1 Introduction to Computer Vision

1.1 What is Computer Vision?

Computer Vision is a field of artificial intelligence that enables computers to interpret and make decisions based on visual data, such as images and videos. It seeks to automate tasks that the human visual system can perform.

1.2 How is Computer Vision used Today?

Computer Vision is applied in various industries, such as:

- **Healthcare**: Medical image analysis and diagnostics.
- **Automotive**: Autonomous vehicles and driver-assist systems.
- **Retail**: Inventory management and customer analytics.
- **Security**: Facial recognition and surveillance.
- **Agriculture**: Crop monitoring and precision farming.

1.3 Why is the .jpeg Format Popular?

The JPEG format is popular because:

- It uses lossy compression to significantly reduce file size while maintaining acceptable image quality.
- It is widely supported across devices and platforms.
- It is efficient for web use, enabling faster image loading.

1.4 Example of Lossless Compression Technique

PNG (Portable Network Graphics) is an example of lossless compression. It preserves the exact image data, ensuring no loss in quality.

1.5 Challenges of Computer Vision

- Understanding diverse environments and contexts.
- Handling occlusions and distortions in images.
- Computational resource requirements.
- Interpreting subjective aspects, such as emotion or intent in images.

2 Image Processing: Introduction and Fundamentals

2.1 Basic Relationship Between Pixels

- **Neighbors**: Pixels directly adjacent to a given pixel.
- **Adjacency**: Two pixels are adjacent if they share a common edge or vertex.

Comfort

Fun

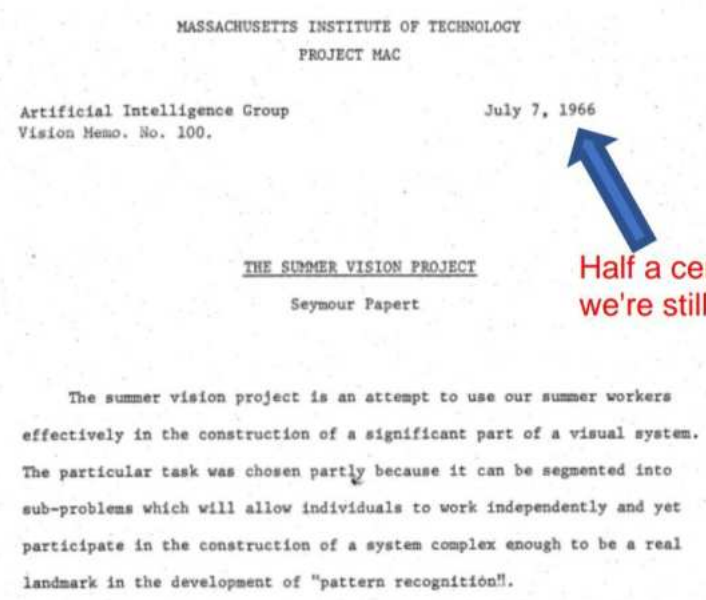
Access

History of Computer Vision

Marvin Minsky, MIT
Turing award, 1969

"In 1966, Minsky hired a first-year undergraduate student and assigned him a problem to solve over the summer:

connect a camera to a computer and get the machine to describe what it sees."



MASSACHUSETTS INSTITUTE OF TECHNOLOGY
PROJECT MAC

Artificial Intelligence Group
Vision Memo. No. 100.

July 7, 1966

THE SUMMER VISION PROJECT

Seymour Papert

The summer vision project is an attempt to use our summer workers effectively in the construction of a significant part of a visual system. The particular task was chosen partly because it can be segmented into sub-problems which will allow individuals to work independently and yet participate in the construction of a system complex enough to be a real landmark in the development of "pattern recognition".

Half a century late
we're still working

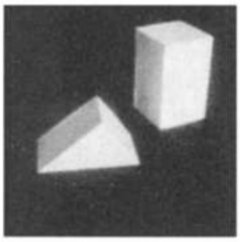
History of Computer Vision

Marvin Minsky, MIT
Turing award, 1969

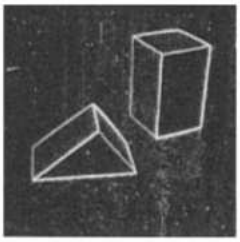
Gerald Sussman, MIT
AI Researcher since 1964

1960's: interpretation of synthetic worlds

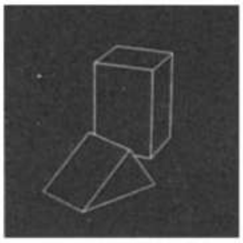
Larry Roberts
"Father of Computer Vision"



Input image



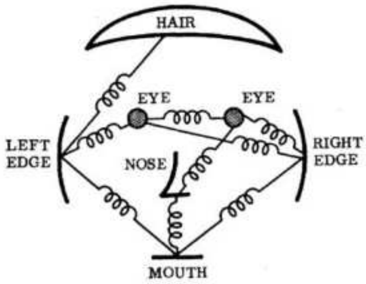
2x2 gradient operator



computed 3D model
rendered from new viewpoint

Larry Roberts PhD Thesis, MIT, 1963,
Machine Perception of Three-Dimensional Solids

1970's: some progress on interpreting selected images



The representation and matching of pictorial structures
Fischler and Elschlager, 1973

The .jpg format is popular because it offers a great balance between image quality and file size due to its "lossy compression" method, making it ideal for sharing photos online where fast loading times are crucial, as it significantly reduces file size without sacrificing too much visual detail; additionally, almost all devices and software support .jpg files, making it universally compatible.

Key points about .jpg format:

Small file size:

Uses lossy compression to significantly reduce file size, allowing for faster loading on websites and easier sharing online.

Wide compatibility:

Supported by nearly all image viewing software and devices, making it the standard for sharing photos.

Suitable for photos:

Well-suited for capturing the richness of colors in photographs with minimal quality loss when compressed.

Developed by Joint Photographic Experts Group:

The format originated from a committee focused on image compression standards.

JPEG - Wikipedia

The Joint Photographic Experts Group created the standard in...

What is a
Understa

28 Mar 2024

In the context of lossless compression, "PNG" stands for "Portable Network Graphics," which is a file format that utilizes lossless compression techniques, meaning that when a PNG image is compressed, no data is lost and the original image can be fully reconstructed upon decompression; making it ideal for preserving image quality in applications like logos, graphics, and detailed images where pixel precision is crucial.

Key points about PNG and lossless compression:

No data loss:

Unlike lossy compression formats like JPEG, PNG ensures that no image information is discarded during compression, resulting in perfect image fidelity.

Wide color support:

PNG can handle a wide range of colors, including transparency, making it suitable for various graphic design needs.

Compression algorithm:

PNG uses a compression algorithm called "Deflate" which effectively reduces file size while maintaining image quality.

Lossless Compression: A Complete Guide -...

Is PNG lossless? Yes. PNG - which

PNG - Wi

PNG * Portab
(PNG, officia

1.4 Example of Lossless Compression Technique

PNG (Portable Network Graphics) is an example of lossless compression. It preserves the exact image data, ensuring no loss in quality.

1.5 Challenges of Computer Vision

- Understanding diverse environments and contexts.
- Handling occlusions and distortions in images.
- Computational resource requirements.
- Interpreting subjective aspects, such as emotion or intent in images.

2 Image Processing: Introduction and Fundamentals

2.1 Basic Relationship Between Pixels

- **Neighbors**: Pixels directly adjacent to a given pixel.
- **Adjacency**: Two pixels are adjacent if they share a common edge or vertex.

2.2 Path, Foreground, and Background

- **Path**: A sequence of adjacent pixels.
- **Foreground**: The primary objects of interest in an image.
- **Background**: The surrounding area of the foreground.
- **Boundary**: The edge separating the foreground from the background.

2.3 Distance Measures

- **Euclidean Distance**: Straight-line distance between two points.
- **City Block Distance**: Distance measured along grid lines (Manhattan distance).
- **Chessboard Distance**: Maximum of horizontal and vertical distances.

3 Digital Image: Definition and Representation

A digital image is a numerical representation of a 2D visual pattern. It consists of pixels, each represented by a numeric value corresponding to intensity or color.

4 Electromagnetic Spectrum

Images are formed by capturing specific ranges of the electromagnetic spectrum, including visible light, infrared, and ultraviolet.

challenges of com...

Computer vision has several challenges, including:

- **Data quality**: The quality of the data used to train computer vision systems is important to ensure accurate results. Data can be inaccurate, missing, or unbalanced, which can lead to bias.
- **Data leakage**: Including inappropriate information in the training data can compromise the model's performance.
- **Lighting conditions**: Changes in lighting can make it difficult for the system to recognize objects. A model trained on high-resolution images might not recognize objects in low-light conditions.
- **Occlusion**: Objects partially blocking the view can impact a model's performance.
- **Labeling data**: Labeling data can be time-consuming.
- **Scaling**: Computer vision systems need to be able to scale.
- **Privacy and ethics**: Computer vision raises privacy and ethical issues.
- **Cost**: Computer vision can be expensive.
- **Experienced professionals**: There may be a lack of experienced professionals in the field.
- **Regular monitoring**: Computer vision systems may need to be monitored regularly.

5 Common Problems with Computer Vision...

Top Comp
Opportu

Introduction

Digital Image

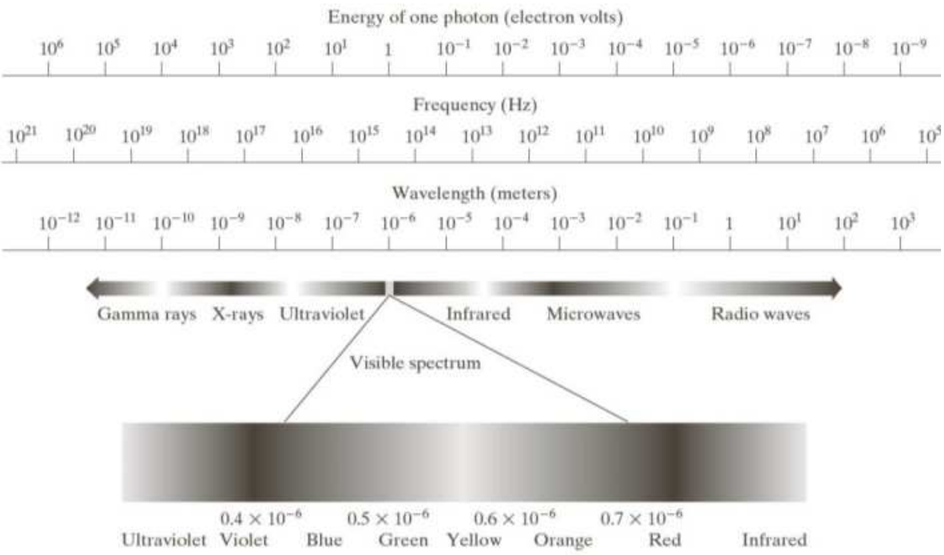
is a two-dimensional function

$$f(x, y)$$

where x and y are spatial coordinates, and

amplitude of f is called intensity or gray level at the point (x, y)

Light and EM Spectrum



Light and EM Spectrum

The colors that humans perceive in an object are determined by the nature of the light reflected from the object.

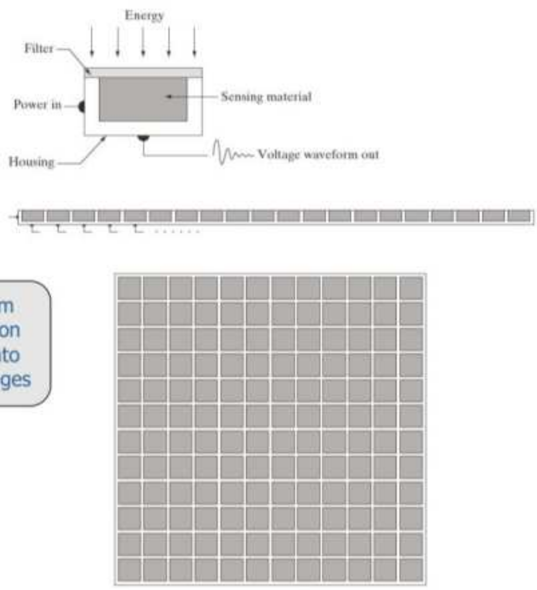
e.g. green objects reflect light with wavelengths primarily in the 500 to 570 nm range while absorbing most of the energy at other wavelength

..Light and EM Spectrum

- Monochromatic light: void of color
  Intensity is the only attribute, from black to white
  Monochromatic images are referred to as **gray-scale** images

- Chromatic light bands: 0.43 to 0.79 μm
  The quality of a chromatic light source:
  Radiance: total amount of energy
  Luminance (lm): the amount of energy an observer perceives from a light source
  Brightness: a subjective descriptor of light perception that is impossible to measure. It embodies the achromatic notion of intensity and one of the key factors in describing color sensation.

Image Acquisition



a
b
c

FIGURE 2.12
(a) Single imaging sensor.
(b) Line sensor.
(c) Array sensor.

Transform illumination energy into digital images

Image Acquisition Using a Single Sensor

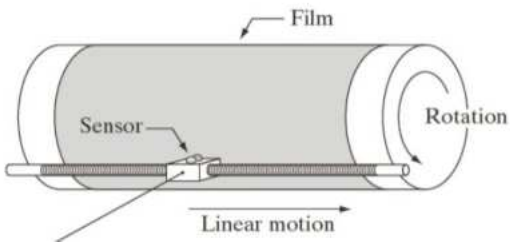


FIGURE 2.13 Combining a single sensor with motion to generate a 2-D image.

3 Digital Image: Definition and Representation

A digital image is a numerical representation of a 2D visual pattern. It consists of pixels, each represented by a numeric value corresponding to intensity or color.

4 Electromagnetic Spectrum

Images are formed by capturing specific ranges of the electromagnetic spectrum, including visible light, infrared, and ultraviolet.

5 Image Acquisition Techniques

5.1 Single Sensor

Uses a single photosensitive element, often with a rotating mirror or a moving sensor.

5.2 Sensor Strips

Linear arrays of sensors used in scanners.

5.3 2D Array of Sensors

Used in cameras to capture images in one shot.

6 Sample Image Formation Model

The formation of an image involves:

- **Illumination**: Light falling on an object.
- **Reflectance**: Light reflected by the object.

The image intensity is a function of both illumination and reflectance.

7 Image Sampling and Quantization

Sampling converts a continuous signal into discrete values by selecting intervals, while **Quantization** maps these values to finite levels.

8 Smoothing Spatial Filters

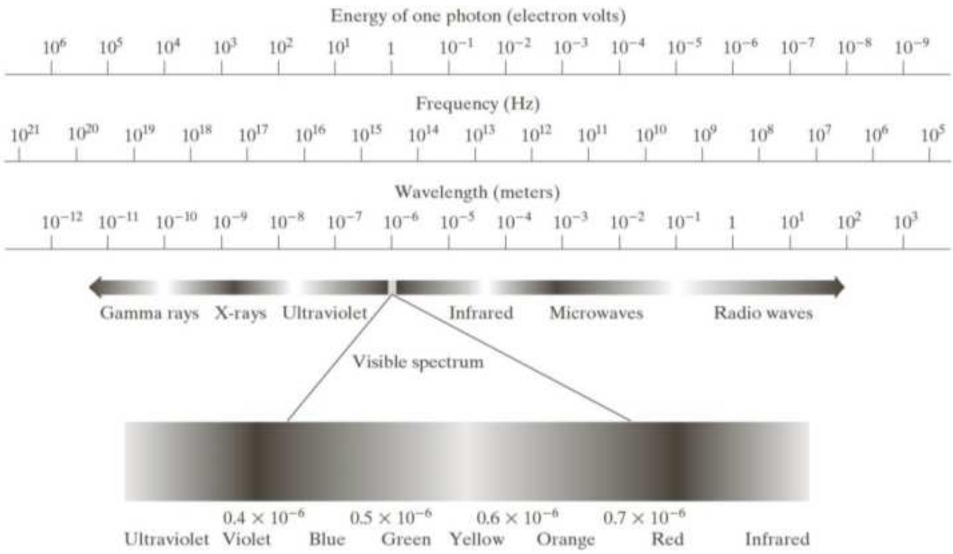
- **Linear Filters**: Use weighted averages for smoothing.
- **Average Filter Masks**: Assign equal weights to neighboring pixels.

9 Order Statistic (Non-linear) Filters

Filters such as the **Median Filter** reorder pixel values to remove noise without blurring edges.

10 Representation of Objects

Objects are represented in terms of boundaries, regions, and key points within



Light and EM Spectrum

The colors that humans perceive in an object are determined by the nature of the light reflected from the object.

e.g. green objects reflect light with wavelengths primarily in the 500 to 570 nm range while absorbing most of the energy at other wavelength

..Light and EM Spectrum

- Monochromatic light: void of color
  Intensity is the only attribute, from black to white
  Monochromatic images are referred to as **gray-scale** imag
- Chromatic light bands: 0.43 to 0.79 μm

Image Acquisition Using a Single Sensor

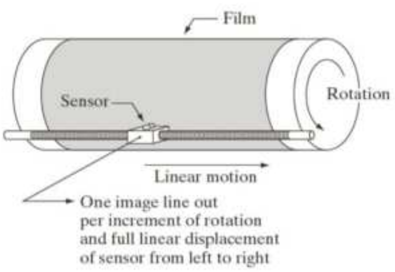
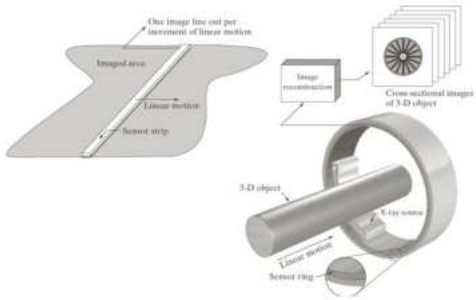


FIGURE 2.13 Combining a single sensor with motion to generate a 2-D image.

