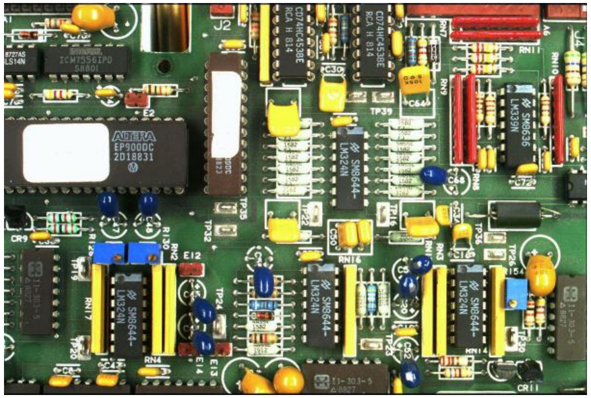


Examples: PCB Inspection

- Printed Circuit Board (PCB) inspection
  - Machine inspection is used to determine that all components are present and that all solder joints are acceptable
  - Both conventional imaging and x-ray imaging are used

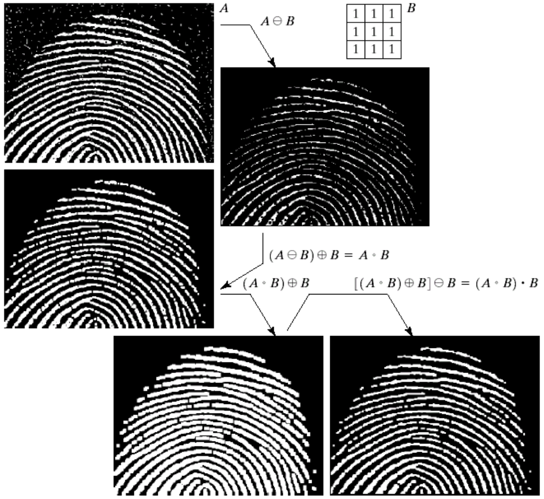






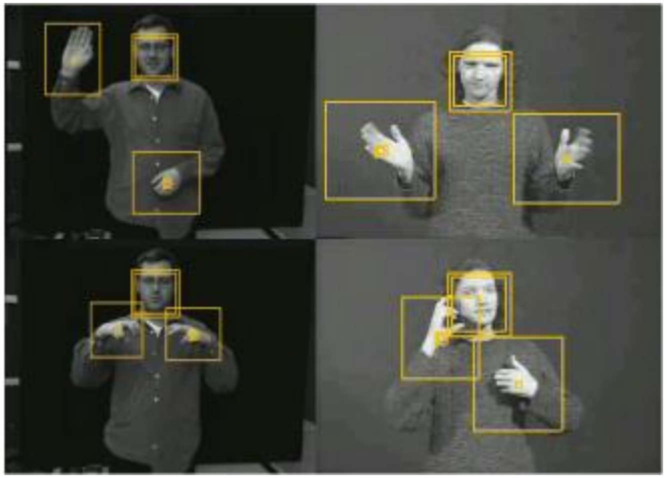
Examples: Law Enforcement

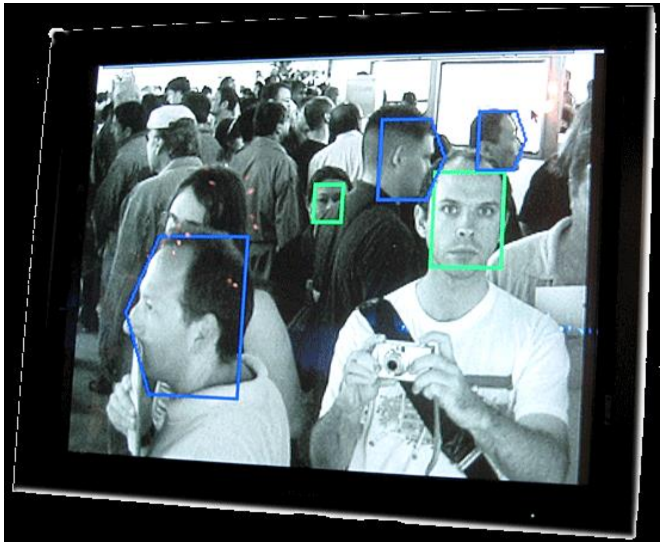
- Image processing techniques are used extensively by law enforcers
  - Number plate recognition for speed cameras/automated toll systems
  - Fingerprint recognition
  - Enhancement of CCTV images



