UNIT-4

Binarization:- It is thresholding technique in image processing that simplifies an image by converting its grayscale pixels value (ranging from 0 to 255) into Binary values. It

* It is widely used as preprocessing step in tasks like OCR, document scanning & Object segmentation

* Main goal is to separate foreground from background
* Its based on idea that important visual information is captured only 2 levels- on/off, black/white

Formula

Let $I(x,y)$ be the grayscale intensity at pixel (x,y). Let T be the threshold

The Binary Image $B(x,y)$ be given by-

$$B(x,y) = \begin{cases} 1 & \text{if } I(x,y) > T \\ 0 & \text{otherwise} \end{cases}$$

Types of Binarization

1) Global Thresholding
↳ A single threshold T is applied to the whole image

* Simple and fast
* Not suitable for image c uneven lighting or shadows

Formula:- $B(x,y) = \begin{cases} 1 & \text{if } I(x,y) > T \\ 0 & \text{if } I(x,y) \leq T \end{cases}$

Adaptive Thresholding
* The threshold $T(x,y)$ is computed locally for each pixel based on intensity of neighbouring pixels
* Better for Non-uniform illumination shadows

Common approaches
* Mean Adaptive Thresholding $T(x,y) \equiv$ av surrounding pixels
* Gaussian Adaptive Thresholding - weighted using a Gaussian window

Otsu's Method

→ An automatic global thresholding that minimizes intra-class va maximizes inter class varianc

2] Basic set theory

1) what is set (first Principles)

↳ A set is a fundamental concept in mathematics - its collection of distinct Objects.

↳ Sets r usually written as:-

$$A = \{a, b, c, d\}$$

meaning set A containing elements a, b, c, d

- The order doesnot matter in a set duplicates are not counted.
- If an element x belongs to set A, we write :

$$x \in A$$

- If x doesn't belong to A, we write:

$$x \notin A$$

2] Why set theory is important in Image Processing

Images are made up of pixels can be considered as elem. in a set. Many image processing operations can be designed / described as ops on set of pixels

eg :- foregrd pixels of binary image can be viewed as a set A

* Bgrd pixel form another set B

Many ops like morphological ops, region analysis r based on manipulating these pixel

a) Union : In img processing if A & B r set of pixel then union reps comb. of pixels in both regions

b) Intersecn :- common pixels of both sets - like overlap of 2 shapes

c) Difference (A - B)
↳ contains elem in A & not in B
A-B ↳ subtracting pixel from one shape from other

d) Compliment (A^C)
↳ The compliment of A cont all elem not in
↳ compliment of foregrd is bgrd & vice versa

4) Set Theory & Binary Images

~~Morphology~~

* Consider a binary image as set of points

$A = \{(x, y) : \text{pixel at } (x,y)$

A cont. all foregrd pixels
Backgrd pixels are the compliment of A

5 : Erosion: Shrinks foreground set A by B

Dialation : Expands foregrd set By B

6 Maths of Morphological Ops.

$$A \ominus B = (z : B_z \subseteq A)$$

B_z is B translated by z

- Dialatn of set A by B

$$A \oplus B \quad z : (B^{ref})_z \cap A \neq \emptyset$$

B^{ref} = ref of B

Intuitive Explantn

~~* Imagine the image~~

Set A ≡ all points where obj exist
Set B ≡ Small shape

Erosion

Morphology means: study of shape & structure

In image processing, it refers to operations that process images based on shapes

Binary Image = set of foreground (white) pixels

eg $A = \{(x,y) : \text{pixel at } (x,y) = 1\}$

Erosion :-

Goal:- Shrink white regions, Remove boundary p
useful for removing noise.

Intuition:- Maths: $A \ominus B = \{z \in E : B_z \subseteq A\}$

* Erosion never adds pixel
* $\boxed{A \ominus B \subseteq A}$

2 Dialation

Grow the white region add boundar
pixel useful for bridging gap

$A \oplus = \{z \in E : (B^{ref})_z \cap A \neq \emptyset$

* Dilation never reduce white pixels
$A \subseteq A \oplus B$

3 Opening $(A \circ B)$ (7)

Date / /
Page No:

Goal :- Remove small white noise while preserving main shape.