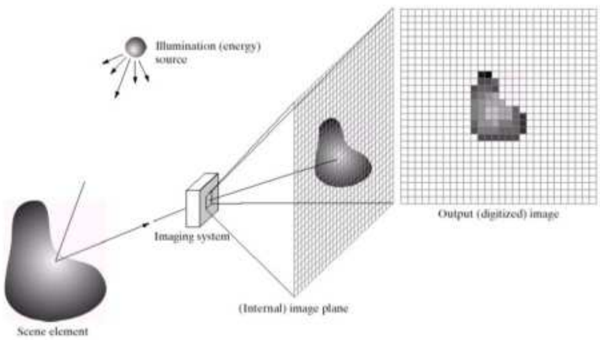
Image Acquisition Using Sensor Strips



a b

FIGURE 2.14 (a) Image acquisition using a linear sensor strip. (b) Image acquisition using a circular sensor strip.

Image Acquisition Process

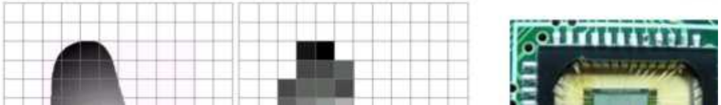


a
b c d e

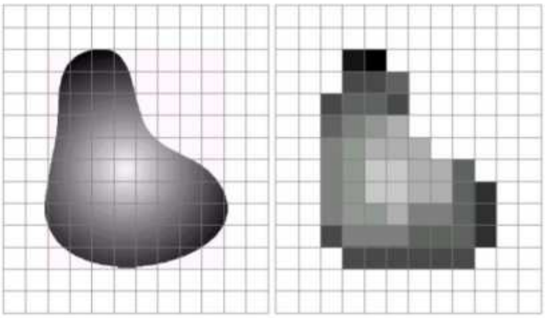
FIGURE 2.15 An example of the digital image acquisition process. (a) Energy ("illumination") source. (b) An element of a scene. (c) Imaging system. (d) Projection of the scene onto the image plane. (e) Digitized image.

Sensor Array

10–11/30

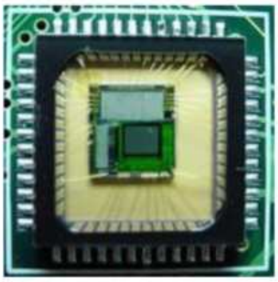


Sensor Array



a b

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



CMOS sensor

A Simple Image Formation Model

$$f(x,y) = i(x,y) * r(x,y)$$

$f(x,y)$: intensity at the point (x,y)

$i(x,y)$: illumination at the point (x,y)

(the amount of source illumination incident on the scene)

$r(x,y)$: reflectance/transmissivity at the point (x,y)

(the amount of illumination reflected/transmitted by the object)

where $0 < i(x,y) < \infty$ and $0 < r(x,y) < 1$

Some Typical Ranges of illumination

- Illumination

  Lumen — A unit of light flow or luminous flux

  Lumen per square meter (lm/m^2) — The metric unit of measure for illuminance of a surface

  - On a clear day, the sun may produce in excess of 90,000 lm/m^2 of illumination on the surface of the Earth
  - On a cloudy day, the sun may produce less than 10,000 lm/m^2 of illumination on the surface of the Earth
  - On a clear evening, the moon yields about 0.1 lm/m^2 of illumination
  - The typical illumination level in a commercial office is about 1000 lm/m^2

(c) Array sensor.

Figure(3):

1) Image Acquisition using Single Sensor:- Fig(a) shows the Components of image (Single) sensor. Eg:- Photodiode. The use of filter improves selectivity. If a green filter on light sensor favours light in green band.

Cont...

In order to generate a 2-D image using Single sensor, there has to be relative displacement in both x-axis & 11
y-axis between the sensor & area to be imaged.

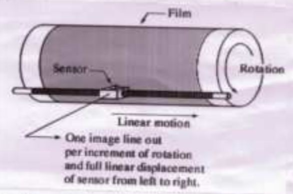
To perform this the arrangement is done as shown in below figure.

The film negative (image to be scanned) is mounted on a drum whose rotation makes one dimensional displacement.

The single sensor is mounted on a lead screw which provides displacement in another dimension.

Such arrangement provides image capturing line by line. This is a slow and inexpensive method of obtaining High-resolution Images.

Also we can replace the drum with flat bed to perform "to and fro" motion.



2) Image Acquisition using Sensor Strip:- Here it consists of an in-line sensor arrangement of in the form of a sensor strip as shown in Fig 1(b).

The strip provides imaging in one dimension and its motion (6)

in perpendicular direction provides imaging in other 12
dimension.

Figure below shows the arrangement of sensor strip which is mostly used in flat bed scanner.

Sensing devices with 4000 or more in-line sensors are possibly used.

It gives one line of image at a time and its "In" motion makes 2-D scanning.

Sensor strips mounted in a ring configuration are frequently used in medical applications which gives cross-sectional images of 3-D objects as shown below.

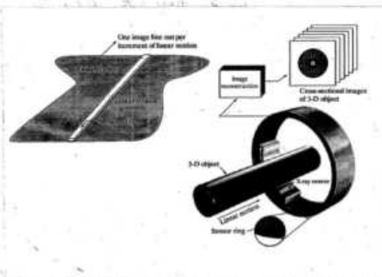


Fig:- Image Acquisition using line sensor.

Cont...

3) Image Acquisition using Sensor Arrays:- Fig 1(c) shows individual sensors are arranged in 2-D array. 13

This is also the predominant arrangement found in digital cameras, electromagnetic & ultrasonic sensing devices.

A typical sensor for these array arrangement is CCD array that contains a packaged rugged arrays of 4000×4000 elements or more. (CCD - Charged Coupled Devices).

The principal manner in which array sensors are used is shown below.

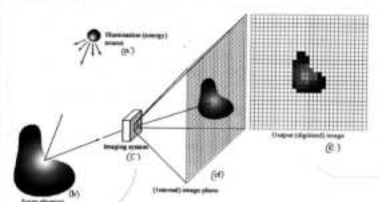


Fig : Image Acquisition using Sensor Array.

→ The fig. shows the energy from illuminating source is reflected by scene element.

→ The Imaging System collects the incoming energy from scene element and focus it on to an image plane.

→ The sensor array (which is coincident with the focal plane) produces outputs proportional to the integral of light received at each sensor.

→ The digitizer ckt then converts these outputs into digital form results in a digital image (shown in 'e'). (7)

* Image Sampling and Quantization:- The o/p of most sensors is a continuous voltage waveform whose amplitude and spatial behavior are related to the physical phenomenon being sensed. 14

To create a digital image, we need to convert the continuous sensed data into digital form.

This involves two processes: 1) Sampling
2) Quantization.

→ Digitizing the coordinate values is called Sampling.

→ Digitizing the amplitude (intensity) values is called quantization.

Basic Relationships Between Pixels

- **Neighbors** of a pixel p at coordinates (x,y)

➢ **4-neighbors of p**, denoted by $\mathbf{N_4(p)}$:
(x-1, y), (x+1, y), (x,y-1), and (x, y+1).

➢ **4 diagonal neighbors of p**, denoted by $\mathbf{N_D(p)}$:
(x-1, y-1), (x+1, y+1), (x+1,y-1), and (x-1, y+1).

➢ **8 neighbors of p**, denoted $\mathbf{N_8(p)}$
$N_8(p) = N_4(p) \cup N_D(p)$

Basic Relationships Between Pixels

- **Adjacency**
  Let V be the set of intensity values

➢ **4-adjacency**: Two pixels p and q with values from V are 4-adjacent if q is in the set $N_4(p)$.

➢ **8-adjacency**: Two pixels p and q with values from V are 8-adjacent if q is in the set $N_8(p)$.

Basic Relationships Between Pixels

- **Adjacency**
  Let V be the set of intensity values

➢ **m-adjacency**: Two pixels p and q with values from V are m-adjacent if

(i) q is in the set $N_4(p)$, or

(ii) q is in the set $N_D(p)$ and the set $N_4(p) \cap N_4(q)$ has no pixels whose values are from V.

• Clearly, the quality of a digital image is determined to a large degree by no. of Samples and discrete gray levels used in Sampling & quantization.

---

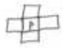
* Some basic Relationships between Pixels:-

Neighbors of a Pixel:-

• A pixel 'p' at coordinates (x,y) has four horizontal and vertical neighbors whose coordinates are given by $(x+1,y)$, $(x-1,y)$ $(x,y+1)$, $(x,y-1)$. This set of pixels called the 4-neighbors of 'p' and is denoted by $N_4(P)$.

• Each pixel is a unit distance from (x,y).

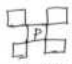
• Some Neighbors of p lie outside the digital image if (x,y) is on the border of the image.



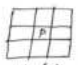
$N_4(P)$ Set:

• The four diagonal neighbors of 'p' have coordinates $(x+1,y+1)$ $(x+1,y-1)$, $(x-1,y+1)$, $(x-1,y-1)$ and are denoted by $N_D(P)$. 17

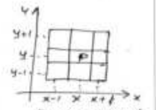
• These points, together with the 4-neighbors, are called the 8-neighbors of 'P', and is denoted by $N_8(P)$.



$N_D(P)$



$N_8(P)$.



Pixel Coordinates.

• Some of the points in $N_D(P)$ & $N_8(P)$ fall outside the image if (x,y) is on the border of image.

Adjacency, Connectivity, Regions and Boundaries:

• Connectivity b/w pixels is a fundamental concept the simplifies the definitions of such as regions and boundaries

• Two neighbor pixels satisfy a specified criterion of similarity in their gray levels (equal) then they are said to be "Connected."

• Let 'V' be the set of gray-level values used to define adjacency.

• In a binary image, $V=\{1\}$ if we are referring to adjacency of pixels with value '1'.

• Same idea in the grayscale image, but here 'V' contains more elements in b/w 0 - 255.

• three types of adjacency:

a) 4-adjacency: 2 pixels 'P' and 'q' with values from V are 4-adjacent if 'q' is in the set $N_4(P)$.

(b) 8-adjacency: 2-pixels 'P' and 'q' with values from V are 8-adjacent if 'q' is in the set $N_8(P)$.

(c) m-adjacency (mixed): Two pixels 'P' & 'q' with values from 'V' are m-adjacent if, 18

i) 'q' is in $N_4(P)$, or

ii) 'q' is in $N_D(P)$ & the set $N_4(P) \cap N_4(q)$ has no pixel whose values are from 'V'.

• mixed adjacency is a modification of 8-adjacency which is introduced to eliminate ambiguities that often arise with 8-adjacency.

• Consider the following pixel arrangement to show the ambiguity of 8-adjacency and its elimination is m-adjacency.

Binary Image Pixel arrangement	8-adjacency	m-adjacency.
0 1 1	0 1 1	0 1 1
0 1 0	0 1 0	0 1 0
0 0 1	0 0 1	0 0 1

• Let 'R' be a subset of pixels in an image

• If 'R' is a Connected set the 'R' is said

• If one or more neighbors in the set 'R' are not connected, then it is said to be "boundary" of the Region 'R'.

Distance Measures:- For pixels p, q & z, with coordinates (x,y), (s,t), (v,w), respectively, 'D' is a distance function or metric if

(a) $D(P,q) \geq 0$ $(D(P,q)=0$ iff $P=q)$,

(b) $D(P,q) = D(q,P)$ &

(c) $D(P,z) \leq D(P,q) + D(q,z)$.

Cont...

• The "Euclidean Distance" between P & q,

$$D_e(P,q) = \left[(x-s)^2 + (y-t)^2\right]^{1/2}$$ 19

• D_4 distance (city-block distance) between P & q,

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

Basic Relationships Between Pixels

- **Path**
- A (digital) path (or curve) from pixel p with coordinates (x_0, y_0) to pixel q with coordinates (x_n, y_n) is a sequence of distinct pixels with coordinates

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

  where (x_i, y_i) and (x_{i-1}, y_{i-1}) are adjacent for $1 \le i \le n$.

- Here *n* is the *length* of the path.
- If $(x_0, y_0) = (x_n, y_n)$, the path is ***closed*** path.
- We can define 4-, 8-, and m-paths based on the type of adjacency used.

Examples: Adjacency and Path

V = {1, 2}

$$\begin{matrix} 0 & 1 & 1 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{matrix} \qquad \begin{matrix} 0 & 1 & 1 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{matrix} \qquad \begin{matrix} 0 & 1 & 1 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{matrix}$$

Examples: Adjacency and Path

V = {1, 2}

$$\begin{matrix} 0 & 1 & 1 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{matrix} \qquad \begin{matrix} 0 & 1 & 1 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{matrix} \qquad \begin{matrix} 0 & 1 & 1 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{matrix}$$

8-adjacent

Examples: Adjacency and Path

V = {1, 2}

0	1	1
0	2	0
0	0	1

0	1	1
0	2	0
0	0	1

8-adjacent

0	1	1
0	2	0
0	0	1

m-adjacent

Examples: Adjacency and Path

V = {1, 2}

$$\begin{matrix} 0_{1,1} & 1_{1,2} & 1_{1,3} \\ 0_{2,1} & 2_{2,2} & 0_{2,3} \\ 0_{3,1} & 0_{3,2} & 1_{3,3} \end{matrix}$$

0	1	1
0	2	0
0	0	1

8-adjacent

0	1	1
0	2	0
0	0	1

m-adjacent

The 8-path from (1,3) to (3,3):
(i) (1,3), (1,2), (2,2), (3,3)
(ii) (1,3), (2,2), (3,3)

The m-path from (1,3) to (3,3):
(1,3), (1,2), (2,2), (3,3)

Basic Relationships Between Pixels

- **Connected in S**

  Let S represent a subset of pixels in an image. Two pixels p with coordinates (x_0, y_0) and q with coordinates (x_n, y_n) are said to be **connected in S** if there exists a path

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

  where $\forall i, 0 \le i \le n, (x_i, y_i) \in S$

Basic Relationships Between Pixels

Let S represent a subset of pixels in an image

- For every pixel *p* in S, the set of pixels in S that are connected to *p* is called a ***connected component*** of S.
- If S has only one connected component, then S is called ***Connected Set.***
- We call R a **region** of the image if R is a connected set
- Two regions, R_i and R_j are said to be ***adjacent*** if their union forms a connected set.
- Regions that are not to be adjacent are said to be ***disjoint.***

Basic Relationships Between Pixels

- **Boundary (or border)**

➢ The ***boundary*** of the region R is the set of pixels in the region that have one or more neighbors that are not in R.

➢ If R happens to be an entire image, then its boundary is defined as the set of pixels in the first and last rows and columns of the image.

- **Foreground and background**

➢ An image contains K disjoint regions, R_k, $k = 1, 2, \ldots, K$. Let R_u denote the union of all the K regions, and let $(R_u)^c$ denote its complement.

All the points in R_u is called **foreground**;

All the points in $(R_u)^c$ is called **background**.

Question 1

- **In the following arrangement of pixels, are the two regions (of 1s) adjacent? (if 8-adjacency is used)**

1	1	1
1	0	1
0	1	0
0	0	1
1	1	1
1	1	1

Region 1

Region 2

Question 2

- **In the following arrangement of pixels, the two regions (of 1s) are disjoint (if 4-adjacency is used)**

1	1	1
1	0	1
0	1	0
0	0	1
1	1	1
1	1	1

Region 1

Region 2

- **In the following arrangement of pixels, the two regions (of 1s) are disjoint (if 4-adjacency is used)**

1	1	1
1	0	1
0	1	0
0	0	1
1	1	1
1	1	1

foreground

background

Question 3

- **In the following arrangement of pixels, the circled point is part of the boundary of the 1-valued pixels if 8-adjacency is used, true or false?**

0	0	0	0	0
0	1	1	0	0
0	1	1	0	0
0	1	1	1	0
0	1	1	1	0
0	0	0	0	0

Question 4

• If R is a connected set the R is said to be Region.

• If one or more neighbors in the set 'R' are not connected, then it is said to be "boundary" of the Region 'R'.

Distance Measures:- For pixels P, q & z, with coordinates (x, y), (s, t), (v, w), respectively, 'D' is a distance function or metric if

(a) $D(P,q) \geq 0$ $\quad (D(P,q) = 0$ iff $P = q)$,
(b) $D(P,q) = D(q,P)$ &
(c) $D(P,z) \leq D(P,q) + D(q,z)$.

Cont...

• The "Euclidean Distance" between P & q,

$$D_e(P,q) = \left[(x-s)^2 + (y-t)^2\right]^{1/2}.$$

• D_4 distance (city-block distance) between P & q,

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

		2		
	2	1	2	
2	1	0	1	2
	2	1	2	
		2		

Fig: $D_4(P,q) \leq 2$

• D_8 distance (chess board Distance) b/w P & q,

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

2	2	2	2	2
2	1	1	1	2
2	1	0	1	2
2	1	1	1	2
2	2	2	2	2

Fig:- $D_8(P,q) \leq 2$

o

* **Introduction to the mathematical tools used in DIP:-**

I. Array verses Matrix Operations:-

• Consider two 2×2 images: $\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}$ & $\begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix}$.

• Array Product : $\begin{bmatrix} a_{11}b_{11} & a_{12}b_{12} \\ a_{21}b_{21} & a_{22}b_{22} \end{bmatrix}$.

• Matrix Product : $\begin{bmatrix} a_{11}b_{11} + a_{12}b_{21} & a_{11}b_{12} + a_{12}b_{22} \\ a_{21}b_{11} + a_{22}b_{21} & a_{21}b_{12} + a_{22}b_{22} \end{bmatrix}$.

