Machine Learning

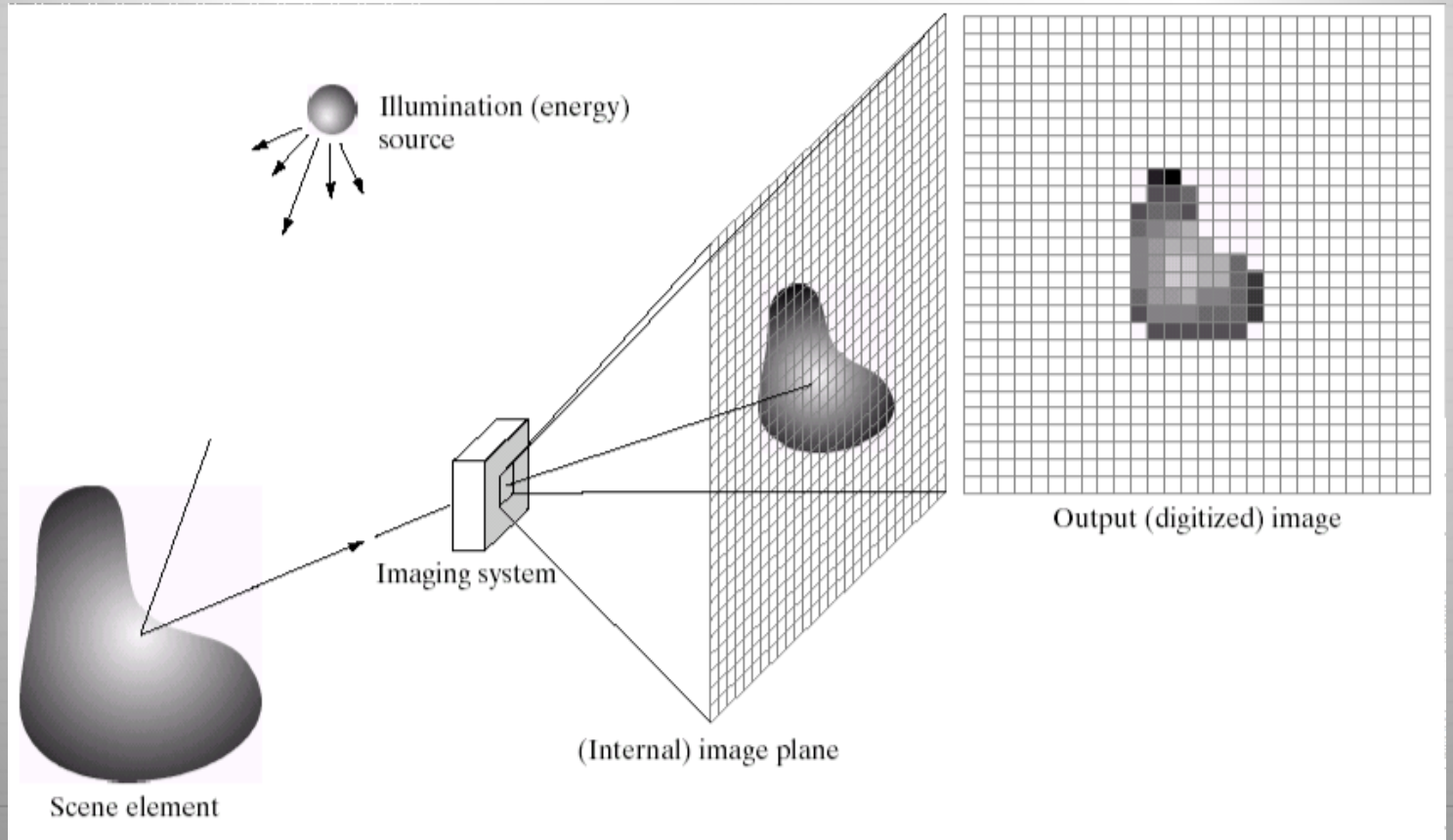
Introduction to Image Processing

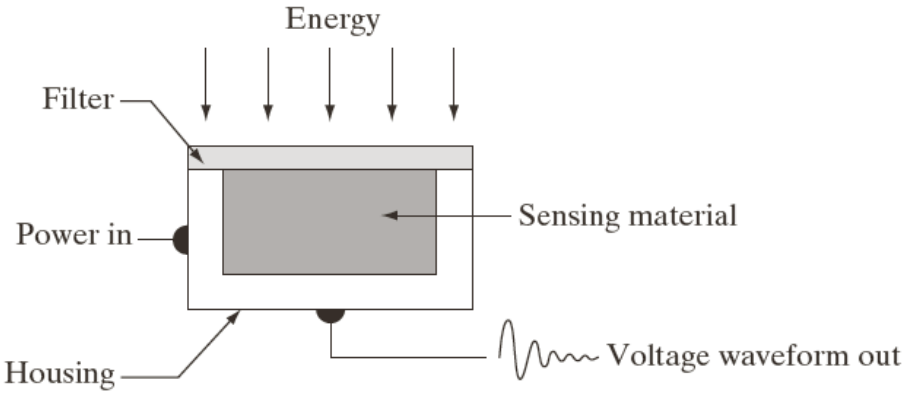
Content

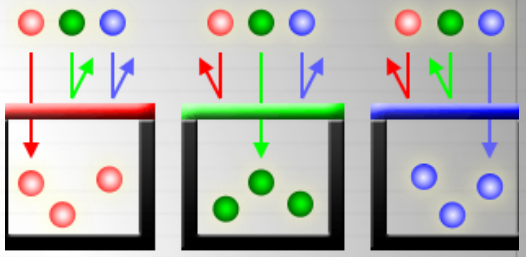
- Image Formation
- Processing Digital Images
  - Applications
  - Three levels of processing images
- Basic Spatial Operations
- Image Filtering
  - Low-pass filtering
  - High-pass filtering
- Convolution
  - Filtering as convolutional operation

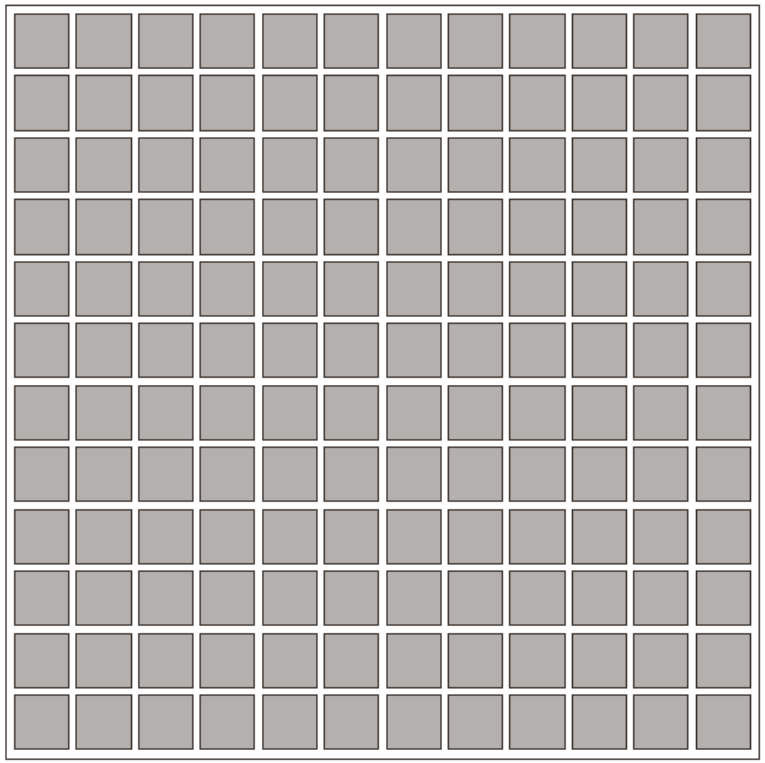
Image Formation

- Image formation is the process of measuring the intensity of the reflected light by a group of sensors



