Date:
Page No.

~~30~~ 01/Oct/2019.

→. Image segmentation

Done using two approaches :-
(i). Discontinuity Based Approach
(ii) Similarity Based Approach.

① Sub-Division is carried out based on abrupt changes in gray level of image.
②. We try to group pixels which are similiar in some sense.

① Discontinuity Based.
- Point Detection
- Edge detection
- Line detection

② Similarity Based.
- Threshold operation
- Region Growing
- Region Splitting / Merging.

→ Region Growing :- We start from a pixel & then we group all other pixels connected to this pixel which are adjacent and similiar in intensity values.

→ Region Splitting & Merging :- In this, we split the image into numbers of different components & then merge the similiar components.

# Point Detection

| | | |
|---|---|---|
| -1 | -1 | -1 |
| -1 | 8 | -1 |
| -1 | -1 | -1 |

Mask.

If $|R| > T$

Apply Mask on pt. $(x,y)$ say it is $R(x,y)$

If $|R| > T$ (Threshold, then $(x,y)$ is considered as a pt.

# Line Detection

Horizontal

| | | |
|---|---|---|
| -1 | -1 | -1 |
| 2 | 2 | 2 |
| -1 | -1 | -1 |

(mask)

Rotate 45°

| | | |
|---|---|---|
| -1 | -1 | 2 |
| -1 | 2 | -1 |
| 2 | -1 | -1 |

Verticle

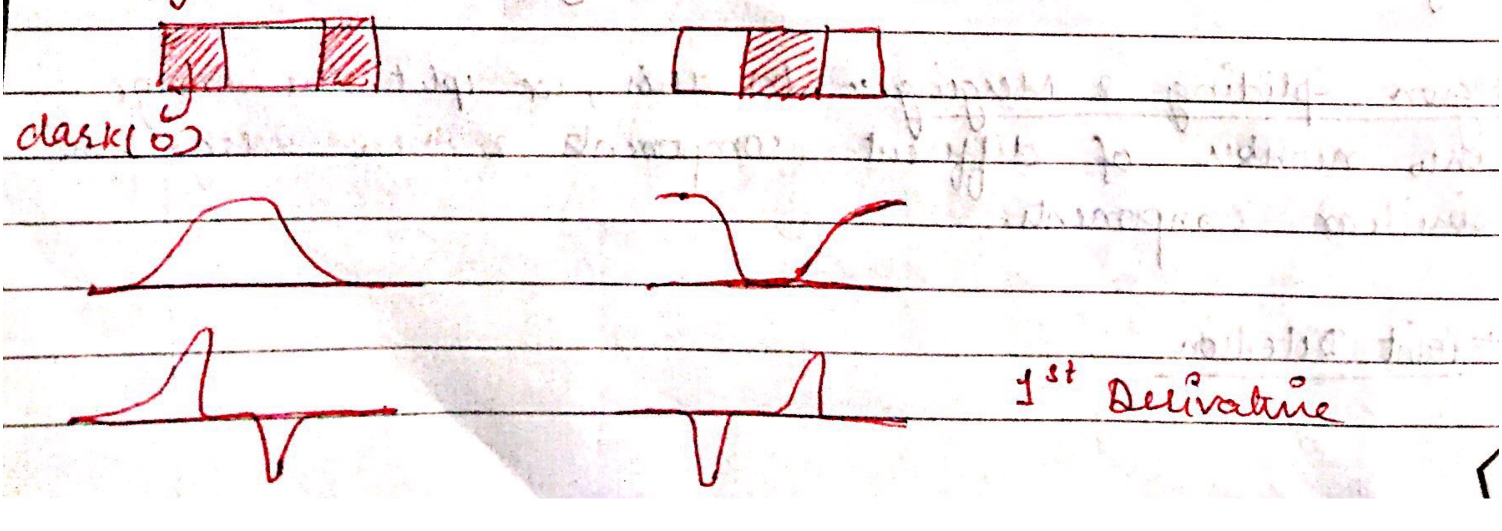
| | | |
|---|---|---|
| -1 | 2 | -1 |
| -1 | 2 | -1 |
| -1 | 2 | -1 |

Rotate -45°

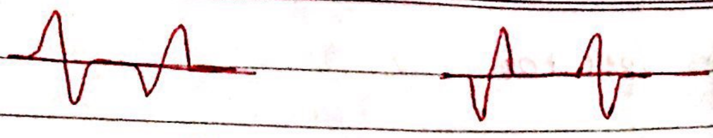
| | | |
|---|---|---|
| 2 | -1 | -1 |
| -1 | 2 | -1 |
| -1 | -1 | 2 |

We apply all four masks on an image. For two different masks i & j, where $i \neq j$ if $R_i > R_j$, then this means the corresponding pt is more likely to lie in the direction of mask i, rather than j

# Edge Detection



Date:
Page No.



