Module 4: Image Segmentation & Representation

★ Image Segmentation

Image Segmentation divides an image into regions that are connected & have some similarity within the region and some difference between adjacent regions.

★ Approaches to Segmentation

1] Discontinuity — point, line, edge
2] Similarity — Threshold, Region based

1] Discontinuity

Sudden change in boundary line, ~~especially~~ called an edge is referred to as discontinuity.

i] Point Detection

Mask =>
$$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}.$$

ii] Line detection — find one pixel wide line :- horizontal
vertical
diagonal (+45°/-45°)

Line Masks :-

$$\begin{bmatrix} -1 & -1 & -1 \\ 2 & 2 & 2 \\ -1 & -1 & -1 \end{bmatrix} \quad \begin{bmatrix} -1 & 2 & -1 \\ -1 & 2 & -1 \\ -1 & 2 & -1 \end{bmatrix} \quad \begin{bmatrix} -1 & -1 & 2 \\ -1 & 2 & -1 \\ 2 & -1 & -1 \end{bmatrix} \quad \begin{bmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \end{bmatrix}$$

Horizontal   Vertical   +45°   −45°

iii] Edge detection.

1] First - order derivatives.

$$\text{Gradient} = \nabla f = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix} = \begin{bmatrix} G_x \\ G_y \end{bmatrix}$$

; Magnitude $= mag(\nabla f) = \sqrt{G_x^2 + G_y^2}$

Direction :- $\alpha(x,y) = \tan^{-1}\left(\frac{G_x}{G_y}\right)$

Robert :- $\underset{G_x}{\begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}} \quad \underset{G_y}{\begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}}$

suitable for small images, detects fine edges but sensitive to noise.

Prewitt :- $\underset{G_x}{\begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}} \quad \underset{G_y}{\begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}}$

Detects smooth edges but misses finer details, better noise suppression.

Page No.
Date

Diagonal Operators :- $\begin{bmatrix} 0 & 1 & 1 \\ -1 & 0 & 1 \\ -1 & -1 & 0 \end{bmatrix}$ $\begin{bmatrix} -1 & -1 & 0 \\ -1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix}$

Sobel :- $\begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$ $\begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$
G_x G_y

Detects sharp edges, preserves noise details, uses weighted average, better for natural images or raw data.

Diagonal Operators :- $\begin{bmatrix} 0 & 1 & 2 \\ -1 & 0 & 1 \\ -2 & -1 & 0 \end{bmatrix}$ $\begin{bmatrix} -2 & -1 & 0 \\ -1 & 0 & 1 \\ 0 & 1 & 2 \end{bmatrix}$

2] Second-order derivatives.

Laplacian :- detects fine edges.

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} \quad \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix}$$

Gaussian function :- $h(r) = -e^{\frac{-r^2}{2\sigma^2}}$
(blurs smoothly) $r^2 = x^2 + y^2$.

LoG :- $\nabla^2 h(r) = -\left[\frac{r^2 - \sigma^2}{\sigma^4}\right] e^{-\frac{r^2}{2\sigma^2}}$

Why LoG? → Gaussian smoothens the image
Laplacian finds ~~at~~ out edges.

Page No.
Date

★ Edge Linking.

After detecting edges through different methods, we wont always get a perfect connected edge, there might be broken or disjoint edges.

Two properties :- ① Magnitude of edge
② Directions
points are useful for edge linking.

Similar magnitude & direction are linked.

★ Canny Edge Detection.

Canny Edge detection is a multi-stage image processing technique used to identify & extract edges in an image.

It aims to find boundaries of objects within an image by detecting abrupt changes in pixel intensity.

Steps :-

1] Smoothens image with 2D Gaussian.
$\underline{G * I}$

2] Gradient calculations and directions

3] Compute edge magnitudes

4] Non - maximum supression.

* Hough Transform

A way of finding edge points in an image that lies along a straight line.

$$y = mx + c \quad \text{—(i)}$$

① Substitute the given points in eq (I)

② Calculate for m when $c = 0$
" c when $m = 0$.

③ Plot (m, c) on the graph.

④ Point of intersection is the final (m, c).

However, $y = mx + c$, won't be able to detect vertical lines, because slope is undefined for vertical lines.

To overcome this issue, we use a normal line passing through origin and perpendicular to straight line :-

$$p = x\cos\theta + y\sin\theta.$$

Page No.
Date

2] Similarity

★ Thresholding :- separates objects from BG based on pixel intensity.

$$g(x,y) = \begin{cases} 1 & \text{if } f(x,y) > T \\ 0 & \text{if } f(x,y) \leq T \end{cases}$$

1] Basic Global Thresholding

Only one threshold value for the full image.
Works only if lighting is even & BG and object have clear intensity separation.

2] Basic Adaptive Thresholding

Different threshold for each pixel based on its local region, adapts according to the different intensities of the object. Works when lighting is uneven.

✱ Region Based Segmentation:-

Based on connectivity of similar pixels in a region, each region must be uniform.