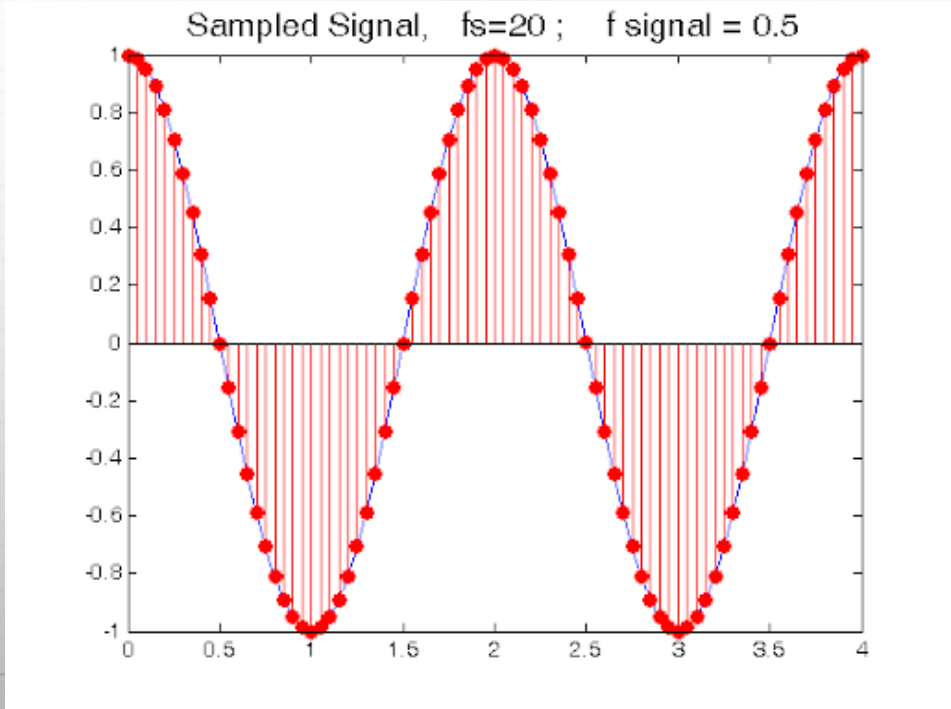

Energy
Filter
Power in
Sensing material
Housing
Voltage waveform out





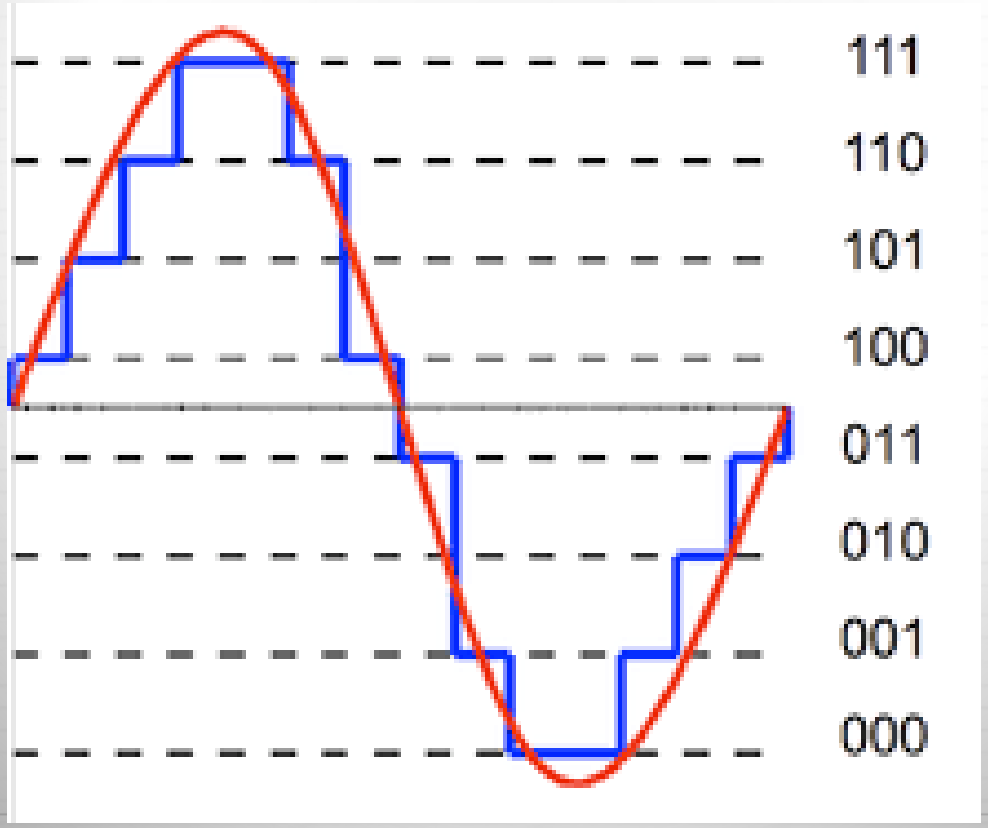
Sampling

- Each sensor on the image plane measures the intensity of the light, hitting its surface during time Δt.
- This is the same as sampling a signal at a point of time

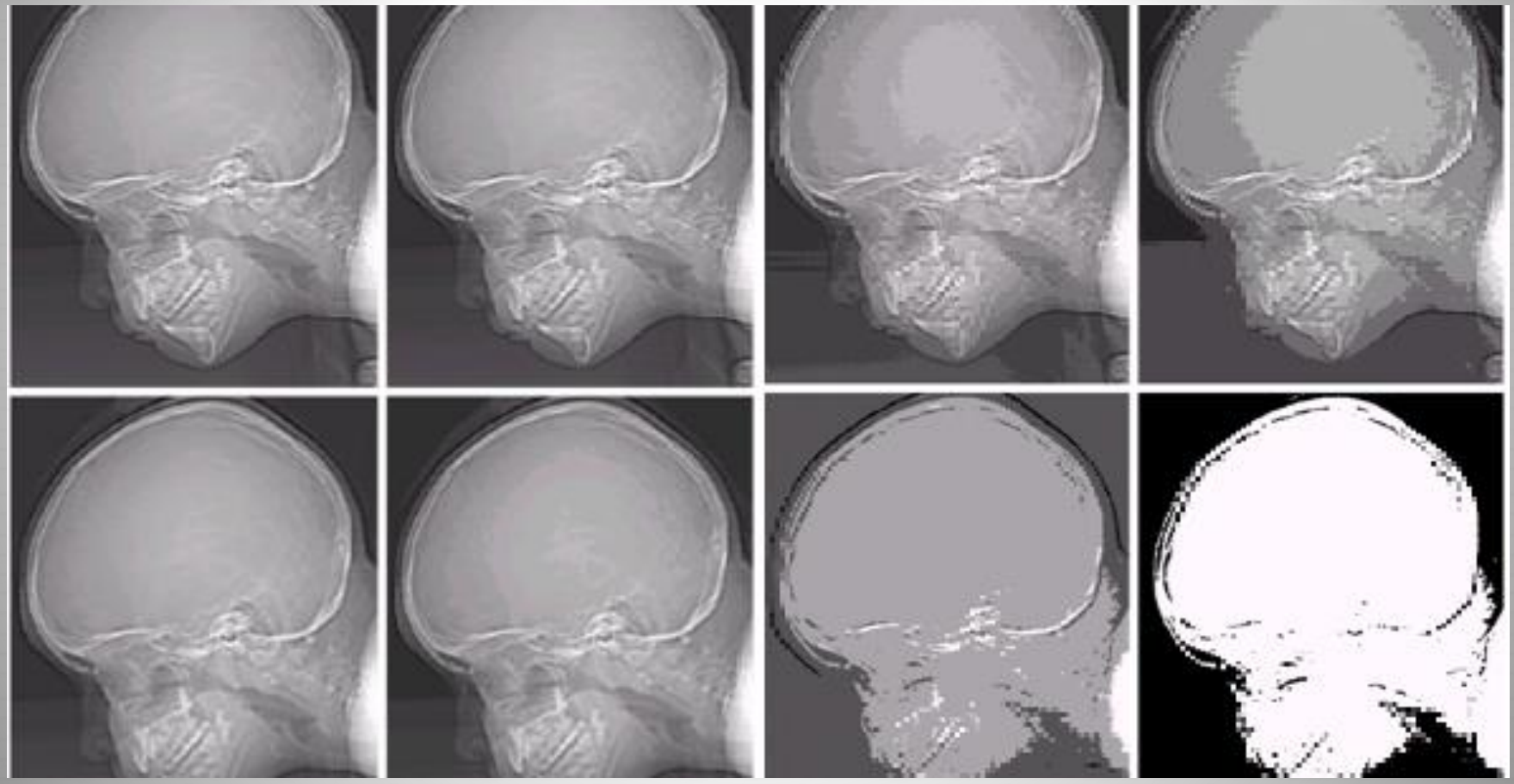


Quantization

- The values measured by each sensor are quantized to a predefined range (generally 0-255)

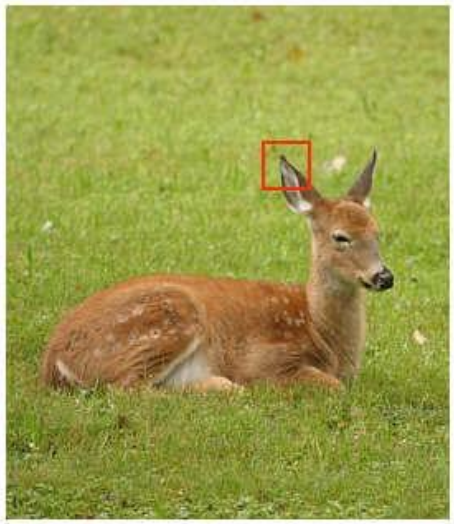


Quantization Levels

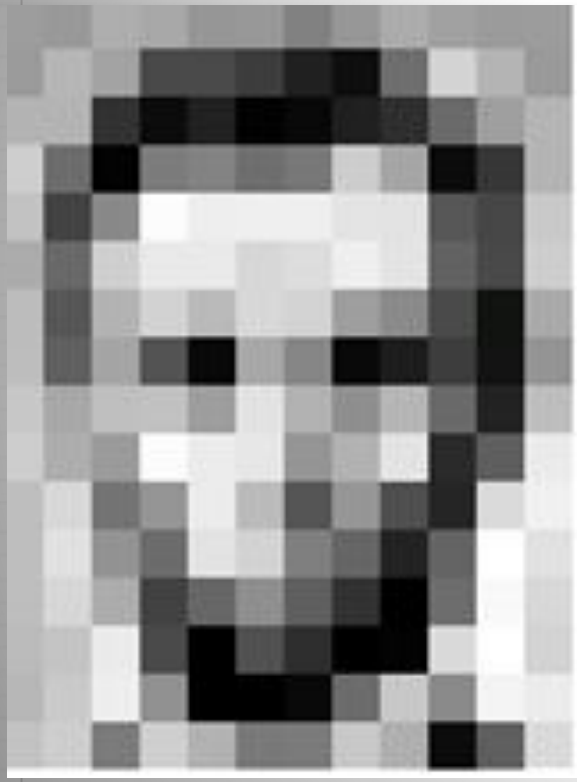


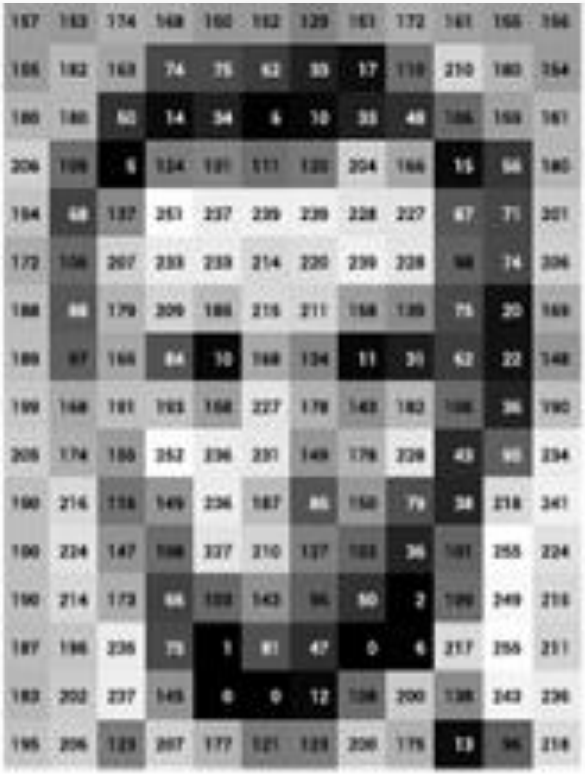
Digital Image

- Digital Image therefore, is a two-dimensional matrix of intensity values measured by the camera sensors.
- Each element of this matrix is called a picture element or a pixel.
- The matrix itself is called pixel map or pixmap