• Array operation involving one or more images is carried out on a pixel-by-pixel basis.

II. Linear verses Non-Linear Operations:-

Distance Measures

The following are the different Distance measures:

a. Euclidean Distance :
$D_e(p, q) = [(x-s)^2 + (y-t)^2]^{1/2}$

b. City Block Distance:
$D_4(p, q) = |x-s| + |y-t|$

		2		
	2	1	2	
2	1	0	1	2
	2	1	2	
		2		

c. Chess Board Distance:
$D_8(p, q) = \max(|x-s|, |y-t|)$

2	2	2	2	2
2	1	1	1	2
2	1	0	1	2
2	1	1	1	2
2	2	2	2	2

Question 5

- **In the following arrangement of pixels, what's the value of the chessboard distance between the circled two points?**

0	0	0	0	0
0	0	1	1	0
0	1	1	0	0
0	1	0	0	0
0	0	0	0	0
0	0	0	0	0

Question 6

- **In the following arrangement of pixels, what's the value of the city-block distance between the circled two points?**

0	0	0	0	0
0	0	1	1	0
0	1	1	0	0
0	1	0	0	0
0	0	0	0	0
0	0	0	0	0

Example: Addition of Noisy Images for Noise Reduction

Noiseless image: f(x,y)

Noise: n(x,y) (at every pair of coordinates (x,y), the noise is uncorrelated and has zero average value)

Corrupted image: g(x,y)

$$g(x,y) = f(x,y) + n(x,y)$$

Reducing the noise by adding a set of noisy images, $\{g_i(x,y)\}$

$$\bar{g}(x,y) = \frac{1}{K}\sum_{i=1}^{K} g_i(x,y)$$

Example: Addition of Noisy Images for Noise Reduction

$$\bar{g}(x,y) = \frac{1}{K}\sum_{i=1}^{K} g_i(x,y)$$

$$E\{\bar{g}(x,y)\} = E\left\{\frac{1}{K}\sum_{i=1}^{K} g_i(x,y)\right\}$$

$$= E\left\{\frac{1}{K}\sum_{i=1}^{K} [f(x,y) + n_i(x,y)]\right\}$$

$$= f(x,y) + E\left\{\frac{1}{K}\sum_{i=1}^{K} n_i(x,y)\right\}$$

$$= f(x,y)$$

Example: Addition of Noisy Images for Noise Reduction

- In astronomy, imaging under very low light levels frequently causes sensor noise to render single images virtually useless for analysis.
- In astronomical observations, similar sensors for noise reduction by observing the same scene over long periods of time. Image averaging is then used to reduce the noise.

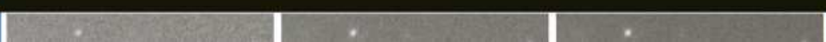


FIGURE 2.26 (a) Image of Galaxy Pair NGC 3314 corrupted by additive Gaussian noise. (b)–(f) Results of averaging 5, 10, 20, 50, and 100 noisy images, respectively. (Original image courtesy of NASA.)

An Example of Image Subtraction: Mask Mode Radiography

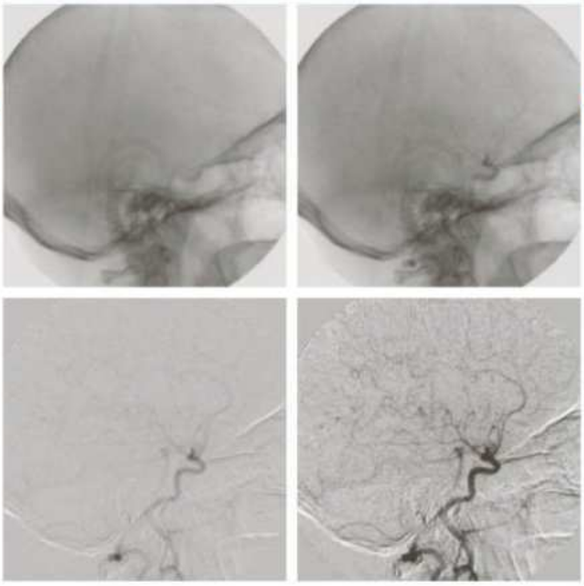
Mask h(x,y): an X-ray image of a region of a patient's body

Live images f(x,y): X-ray images after injection of the contrast medium

Enhanced detail g(x,y)

$$g(x,y) = f(x,y) - h(x,y)$$

The procedure gives a movie showing how the contrast medium propagates through the various arteries in the area being observed.



a b
c d

FIGURE 2.28 Digital subtraction angiography. (a) Mask image. (b) A live image. (c) Difference between (a) and (b). (d) Enhanced difference image. (Figures (a) and (b) courtesy of The Image Sciences Institute, University Medical Center, Utrecht, The Netherlands.)

An Example of Image Multiplication

a b c

FIGURE 2.29 Shading correction. (a) Shaded SEM image of a tungsten filament and support, magnified approximately 130 times. (b) The shading pattern. (c) Product of (a) by the reciprocal of (b). (Original image courtesy of Mr. Michael Shaffer, Department of Geological Sciences, University of Oregon, Eugene.)

Set and Logical Operations

- Let A be the elements of a gray-scale image
  The elements of A are triplets of the form (x, y, z), where x and y are spatial coordinates and z denotes the intensity at the point (x, y).

$$A = \{(x, y, z) \mid z = f(x, y)\}$$

- The complement of A is denoted A^c

$$A^c = \{(x, y, K - z) \mid (x, y, z) \in A\}$$

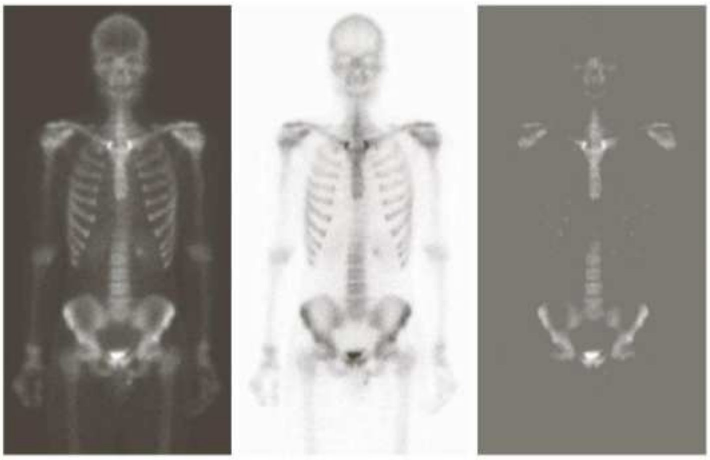
$K = 2^k - 1; k$ is the number of intensity bits used to represent z

Set and Logical Operations

- The union of two gray-scale images (sets) A and B is defined as the set

$$A \cup B = \{\max_z (a, b) \mid a \in A, b \in B\}$$

Set and Logical Operations



a b c

FIGURE 2.32 Set operations involving gray-scale images. (a) Original image. (b) Image negative obtained using set complementation. (c) The union of (a) and a constant image. (Original image courtesy of G.E. Medical Systems.)

of the result.

Spatial Operations

- Single-pixel operations

Alter the values of an image's pixels based on the intensity.

$$s = T(z)$$

e.g.,

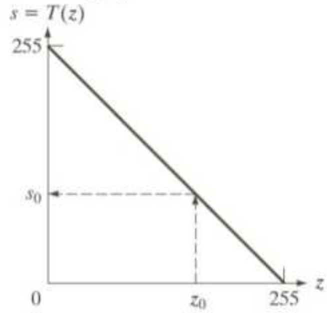
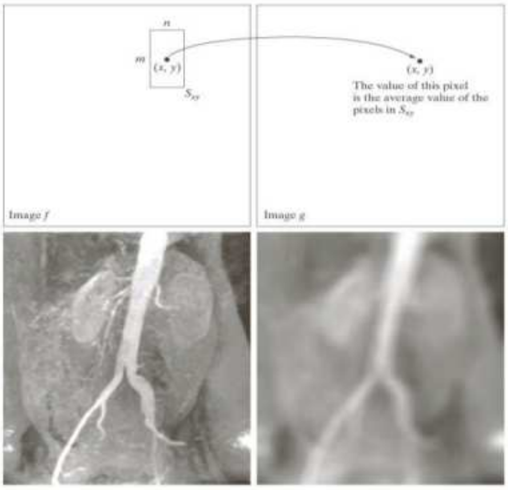


FIGURE 2.34 Intensity transformation function used to obtain the negative of an 8-bit image. The dashed arrows show transformation of an arbitrary input intensity value z_0 into its corresponding output value s_0.

Spatial Operations

- Neighborhood operations



Thank You

Fig:- Logic Operations on Image:

* Need for Image Transform:-

"Transform" is basically a mathematical tool to represent a signal. The need for transforms is as follows:

i) Mathematical Convenience: Every action in time domain will have an impact in frequency domain.

Convolution in Time Domain $\Longleftrightarrow$ Multiplication in frequency Domain.

$\downarrow$ Complex. $\downarrow$ Simple.

ii) To Extract more Information:- It allow us to extract more relevant information.

- To illustrate this, consider the example of Prism experiment. (12)

Cont....

- Person 'X' on the left side sees the light as white light;
- Person 'Y' on the right side sees the white light as a combination of seven colors (VIBGYOR). 24

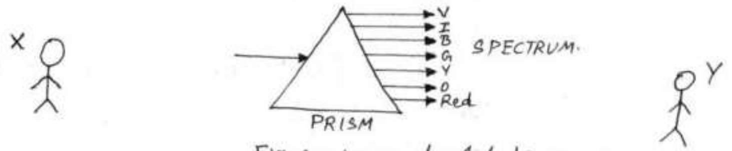


Fig:- Spectrum of white Light:

Fig:- Concept of Transform:

- Here, 'Y' is getting more information than 'X', by using prism.
- Similarly the T/f tool allows us to extract more information

iii) Other needs:-

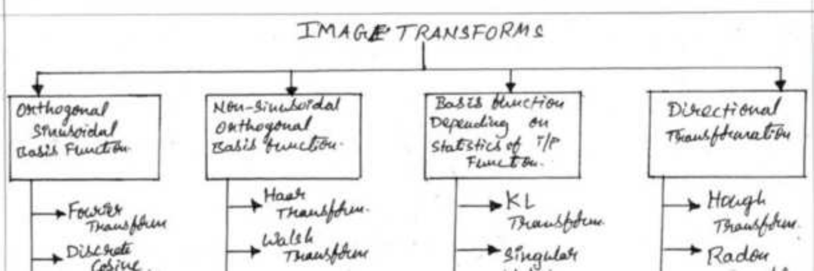
- The transformation may isolate critical components of the image pattern, so that they are directly accessible to analysis.
- The transformation may place the image data in a more compact form so that they can be stored & transmitted efficiently.

➤ Classification of Image Transforms:- Image transforms can be classified on the nature of the "basis Function".

i) Transforms with orthogonal basis function.
ii) Transforms with non-sinusoidal orthogonal basis function.
iii) Transforms whose basis functions depend on the statistics of the input data.
iv) Directional Transformation.

Cont...

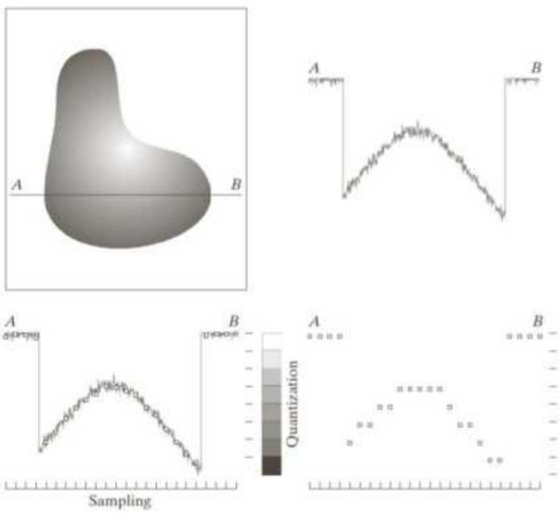
25



Some Typical Ranges of Reflectance

- Reflectance
  - 0.01 for black velvet
  - 0.65 for stainless steel
  - 0.80 for flat-white wall paint
  - 0.90 for silver-plated metal
  - 0.93 for snow

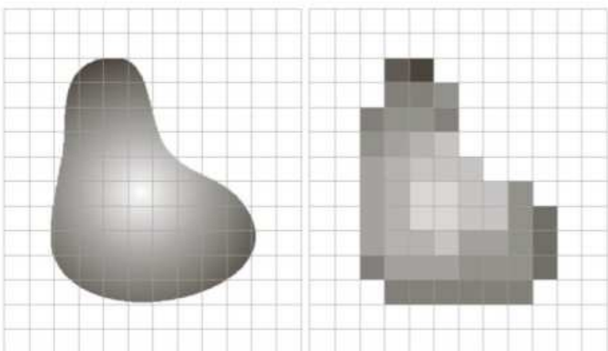
Image Sampling and Quantization



a b
c d

FIGURE 2.16 Generating a digital image. (a) Continuous image. (b) A scan line from A to B in the continuous image, used to illustrate the concepts of sampling and quantization. (c) Sampling and quantization. (d) Digital scan line.

Image Sampling and Quantization



a b

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

digital form results in a digital image (shown in 'e').

* Image Sampling and Quantization:- The o/p of most sensors is a continuous voltage waveform whose amplitude and spatial behavior are related to the physical phenomenon being sensed. 14

To create a digital image, we need to convert the continuous sensed data into digital form.

This involves two processes: 1) Sampling
2) Quantization.

→ Digitizing the coordinate values is called Sampling.

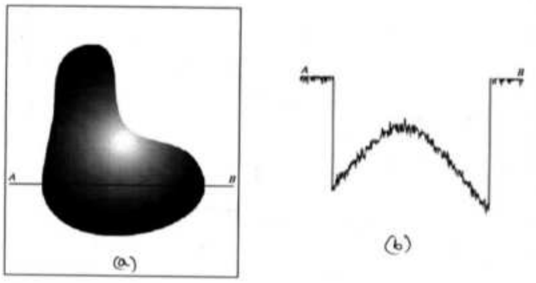
→ Digitizing the amplitude (intensity) values is called quantization.

• Let us consider a continuous image, $f(x, y)$ whose x, y coordinates and amplitude are continuous as shown in fig(a).

• To convert the image into digital form, we have to sample the function in both the amplitude & coordinates.

• As per the definitions of sampling & quantization, we perform the operations on the image in fig(a).

• The 1-D function in fig(b) is a plot of amplitude (gray level) values of the continuous image along the line segment A→B in fig(a).



(a) (b)

Fig:- Generation of Digital Image:

Cont...

• To sample this function, we take equally spaced samples along line AB as shown in fig(c). 15

• The location of each sample is given by a tick mark in bottom part of the figure.

• The samples are shown as small white squares superimposed on the function. The set of these discrete location gives sampled function.

• The values of the samples are still a continuous gray level values.

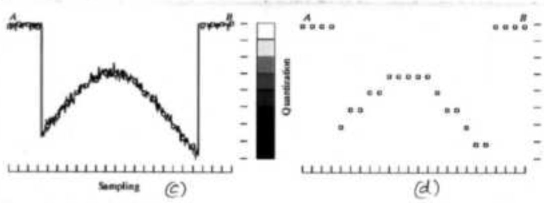
• In order to form a digital function, the gray-level values must be converted (quantized) into discrete quantities.

• The right side of fig(c) shows eight discrete gray-levels ranging from black to white.

• These gray-levels are assigned to each corresponding sample to obtain the quantized value.

• The digital samples obtained in sampling and quantization is represented in fig(d).

• Starting at the top of the image and carrying out this procedure line by line produces a digital image.



(c) (d)

Fig:- Generation of Digital Image:

• When a sensing array is used for image acquisition, there is no motion & the no. of sensors in array establishes the limits of sampling in both directions.

• The figure(2a) shows continuous image projected onto the plane ⑧

P.T.O.

of array sensor and fig (2b) shows the image after sampling and quantization.

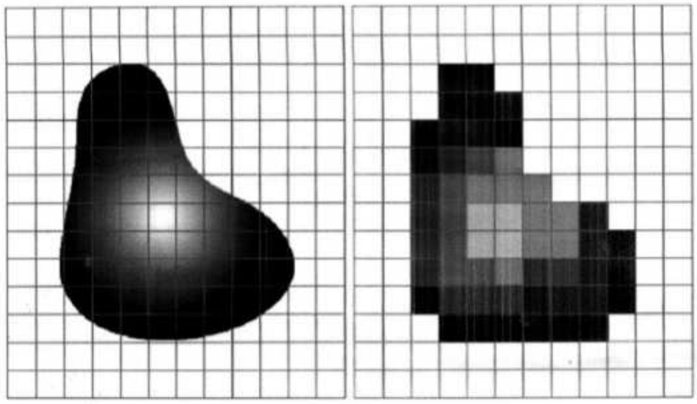


Fig 2:- (a) Continuous Image (b) Digital Image:

• Clearly, the quality of a digital image is determined to a large degree by no. of samples and discrete gray levels used in sampling & quantization.

---

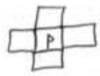
* Some basic Relationships between Pixels:-

Neighbors of a Pixel:-

• A pixel 'p' at coordinates (x,y) has four horizontal and vertical neighbors whose coordinates are given by $(x+1,y)$, $(x-1,y)$ $(x,y+1)$, $(x,y-1)$. This set of pixels called the 4 neighbors of 'p' and is denoted by $N_4(P)$.

• Each pixel is a unit distance from (x,y).

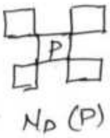
• Some Neighbors of p lie outside the digital image if (x,y) is on the border of the image.