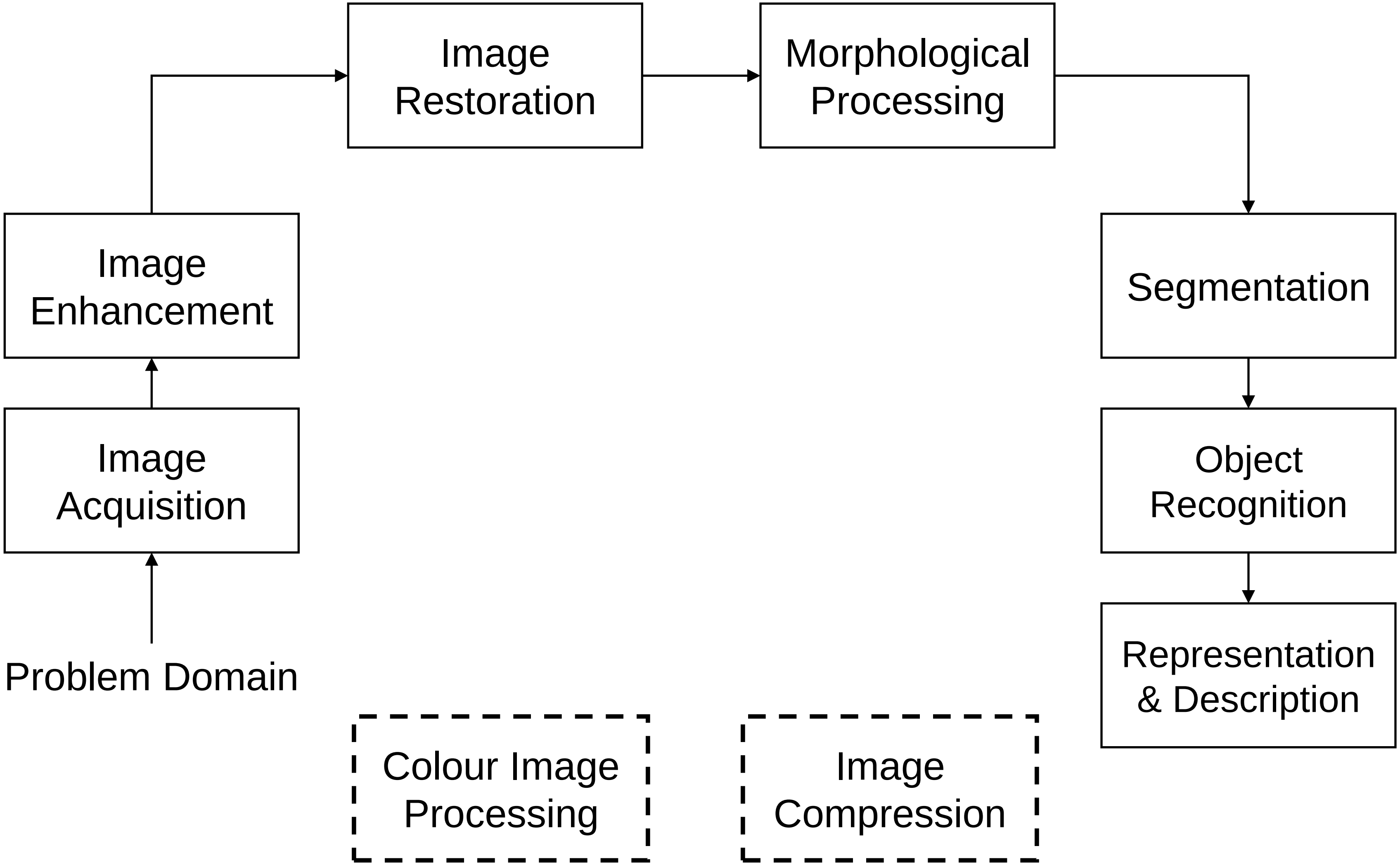
Examples: HCI

- Try to make human computer interfaces more natural
  - Face recognition
  - Gesture recognition