Digital Image





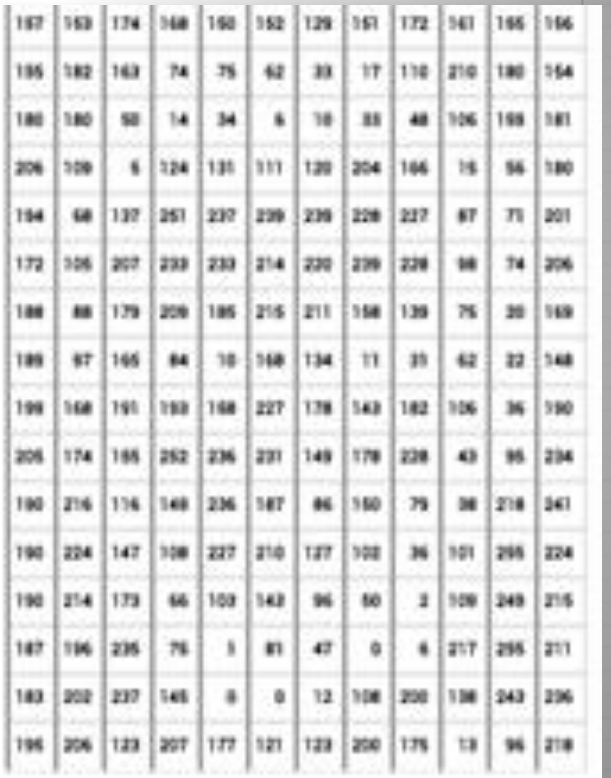
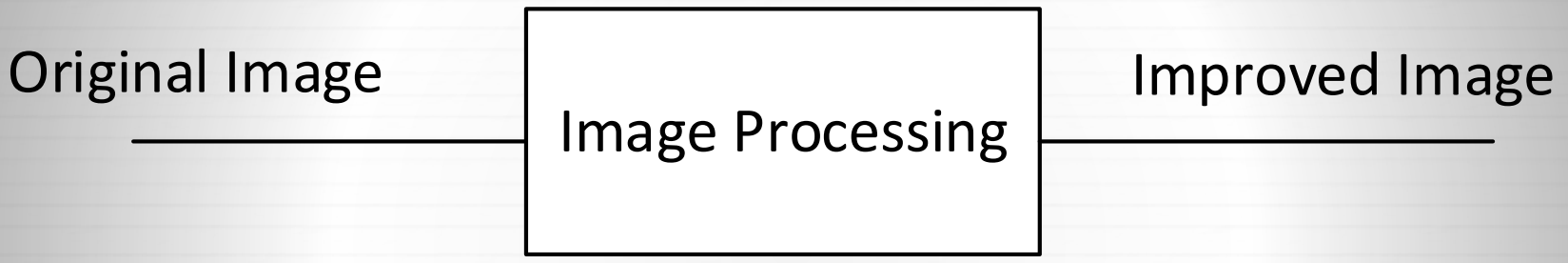


Image Processing

- Digital image processing refers to manipulating images to:
  - Improve the quality of visual information for human interpretation
    - Example: Improving image contrast
  - Processing of image data for storage, transmission, or processing by machines
    - Example: Determining region of interest
  - Extracting information about the content of the image
    - Examples: License plate detection/reading

Image Processing



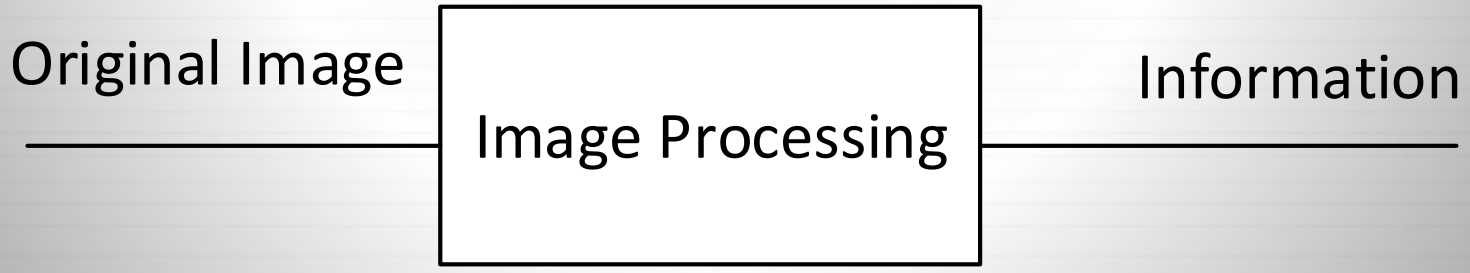


Image Processing Levels

- Images can be processed in three different levels

1. Low level: In this level a pixel and its neighbors are considered for processing

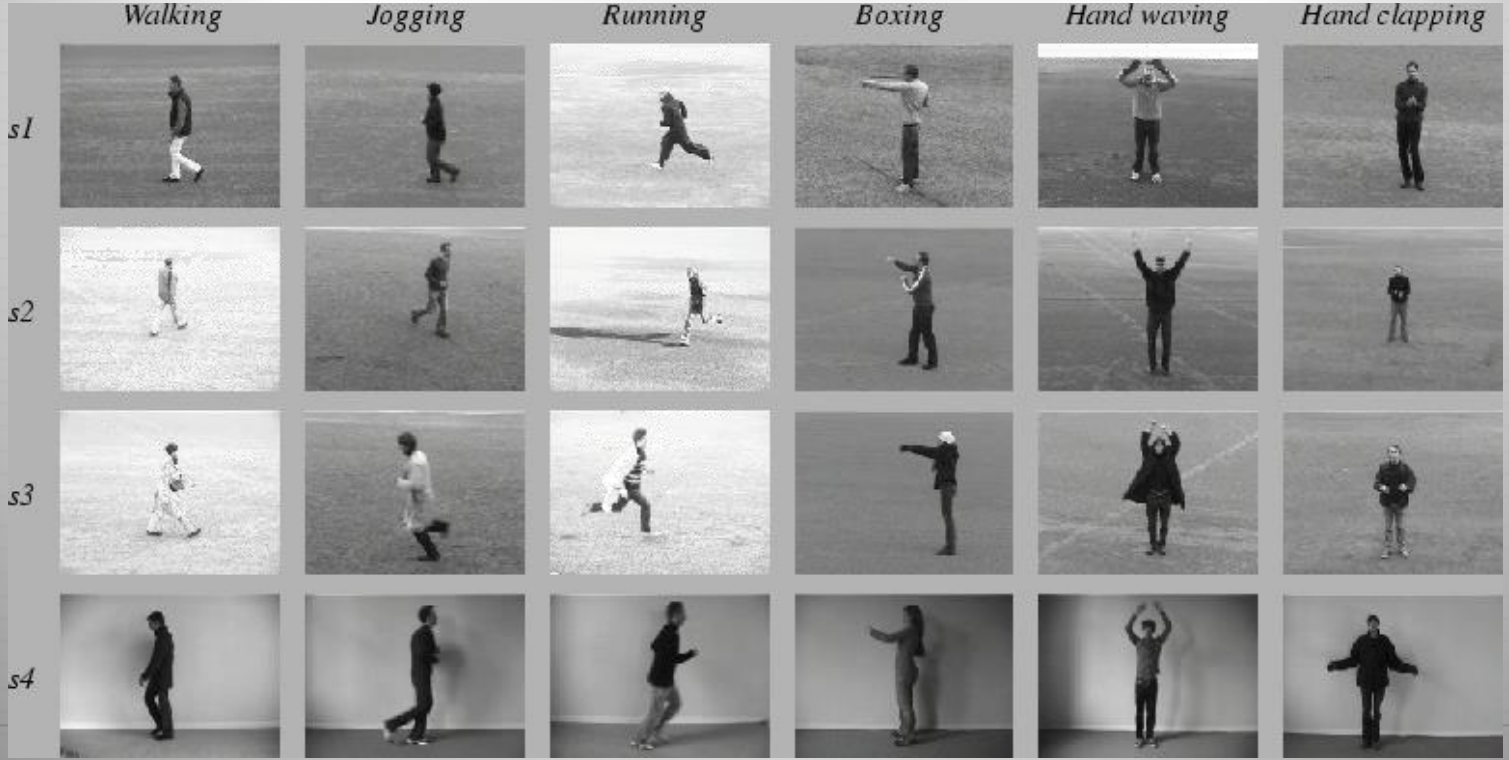
   Examples: Noise removal, Edge detection, Smoothing

Image Processing Levels

- Mid-level: In this level the results of low-level image processing is used to detect objects.
- Examples: Segmentation, Face detection

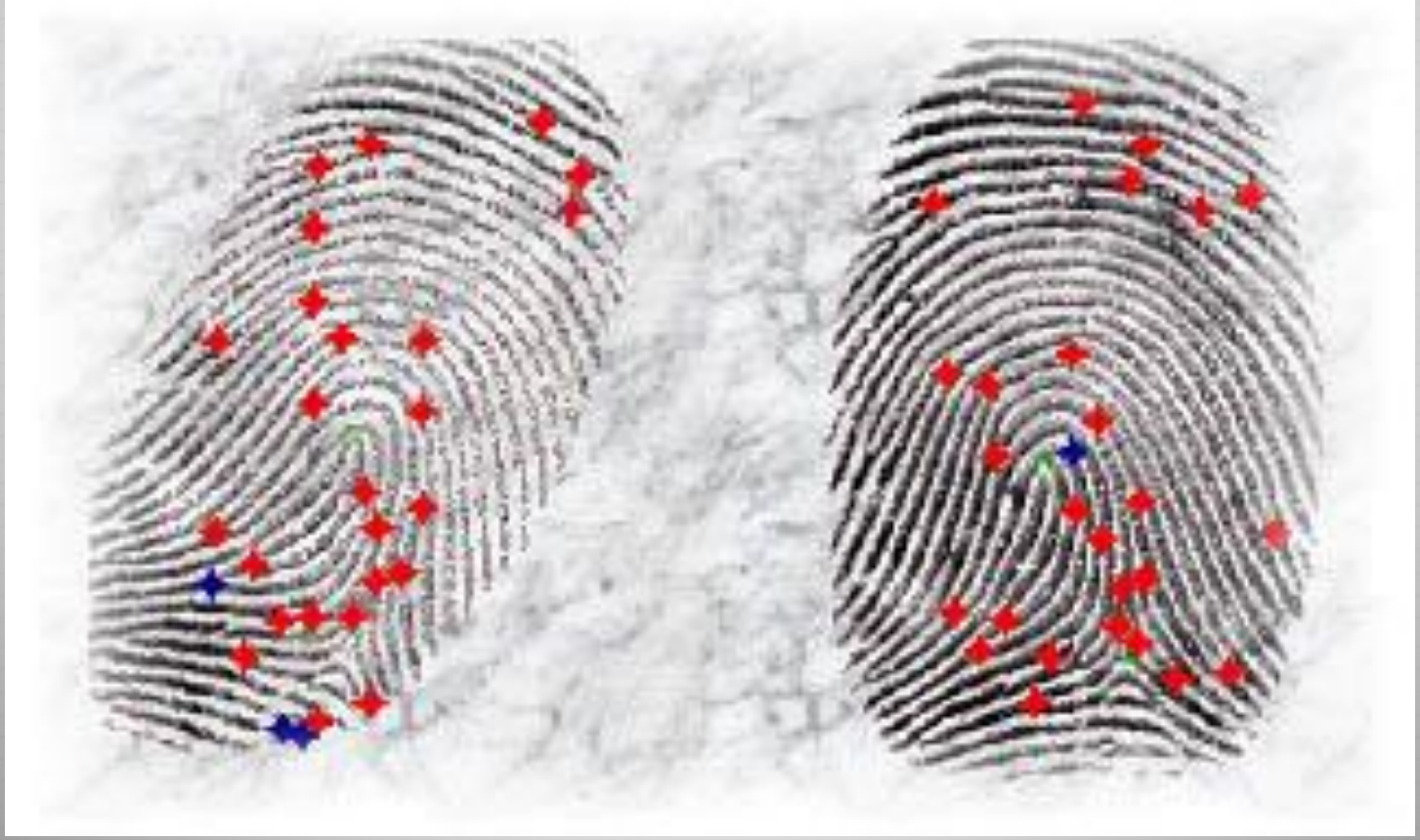
Image Processing Levels

- High-level: In this level the results of mid-level processing is used to identify objects, and describe scenes.
- Examples: Face identification, Human activity description