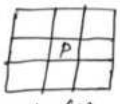
$N_4(P)$ Set:

• The four diagonal neighbors of 'p' have coordinates $(x+1,y+1)$ $(x+1,y-1)$, $(x-1,y+1)$, $(x-1,y-1)$ and are denoted by $N_D(P)$.

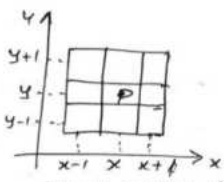
• These points, together with the 4-neighbors, are called the 8-neighbors of 'p', and is denoted by $N_8(P)$.



$N_D(P)$



$N_8(P)$.



Pixel Coordinates.

• Some of the points in $N_D(P)$ & $N_8(P)$ fall outside the image if (x,y) is on the border of image.

Adjacency, Connectivity, Regions and Boundaries:

• Connectivity b/w pixels is a fundamental concept the simplifies the definitions of such as regions and boundaries.

• Two neighbor pixels satisfy a specified criterion of similarity in their gray levels (equal) then they are said to be "Connected."

8 Smoothing Spatial Filters

- **Linear Filters**: Use weighted averages for smoothing.
- **Average Filter Masks**: Assign equal weights to neighboring pixels.

9 Order Statistic (Non-linear) Filters

Filters such as the **Median Filter** reorder pixel values to remove noise without blurring edges.

10 Representation of Objects

Objects are represented in terms of boundaries, regions, and key points within an image.

11 Median Filtering for Noise Reduction and Sharpening

Median Filtering reduces noise while preserving edges, useful for tasks such as sharpening.

12 Laplace Operator

A second-order derivative operator used for edge detection by identifying areas of rapid intensity change.

13 Unsharp Masking and High-Boost Filtering

These techniques enhance edges by subtracting a smoothed version of the image from the original.

14 Spatial Correlation and Convolution

- **Spatial Correlation**: Measures similarity between patterns.

- **Convolution**: Applies a filter to an image, combining values in a local neighborhood.

15 Image Segmentation (Concept)

Image segmentation partitions an image into regions or objects based on properties like color, intensity, and texture.

Spatial Domain vs. Transform Domain

- Spatial domain

  image plane itself, directly process the intensity values of the image plane

- Transform domain

  process the transform coefficients, not directly process the intensity values of the image plane

Spatial Domain Process

$$g(x, y) = T[f(x, y)])$$

$$f(x, y): \text{input image}$$

$$g(x, y): \text{output image}$$

$$T: \text{an operator on } f \text{ defined over a neighborhood of point } (x, y)$$

Spatial Domain Process

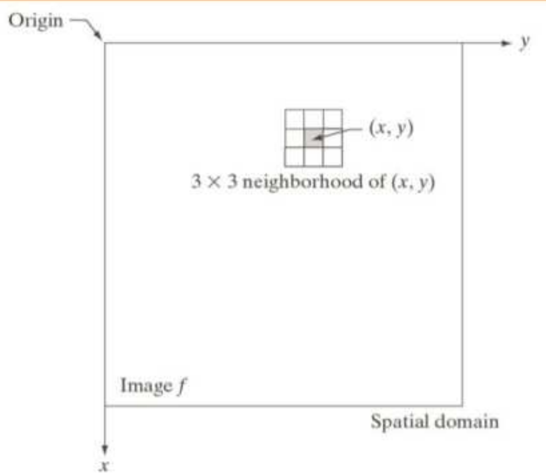


FIGURE 3.1 A 3 × 3 neighborhood about a point (x, y) in an image in the spatial domain. The neighborhood is moved from pixel to pixel in the image to generate an output image.

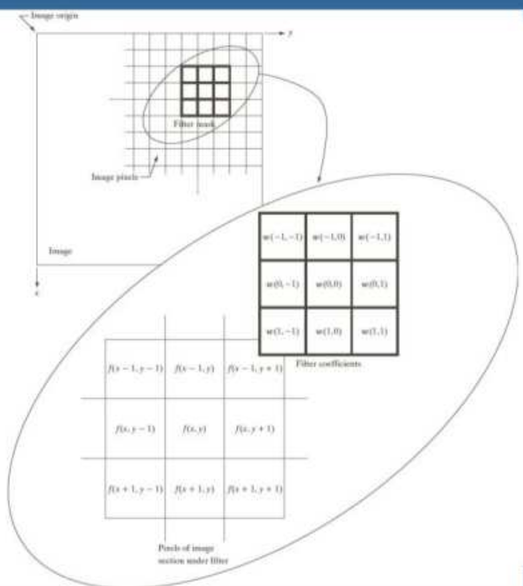
Spatial Filtering

A spatial filter consists of (a) **a neighborhood**, and (b) **a predefined operation**

Linear spatial filtering of an image of size MxN with a filter of size mxn is given by the expression

$$g(x, y) = \sum_{s=-a}^{a} \sum_{t=-b}^{b} w(s,t) f(x+s, y+t)$$

Spatial Filtering



Spatial Correlation

The correlation of a filter $w(x, y)$ of size $m \times n$ with an image $f(x, y)$, denoted as $w(x, y) \star f(x, y)$

$$w(x, y) \star f(x, y) = \sum_{s=-a}^{a} \sum_{t=-b}^{b} w(s,t) f(x+s, y+t)$$

Spatial Convolution

The convolution of a filter $w(x, y)$ of size $m \times n$ with an image $f(x, y)$, denoted as $w(x, y) \star f(x, y)$

$$w(x, y) \star f(x, y) = \sum_{s=-a}^{a} \sum_{t=-b}^{b} w(s,t) f(x-s, y-t)$$

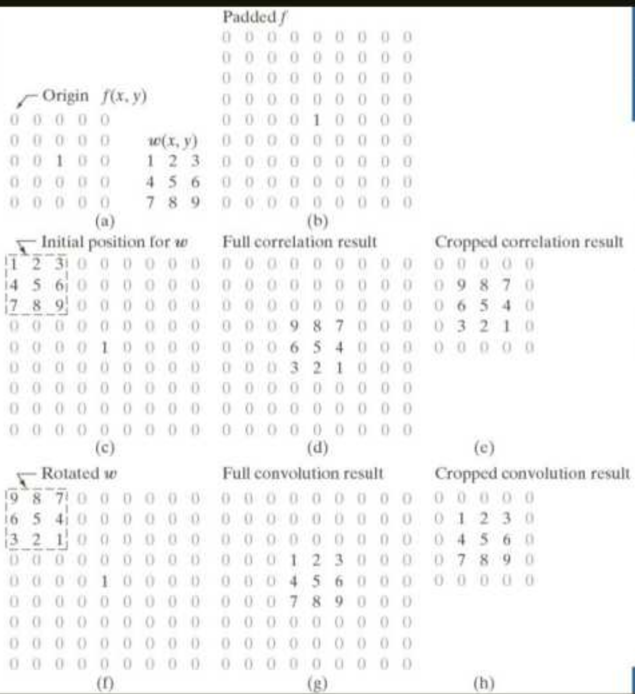


FIGURE 3.30 Correlation (middle row) and convolution (last row) of a 2-D filter with a 2-D discrete, unit impulse. The 0s are shown in gray to simplify visual analysis.

Smoothing Spatial Filters

Smoothing filters are used for blurring and for noise reduction

Blurring is used in removal of small details and bridging of small gaps in lines or curves

Smoothing spatial filters include linear filters and nonlinear filters.

Spatial Smoothing Linear Filters

(3) Periodicity Property :- The 2D-DFT of a function $f(m,n)$ is said to be periodic with a period "N" if, $F(k,l) \longrightarrow F(k+pN, l+qN)$.

Proof :- Consider, $F(k+pN, l+qN) = \sum_{m=0}^{N-1}\sum_{n=0}^{N-1} f(m,n)\, e^{-j\frac{2\pi}{N}(k+pN)m}\, e^{-j\frac{2\pi}{N}(l+qN)n}$.

$$= \sum_{m=0}^{N-1}\sum_{n=0}^{N-1} f(m,n)\, e^{-j\frac{2\pi}{N}km} \cdot e^{-j\frac{2\pi}{N}pNm} \cdot e^{-j\frac{2\pi}{N}ln} \cdot e^{-j\frac{2\pi}{N}qNn}$$

$$= \left[\sum_{m=0}^{N-1}\sum_{n=0}^{N-1} f(m,n)\, e^{-j\frac{2\pi}{N}km}\, e^{-j\frac{2\pi}{N}ln}\right] e^{-j2\pi pm}\, e^{-j2\pi qn}$$

$\therefore F(k+pN, l+qN) = F(k,l)\, e^{-j2\pi pm}\, e^{-j2\pi qn}$.

Here, $e^{-j2\pi pm}$ & $e^{-j2\pi qn}$ are always '1' $\forall$ q & n

Hence, $\boxed{F(k+pN, l+qN) = F(k,l)}$ $\left[\because \exp e^{-j2\pi} = \cos 2\pi - j\sin 2\pi = 1\right]$

---

(4) Convolution Property :- Convolution is one of the most important tool in DIP.

"Convolution in spatial domain is equals to multiplication in time domain".

Proof :- Convolution of two sequences $x(n)$ & $h(n)$ is defined as,

$$x(n) * h(n) = \sum_{k=-\infty}^{\infty} x(k)\, h(n-k) \longrightarrow (1).$$

From (1) we can write 2D-convolution of arrays (or) Matrices $f(m,n)$ & $g(m,n)$ as,

$$f(m,n) * g(m,n) = \sum_{a=0}^{N-1}\sum_{b=0}^{N-1} f(a,b)\, g(m-a, n-b).$$

[$\because$ array/matrix has finite no. of elements]

Now,

$$DFT\{f(m,n) * g(m,n)\} = \sum_{m=0}^{N-1}\sum_{n=0}^{N-1}\left\{\sum_{a=0}^{N-1}\sum_{b=0}^{N-1} f(a,b)\, g(m-a, n-b)\right\} e^{-j\frac{2\pi}{N}mk}\, e^{-j\frac{2\pi}{N}nl}$$

$$= \sum_{m=0}^{N-1}\sum_{n=0}^{N-1}\sum_{a=0}^{N-1}\sum_{b=0}^{N-1} f(a,b)\, g(m-a, n-b)\, e^{-j\frac{2\pi}{N}(m+a-a)k}\, e^{-j\frac{2\pi}{N}(n-b+b)l}$$

$$= \sum_{m=0}^{N-1}\sum_{n=0}^{N-1}\sum_{a=0}^{N-1}\sum_{b=0}^{N-1} f(a,b)\, g(m-a, n-b)\, e^{-j\frac{2\pi}{N}(m-a)k}\, e^{-j\frac{2\pi}{N}ak}\, e^{-j\frac{2\pi}{N}(n-b)l}\, e^{-j\frac{2\pi}{N}bl}$$

$$= \left(\sum_{a=0}^{N-1}\sum_{b=0}^{N-1} f(a,b)\, e^{-j\frac{2\pi}{N}ak}\, e^{-j\frac{2\pi}{N}bl}\right)\left(\sum_{m=0}^{N-1}\sum_{n=0}^{N-1} g(m-a, n-b)\, e^{-j\frac{2\pi}{N}(m-a)k}\, e^{-j\frac{2\pi}{N}(n-b)l}\right) \quad (15)$$

$\boxed{\therefore DFT\{f(m,n) * g(m,n)\} = F(k,l) \times G(k,l)}$ Hence Proved.

---

(5) Correlation Property :- The cross correlation of two sequences $x(n)$ & $h(n)$ is equivalent to performing the convolution of one sequence with the folded version of the other sequence.

Proof :- The DFT of Correlation of two sequences $x(n)$ & $h(n)$ is defined as

$$DFT\{R_{x,h}\} = \sum_{m=0}^{N-1}\left\{\sum_{n=0}^{N-1} x(n)\, h(n+m)\right\} e^{-j\frac{2\pi}{N}mk}.$$

$$= \sum_{m=0}^{N-1}\left\{\sum_{n=0}^{N-1} x(n)\, h(n+m)\right\} e^{-j\frac{2\pi}{N}(m+n-n)k}.$$

$$= \sum_{m=0}^{N-1}\sum_{n=0}^{N-1} x(n)\, h(n+m)\, e^{-j\frac{2\pi}{N}(n+m)k}\, e^{-j\frac{2\pi}{N}(-n)k}.$$

$$= \sum_{n=0}^{N-1} x(n)\, e^{-j\frac{2\pi}{N}(-n)k} \cdot \sum_{m=0}^{N-1} h(n+m)\, e^{-j\frac{2\pi}{N}(n+m)k}$$

$$= H(k) \sum_{n=0}^{N-1} x(n)\, e^{-j\frac{2\pi}{N}n(-k)}$$

$$\boxed{DFT\{R_{x,h}\} = H(k) \cdot X(-k)}$$

It shows that the correlation in time domain equals to multiplication of DFT of one seq. & time reversal of DFT of another sequence.

Note :- Correlation is basically used to find relative similarity b/w two signals.

---

(6) Scaling Property :- It is used to increase or decrease the size of an image.

[illegible]

Smoothing Spatial Filters

Smoothing filters are used for blurring and for noise reduction

Blurring is used in removal of small details and bridging of small gaps in lines or curves

Smoothing spatial filters include linear filters and nonlinear filters.

Spatial Smoothing Linear Filters

The general implementation for filtering an $M \times N$ image with a weighted averaging filter of size $m \times n$ is given

$$g(x,y) = \frac{\sum_{s=-a}^{a} \sum_{t=-b}^{b} w(s,t) f(x+s, y+t)}{\sum_{s=-a}^{a} \sum_{t=-b}^{b} w(s,t)}$$

where $m = 2a+1, \quad n = 2b+1.$

Two Smoothing Averaging Filter Masks

$\frac{1}{9} \times$

1	1	1
1	1	1
1	1	1

$\frac{1}{16} \times$

1	2	1
2	4	2
1	2	1

a b

FIGURE 3.32 Two 3×3 smoothing (averaging) filter masks. The constant multiplier in front of each mask is equal to 1 divided by the sum of the values of its coefficients, as is required to compute an average.

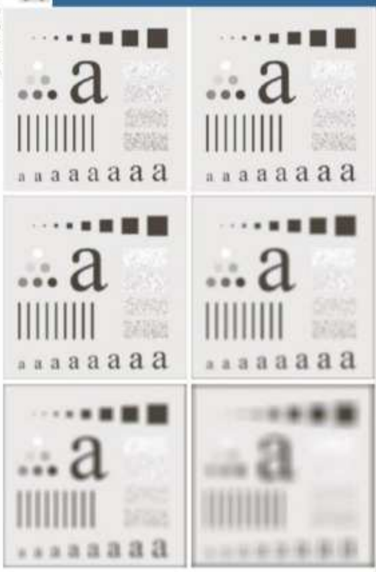
FIGURE 3.33 (a) Original image, of size 500×500 pixels. (b)–(f) Results of smoothing with square averaging filter masks of sizes $m = 3, 5, 9, 15$, and 35, respectively. The black squares at the top are of sizes 3, 5, 9, 15, 25, 35, 45, and 55 pixels, respectively; their borders

a b
c d
e f

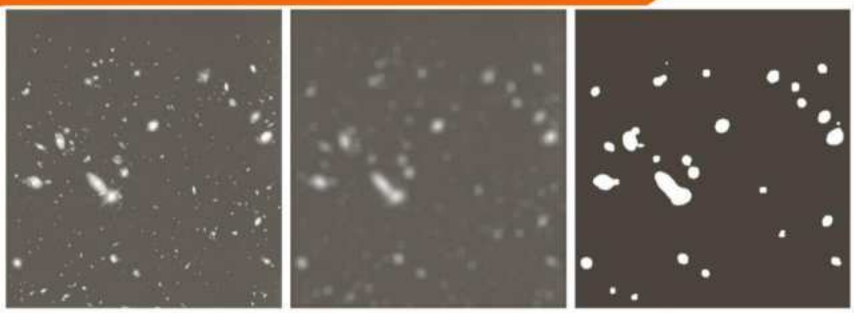
average.

FIGURE 3.33 (a) Original image, of size 500×500 pixels. (b)-(f) Results of smoothing with square averaging filter masks of sizes $m = 3, 5, 9, 15$, and 35, respectively. The black squares at the top are of sizes 3, 5, 9, 15, 25, 35, 45, and 55 pixels, respectively; their borders are 25 pixels apart. The letters at the bottom range in size from 10 to 24 points, in increments of 2 points; the large letter at the top is 60 points. The vertical bars are 5 pixels wide and 100 pixels high; their separation is 20 pixels. The diameter of the circles is 25 pixels, and their borders are 15 pixels apart; their intensity levels range from 0% to 100% black in increments of 20%. The background of the image is 10% black. The noisy rectangles are of size 50×120 pixels.

a b
c d
e f



Example: Gross Representation of Objects



a b c

FIGURE 3.34 (a) Image of size 528×485 pixels from the Hubble Space Telescope. (b) Image filtered with a 15×15 averaging mask. (c) Result of thresholding (b). (Original image courtesy of NASA.)

Order-statistic (Nonlinear) Filters

— Nonlinear

— Based on ordering (ranking) the pixels contained i
filter mask

An "average filter mask" in computer vision is a small matrix of numbers, often called a "kernel," used to calculate the average value of a pixel's neighboring pixels in an image, effectively smoothing out the image by replacing each pixel with the average of its surrounding pixels; essentially, it's a filter where every element in the mask has the same value (usually 1/N, where N is the number of pixels in the mask), allowing for a simple averaging operation when applied to an image.