2nd Derivative

We use 1st derivative for edge detection :- 2nd Derivative is very sensitive.

We use gradient operator :-

$$\vec{\nabla} f = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \partial f / \partial x \\ \partial f / \partial y \end{bmatrix}$$

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

$$\simeq |G_x| + |G_y|.$$

The magnitude tells the strength of edge at $(x, y)$.

⟹ <u>Direction of edge</u> :-

$$\angle(x, y) = \tan^{-1}\left(\frac{G_y}{G_x}\right).$$

# <u>Perwitt Edge Operator</u>

| | | |
|---|---|---|
| -1 | -1 | -1 |
| 0 | 0 | 0 |
| +1 | +1 | +1 |

Horizontal
$(G_x)$

| | | |
|---|---|---|
| -1 | 0 | +1 |
| -1 | 0 | +1 |
| -1 | 0 | +1 |

Verticle
$(G_y)$

# <u>Sobel Edge Operator</u> :-

| | | |
|---|---|---|
| -1 | -2 | -1 |
| 0 | 0 | 0 |
| 1 | 2 | 1 |

| | | |
|---|---|---|
| -1 | 0 | 1 |
| -2 | 0 | 2 |
| -1 | 0 | 1 |

Effect due to spurious noise is taken care to some extent

Scanned with CamScanner

using the sobel edge operator

# Edge Linking :-
① Local Processing ( rigid only)
② Global Processing.

In edge linking, all points that are similar in nature are linked. This forms a boundary of pixel that are similar

Similarity of pixel is based on
① strength of response and
② Direction of Gradient.

Pixels $(x,y)$ & $(x'y')$ are similar if :- ← Local Processing

$$|\nabla f(x,y) - \nabla f(x',y')| \le T_c.$$

where $T_c$ is some threshold value.

&

$$|\alpha(x,y) - \alpha(x',y')| < A.$$

$$(x,y) \in N(x',y')$$

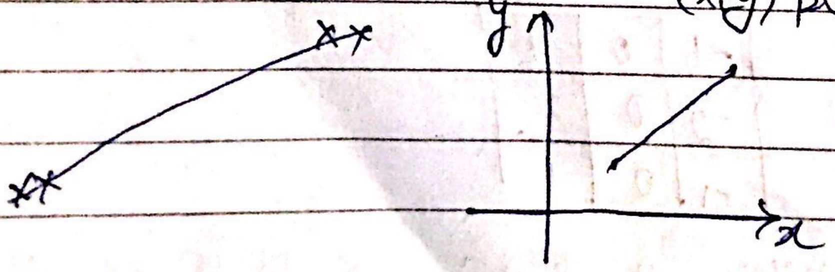
A is angle.

03/oct/2019.

## Edge Linking

Global Processing (Hough Transform)

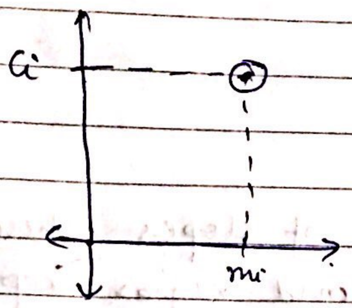
Transformation from spatial Domain → Parameter
(x,y) plane Space (m,c) plane



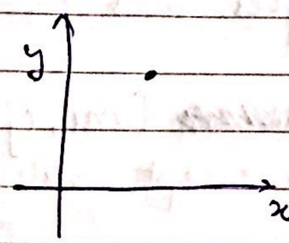
$$y = \underbrace{m}_{\text{slope}}x + \underbrace{c}_{\text{intercept}}$$