Applications

- Fingerprint Detection/Identification

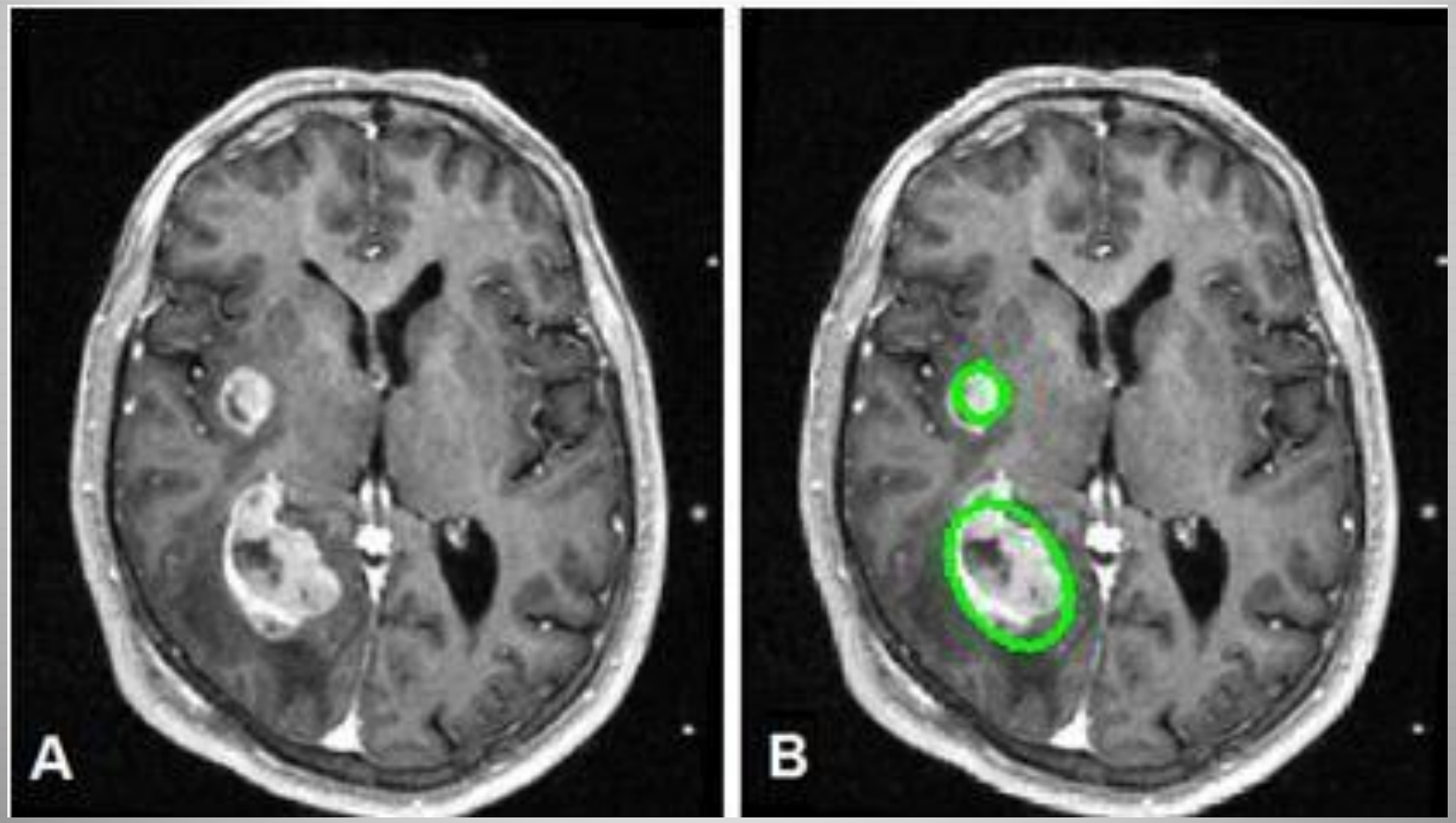


Applications

- License Plate Reading

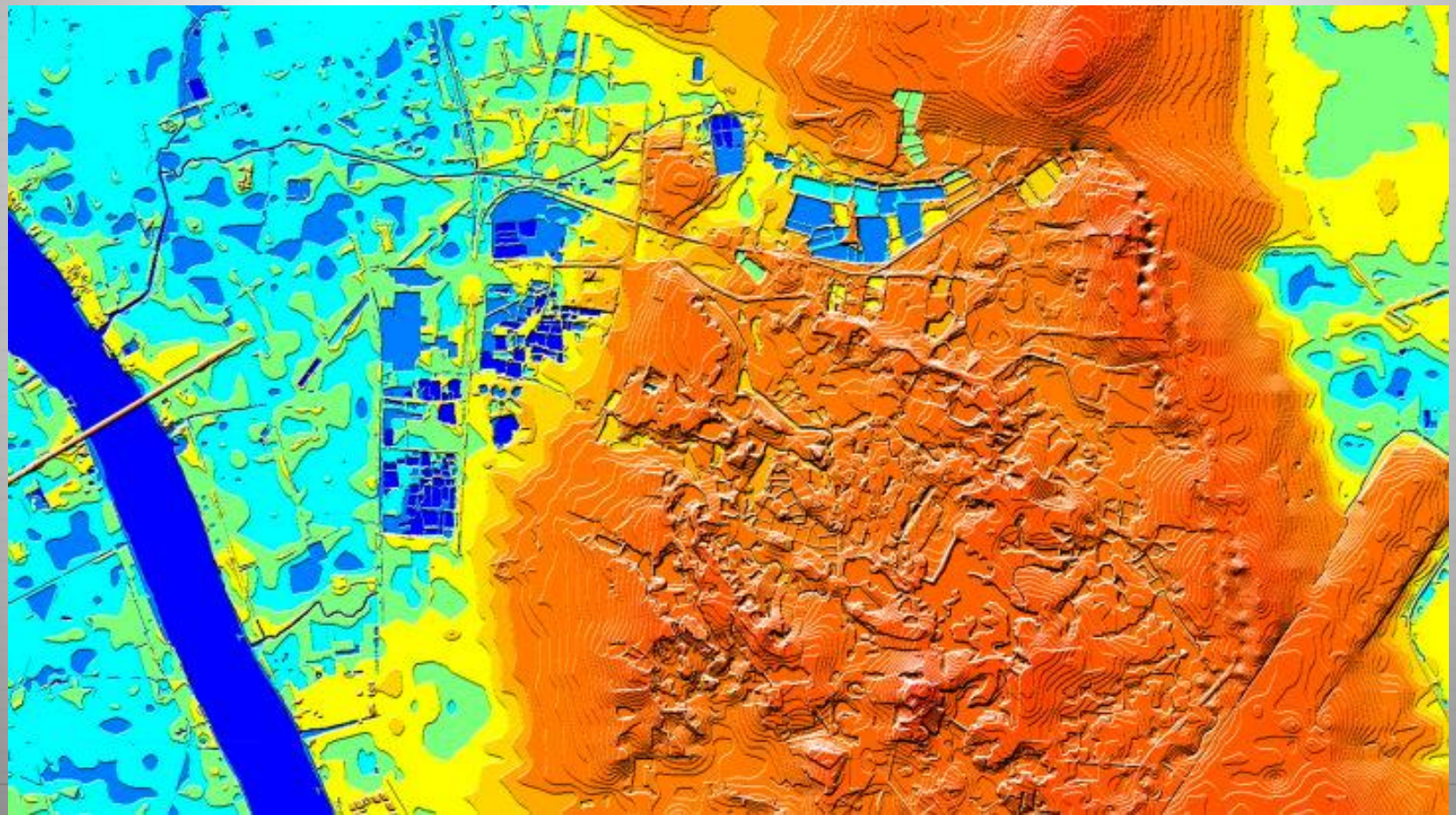
Applications

- Processing Medical Images



Applications

- Remote Sensing (Satellite Image Processing)

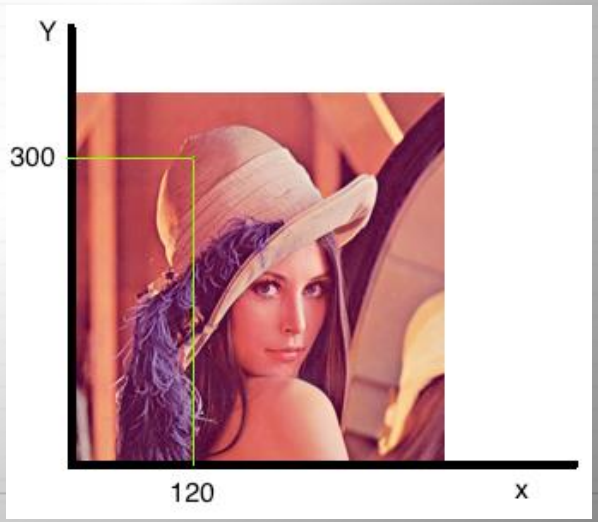


Spatial Operations on Images

- Spatial operations are operations that are performed on a pixel and its neighbors.
- For example,
  - by comparing a pixel with its neighbors we can decide if it is on the boundary of an object or not.
  - Thresholding is another important spatial operation.

Manipulating Pixels

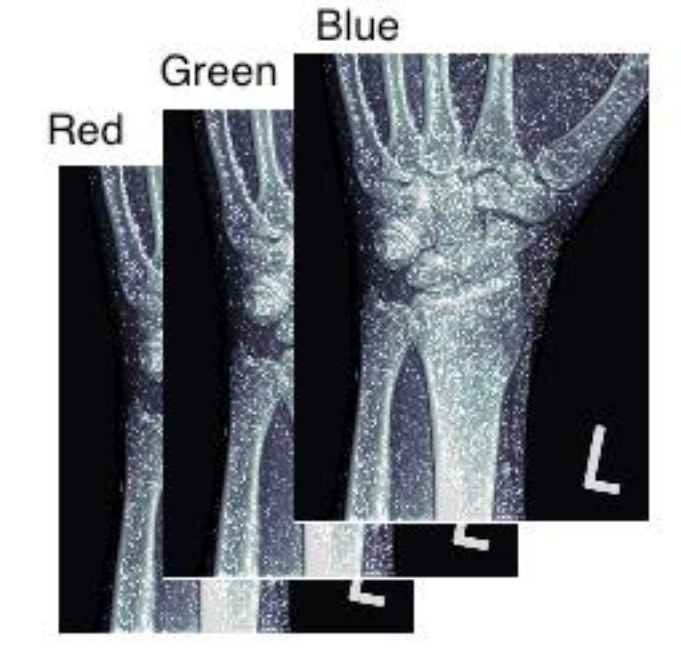
- A color digital image is 3-dimensional numeric array.
- The first index (x axis) and the second (y axis) are coordinate of the pixel, and the third index is the color (Red, Green, Blue)
- Therefore, each element can be modified using its row and column indexes.
  - Example: Image[120,300,1]=35



Example

Pseudo-coloring images can make important parts better visible.