Key Stages in Digital Image Processing



Key Stages in Digital Image Processing: Image Aquisition

Image Restoration

Morphological Processing

Image Enhancement

Segmentation

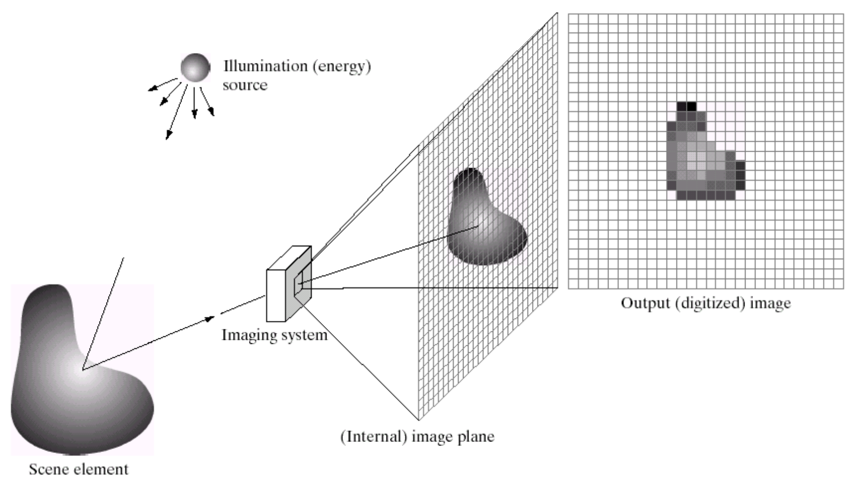


Image Acquisition

Object Recognition

Representation & Description

Problem Domain

Colour Image Processing

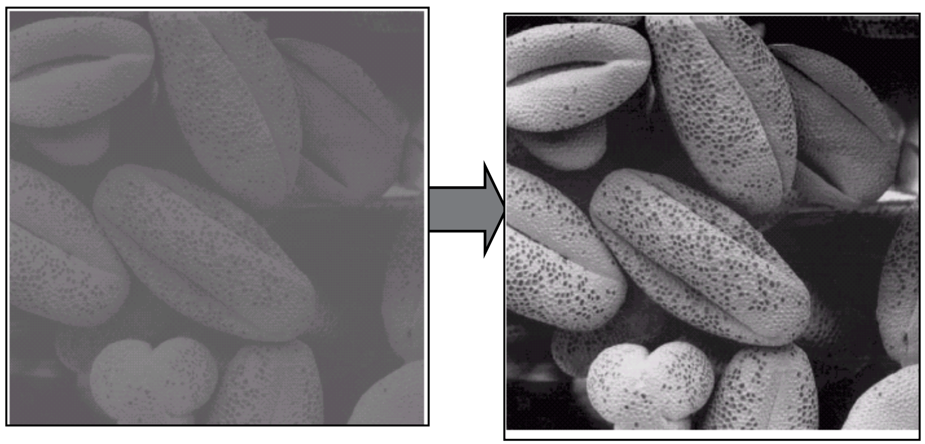
Image Compression

Key Stages in Digital Image Processing: Image Enhancement