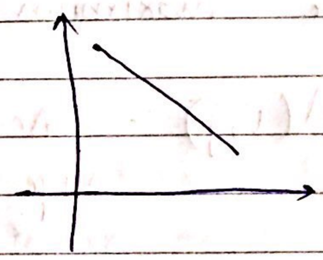
$m_i$ $c_i$

$$\boxed{y = m_i x + c_i}$$

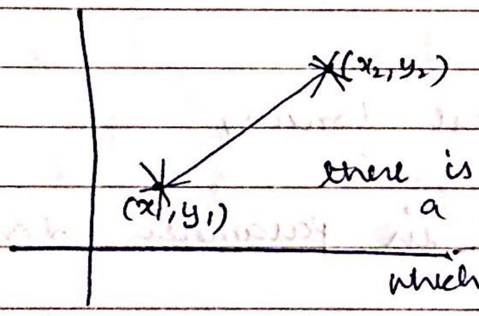



$c = -(mx + y).$

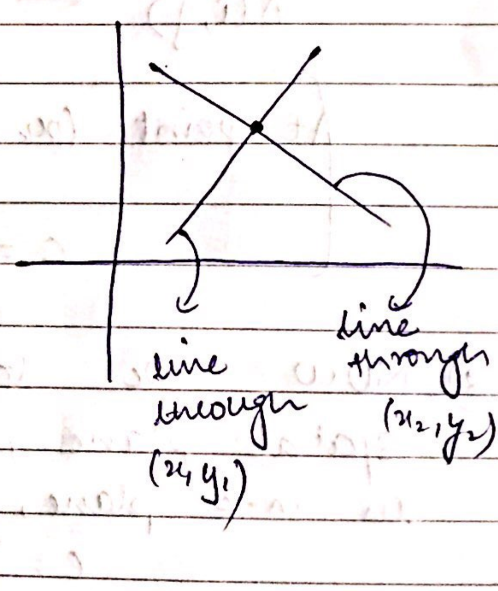


a dot in x-y plane will be represented as Line in m-c plane.



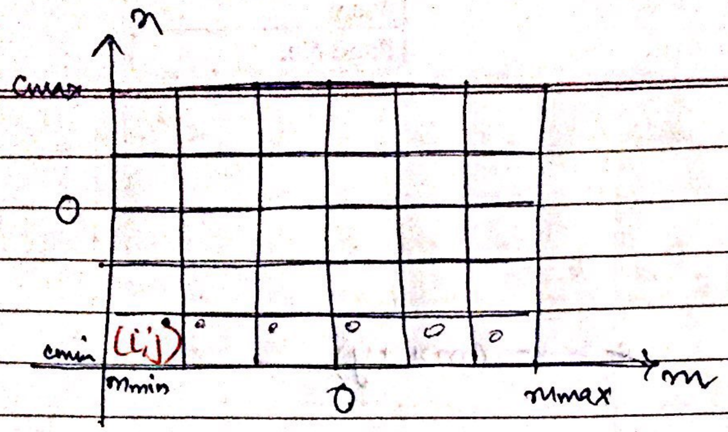

there is a line which slope & intercept is same, times line of $(x_1, y_1)$ & $(x_2, y_2)$ in m-c plane will be intersecting in nature.




**Q How to compute Hough Transform?**

:- M-c plane is subdivided into no. of accumulator cells.

Date:
Page No.




Range of slope is expected range of slopes where mmin represents minimum slope value and mmax represents the maximum slope value same goes with c.

A(i,j) :- Accumulated cell ~~[illegible]~~ (mi, cj) where slope value is mi & intercept value is cj.

① Initialise all cells with value 0.

$A(i,j) \leftarrow 0.$

At point $(x_k, y_k)$ in spatial domain,

$c = -mx_k + y_k$ in parameter domain.

② NOW we take at boundary pt say $(x_k, y_k)$ in spatial plane, this will be represented by a line in m-c plane,

$$c = -mx_k + y_k.$$

Assume every possible value of m & find the corresponding value of c. Value of 'c' has to be rounded off to the nearest allowed value of 'c'. Suppose $m_p \Rightarrow c_q$ ( $m_p$ gives value $c_q$), then corresponding accumulator cell value is incremented by 1.

$$A(p,q) \leftarrow A(p,q) + 1.$$

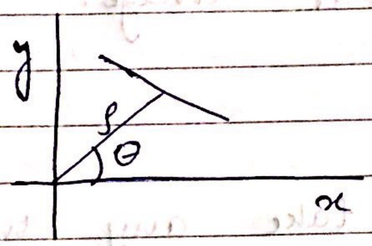
Date:
Page No.