Key points about average filter masks:

- **Function:** To blur or smooth an image by replacing each pixel with the average of its nearby pixels.
- **Structure:** The mask is typically a square matrix with all elements equal, meaning each pixel in the neighborhood contributes equally to the average.
- **Example:** A 3x3 average filter mask would look like this:

Code

```
[ 1/9 1/9 1/9 ]

[ 1/9 1/9 1/9 ]
```

Order-statistic (Nonlinear) Filters

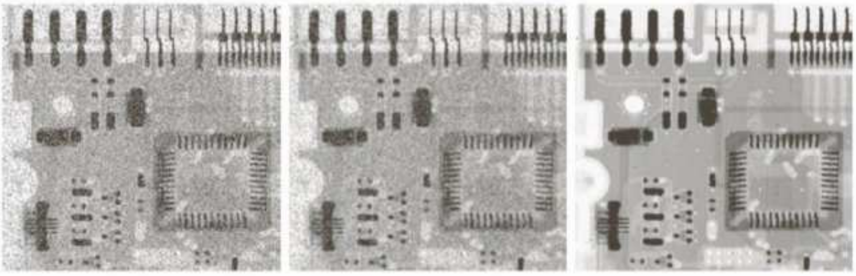
— Nonlinear

— Based on ordering (ranking) the pixels contained in the filter mask

— Replacing the value of the center pixel with the value determined by the ranking result

E.g., median filter, max filter, min filter

Example: Use of Median Filtering for Noise Reduction



a b c

FIGURE 3.35 (a) X-ray image of circuit board corrupted by salt-and-pepper noise. (b) Noise reduction with a 3 × 3 averaging mask. (c) Noise reduction with a 3 × 3 median filter. (Original image courtesy of Mr. Joseph E. Pascente, Lixi, Inc.)

Sharpening Spatial Filters

- Foundation
- Laplacian Operator
- Unsharp Masking and Highboost Filtering
- Using First-Order Derivatives for Nonlinear Image Sharpening — The Gradient

Sharpening Spatial Filters

- Foundation
- Laplacian Operator
- Unsharp Masking and Highboost Filtering
- Using First-Order Derivatives for Nonlinear Image Sharpening — The Gradient

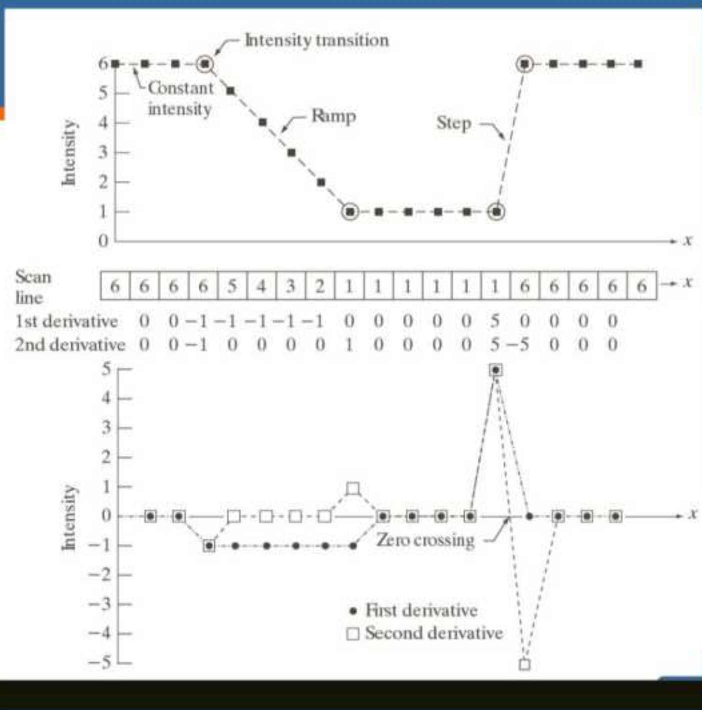
Sharpening Spatial Filters: Foundation

- The first-order derivative of a one-dimensional function f(x) is the difference

$$\frac{\partial f}{\partial x} = f(x+1) - f(x)$$

- The second-order derivative of f(x) as the difference

$$\frac{\partial^2 f}{\partial x^2} = f(x+1) + f(x-1) - 2f(x)$$



a
b
c

FIGURE 3.36 Illustration of the first and second derivatives of a 1-D digital function representing a section of a horizontal intensity profile from an image. In (a) and (c) data points are joined by dashed lines as a visualization aid.

In computer vision, a "median filter" replaces a pixel value with the median value of its neighboring pixels within a specified window, effectively removing noise while preserving edges, while a "max filter" replaces a pixel with the highest value in its neighborhood, highlighting bright areas, and a "min filter" replaces a pixel with the lowest value, effectively darkening the image and highlighting dark areas; all three are considered "non-linear" filters based on order statistics.

Key points about each filter:

Median filter:

- Most commonly used for removing "salt and pepper" noise, which appears as random bright and dark pixels.

Key points about each filter:

Median filter:

- Most commonly used for removing "salt and pepper" noise, which appears as random bright and dark pixels.
- Calculates the middle value of a set of neighboring pixels, replacing the central pixel with that value.
- Considered good for preserving edges while removing noise.

Max filter:

- Used to identify the brightest points in an image, often for edge detection or object boundary extraction.
- Replaces a pixel with the highest value in its neighborhood.
- Can be used to "dilate" objects in an image.

Min filter:

10:41

what is median ,ma...

Min filter:

- Used to identify the darkest points in an image, often for edge detection or object boundary extraction.
- Replaces a pixel with the lowest value in its neighborhood.
- Can be used to "erode" objects in an image.

Minimum, Maximum, and Median Filters -...
The minimum filter erodes shapes on the image, whereas...
Graphics Mill

Median Fi overview
Bovik, Huang introduced a
ScienceD

Generative AI is experimental. Learn more

Export

বাংলায় In English

Home Search Saved Notifications

representation of o...

In computer vision, "object representation" refers to the process of encoding information about an object within an image using numerical or symbolic data, typically including its location, shape, features, and class label, allowing a computer to understand and identify the object within a scene; this is often achieved through features extracted from the image like pixel values, edges, textures, or more complex descriptors depending on the task and chosen algorithm.

Key aspects of object representation:

Bounding Boxes:

A common way to represent an object is by defining a rectangular bounding box around it, specifying its position and size within the image.

Feature Vectors:

Extracting features like color histograms, texture descriptors, or local invariant features (e.g., SIFT, SURF) from the object region to create a numerical representation that captures its unique characteristics.

Object Class Labels:

Assigning a categorical label to the object, such as "car," "person," "dog," to identify its type.

Coordinate Systems:

Representing object position in a chosen coordinate system, like image-centric (pixel coordinates) or world-centric (3D space) depending on the application.

Different approaches to object representation:

Traditional Methods:

- **Edge Detection:** Identifying object boundaries using edge filters.
- **Region-based Analysis:**

10:46 0.00 KB/S 92

median filters for n...

A median filter is a non-linear digital filter that can be used in computer vision to reduce noise and sharpen images:

How it works

A median filter analyzes an image pixel by pixel, replacing each pixel with the median value of its neighboring pixels. The neighboring pixels are usually arranged in a box or cross pattern centered on the pixel being analyzed. The median is the middle value of the sorted pixel values in the window.

Benefits

Median filtering can effectively reduce noise while preserving important image features like edges and fine details. It's often used as a pre-processing step to improve the results of other image processing tasks, such as edge detection.

Applications

Median filtering is used in a wide range of applications, including computer vision, medical imaging, remote sensing, and video processing.

Median filter - Wikipedia

The median filter is a non-linear digital filtering technique, often...

Median Fi overview

Another type

Sharpening Spatial Filters: Laplace Operator

The second-order isotropic derivative operator is the Laplacian for a function (image) f(x,y)

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

$$\frac{\partial^2 f}{\partial x^2} = f(x+1, y) + f(x-1, y) - 2f(x, y)$$

$$\frac{\partial^2 f}{\partial y^2} = f(x, y+1) + f(x, y-1) - 2f(x, y)$$

$$\nabla^2 f = f(x+1, y) + f(x-1, y) + f(x, y+1) + f(x, y-1)$$
$$-4f(x, y)$$

Sharpening Spatial Filters: Laplace Operator

(a)

0	1	0
1	−4	1
0	1	0

(b)

1	1	1
1	−8	1
1	1	1

(c)

0	−1	0
−1	4	−1
0	−1	0

(d)

−1	−1	−1
−1	8	−1
−1	−1	−1

a b
c d

FIGURE 3.37 (a) Filter mask used to implement Eq. (3.6-6). (b) Mask used to implement an extension of this equation that includes the diagonal terms. (c) and (d) Two other implementations of the Laplacian found frequently in practice.

Sharpening Spatial Filters: Laplace Operator

Image sharpening in the way of using the Laplacian:

$$g(x, y) = f(x, y) + c\left[\nabla^2 f(x, y)\right]$$

where,

$f(x, y)$ is input image,

$g(x, y)$ is sharpenend images,

$c =$ -1 if $\nabla^2 f(x, y)$ corresponding to Fig. 3.37(a) or (b)

and $c = 1$ if either of the other two filters is used.

10:48 4.00 KB/S 92

what is laplace ord...

Show more

বাংলায় In English

Introduction. The Laplacian operator is **a second-order differential operator in n-dimensional Euclidean space**, denoted as ∇^2. It is the divergence of the gradient of a function. 16 Dec 2023

https://medium.com › aimonks › e...

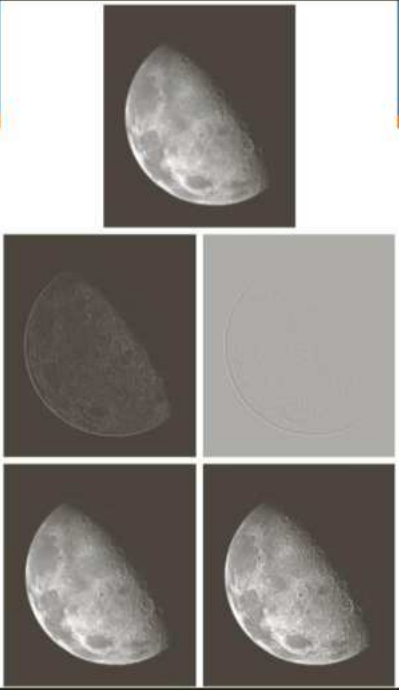
Exploring the Laplacian Operator: A Key Tool in Computer Vision for ...

About featured snippets Feedback

People also ask

What is the Laplace operator in computer vision?

Home Search Saved Notifications



a
b c
d e

FIGURE 3.38 (a) Blurred image of the North Pole of the moon. (b) Laplacian without scaling. (c) Laplacian with scaling. (d) Image sharpened using the mask in Fig. 3.37(a). (e) Result of using the mask in Fig. 3.37(b). (Original image courtesy of NASA.)

Unsharp Masking and Highboost Filtering

- Unsharp masking

  Sharpen images consists of subtracting an unsharp (smoothed) version of an image from the original image

  e.g., printing and publishing industry

- Steps

  1. Blur the original image
  2. Subtract the blurred image from the original
  3. Add the mask to the original

Unsharp Masking and Highboost Filtering

Let $\overline{f}(x, y)$ denote the blurred image, unsharp masking is

$$g_{mask}(x, y) = f(x, y) - \overline{f}(x, y)$$

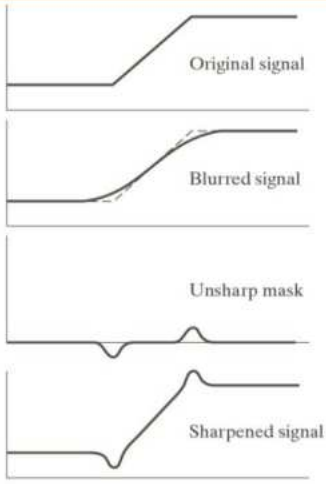
Then add a weighted portion of the mask back to the original

$$g(x, y) = f(x, y) + k * g_{mask}(x, y) \qquad k \geq 0$$

when $k > 1$, the process is referred to as highboost filtering.

Unsharp Masking: Demo

Unsharp Masking: Demo



a b c d

FIGURE 3.39 1-D illustration of the mechanics of unsharp masking. (a) Original signal. (b) Blurred signal with original shown dashed for reference. (c) Unsharp mask. (d) Sharpened signal, obtained by adding (c) to (a).

Unsharp Masking and Highboost Filtering: Example

a b c d e

FIGURE 3.40 (a) Original image. (b) Result of blurring with a Gaussian filter. (c) Unsharp mask. (d) Result of using unsharp masking. (e) Result of using highboost filtering.

Image Sharpening based on First-Order Derivatives

For function $f(x, y)$, the gradient of f at coordinates (x, y) is defined as

$$\nabla f \equiv \text{grad}(f) \equiv \begin{bmatrix} g_x \\ g_y \end{bmatrix} = \begin{bmatrix} \dfrac{\partial f}{\partial x} \\ \dfrac{\partial f}{\partial y} \end{bmatrix}$$

The *magnitude* of vector ∇f, denoted as $M(x, y)$

Gradient Image $\quad M(x, y) = \text{mag}(\nabla f) = \sqrt{g_x^{\,2} + g_y^{\,2}}$

Image Sharpening based on First-Order Derivatives

The *magnitude* of vector ∇f, denoted as $M(x, y)$

$$M(x, y) = \text{mag}(\nabla f) = \sqrt{g_x^{\ 2} + g_y^{\ 2}}$$

$$M(x, y) \approx |g_x| + |g_y|$$

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

$$M(x, y) = |z_8 - z_5| + |z_6 - z_5|$$

Image Sharpening based on First-Order Derivatives

Roberts Cross-gradient Operators

$$M(x, y) \approx |z_9 - z_5| + |z_8 - z_6|$$

Sobel Operators

$$M(x, y) \approx |(z_7 + 2z_8 + z_9) - (z_1 + 2z_2 + z_3)| \\ + |(z_3 + 2z_6 + z_9) - (z_1 + 2z_4 + z_7)|$$

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

Image Sharpening based on First-Order Derivatives

(a)

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

(b)

−1	0
0	1

(c)

0	−1
1	0

(d)

−1	−2	−1
0	0	0
1	2	1

(e)

−1	0	1
−2	0	2
−1	0	1

a
b c
d e

FIGURE 3.41 A 3 × 3 region of an image (the zs are intensity values). (b)–(c) Roberts cross gradient operators. (d)–(e) Sobel operators. All the mask coefficients sum to zero, as expected of a derivative operator.

a b

FIGURE 3.42 (a) Optical image

10:50 124 KB/S 92

what is high boost...

বাংলায় In English

Definition. High-boost filtering is a technique used in image processing to enhance the details of an image by amplifying high-frequency components while reducing lower frequencies.

https://library.fiveable.me › high-b...

High-boost filtering - (Computer Vision and Image Processing)

About featured snippets Feedback

People also ask

What is high boost filtering?

What is Unsharp masking and high boost filtering?

Home Search Saved Notifications

neighborhood.

15 Image Segmentation (Concept)

Image segmentation partitions an image into regions or objects based on properties like color, intensity, and texture.

16 Addition of Two Images

Combining pixel values of two images to create a composite image.

17 Spatial and Intensity Resolution

- **Spatial Resolution**: The smallest discernible detail in an image.
- **Intensity Resolution**: The smallest discernible change in intensity.

18 Connected Component Analysis

Identifies and labels connected regions in a binary image.

19 Light-Sensitive Receptors

- **Rods**: Sensitive to low light levels.
- **Cones**: Detect color and function in bright light.

20 Edge Detection Using Prewitt and Sobel Operators

- **Prewitt and Sobel**: First-order derivative operators for edge detection.
- **Marr-Hildreth**: A second-order method using Laplacian of Gaussian.

21 Line and Point Detection

Algorithms designed to identify specific structures such as lines and points in an image.

22 Image Classification

22.1 Problem and Challenges

- Variability in illumination, perspective, and noise.

22.2 Nearest Neighbor Classifier

- **L1 Distance**: Sum of absolute differences.

22.3 Hyperparameters

Values set before the learning process to optimize performance.

22.4 Kth Nearest Neighbor

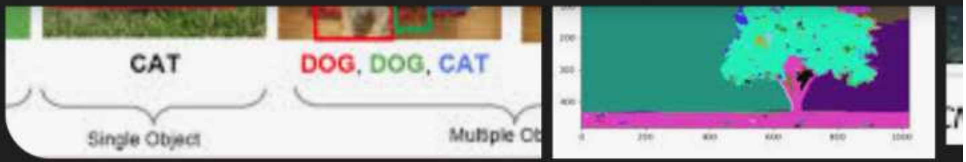
Considers the majority class among k closest points.

22.5 Linear Classification (Parametric Approach)

Models the decision boundary as a linear function.

10:58

what is image seg...



বাংলায় In English

Image segmentation is **a computer vision technique that partitions a digital image into discrete groups of pixels—image segments—to inform object detection and related tasks**. By parsing an image's complex visual data into specifically shaped segments, image segmentation enables faster, more advanced image processing.

https://www.ibm.com › topics › im...

What Is Image Segmentation? | IBM

Home Search Saved Notifications

10:58 0.05 KB/S Vo LTE 5G 91

what is image seg...

Image segmentation is used in many fields, including:

- **Medical imaging**: To detect and label pixels that represent tumors in a patient's brain or other organs
- **Agriculture**: To use image segmentation in agricultural applications
- **Satellite images**: To use image segmentation in satellite images
- **Autonomous vehicles**: To use image segmentation in AI for autonomous vehicles

Some techniques used in image segmentation include:

- **Watershed segmentation**: Treats the image as a topographic surface, where pixel values represent elevation. This technique is useful for separating overlapping

Home Search Saved Notifications

To add two images in computer vision, you essentially perform a pixel-by-pixel addition of the corresponding values from each image, creating a new image where each pixel is the sum of the corresponding pixels from the original two images; this operation requires both images to have the same dimensions and color space, and is typically done using a function like "cv2.add()" in OpenCV libraries, where each pixel in the output image is calculated by adding the values of the corresponding pixels from the input images.