Image Restoration

Morphological Processing

Image Enhancement



Segmentation

Image Acquisition

Object Recognition

Problem Domain

Representation & Description

Colour Image Processing

Image Compression

Key Stages in Digital Image Processing: Image Restoration

Image Restoration

Morphological Processing

Image Enhancement

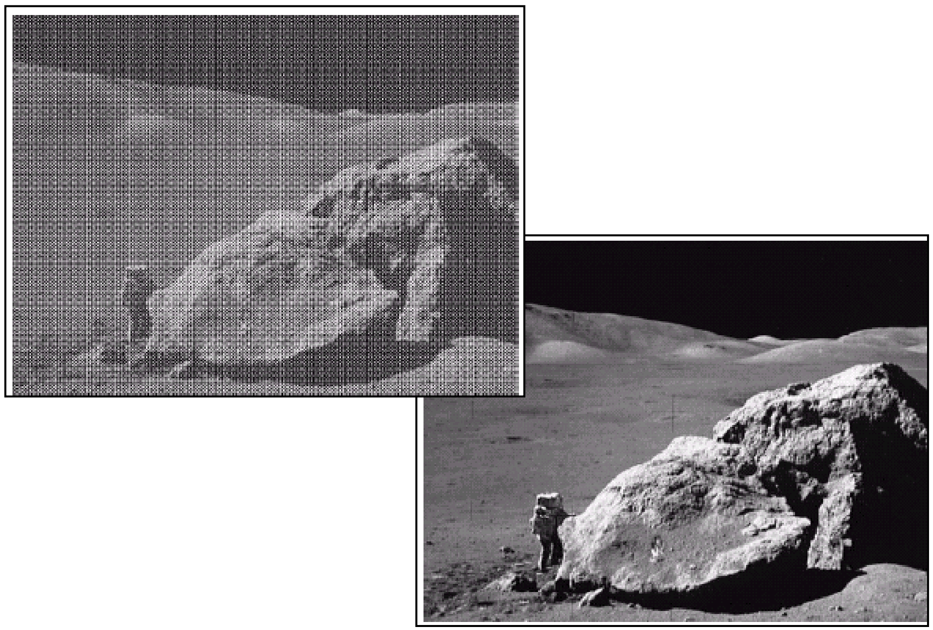
Segmentation

Image Acquisition