This has to be done for all boundary points. At the end, A(i,j) contains value 'q'. This indicates that $\boxed{'q'}$ no of points lie on straight line $\boxed{y = m_i x + c_j}$

So, by taking an appropriate threshold we can link the edges.

"Problem" comes when the line is vertical & $m = \infty$. To solve this problem, we use 'normal representation' of straight line instead of "m-c plane".

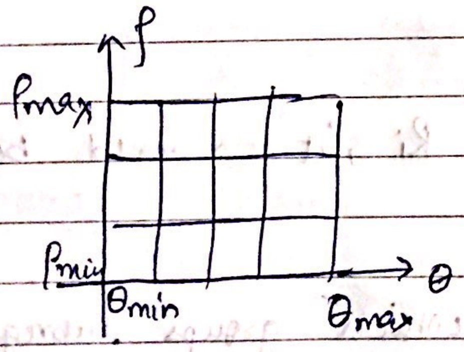
$$\rho = x\cos\theta + y\sin\theta$$




$\rho$ :- length of $\perp$ dropped on line from origin

$\theta$ :- angle made by $\perp$ with x-axis.

So, now the accumulator cell will be :-

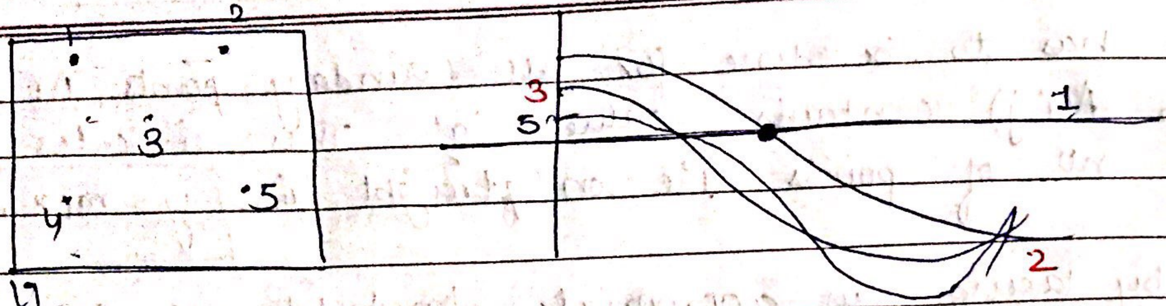



So, the maximum value of $\theta$ can be $\pm 90°$. maximum value of range of '$\rho$' is $\sqrt{M^2+N^2}$.

where $M \times N$ is the image size.

A Particular point in xy-plane is represented as sinusoidal wave in $\rho$-$\theta$ plane. So, 'q' no of collinear points will be mapped to 'q' no of sinusoidal curves in $\rho$-$\theta$ plane intersecting at a point.

Date:
Page No.




17/Oct/2017

## Region Growing Technique

All the pixels belonging to an image form a set R. It partitions R into no. of subregions like R1, R2, ... Rn. This partioning should follow some properties :

1. If we take union of Ri, $\bigcup_{i=1}^{n} R_i$ ($1 \le i \le n$), then it should give the original image.

$$\cup R_i = R$$

2. Ri is connected. If we take any two points in region Ri, then we should be able to identify a path b/w these two points.

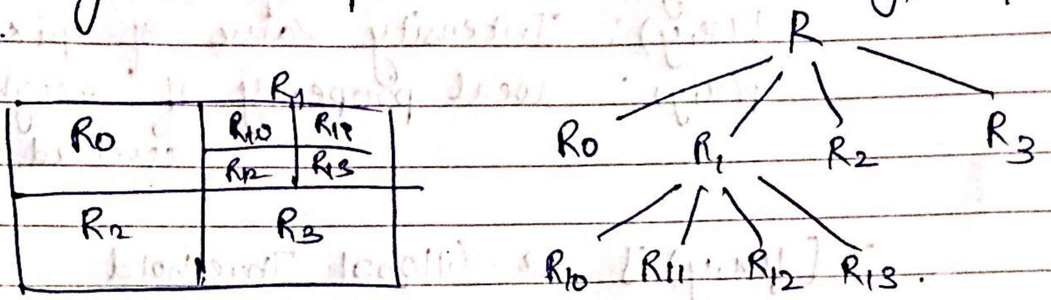
3. $R_i \cap R_j = \phi$ for $i \neq j$.