* first erode :- shrink all white obj
Then dialate :- restore them (big ones would come back)

Operation :-

$$A \circ B = (A \ominus B) \oplus B$$

Use Case :-
- Smooth Boundaries
- Remove small blobs
- Disconnect thin Bridges

~~Prop:~~ ~~Res~~

Closing $A \bullet B$
↳ fill small Blackholes inside region

first dialate :- expand all white objects
Then erode :- shrink them back

$$A \bullet B = (A \oplus B) \ominus B$$

Use :- (1) fill fingrale
(2) Smooth boundaris

#9| COLOR IMG REPRESENTN

→ A color image is not just matrix of no its per pixel desc of how human perciéve color. But computers can't see color. So we represent color models which break down each pixel's color into componnts

1] RGB (Red Green Blue)

Perception of color :-

Our eye ~~is sensitive~~ has 3 types of cone cells, each sensitive to :-

- Red (R)
- Green (G)
- Blue (B)

#1 RGB model)

→ Each pixel is stored as a triplet (R, G
Each R G B range rom 0 to 255 (i

Red = (255, 0, 0)
Green = (0, 255, 0)
Blue = (0, 0, 255)
White = (255, 255, 255)
Black = (0, 0, 0)

③

HSV (Hue Sat Value)

More human intuitive color

Hue → color type (0-360°)

Saturation :- intensity / purity (0 = gray, 1 = full color)

Value = brightness (0 = black, 1 = bright)

RGB ⟷ HSV :

$$V = \max(R, G, B)$$

$$S = \begin{cases} 0 & \text{if } V = 0 \\ \dfrac{V - \min(R,G,B)}{V} \end{cases}$$

~~H = { 0 if~~

$$H = \begin{cases} 0 & \text{if } \Delta = 0 \\ 60 \times \left(\dfrac{G-B}{\Delta} \% 6\right) & V = R \\ 60 \times \left(\dfrac{B-R}{\Delta} + 2\right) & V = G \\ 60 \times \left(\dfrac{R-G}{\Delta} + 4\right) & V = B \end{cases}$$

$$\Delta = V - \min(R, G, B)$$

Mathematical model

RGB

↳ Memory = 3 * width * Height * 8 bits

RGB → Grayscale

$$Gray = 0.2989 \cdot R + 0.5890 \cdot G + 0.1140 \cdot$$

eg $R = 0.8 \quad G = 0.6 \quad B = 0.2$

$$C_{max} = \max(R, G, B) = 0.8$$

$$C_{min} = \min(R, G, B) = 0.2$$

$$\Delta = C_{max} - C_{min} = 0.8 - 0.2 = .6$$

$$\boxed{V = C_{max}}$$

$$S = \begin{cases} 0 & \text{if } C_{max} = 0 \\ \dfrac{\Delta}{C_{max}} = \dfrac{0.6}{0.8} = 0.75 \end{cases}$$

$$Hue = \begin{cases} 0 & \text{if } \Delta = 0 \\ 60 * \left(\dfrac{G-B}{\Delta} \% 6\right) & C_{max} = R \\ 60 \left(\dfrac{B-R}{\Delta} + 2\right) & C_{max} = G \\ 60 * \left(\dfrac{R-G}{\Delta} + 4\right) & B \end{cases}$$

$$*\left(\frac{0.4}{0.6}\right) \% 6$$

$$\left(\frac{2}{3} \% 6\right)$$

$$0$$

Color Image Processing

- color image processing is branch of image processing that handles image in color not just gray scale
- Gray scal has just one channel (just intensity)
- A color imag typically has 3 channels. RGB, HSV etc.

Each pixel carries more information - not just how bright it is what color it is

Why Color?

- color provides more info. to distinguish objects, regions, & textures.
- you can't ripe tomato from a green one in grayscale.