4 major approaches:-

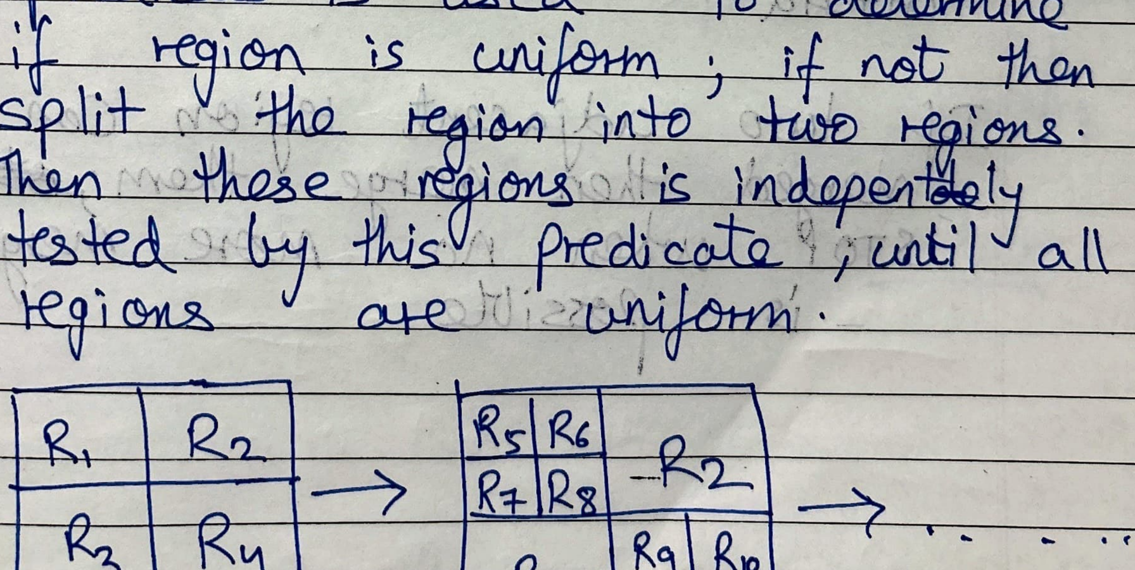
1] Region Growing

Groups pixels or subregions into larger regions.

Starts from seed pixels & adds neighbours with similar properties.

2] Region Splitting

Opposite of region growing, a predicate is used to determine if region is uniform; if not then split the region into two regions. Then these regions is indepentely tested by this predicate; until all regions are uniform.



Predicate :- max − min ≤ Threshold.

3] Region Merging

If the 2x2 matrix satisfy the homogeneity property

Yes, if they satisfy the predicate condition, then merge.

Eg:- Max - Min ≤ threshold : if yes, then merge.

4] Region Splitting & Merging

If region satisfies homogeneity criteria, ~~it~~ then unmodified.

If not then split the regions & recursively apply this step. STOP when all match the homogeneity criteria.

If two adjacent region can be merged then merge them; ~~or~~ STOP when no more merging is possible.

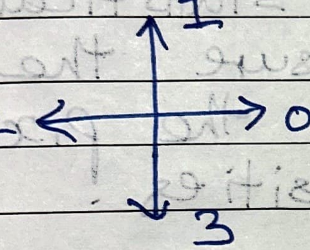
~~$x(k)$:- 1, 2, 3, 0~~
~~$h(n) = \{-1, -1, 0, 0$~~

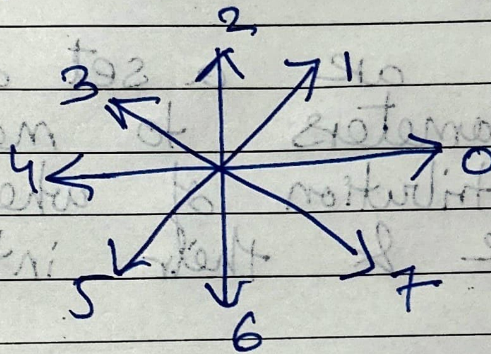
~~$x(k) = \{2, 3, 4, 5\}$~~
~~$h(k) = \{0, 1, 1, 0\}$~~
~~$y(k) = \{2\times 0, 3\times 1, 4\times 1, 5, 0\}$~~
~~$= \{0, 3, 4, 0$~~

★ Boundary Descriptors

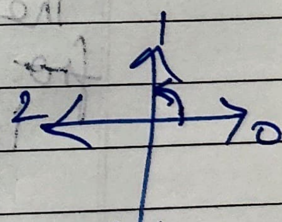
1] Chain Codes

Represent an object boundary by a connected sequence of straight lines. segments of specified length & direction.





① Find chain code.
② Calculate chain first difference



③ Then normalise the difference.
& also called shape number.

Page No.
Date

★ Signature.

A simple functional representation that can be used to describe & reconstruct the boundary with appropriate accuracy.

Simple way – to plot the distance from the centroid.

★ Image Moments

They are a set of statistical parameters to measure the distribution of where the pixels are & their intensities!

The image moment M_{ij} of order (i,j) ~~fro~~ for grayscale with pixel intensities $I(x,y)$ will be :-

$$M_{ij} = \sum_x \sum_y x^i y^i I(x,y)$$

Binary :- $M_{00} = \sum_x \sum_y I(x,y)$.

Corresponds to count of all non-zero pixels

In terms of Image Moments, Centroid is :-

$$\text{Centroid } (\bar{x}, \bar{y}) = \left\{ \frac{M_{10}}{M_{00}}, \frac{M_{01}}{M_{00}} \right\}.$$