Object Recognition

Problem Domain

Representation & Description



Colour Image Processing

Image Compression

Key Stages in Digital Image Processing: Morphological Processing

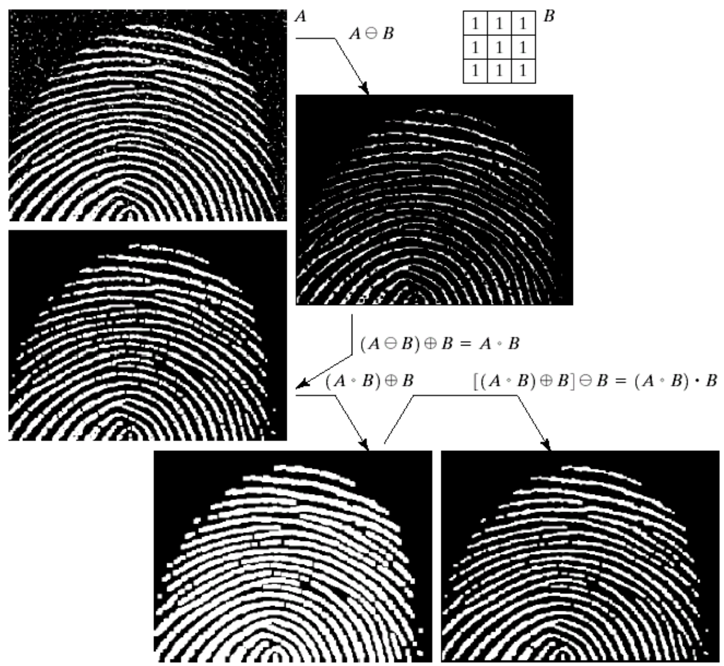
Image Restoration

Morphological Processing

Image Enhancement

Image Acquisition

Problem Domain



Segmentation

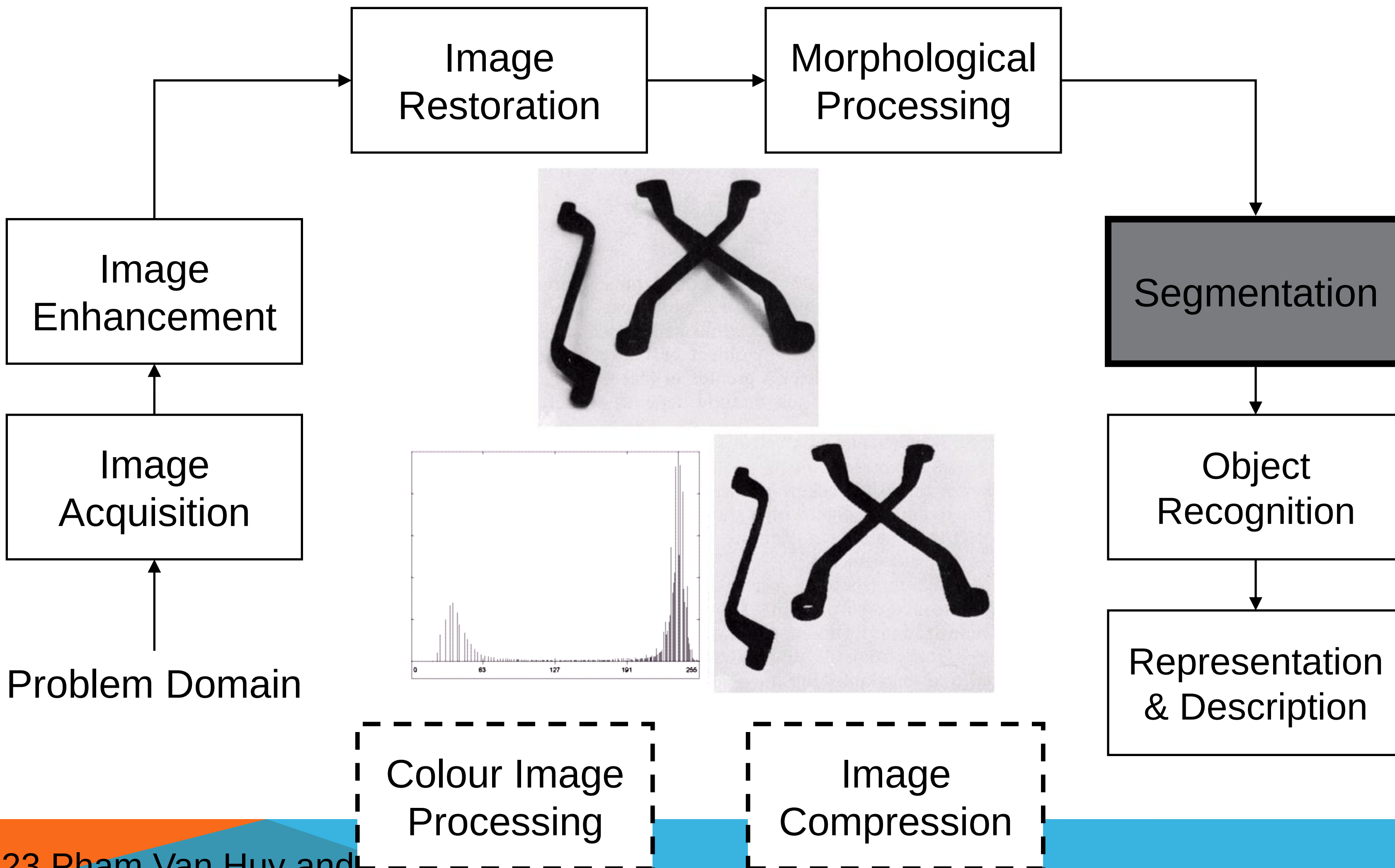
Object Recognition

Representation & Description

Colour Image Processing