Basic Processes in CIP

↳ Color enhancement refers to improving appearance, visual perception or interpretation of a color image by adjusting its color related features

↳ Real world images suffer from :-

- poor lightning
- lower contrast
- under exposure / over exposes
- color bias

Techniques

① Contrast & Brightness Adjustment

$$I'(x,y) = \alpha I(x,y) + \beta$$

$I(x,y)$ = original pixel value at position (x,y) for RGB

$\alpha > 1$ — Inc. contrast

β = control brightness

Contrast = how different r from each color

Date: / /
Page No.

② Histogram Equalization

Theory
- Redistributes pixels intensities to flatten the histogram & increase contrast
- Applied separately to each of R, G, B

Problem
- Doing this independently might shift color unnaturally

cdf (i) = cumulative distribution fxn of pixel histogram

$$\text{eq. value} = \frac{cdf(i) - cdf_{min}}{(M \times N) - cdf_{min}} \times (L-1)$$

M×N = image size
L = no. of intensity levels

(3) Saturation Adj (HSL or HSV)
$S \uparrow\uparrow$ → color pop more
Decreas S

(4) Colour Normalization

(1) compute mean of R, G, B
μ_R, μ_G, μ_B

(2) find global averay
$$\mu_{gray} = \frac{\mu_R + \mu_G + \mu_B}{3}$$

$$R' = R \cdot \frac{\mu_{gray}}{\mu_R}$$

$$G' = G \cdot \frac{\mu_{gray}}{\mu_g}$$

$$B' = B \cdot \frac{\mu_{gray}}{\mu_{Blue}}$$

(5) Gamma correctn

$$I'(x,y) = I(x,y)^{\gamma}$$

$\gamma < 1$:- lightens img
$\gamma > 1$ Darkens

Color Restoration : Correcting color distortions due to noise
or color Balance

Color Seg : Partitioning of an image into regions based on color similarity.

Spatial filtering

Every Image is 2D fxn of intensity.
In digital images, intensity values are sampled and quantised into pixels.

A spatial filter modifies the intensity of each pixel by taking account of the pixels neighbourhood.

Steps for spatial filtering

1) choose filter kernel eg (3×3 matrix)
2) slide over kernel over each pixel
3) At each position compute sum of products of kernal values and pixel values

This operatn is called convolution or correlation

$\rightarrow g(x,y)$

For gray scale image

$$g(x,y) = \sum_{i=-K}^{K} \sum_{j=-K}^{K} f(x+i, y+j) \cdot h(i,j)$$

$f(x,y)$ = original image
$h(i,j)$ = filter mask or kernel
$g(x,y)$ = output image.

$$g_c(x,y) = \sum_{i=-K}^{K} \sum_{j=-K}^{K} f_c(x+i, y+j) h(i,j)$$

$c \in (R, G, B)$

~~Types of Spatial Filtering~~

3) Filtering in RGB Vector Space

$$p'(x,y) = [R(x,y), G(x,y), B(x,y)]$$

- vector mean (averaging RGB vector)
- Vector median (reducing noise)
- Euclidean dist for filtering

Working with RGB Images Vector

Digital Image are typically represented in RGB model i.e each pixel having 3 values R, G, B

$$\vec{p} = [R, G, B]^T$$

→ Instead of processing R, G, B indirectly we process them jointly as a vector this is what is meant by "working in RGB vector space"

Vector Rep = $P_i = \begin{bmatrix} R \\ G \\ B \end{bmatrix}$

$$\|\vec{P_i}\| = \sqrt{R_i^2 + G_i^2 + B_i^2}$$

$$D(P_i, P_j) = \|P_i - P_2\|$$

Ange (color similarity based on directn)

Types of Vector Based filtering

A) Vector mean filter
used for smoothing or blurring

$$\vec{P}_{mean} = \frac{1}{N} \sum_{i=1}^{N} P_i$$

- each pixel is averaged out over a neighbourhood.
- result is new RGB filter that replaces central pi

B) Vector Median filter.

$$\vec{P}_{median} = \arg\left(\min \sum_{i=1}^{N} (P_i - P_j)\right)$$