Repeat intensity value to create an image with three channels (Red, Green, Blue channels)



Example (cont.)

- If the intensity of a pixel in gray image is between x1 and x2, change its color to Red in color image.
  - if Image1[i, j] > x1 and Image1[i, j] < x2 :
    - Image2[i,j,1] = 255
    - Image2[i,j,2] = 0
    - Image2[i,j,3] = 0

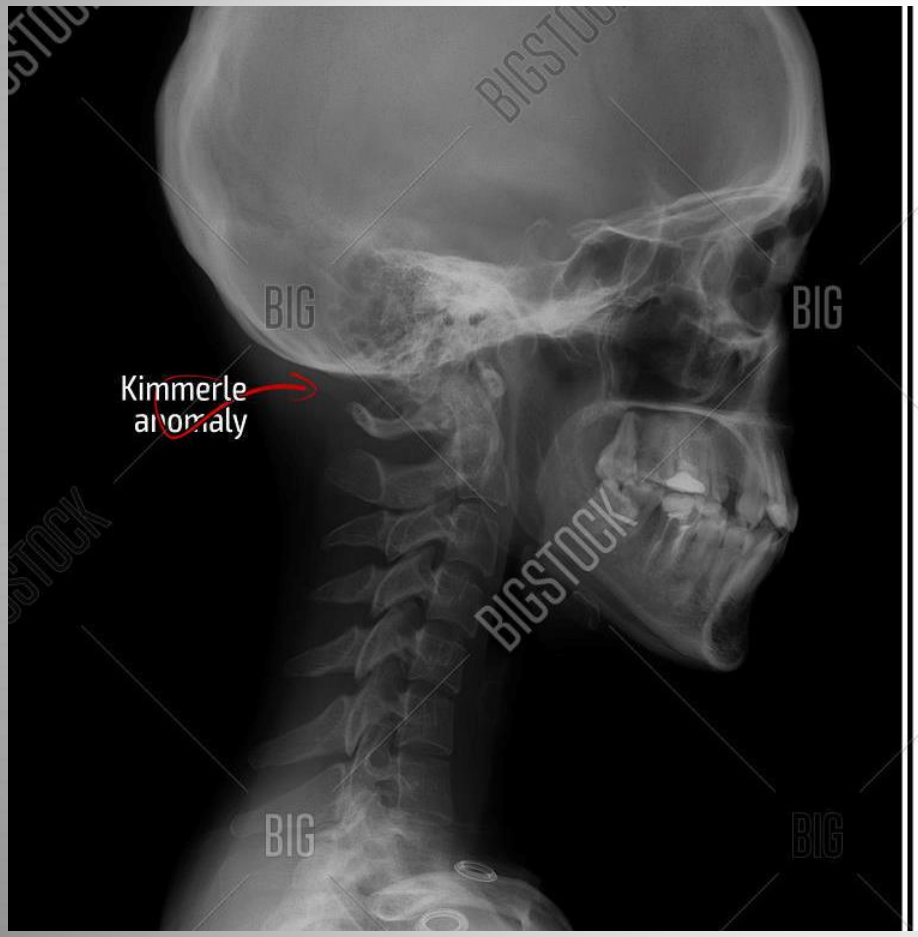
    else :

Image Negation

- In gray images, a white object in a black/gray background is better visible than a black object in a white background.

- This fact motivated radiologists to use negative images instead of the original ones

Image Negation



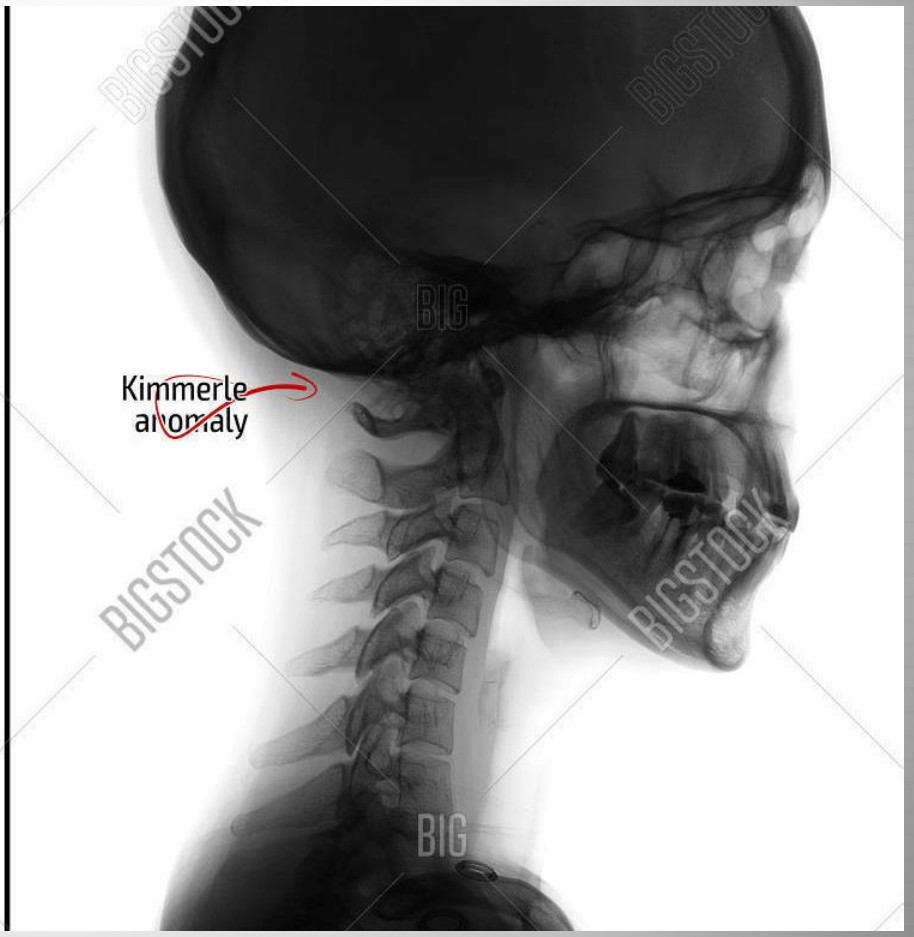


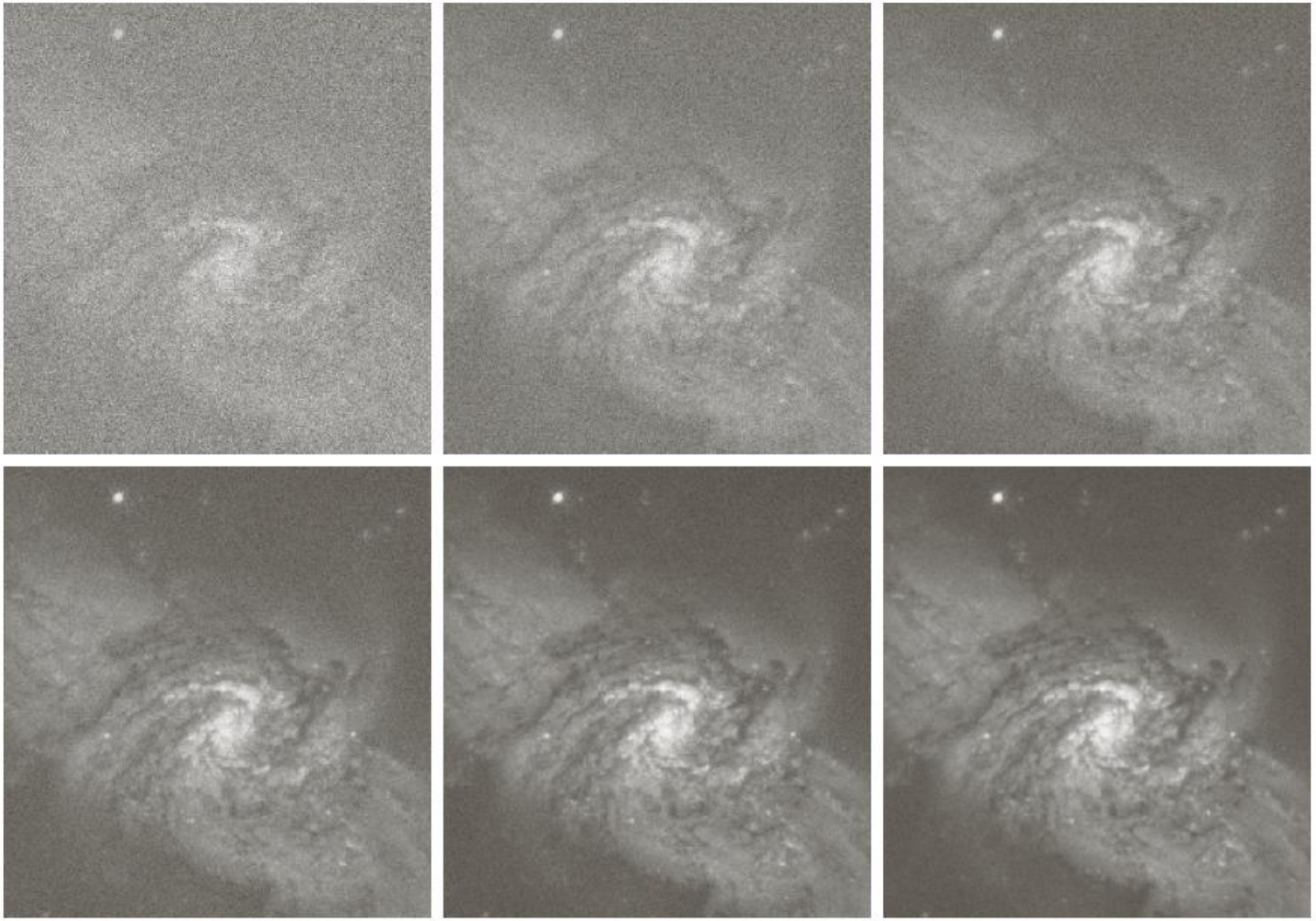
Image Negation

- In gray images with 256 quantization levels, 0 is considered as black, and 255 as white.
- Other values correspond to gray shades.
- Therefore, assuming x is the pixel value, 255-x is its negation.
  - NegImage = 255 – Image

Image Averaging

- Assume we have ***n*** noisy images of the same scene.
- As noise is randomly affects pixels, each image has noise at different pixels.
- Therefore, average of these images will reduce the noise affect.