Image Compression

Key Stages in Digital Image Processing: Segmentation

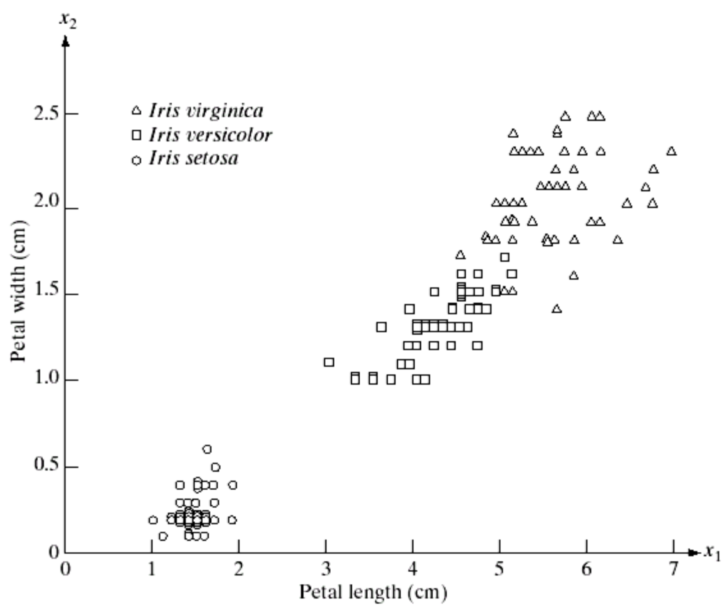


Key Stages in Digital Image Processing: Object Recognition

Image Restoration → Morphological Processing → Segmentation

Problem Domain → Image Acquisition → Image Enhancement → Image Restoration

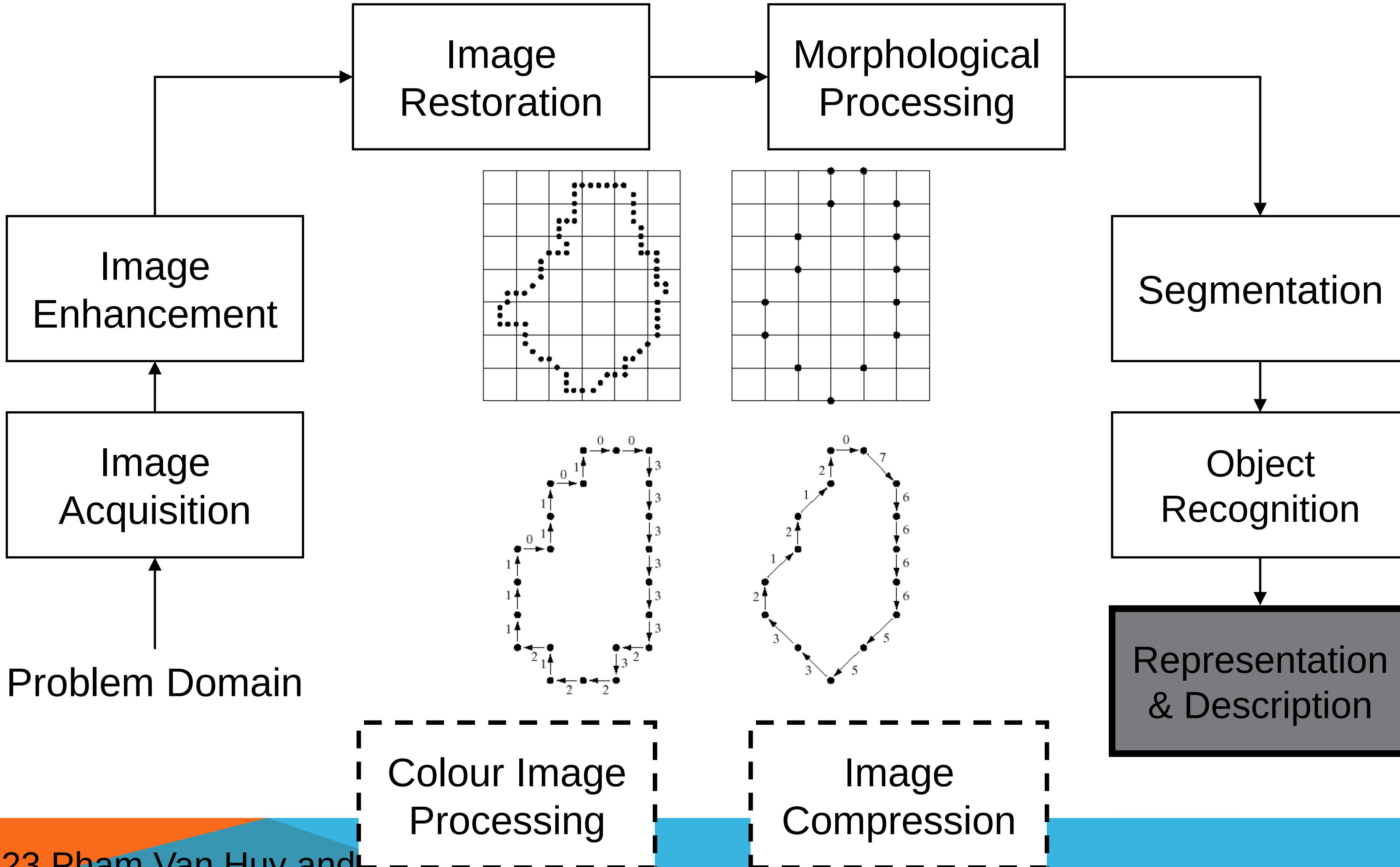
Segmentation → **Object Recognition** → Representation & Description