Key points about image addition:

Pixel-wise operation:

The addition happens at the level of individual pixels, meaning the value of each pixel in the resulting image is the how to do Addition...

Key points about image addition:

Pixel-wise operation:

The addition happens at the level of individual pixels, meaning the value of each pixel in the resulting image is the sum of the values of the corresponding pixels in the original images.

Same dimensions:

Both input images must have the same dimensions (width and height) for the addition to work correctly.

Color space compatibility:

The images should be in the same color space (e.g., RGB, grayscale) to ensure meaningful addition.

Potential overflow:

If the pixel values add up to exceed the maximum value allowed by the data type (e.g., 255 for 8-bit grayscale), the result might be clipped to the maximum value, leading to potential information loss.

Advanced Techniques for Edge Detection

- The Marr-Hildreth edge detector

$$G(x,y) = e^{-\frac{x^2+y^2}{2\sigma^2}}, \quad \sigma : \text{space constant.}$$

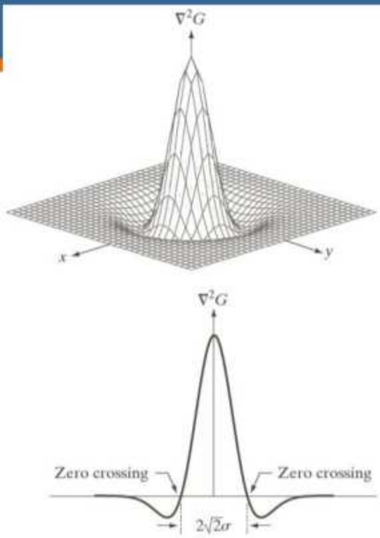
Laplacian of Gaussian (LoG)

$$\nabla^2 G(x,y) = \frac{\partial^2 G(x,y)}{\partial x^2} + \frac{\partial^2 G(x,y)}{\partial y^2}$$

$$= \frac{\partial}{\partial x}\left[\frac{-x}{\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}\right] + \frac{\partial}{\partial y}\left[\frac{-y}{\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}\right]$$

$$= \left[\frac{x^2}{\sigma^4} - \frac{1}{\sigma^2}\right] e^{-\frac{x^2+y^2}{2\sigma^2}} + \left[\frac{y^2}{\sigma^4} - \frac{1}{\sigma^2}\right] e^{-\frac{x^2+y^2}{2\sigma^2}}$$

$$= \left[\frac{x^2+y^2-\sigma^2}{\sigma^4}\right] e^{-\frac{x^2+y^2}{2\sigma^2}}$$



0	0	−1	0	0
0	−1	−2	−1	0
−1	−2	16	−2	−1
0	−1	−2	−1	0
0	0	−1	0	0

a b
c d

FIGURE 10.21 (a) Three-dimensional plot of the *negative* of the LoG. (b) Negative of the LoG displayed as an image. (c) Cross section of (a) showing zero crossings. (d) 5×5 mask approximation to the shape in (a). The negative of this mask would be used in practice.

Marr-Hildreth Algorithm

1. Filter the input image with an nxn Gaussian lowpass filter. n is the smallest odd integer greater than or equal to 6σ
2. Compute the Laplacian of the image resulting from Step 1.
3. Find the zero crossing of the image from Step 2.

$$g(x,y) = \nabla^2 \left[G(x,y) \star f(x,y)\right]$$

③ **Binary Images** :→ Binary images are represented using a single bit per pixel, where each pixel in either black (0) or white (1).

→ Binary images are often generated by thresholding grayscale or color images, where pixels above a certain intensity threshold are set to white, and pixels below the threshold are set to black.

→ Binary images are used in tasks such as object detection, segmentation, and feature extraction.

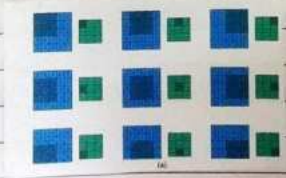
⇒ Image Filtering :-

① **Convolution :**

→ Convolution is a fundamental operation in image processing where a Kernel (also Known as a filter or mask) is applied to an image.

→ The Kernel is a small matrix used to modify the values of pixels in the original image.

→ During Convolution, the Kernel slides over the entire image, and at each position, the sum of element-wise products between the Kernel and the overlapping image region is computed to produce the output pixel value.



② **Image Filtering Techniques :-**

ⓐ **Blurring** :→ Blurring is a common image filtering technique used to reduce noise and detail in images.

→ It works by averaging pixel values within a local neighborhood defined by the kernel.

→ Popular blurring filters include the Gaussian blur, which assigns higher weights to central pixels and lower weights to surrounding pixels, creating a smoother effect.

ⓑ **Sharpening** :- → Sharpening enhances the edges and details in an image.

→ It works by accentuating the difference between neighboring pixel values.

of gradient magnitude after applying Gaussian smoothing and gradient calculation.

(d) Noise Reduction:-

→ Noise reduction filters are used to remove unwanted artifacts or irregularities from images

→ Gaussian filters smooth the image by convolving it with a Gaussian Kernel, which effectively reduces high-frequency noise.

→ Median filters replace each pixel's value with the median value in its neighborhood, making them robust to outliers and preserving edges.

(e) Gaussian Filter:- → The Gaussian Filter is a commonly used image smoothing filter.

→ It applies a weighted average to the pixels in the images, with the weights determined by a Gaussian function.
The Gaussian filter effectively reduces high-frequency noise in the image while preserving important edges and structures.

→ It is widely used as a preprocessing step before applying other image processing techniques.
The size of the Gaussian Kernel and the standard deviation of the Gaussian function are parameters that affect the smoothing effect.

(f) Median Filter:- → The median filter replaces each pixel's value with the median value within its neighborhood.

→ It is effective at removing impulsive noise (salt-and-pepper noise) while preserving edges and fine details.

→ Unlike the Gaussian filter, the median filter is non-linear and does not smooth the image in a uniform manner.
It is computationally efficient and robust to outliers in the image.

② Image Segmentation :-

→ Image segmentation involves partitioning an image into multiple segments or regions based on pixel-level classification.

→ The goal is to assign a label to each pixel in the image, thereby dividing the image into semantically meaningful regions.

→ Image segmentation allows for a more detailed understanding of the spatial structure of objects within an image.

→ Examples include segmenting objects from backgrounds, medical image segmentation, and scene parsing.

→ Architectures commonly used for image (semantic) segmentation include U-Net, Fully Convolutional Networks (FCNs), and DeepLab.

→ Semantic segmentation assigns a single class label to each pixel, while instance segmentation distinguishes between individual object instances within the same class

→ ~~Image~~ Instance Segmentation goes a step further by not only identifying object classes but also distinguishing individual object instances within an image.
Mask R-CNN and its variants are popular architectures for instance segmentation.

⇒ Popular datasets used for image classification tasks include MNIST, CIFAR-10, and ImageNet.

① MNIST : MNIST is a dataset containing 28X28 grayscale images of handwritten digits (0-9). It is widely used as a benchmark dataset for testing various machine learning algorithms, including SVMs and decision trees.

② CIFAR-10 : CIFAR-10 consists of 32X32 color images across 10 classes, such as airplanes, automobiles, birds, cats, etc.
It is another popular dataset for image classification tasks, allowing researchers

what isspatial resol...

Spatial resolution in image processing is a measure of the amount of detail in an image, and is determined by the number of pixels used to create it:

Definition

Spatial resolution is the number of pixels used to construct a digital image.

Higher resolution

A higher spatial resolution means more pixels, which allows for finer details to be reproduced.

Factors that affect spatial resolution

Factors that can affect spatial resolution include the distance between the target and the remote sensor, atmospheric conditions, target signal strength, and contrast between the target and the background.

Use cases

Spatial resolution is important in many applications, including medical imaging, remote sensing, and video production:

- **Medical imaging**: In medical imaging, spatial resolution is important for accurately depicting microstructures, which can be crucial for diagnosis.
- **Remote sensing**: High spatial resolution satellites can be used to capture remote sensing data at specific locations and times.
- **Video production**: Adaptive spatial resolution techniques can be used to optimize rendering and encoding for specific scenes.

Spatial Resolution - an overview | ScienceDire...
Spatial resolution refers to the scale or size of the smallest unit...

Digital Im
- Spatial R
11 Feb 2016 –
a term that re

In computer vision, "intensity resolution" refers to the number of distinct brightness levels a pixel can represent in an image, essentially determining how fine the gradations of light and dark can be within a single pixel; the higher the intensity resolution, the more detailed the image can be in terms of brightness variations.

Key points about intensity resolution:

Measured in bits:

Intensity resolution is usually expressed by the number of bits used to store each pixel's intensity value. For example, an 8-bit image can represent 256 different intensity levels (2^8).

Impact on image quality:

A higher intensity resolution allows for more subtle distinctions between different brightness levels, resulting in

11:17 4.00 KB/S Vo LTE 5G 88

what is intensity re...

In computer vision, "intensity resolution" refers to the number of distinct brightness levels a pixel can represent in an image, essentially determining how fine the gradations of light and dark can be within a single pixel; the higher the intensity resolution, the more detailed the image can be in terms of brightness variations.

Key points about intensity resolution:

Measured in bits:

Intensity resolution is usually expressed by the number of bits used to store each pixel's intensity value. For example, an 8-bit image can represent 256 different intensity levels (2^8).

Impact on image quality:

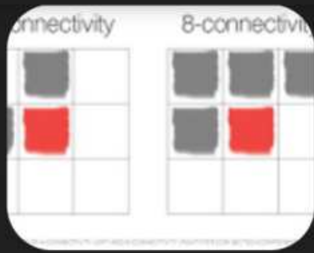
A higher intensity resolution allows for more subtle distinctions between different brightness levels, resulting in a more detailed image.

Relationship to spatial resolution:

While spatial resolution refers to the number of pixels in an image, intensity resolution focuses on the range of brightness values each pixel can hold.

Digital Image Fundamentals - II | PPT ...
11 Jan 2015 — It covers gamma

Understa
Images fo
19 Oct 2023 -



In computer vision, "connected component analysis" (CCA) is a technique used to identify and label distinct regions (or "blobs") within a binary image where pixels are connected to each other, essentially allowing you to separate and analyze individual objects within an image by grouping together neighboring pixels that belong to the same connected area; it's often used for tasks like object counting, segmentation, and feature extraction in images where the foreground objects are clearly separated from the background.

Key points about connected component analysis:

Binary image input:

This technique primarily operates on

Key points about connected component analysis:

Binary image input:

This technique primarily operates on binary images, where each pixel is either considered "on" (foreground) or "off" (background).

Labeling connected regions:

The core function of CCA is to assign unique labels to each separate connected region within the image, allowing you to differentiate between different objects.

Connectivity definition:

When determining if pixels are connected, you can use different connectivity rules like 4-connectivity (only checking adjacent pixels horizontally and vertically) or 8-connectivity (including diagonals).

Applications of connected component analysis:

Applications of connected component analysis:

Object counting:

Counting the number of distinct objects in an image by identifying and counting the labeled connected components.

Character recognition:

Analyzing individual characters on a license plate by isolating each character as a connected component.

Medical image analysis:

Identifying and analyzing individual cells or tissue regions in a medical scan

Defect detection:

Finding flaws or cracks in an image by identifying small connected regions that deviate from the expected pattern

How it works:

Scanning the image:

The algorithm iterates through each pixel in the binary image.

Label assignment:

When a foreground pixel is encountered, it is assigned a label based on its connected neighbors.

Equivalence checking:

If a pixel connects to multiple regions with different labels, the labels are marked as equivalent and later unified to a single label.

Label refinement:

After scanning the entire image, a second pass may be needed to resolve label equivalences and assign consistent labels to all connected components.

Connected-component OpenCV

12:46 68.0 KB/S 78

light sensitive rece...

In computer vision, "rods" refer to the light-sensitive receptors in the human eye that are primarily responsible for vision in low-light conditions (scotopic vision), meaning they are most active in dim environments and are crucial for night vision, detecting shades of gray rather than color; essentially, they are highly sensitive to light but do not distinguish between different wavelengths of color like cones do.

Key points about rods:

- **Function:** Primarily responsible for seeing in low light conditions.
- **Sensitivity:** Extremely sensitive to light, even detecting single photons.
- **Color perception:** Cannot distinguish colors, only detect shades of gray.
- **Location in the eye:** Concentrated in the peripheral retina, away from the fovea.

Rods & Cones

Rods are responsible for vision at low light levels (scotopic vision)....

Photorec
Cones): A
6 May 2024 -
photorecept

Cones are light-sensitive receptors in the retina of the eye that are responsible for color vision:

Location

Cones are concentrated in the fovea, a small pit in the center of the retina.

Number

Each eye contains around 6 million cones.

Types

There are three types of cones, each sensitive to different wavelengths of light:

- **L-cones**: Sensitive to red light
- **M-cones**: Sensitive to green light
- **S-cones**: Sensitive to blue light

Function

Cones are responsible for sharp color vision in daylight. They are active at higher light levels and are responsible for high spatial acuity.

Photopigments

Cones contain photopigments, which are proteins linked to a chromophore that is an aldehyde of vitamin A.

Efficiency

Cones are highly efficient, absorbing up to 75–80% of all incident photons.

Afterimage

Staring at a particular color for a minute or so can exhaust the cone cells that respond to that color, resulting in a vivid color aftereffect.

Rods & Cones

Cones are active at higher light levels (photopic vision), are...

Photorec Cones): A

6 May 2024

Prewitt Operator

The Prewitt operator was developed by Judith M. S. Prewitt. Prewitt operator is used for edge detection in an image. Prewitt operator detects both types of edges, these are:

- Horizontal edges or along the x-axis,
- Vertical Edges or along the y-axis.

Wherever there is a sudden change in pixel intensities, an edge is detected by the mask. Since the edge is defined as the change in pixel intensities, it can be calculated by using differentiation. Prewitt mask is a first-order derivative mask. In the graph representation of Prewitt-mask's result, the edge is represented by the local maxima or local minima.

Both the first and second derivative masks follow these three properties:

- More weight means more edge detection.
- The opposite sign should be present in the mask. (+ and -)
- The Sum of ... values must be

- The opposite sign should be present in the mask. (+ and -)
- The Sum of the mask values must be equal to zero.

Prewitt operator provides us two masks one for detecting edges in the horizontal direction and another for detecting edges in a vertical direction.

```
Prewitt Operator [X-axis] = [ -1 0 1; -1 0 1; -1 0 1]

Prewitt Operator [Y-axis] = [-1 -1 -1; 0 0 0; 1 1 1]
```

Steps:

- Read the image.
- Convert into grayscale if it is colored.
- Convert into the double format.
- Define the mask or filter.
- Detect the edges along X-axis.
- Detect the edges along Y-axis.
- Combine the edges detected along the X and Y axes.
- Display all the images.

Imtool() is the inbuilt function in Matlab. It is used to display the image. It takes 2 parameters; the

Sobel Operator

It is named after Irwin Sobel and Gary Feldman. Like the Prewitt operator Sobel operator is also used to detect two kinds of edges in an image:

- Vertical direction
- Horizontal direction

The difference between Sobel and Prewitt Operator is that in Sobel operator the coefficients of masks are adjustable according to our requirement provided they follow all properties of derivative masks.

Sobel-X Operator = [-1 0 1; -2 0 2; -1 0 1]

Sobel-Y Operator = [-1 -2 -1; 0 0 0; 1 2 1] profiles. Third column: Second derivative images and intensity profiles.

Basic Edge Detection by Using First-Order Derivative

$$\nabla f \equiv grad(f) = \begin{bmatrix} g_x \\ g_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix}$$

The magnitude of ∇f

$$M(x, y) = \text{mag}(\nabla f) = \sqrt{g_x^{\ 2} + g_y^{\ 2}}$$

The direction of ∇f

$$\alpha(x, y) = \arctan(g_y / g_x)$$

The direction of the edge

$$\phi = \alpha - 90^\circ$$

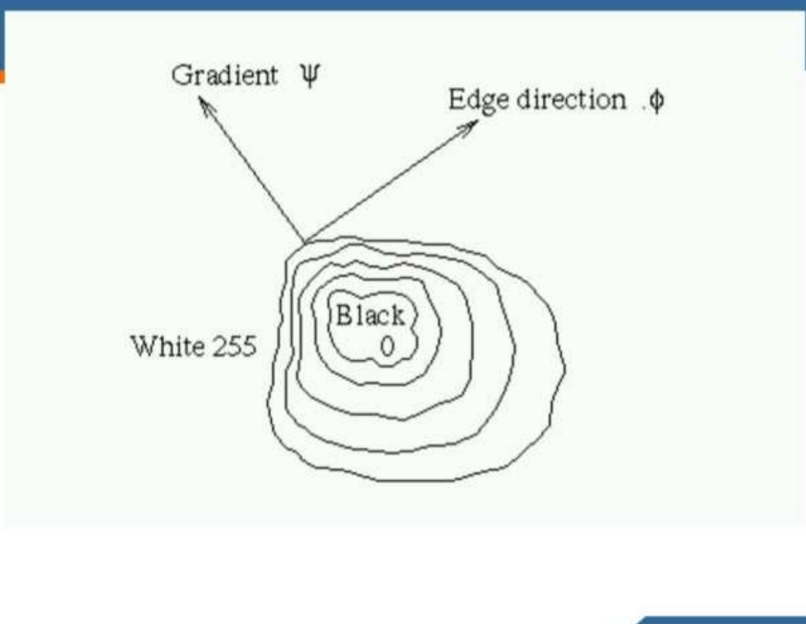
Basic Edge Detection by Using First-Order Derivative

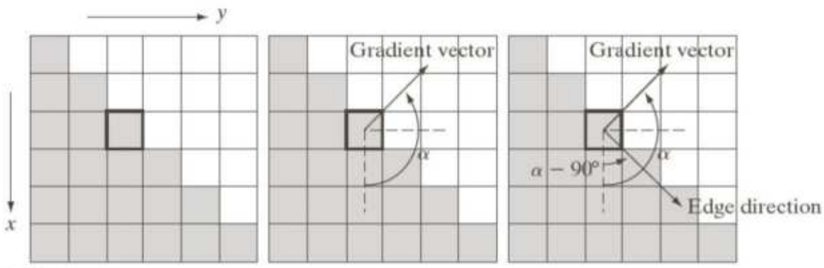
Edge normal: $\nabla f \equiv grad(f) = \begin{bmatrix} g_x \\ g_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix}$

Edge unit normal: $\nabla f / \text{mag}(\nabla f)$

In practice,sometimes the magnitude is approximated by

$$\text{mag}(\nabla f) = \left|\frac{\partial f}{\partial x}\right| + \left|\frac{\partial f}{\partial y}\right| \text{ or } \text{mag}(\nabla f) = \max\left(\left|\frac{\partial f}{\partial x}\right|, \left|\frac{\partial f}{\partial y}\right| \right)$$





a b c

FIGURE 10.12 Using the gradient to determine edge strength and direction at a point. Note that the edge is perpendicular to the direction of the gradient vector at the point where the gradient is computed. Each square in the figure represents one pixel.