$$NoiseReducedImage = \frac{1}{n}\sum_{i} NoisyImage_i$$



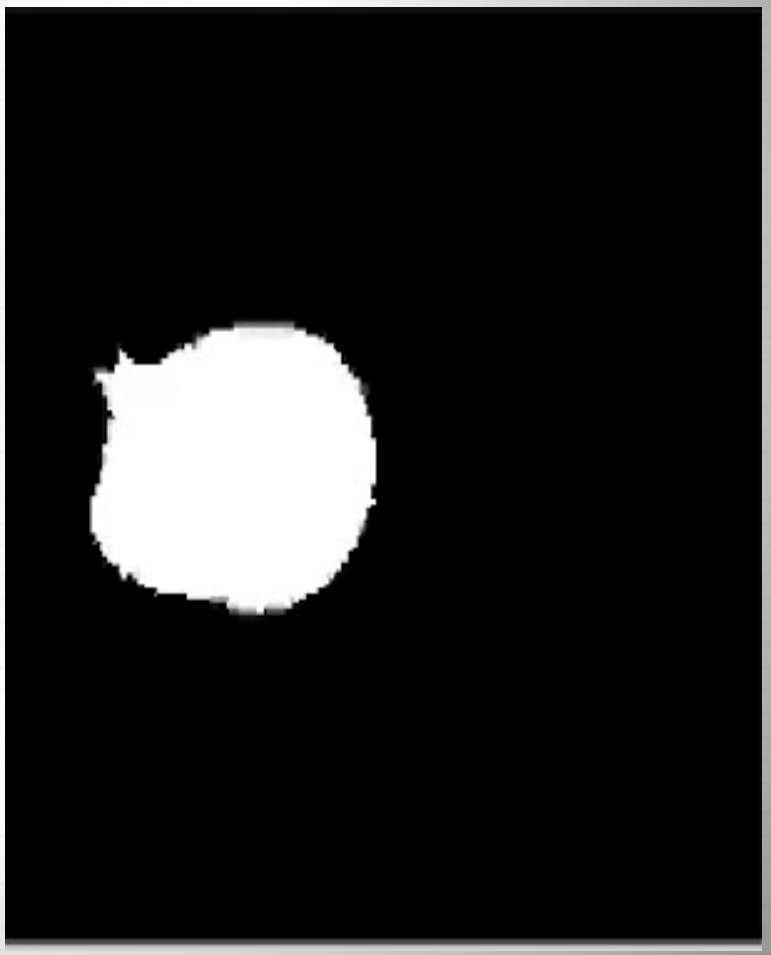
a b c
d e f

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

Thresholding

- Assume we want to replace all pixels brighter than T with white, and all other pixels with black.
- This operation is called thresholding, and T is called threshold value.
- if Image[x,y] > T :
  - Timage[x,y] = 255
- else:
  - Timage[x,y] = 0

Thresholding



Smoothing

- Smoothing is used to eliminate noise
- The (weighted) average of the neighbor pixel values is used in place of a pixel to remove noise.
- Smoothing is also called mean filtering

Smoothing

Original

Mean filter

Edge Detection

- Edge detection is used to find the boundaries of objects
- If we assume the area of an object has a uniform intensity, the edge pixels are pixels that are different than their neighbors.
- Algorithm:
  - Compare each pixel with its neighbors
  - If the difference is larger than a threshold then
    - It is an edge pixel

Edge Detection

Original Image

Edge Image

Image Filtering

- Most of the spatial operations on images can be defined using filters.
- A filter is a matrix with weights at its different positions.
- The weight values are multiplied by the pixel values, and are added up to define the new value of the pixel under the center of the filter

Sample Image Filter

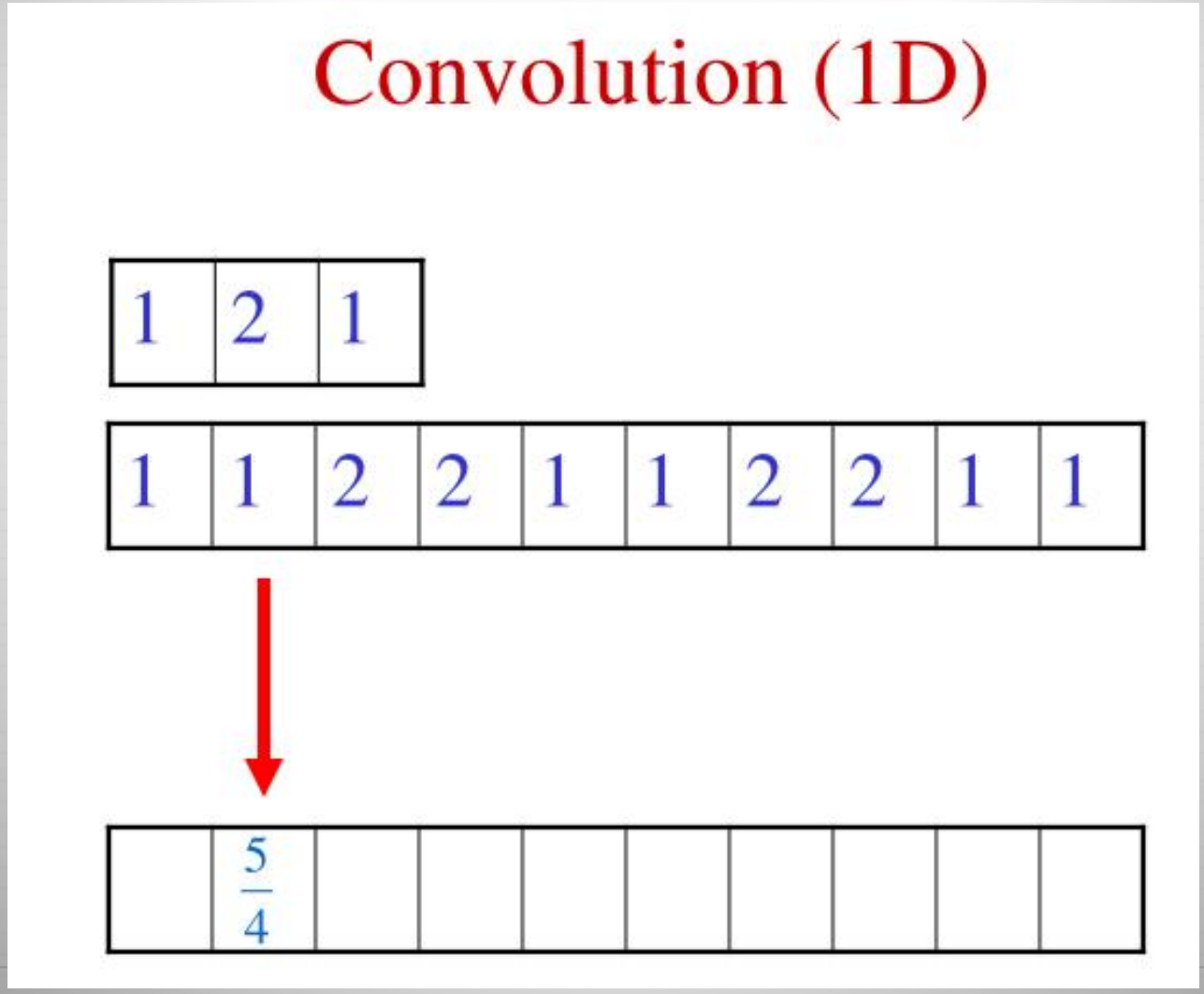
A filter with its weights

$$\begin{bmatrix} 2 & 1 & 3 \\ 4 & 1 & 5 \\ 3 & 2 & 2 \end{bmatrix}$$

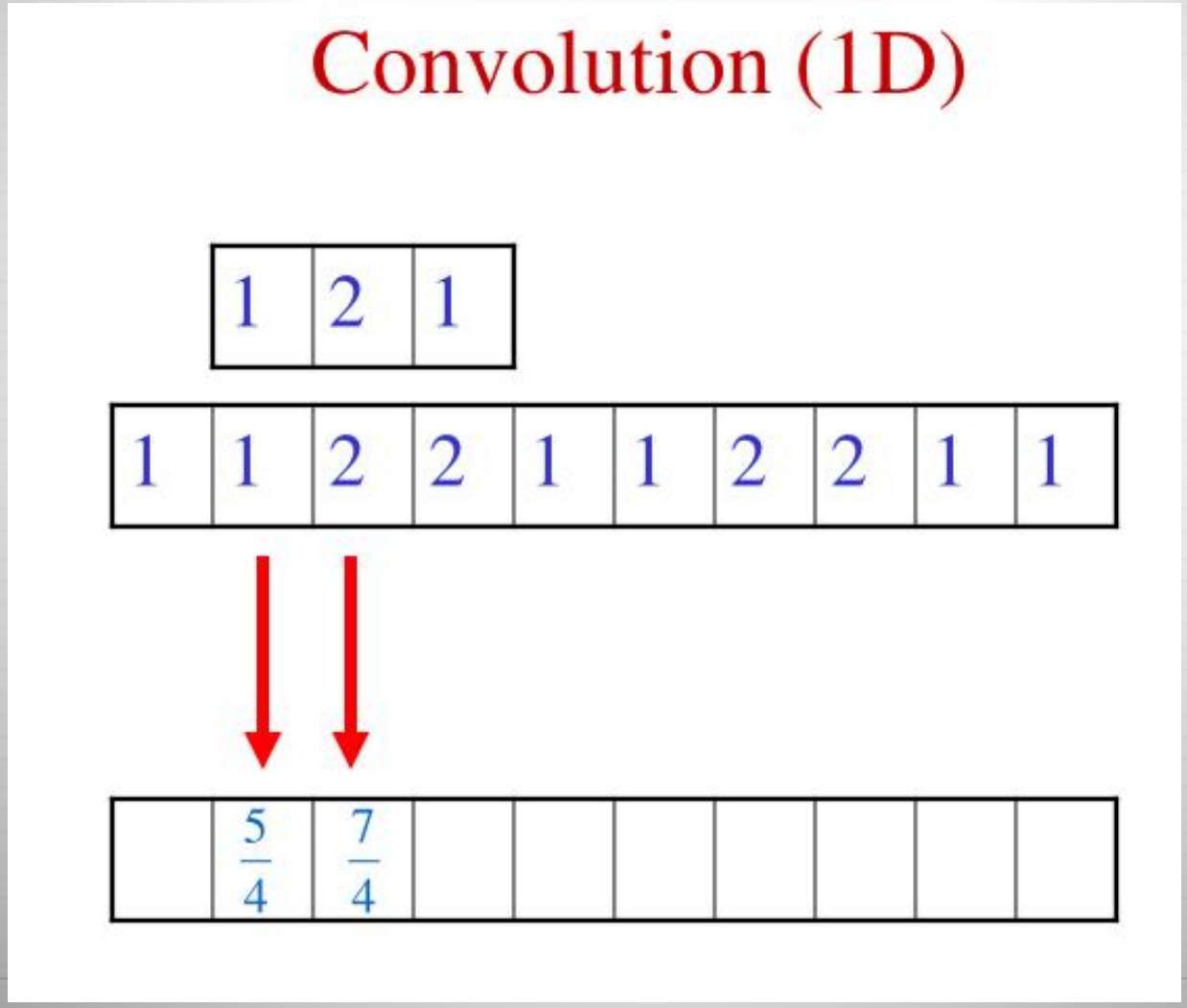
Convolution

- Filtering of an image is carried out by an operation called *convolution*.
- Convolution is a neighborhood operation, in which, each output pixel is the weighted sum of neighboring input pixels.
- The matrix of weights is called the *convolution kernel*, also known as the *filter*.