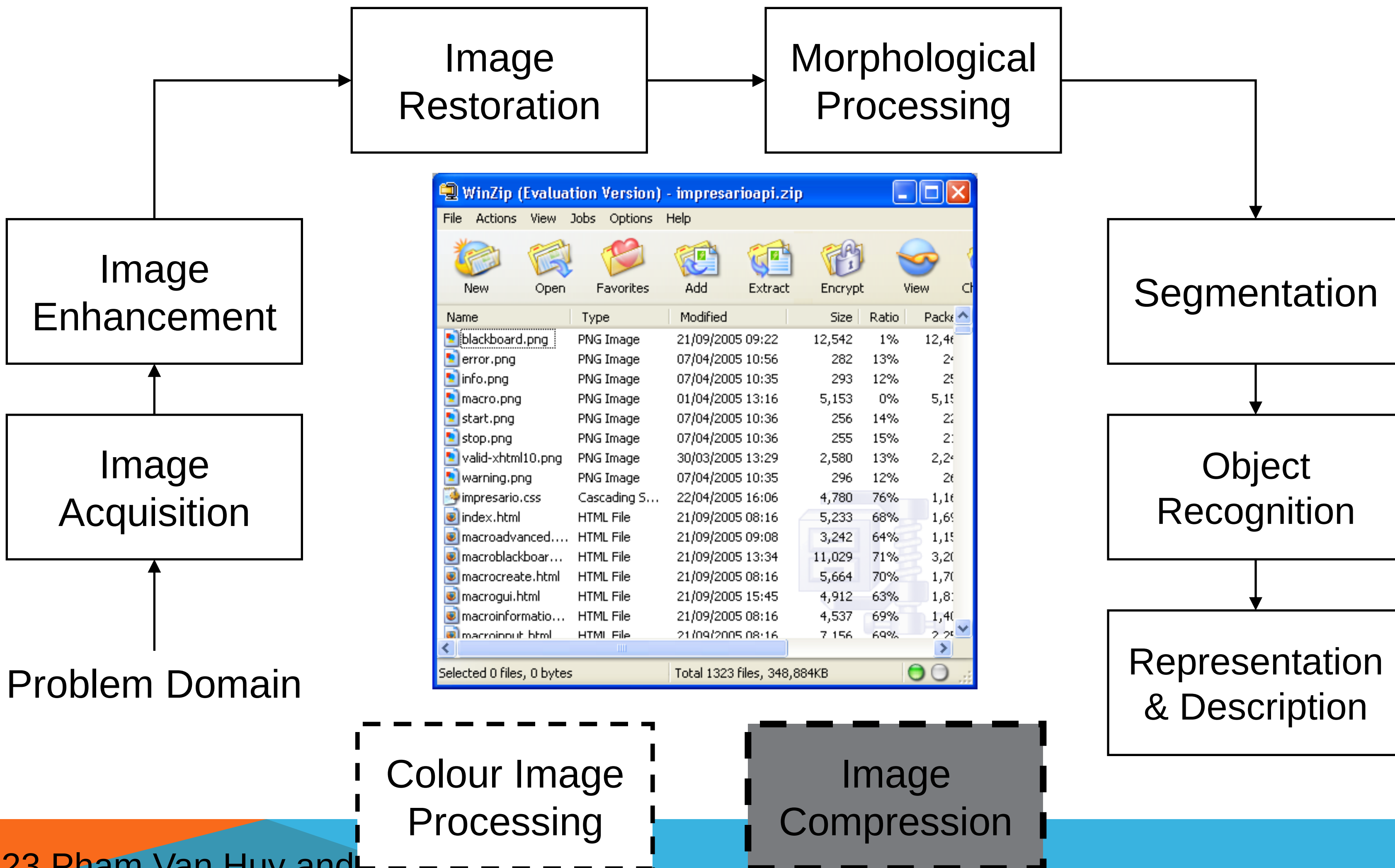
Colour Image Processing

Image Compression

Key Stages in Digital Image Processing: Representation & Description



Key Stages in Digital Image Processing: Image Compression

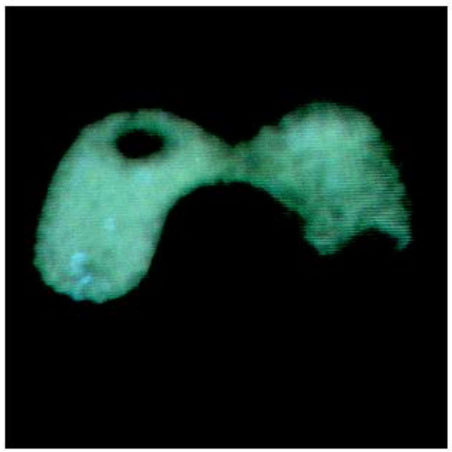


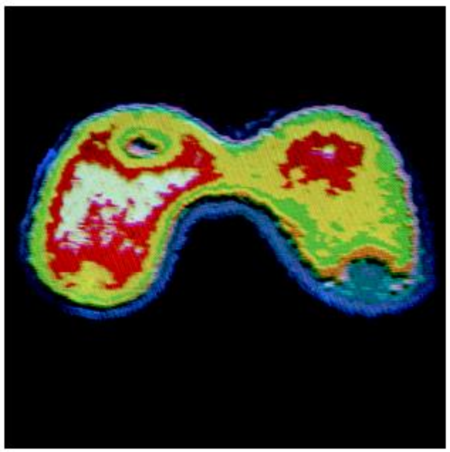
Key Stages in Digital Image Processing: Colour Image Processing

Image Restoration

Morphological Processing

Image Enhancement





Segmentation

Image Acquisition

Object Recognition

Problem Domain

Representation & Description

Colour Image Processing

Image Compression

References

- Rafael C. Gonzalez, Richard E. Woods, [2018], Digital image processing, 4th edition, Pearson.
- https://www.geeksforgeeks.org/digital-image-processing-basics/
- https://www.tutorialspoint.com/dip/index.htm
- https://www.linkedin.com/pulse/what-computer-vision-explanation-types-examples-neil-sahota-%E8%90%A8%E5%86%A0%E5%86%9B-/
- https://www.geeksforgeeks.org/difference-between-image-processing-and-computer-vision/
- https://www.tutorialspoint.com/dip/computer_vision_and_graphics.htm
- https://www.tutorialspoint.com/dip/grayscale_to_rgb_conversion.htm

References (ct)

- https://web.stanford.edu/class/cs101/image-6-grayscale.html