−1
1

−1	1

a b

FIGURE 10.13 One-dimensional masks used to implement Eqs. (10.2-12) and (10.2-13).

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

−1	0
0	1

0	−1
1	0

Roberts

−1	−1	−1
0	0	0
1	1	1

−1	0	1
−1	0	1
−1	0	1

Prewitt

−1	−2	−1
0	0	0
1	2	1

−1	0	1
−2	0	2
−1	0	1

Sobel

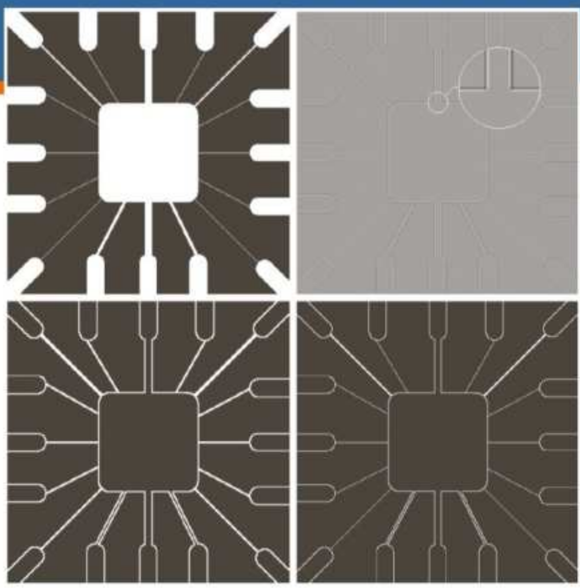
a
b c
d e
f g

FIGURE 10.14 A 3 × 3 region of an image (the z's are intensity values) and various masks used to compute the gradient at the point labeled z_5.

Line Detection

- Second derivatives to result in a stronger response and to produce thinner lines than first derivatives

- Double-line effect of the second derivative must be handled properly



a b
c d

FIGURE 10.5 (a) Original image. (b) Laplacian image; the magnified section shows the positive/negative double-line effect characteristic of the Laplacian. (c) Absolute value of the Laplacian. (d) Positive values of the Laplacian.

Detecting Line in Specified Directions

Horizontal

−1	−1	−1
2	2	2
−1	−1	−1

+45°

2	−1	−1
−1	2	−1
−1	−1	2

Vertical

−1	2	−1
−1	2	−1
−1	2	−1

−45°

−1	−1	2
−1	2	−1
2	−1	−1

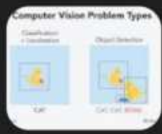
FIGURE 10.6 Line detection masks. Angles are with respect to the axis system in Fig. 2.18(b).

- Let R_1, R_2, R_3, and R_4 denote the responses of the masks in Fig. 10.6. If, at a given point in the image, $|R_k|>|R_j|$, for all $j \neq k$, that point is said to be more likely associated with a line in the direction of mask k.

a b
c d

1:12

what is point detec...



In computer vision, "point detection" refers to the process of identifying specific, salient points of interest within an image, often called "keypoints" or "interest points," which are locations that stand out due to significant changes in intensity or texture, allowing for further analysis of the image like object tracking, pose estimation, or facial recognition; essentially, it's about pinpointing key locations within an image that provide meaningful information about the scene or object depicted.

Key points about point detection:

What it detects:

Corners, junctions, or other distinct features in an image that are relatively invariant to changes in viewpoint, scale, or lighting.

Applications:

- **Object tracking:** Identifying key points on an object to track its movement across frames.
- **Human pose estimation:** Locating joints on a human body.
- **Facial recognition:** Identifying key facial landmarks like eyes, nose, and mouth.
- **Augmented reality:** Precisely placing virtual objects in the real world.

Common algorithms:

- **Harris Corner Detector:** A classic algorithm that uses the local image gradient to identify corners.
- **Scale Invariant Feature Transform (SIFT):** Detects keypoints across different scales and rotations.
- **Oriented FAST and Rotated BRIEF (ORB):** A faster alternative to SIFT with good performance

A Beginners Guide to Computer Vision (Part...

23 Apr 2020 — Here comes the concept of Interest Points. Look...

Medium

Object De
Image Cla

28 Sept 2022

Roboflow

1:14 0.87 KB/S Vo LTE 5G 73

what is image class...

Image classification is a computer vision task that involves assigning a label to an image based on its contents:

How it works

Image classification analyzes an image at the pixel level to determine the most appropriate label. It uses training data of already labeled images to assign a probability to each class.

Why it's important

Image classification is a fundamental task in computer vision and machine learning. It provides valuable data and insights that can help inform decisions.

How it's performed

Image classification is typically performed using classification networks such as CNNs.

Types of classification

There are two types of image classification: supervised and unsupervised, depending on the interaction between the analyst and

1:16

what are problems...

Some challenges and problems in image classification in computer vision include:

Ethical concerns

Biased training data can lead to biased predictions that reinforce social biases.

Poor lighting

Changes in brightness, shadows, and dark spots can make it difficult for algorithms to recognize objects.

Deformation

In the real world, objects can deform, making it difficult for algorithms to identify them.

Scalability

As complexity increases, it's important to ensure scalability.

Accuracy

Inaccurate or mislabeled data can lead to model errors.

Data quantity

There may be a large number of images.

Data dimensionality

The data may have high dimensionality.

Labeled data

There may be a lack of labeled data.

Variations

There may be variations in lighting, scale, and orientation.

Seeing Through the Machine's Eyes: Top...

Poor Lighting. Changes in

Common Image Cla

22 Jul 2024 -

First classifier: **Nearest Neighbor**

```
def train(images, labels):
  # Machine learning!
  return model
```

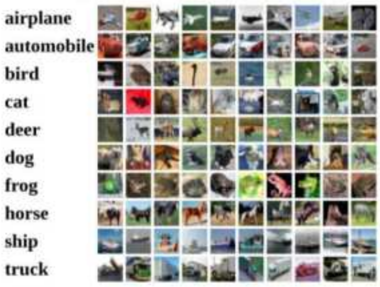
→ Memorize all data and labels

```
def predict(model, test_images):
  # Use model to predict labels
  return test_labels
```

→ Predict the label of the most similar training image

Example Dataset: **CIFAR10**

10 classes
50,000 training images
10,000 testing images



Example Dataset: **CIFAR10**

10 classes
50,000 training images
10,000 testing images

Test images and nearest neighbors

Distance Metric to compare images

L1 distance: $d_1(I_1, I_2) = \sum_p |I_1^p - I_2^p|$

test image

56	32	10	18
90	23	128	133
24	26	178	200
2	0	255	220

\-

training image

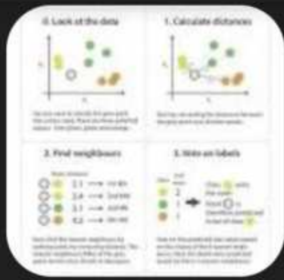
10	20	24	17
8	10	89	100
12	16	178	170
4	32	233	112

=

pixel-wise absolute value differences

46	12	14	1
82	13	39	33
12	10	0	30
2	32	22	108

$\xrightarrow{\text{add}}$ 456



In computer vision, "nearest neighbour classification" refers to a classification technique where an image is assigned a label based on the class of the most similar image (or images) in a training dataset, essentially identifying the "nearest neighbour" to the new image based on a distance metric like pixel intensity comparisons, thus determining its category; this method is often implemented using the "k-Nearest Neighbors (KNN)" algorithm, where "k" represents the number of closest neighbors considered for classification.

Key points about nearest neighbour classification:

Simple concept:

It relies on the idea that similar images

Key points about nearest neighbour classification:

Simple concept:

It relies on the idea that similar images should be close to each other in feature space, allowing new images to be classified based on the class of their closest neighbors.

KNN algorithm:

The most common implementation is the k-Nearest Neighbors algorithm, where "k" determines how many nearest neighbors are considered when making a classification decision.

Distance metric:

To find the "nearest neighbors," a distance metric like Euclidean distance is used to calculate the similarity between images.

Majority vote:

When using multiple nearest neighbors (k > 1), the class label is usually determined by a majority vote among the "k" closest neighbors.

Applications in computer vision:

Image classification:

Identifying the category of an image (e.g., cat, dog, car) based on its

Applications in computer vision:

Image classification:

Identifying the category of an image (e.g., cat, dog, car) based on its similarity to labeled training images.

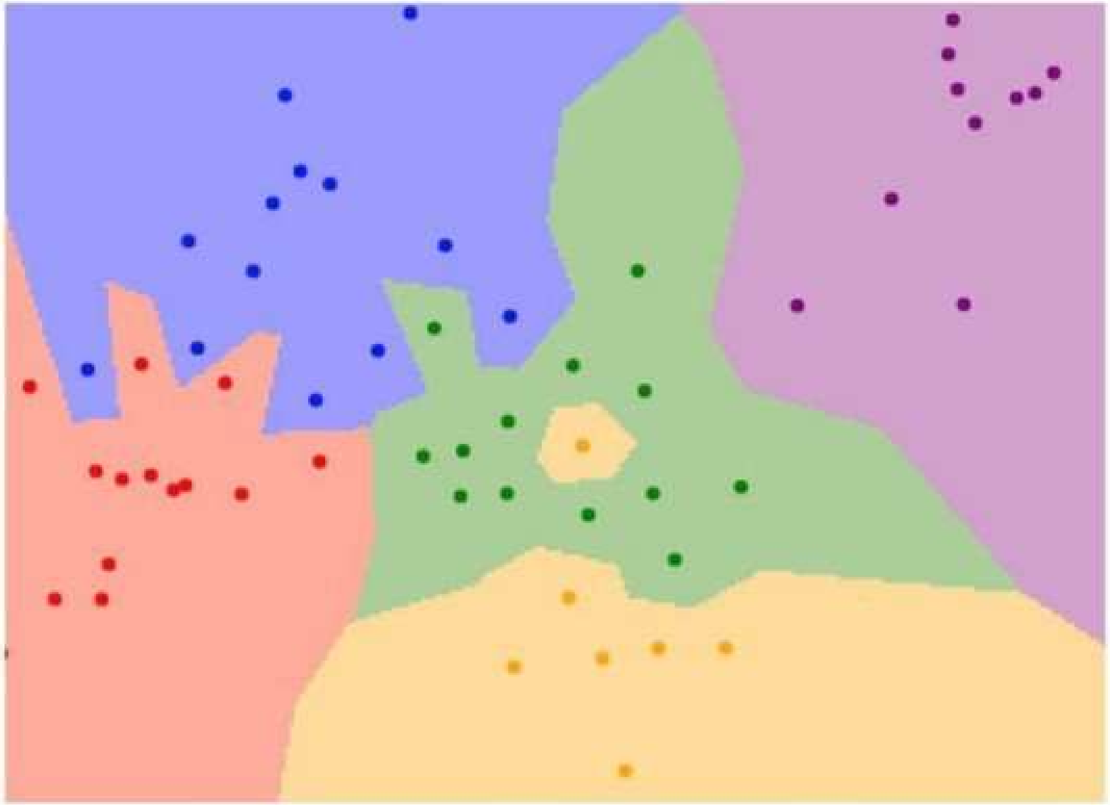
Object detection:

Locating and classifying objects within an image by comparing small image patches to known object templates.

Limitations:

- **Computational cost:** Can be computationally expensive for large datasets as it requires calculating distances to all training images for each new image.
- **Sensitivity to outliers:** Can be sensitive to noise or outliers in the training data.
- **Curse of dimensionality:** Performance can degrade significantly when dealing with

What does this look like?



Hyperparameters

What is the best value of **k** to use?
What is the best **distance** to use?

These are **hyperparameters**: choices about the algorithm that we set rather than learn

Very problem-dependent.
Must try them all out and see what works best.

Setting Hyperparameters

Idea #1: Choose hyperparameters that work best on the data

Your Dataset

Setting Hyperparameters

Idea #1: Choose hyperparameters that work best on the data

BAD: K = 1 always works perfectly on training data

Your Dataset

Setting Hyperparameters

Idea #1: Choose hyperparameters that work best on the data

BAD: K = 1 always works perfectly on training data

Your Dataset

Idea #2: Split data into **train** and **test**, choose hyperparameters that work best on test data

train	test

what is hyperpara...

In computer vision, a hyperparameter is a configuration variable that is manually set before training a machine learning model:

Definition

A hyperparameter is a parameter whose value is set before the machine learning process begins.

Purpose

Hyperparameters define the model's configuration and settings, and guide the learning process.

Examples

Examples of hyperparameters in computer vision include the learning rate, batch size, number of layers, filter sizes, pooling strategies, dropout rates, and activation functions.

Importance

Choosing the right hyperparameters is important because they directly

1:24 40.0 KB/S 71

what is hyperpara...

Importance

Choosing the right hyperparameters is important because they directly impact the performance of the resulting model.

Tuning

Fine-tuning hyperparameters can significantly improve the model's performance. For example, adjusting the learning rate can affect the convergence speed.

Methods

Some methods for hyperparameter tuning include Hyperband and Sequential Model-Based Optimization (SMBO).

What is Hyperparameter Tuning? A Deep Dive. -...
16 Jun 2023 — Hyperparameters define the configuration or...
Roboflow Blog

What is H Tuning? -
Hyperparame configuratio
AWS

Home Search Saved Notifications 1

K-Nearest Neighbors

Instead of copying label from nearest neighbor,
take **majority vote** from K closest points

K = 1 K = 3 K = 5

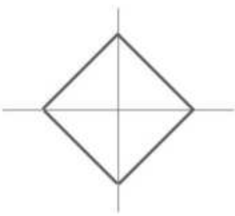
What does this look like?

What does this look like?

K-Nearest Neighbors: Distance Metric

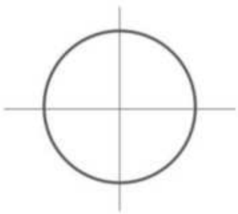
L1 (Manhattan) distance

$$d_1(I_1, I_2) = \sum_p |I_1^p - I_2^p|$$



L2 (Euclidean) distance

$$d_2(I_1, I_2) = \sqrt{\sum_p (I_1^p - I_2^p)^2}$$



K-Nearest Neighbors: Distance Metric

L1 (Manhattan) distance

$$d_1(I_1, I_2) = \sum_p |I_1^p - I_2^p|$$

K = 1

L2 (Euclidean) distance

$$d_2(I_1, I_2) = \sqrt{\sum_p \left(I_1^p - I_2^p\right)^2}$$

K = 1

In computer vision, "K-Nearest Neighbor" (KNN) is a machine learning algorithm used for image classification, where a new image is categorized based on the class labels of the "K" most similar images (its nearest neighbors) from a pre-labeled training dataset, essentially deciding the class of a new image by comparing it to known images that are closest to it in feature space.

Key points about KNN in computer vision:

How it works:

- **Feature extraction:** First, each image in the training set is converted into a feature vector (a set of numerical values representing important

How it works:

- **Feature extraction:** First, each image in the training set is converted into a feature vector (a set of numerical values representing important characteristics like color, texture, shape).
- **Distance calculation:** When classifying a new image, the algorithm calculates the distance between its feature vector and the feature vectors of all images in the training set.
- **K nearest neighbors:** It then identifies the "K" images with the smallest distances (considered the "nearest neighbors").
- **Classification:** The new image is assigned the class label that is most common among its "K" nearest neighbors, usually by majority vote.

1:26 6.00 KB/S Vo LTE 5G 71

what is k nearest n...

"K" value:

The value of "K" is a hyperparameter that needs to be tuned, determining how many neighbors to consider when making a classification decision.

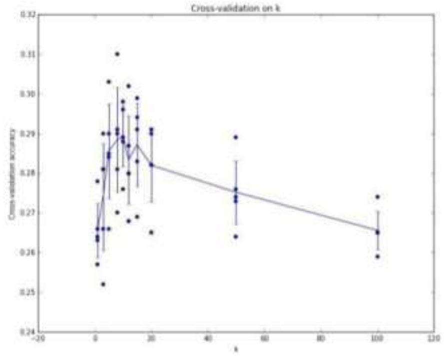
Advantages:

- Simple to implement
- No complex training process needed, as it only stores the training data
- Can be effective for simple classification tasks

Disadvantages:

- Can be computationally expensive for large datasets
- Sensitive to the choice of distance metric
- May not perform well on complex image classification tasks with high dimensionality

Setting Hyperparameters



Example of 5-fold cross-validation for the value of **k.**

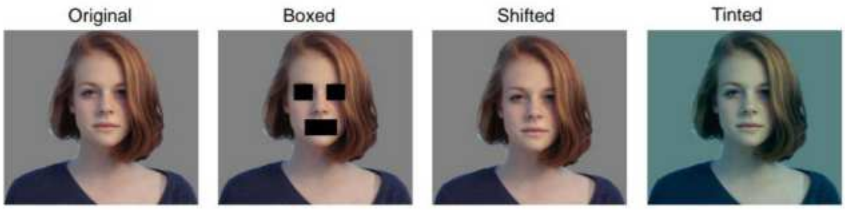
Each point: single outcome.