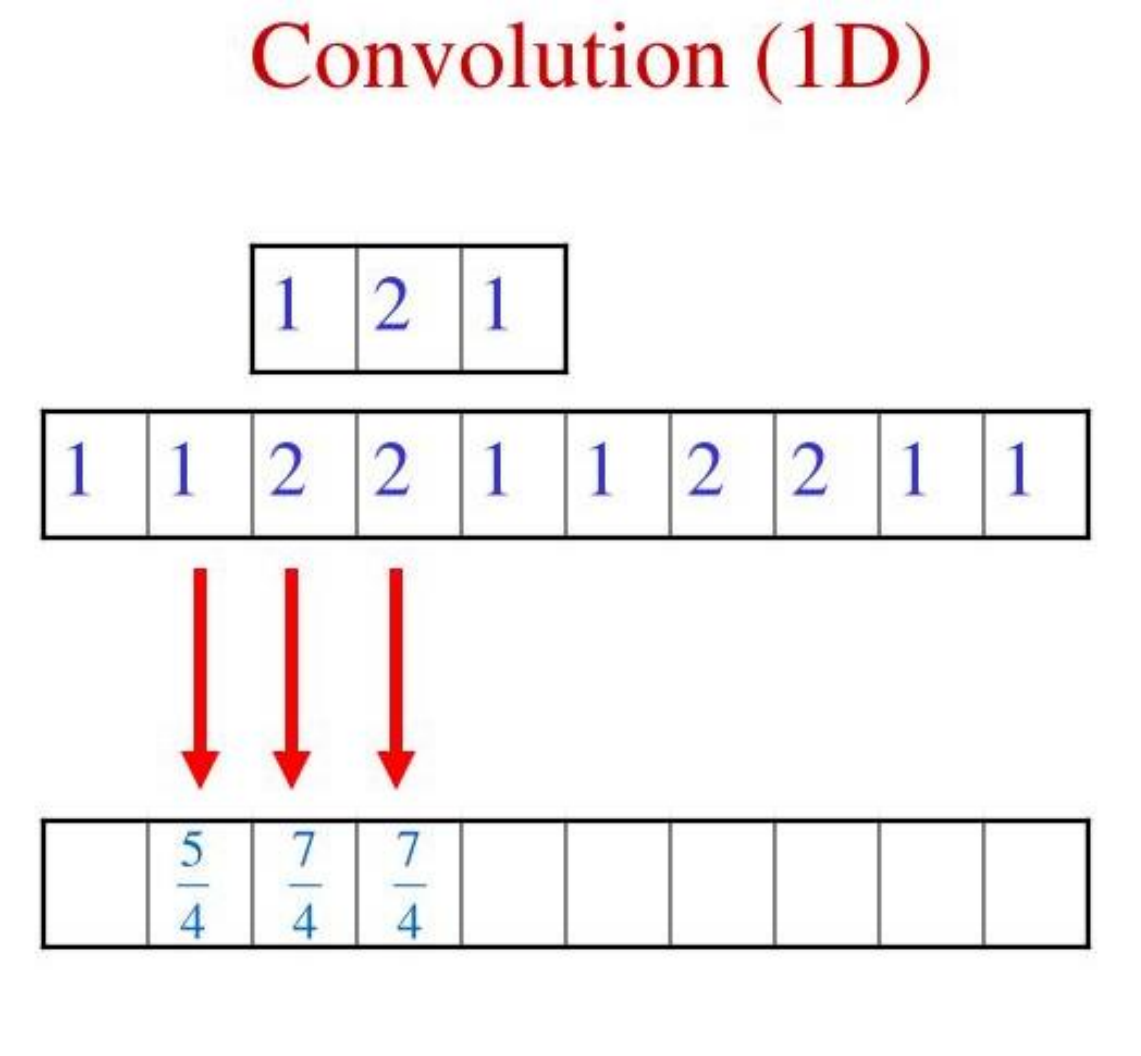
Convolution



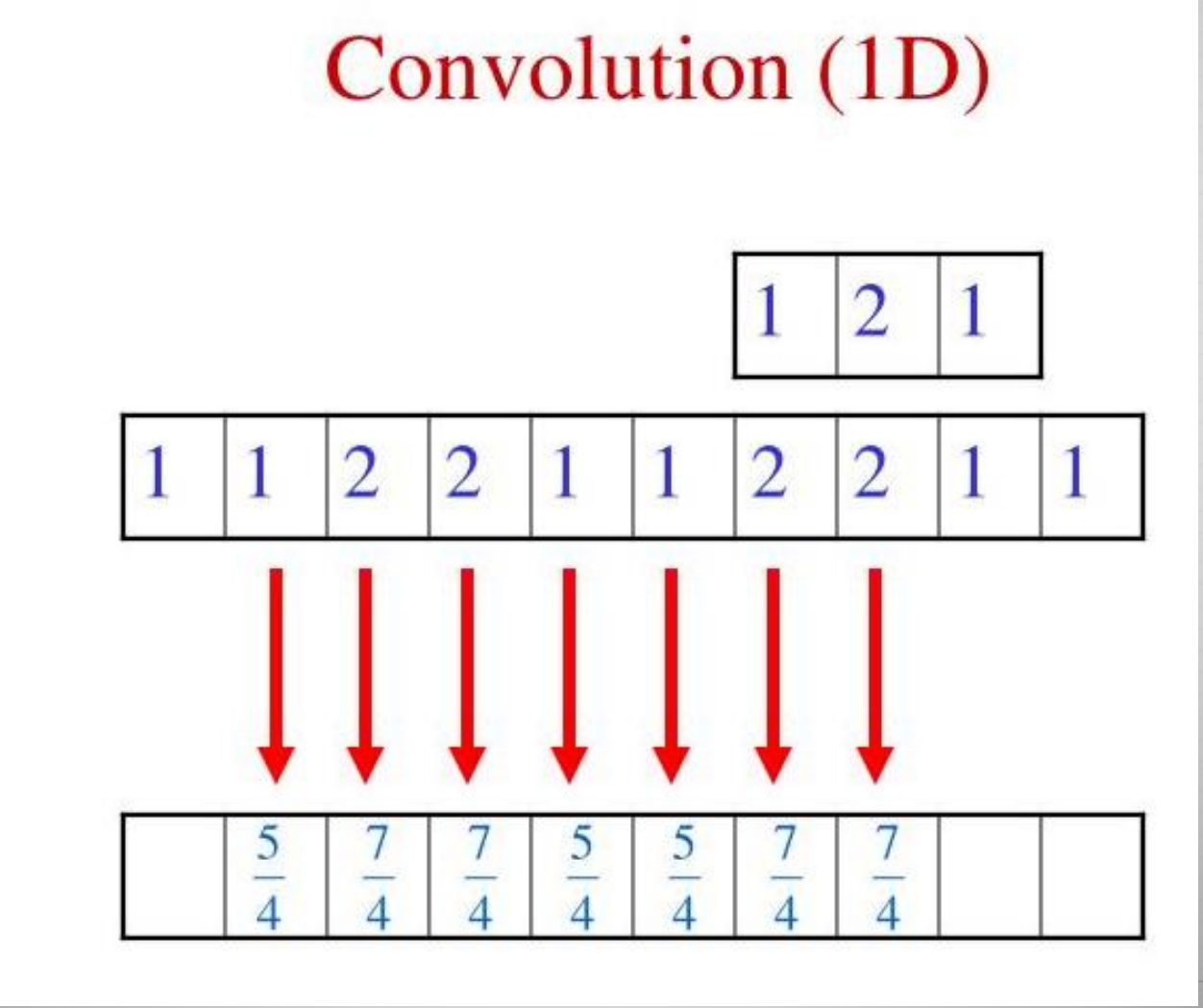
Convolution



Convolution



Convolution

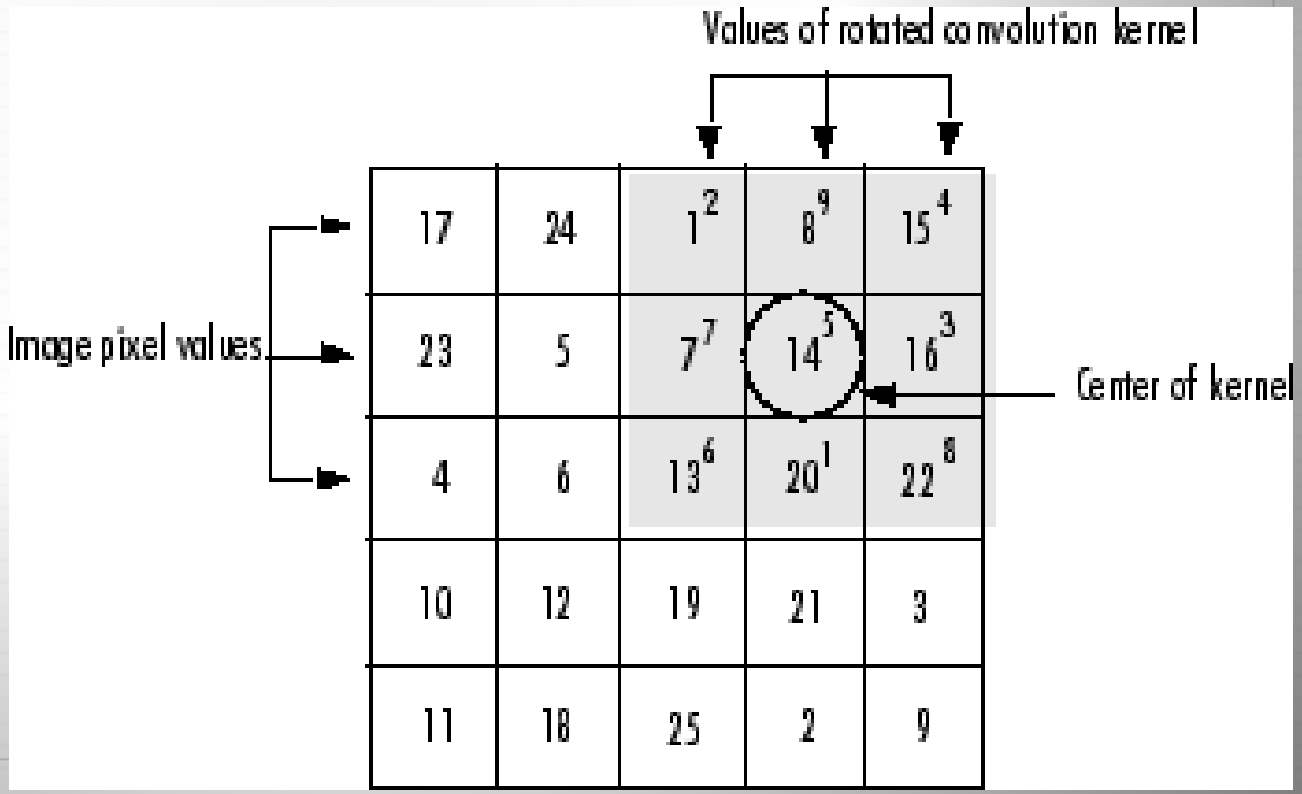


For example, suppose the image is

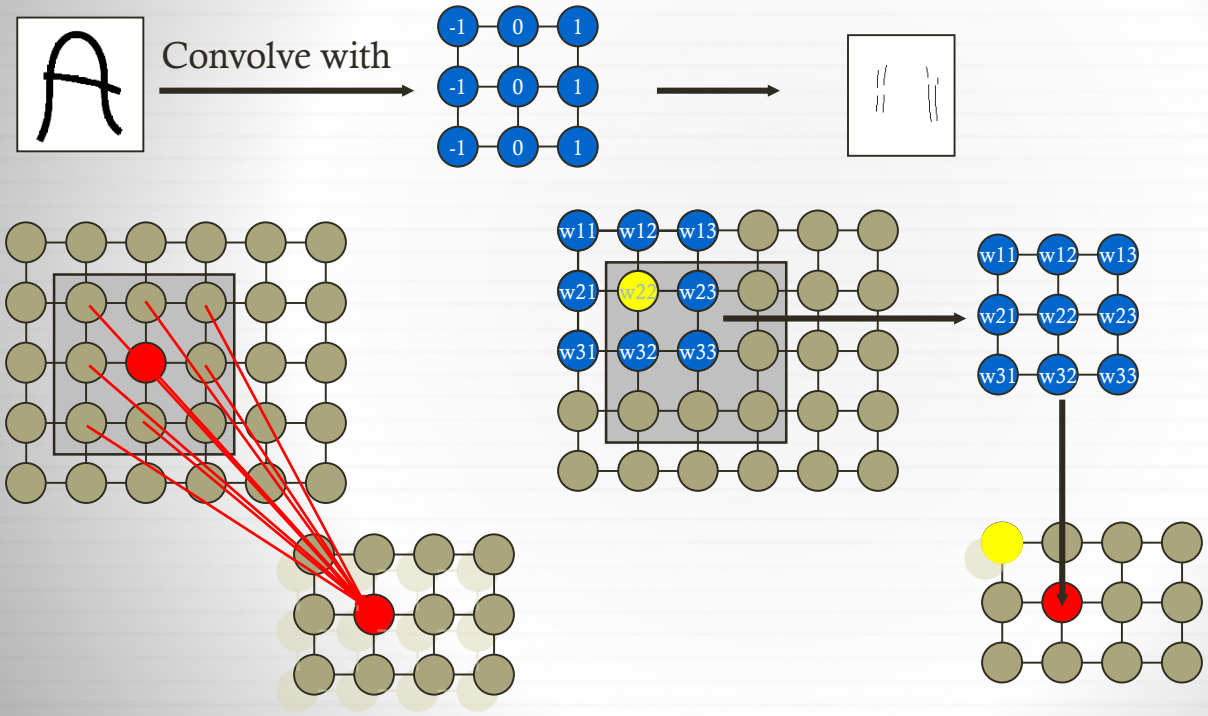
$$A = \begin{bmatrix} 17 & 24 & 1 & 8 & 15 \\ 23 & 5 & 7 & 14 & 16 \\ 4 & 6 & 13 & 20 & 22 \\ 10 & 12 & 19 & 21 & 3 \\ 11 & 18 & 25 & 2 & 9 \end{bmatrix}$$

and the filter is

$$h = \begin{bmatrix} 2 & 9 & 4 \\ 7 & 5 & 3 \\ 6 & 1 & 8 \end{bmatrix}$$

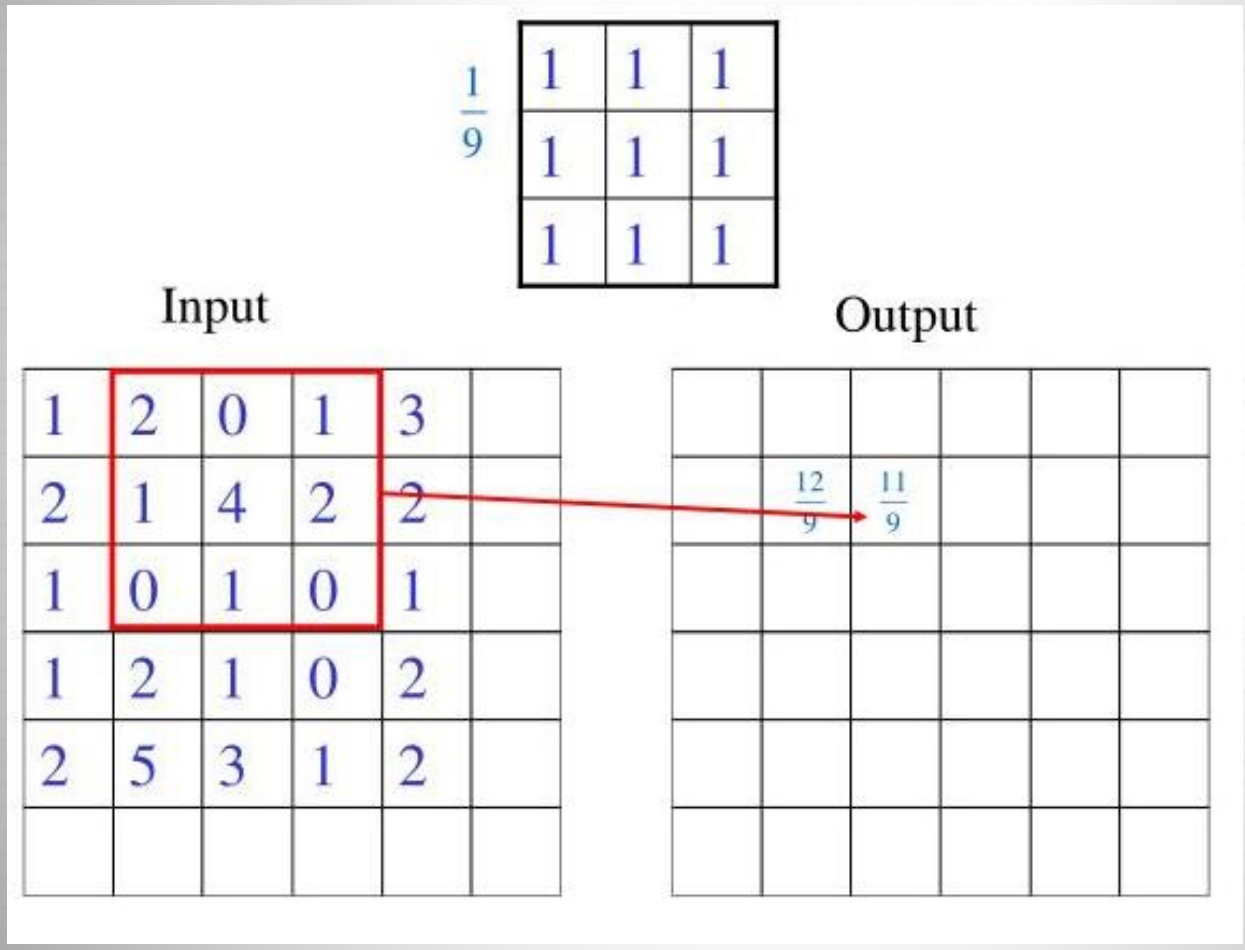


Convolution in 2-D Images



Smoothing Filter

$$\frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$


1/9
1 1 1
1 1 1
1 1 1
Input
1 2 0 1 3
2 1 4 2 2
1 0 1 0 1
1 2 1 0 2
2 5 3 1 2
Output
12/9
11/9

Gaussian Smoothing Filter

- Gaussian smoothing filter assigns different weights to the neighbors of the pixel at the center.

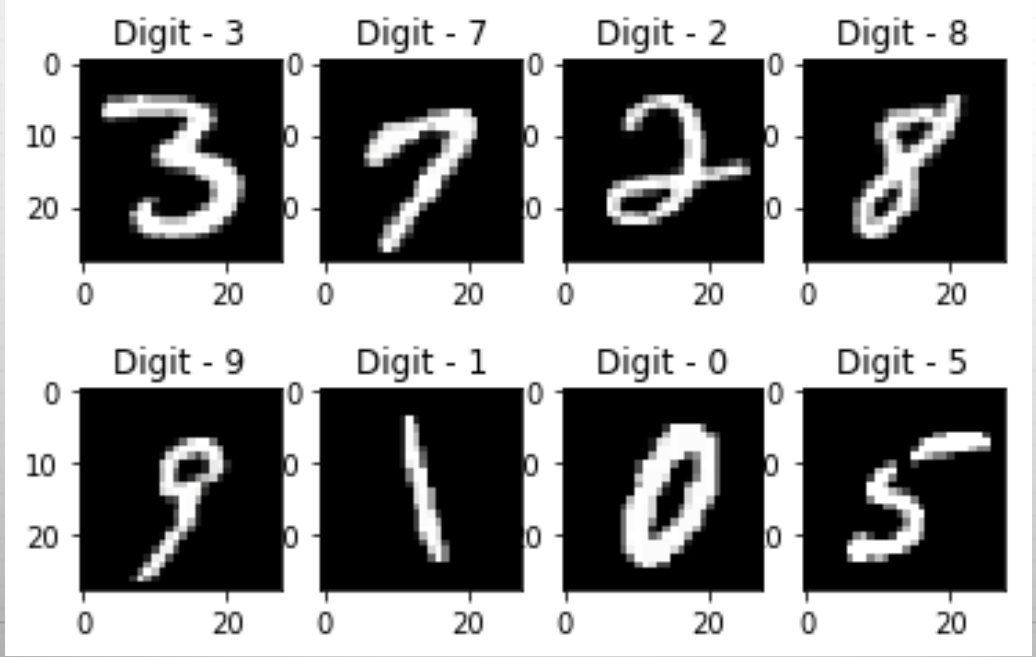
$$\frac{1}{16}\begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$$

Summary

- Digital Images are 2-dimensional matrices created by measuring the intensity of the reflected light.
- Image processing is: using algorithms to change the image matrix to improve its quality and/or extract information
- Many image processing algorithms can be defined using filters
- Applying filters to images is done by an operation called convolution

Assignment

- MNIST dataset includes handwritten digits in gray images of 28x28 (single channel images).
- The dataset has 70000 samples (60000 training and 10000 testing samples).



Assignment

- The aim of the assignment is developing a neural network in Python using keras and tensorflow to identify the digits.
- Try training with different epochs (20,30,50)
- Use L1 and L2 regularizations and discuss their impact
- In a subset of randomly selected 200 images, apply 3 pixels shift. Can the network identify them correctly (do not train your model again!)