The line goes through the mean, bars indicated standard deviation

(Seems that k ~= 7 works best for this data)

k-Nearest Neighbor on images **never used.**

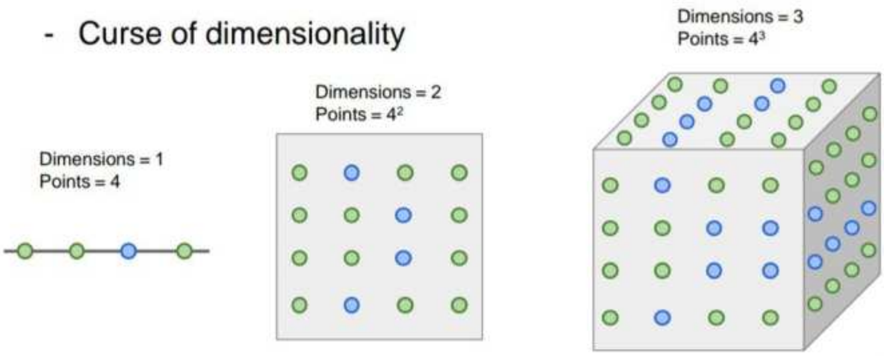
- Very slow at test time
- Distance metrics on pixels are not informative



(all 3 images have same L2 distance to the one on the left)

k-Nearest Neighbor on images **never used.**

- Curse of dimensionality



K-Nearest Neighbors: Summary

In **Image classification** we start with a **training set** of images and labels, and must predict labels on the **test set**

The **K-Nearest Neighbors** classifier predicts labels based on nearest training examples

Distance metric and K are **hyperparameters**

Choose hyperparameters using the **validation set**; only run on the test set once at the very end!

what is linear clad...

In computer vision, "linear classification" refers to a method of categorizing images into different classes by using a linear function to separate the data points, essentially drawing a straight line (or hyperplane in higher dimensions) to distinguish between different classes, where each side of the line represents a different category; it's a simple classification technique that works well when the data is easily separable by a linear boundary, making it easy to interpret and computationally efficient.

Key points about linear classification:

Decision boundary:

The line or hyperplane that separates the classes is called the decision boundary.

Key points about linear classification:

Decision boundary:

The line or hyperplane that separates the classes is called the decision boundary.

Feature vector:

Each image is represented as a vector of features (like pixel values) which are used for classification.

Weight vector:

The linear classifier learns a set of weights that are applied to each feature to determine the class.

Linear combination:

The classification decision is made by calculating a weighted sum of the features, which is essentially a linear combination.

Example algorithms using linear classification:

Example algorithms using linear classification:

- **Perceptron:** A basic linear classifier that updates weights based on misclassified data
- **Logistic Regression:** A probabilistic linear classifier that outputs the probability of belonging to a certain class
- **Linear Discriminant Analysis (LDA):** A dimensionality reduction technique that aims to find the linear combination of features that best separates classes

Limitations of linear classification:

Non-linear data:

If the data cannot be easily separated by a straight line (non-linearly separable), a linear classifier will not perform well.

Complex patterns:

For complex image recognition tasks where features are not linearly related, more advanced non-linear classifiers are often needed.

Linear Classifier - an overview | ScienceDire...

Linear vs. Classifica

28 Jun 2024

1:30 7.00 KB/S 70

what is parametric...

In computer vision, a "parametric approach" in linear classification refers to a method where the classification model is defined by a fixed set of parameters (like weights and biases) that are learned from training data, allowing for efficient prediction by simply applying a linear function to the input image features, unlike non-parametric methods which might require storing the entire training data for comparison; essentially, it means the model assumes a specific form (like a linear equation) to make predictions, with the key focus being on optimizing those parameters to best fit the data.

Key points about parametric linear classification:

Fixed model structure:

Key points about parametric linear classification:

Fixed model structure:

Unlike non-parametric methods, a parametric linear classifier has a predefined structure, typically a linear combination of input features with learned weights and biases.

Parameter learning:

The core process involves training the model by adjusting the parameters (weights and biases) to minimize the error on the training data.

Fast prediction:

Once trained, making predictions on new data is fast as it only requires a simple matrix multiplication with the learned parameters.

Example:

- **Logistic Regression:** A common parametric linear classifier where the input features are linearly

Example:

- **Logistic Regression:** A common parametric linear classifier where the input features are linearly combined with weights, then passed through a sigmoid function to output a probability for a binary classification task.

Benefits of parametric approach:

- **Computational efficiency:** Faster prediction due to the simple mathematical operations involved.
- **Interpretability:** The weights in a linear model can be interpreted as the importance of each feature.

Limitations of parametric approach:

Assumption of linearity:

May not perform well if the relationship between features and class labels is not linear.

Limited complexity:

Might struggle with complex data that requires more intricate relationships between features.

Representing Digital Images

TABLE 2.1
Number of storage bits for various values of N and k.

N/k	1 (L = 2)	2 (L = 4)	3 (L = 8)	4 (L = 16)	5 (L = 32)	6 (L = 64)	7 (L = 128)	8 (L = 256)
32	1,024	2,048	3,072	4,096	5,120	6,144	7,168	8,192
64	4,096	8,192	12,288	16,384	20,480	24,576	28,672	32,768
128	16,384	32,768	49,152	65,536	81,920	98,304	114,688	131,072
256	65,536	131,072	196,608	262,144	327,680	393,216	458,752	524,288
512	262,144	524,288	786,432	1,048,576	1,310,720	1,572,864	1,835,008	2,097,152
1024	1,048,576	2,097,152	3,145,728	4,194,304	5,242,880	6,291,456	7,340,032	8,388,608
2048	4,194,304	8,388,608	12,582,912	16,777,216	20,971,520	25,165,824	29,369,128	33,554,432
4096	16,777,216	33,554,432	50,331,648	67,108,864	83,886,080	100,663,296	117,440,512	134,217,728
8192	67,108,864	134,217,728	201,326,592	268,435,456	335,544,320	402,653,184	469,762,048	536,870,912

Spatial and Intensity Resolution

- Spatial resolution
  - — A measure of the smallest discernible detail in an image
  - — stated with *line pairs per unit distance, dots (pixels) per unit distance, dots per inch (dpi)*
- Intensity resolution
  - — The smallest discernible change in intensity level
  - — stated with *8 bits, 12 bits, 16 bits, etc.*

Spatial and Intensity Resolution

a b
c d

FIGURE 2.20 Typical effects of reducing spatial resolution. Images shown at: (a) 1250

Representing Digital Images

The representation of an M×N numerical array in MATLAB

$$f(x,y)=\begin{bmatrix} f(1,1) & f(1,2) & \dots & f(1,N) \\ f(2,1) & f(2,2) & \dots & f(2,N) \\ \dots & \dots & \dots & \dots \\ f(M,1) & f(M,2) & \dots & f(M,N) \end{bmatrix}$$

Representing Digital Images

- Discrete intensity interval [0, L-1], $L=2^k$

The number b of bits required to store a M × N digitized image

b = M × N × k

Representing Digital Images

TABLE 2.1
Number of storage bits for various values of N and k.

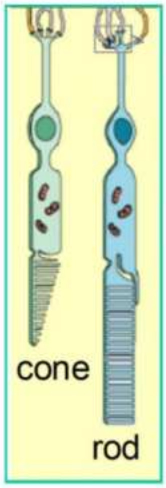
N/k	1 (L = 2)	2 (L = 4)	3 (L = 8)	4 (L = 16)	5 (L = 32)	6 (L = 64)	7 (L = 128)	8 (L = 256)
32	1,024	2,048	3,072	4,096	5,120	6,144	7,168	8,192
64	4,096	8,192	12,288	16,384	20,480	24,576	28,672	32,768
128	16,384	32,768	49,152	65,536	81,920	98,304	114,688	131,072
256	65,536	131,072	196,608	262,144	327,680	393,216	458,752	524,288
512	262,144	524,288	786,432	1,048,576	1,310,720	1,572,864	1,835,008	2,097,152
1024	1,048,576	2,097,152	3,145,728	4,194,304	5,242,880	6,291,456	7,340,032	8,388,608
2048	4,194,304	8,388,608	12,582,912	16,777,216	20,971,520	25,165,824	29,369,128	33,554,432
4096	16,777,216	33,554,432	50,331,648	67,108,864	83,886,080	100,663,296	117,440,512	134,217,728
8192	67,108,864	134,217,728	201,326,592	268,435,456	335,544,320	402,653,184	469,762,048	536,870,912

Light

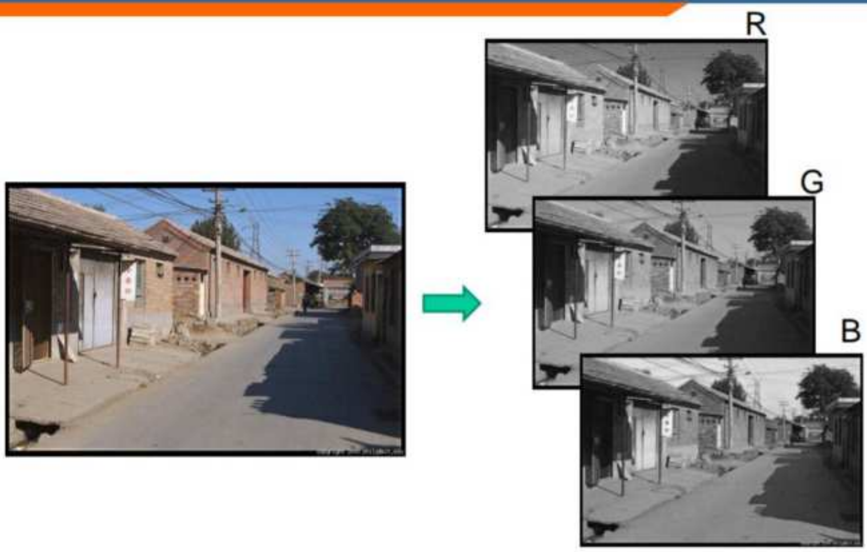
Two types of light-sensitive receptors

Cones
- cone-shaped
- less sensitive
- operate in high light color vision

Rods
- rod-shaped
- highly sensitive
- operate at night
- gray-scale vision
- slower to respond

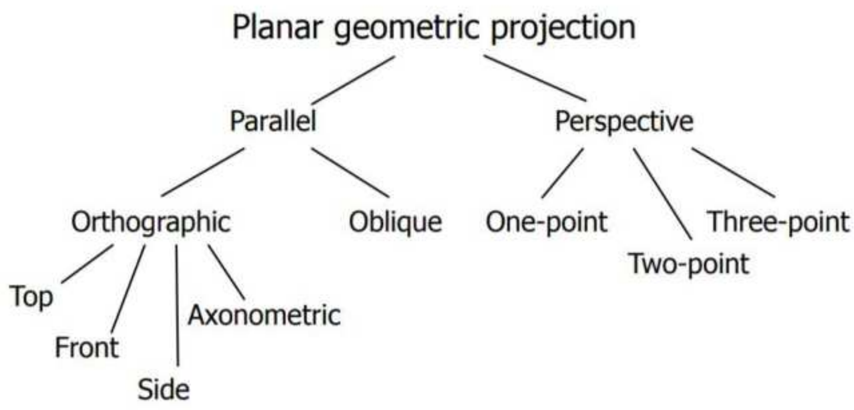


Color Image



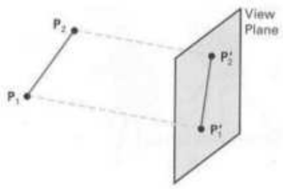
Thank You

Taxonomy of Projections

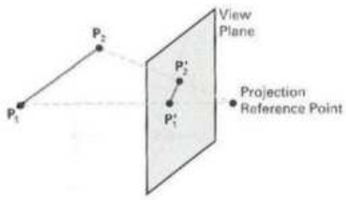


Parallel & Perspective

Parallel Projection

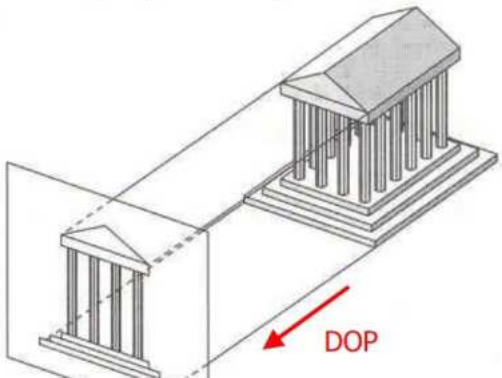


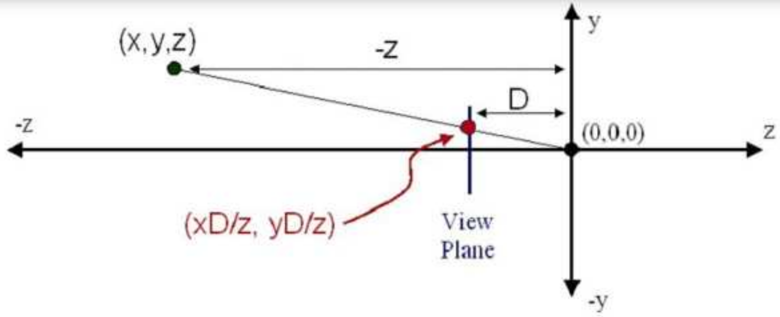
Perspective Projection



Parallel Projection

- Center of projection is at infinity
  - Direction of projection (DOP) same for all points

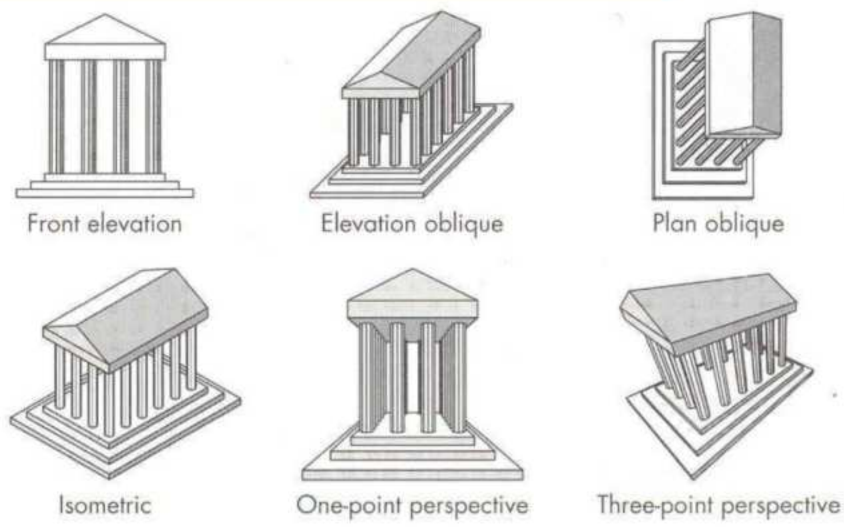




Perspective vs. Parallel

- Perspective projection
  - + Size varies inversely with distance – looks realistic
  - – Distance and angles are not(in general) preserved
  - – Parallel line do not (in general) remain parallel
- Parallel projection
  - + Good for exact measurements
  - + Parallel lines remain parallel
  - – Angles are not (in general) preserved
  - – Less realistic looking

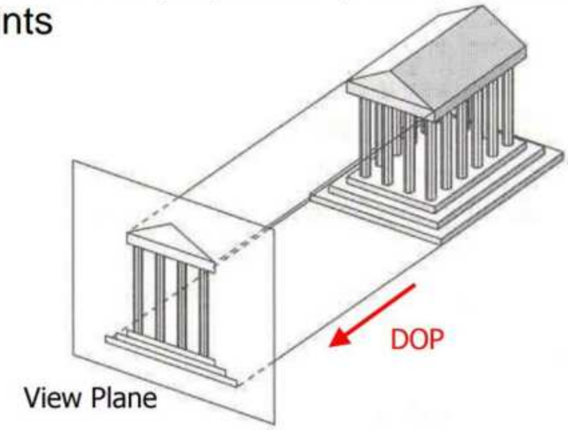
Classical Viewing



(Thank You)

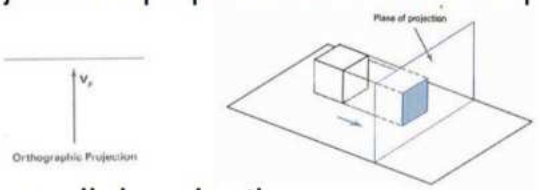
Parallel Projection

- Center of projection is at infinity
  - Direction of projection (DOP) same for all points

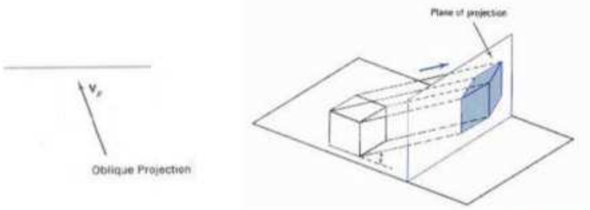


Orthographic & Oblique

- Orthographic parallel projection
  - the projection is perpendicular to the view plane



- Oblique parallel projection
  - The projectors are inclined with respect to the view plane



Orthographic Projections

- DOP perpendicular to view plane