4. If we define predicate 'P' over Ri, it should be true.

Region Growing is a procedure which groups subregion into larger area depending on some criteria. It tries to grow a region starting from a 'seed' point. Two points are similar if intensity is closed. Points should be connected.

## ⇔ Region Splitting & Merging

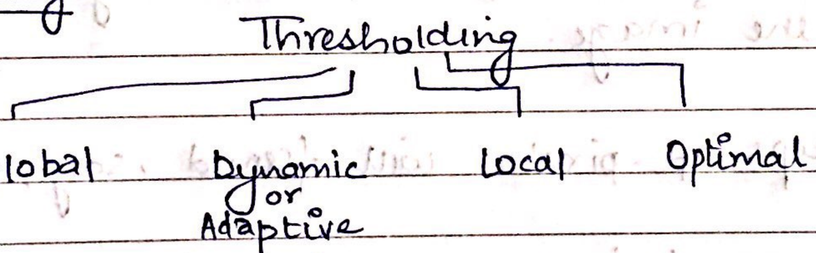
If we have an image say 'R', first try to find if the intensity values are similiar or not. If not, then break the image into partition check this again for each partition.




Img :- 'R'

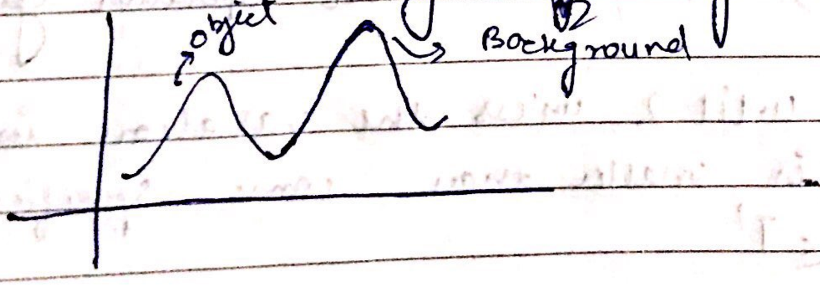
Stop if no more partition can be done, Select adjacent partition and check if they are similiar, then merge them.

## ⇔ Thresholding




## Bi-modal Histogram.

Two peak :- Dark object
Light ~~obj~~ Background




$0 \le f(x,y) < T$ → Object

$T \le f(x,y) \le 255$ → Background

Date:
Page No.

Thresholding function can be defined as :-

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

$(x,y)$ : location of pixel
$f(x,y)$ : Intensity value of pixel $(x,y)$.
$p(x,y)$ : local property of neighbourhood centered at $(x,y)$

$T[f(x,y)] \rightarrow$ Global Threshold
$T[f(x,y), p(x,y)] \rightarrow$ Local Threshold

~~T[f(x,y)]~~ $T[(x,y), f(x,y), p(x,y)] \rightarrow$ Adaptive Threshold.

## Automatic Detection of Threshold Value

1. Choose initial value of threshold, using this value segment the image.

2. Two groups of pixels will formed, say, $G_1$ & $G_2$

3. Compute mean '$\mu_1$' for group '$G_1$' & '$\mu_2$' for group '$G_2$'

4. Choose new threshold '$T$' = $\frac{\mu_1 + \mu_2}{2}$

5. Go back to (step-2) & perform the operation again.

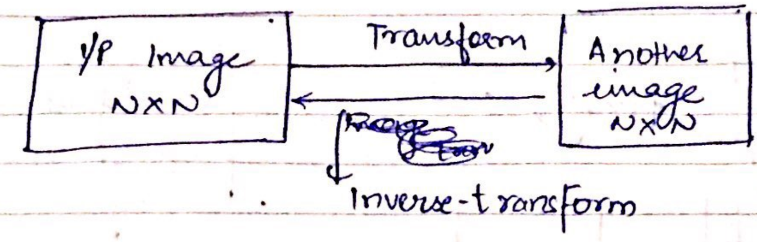
6. Steps are repeated until & unless the variation in the value of '$T$' is smaller than some specified value, $|T_i - T_{in}| \le T'$

04/Nov/2019.

Date:
Page No.

## Image Enhancement

→ can be done in spatial & frequency Domain.




unitary Matrix.

$$\boxed{A^{-1} = A^{*T}}$$

conjugate transpose

→ Orthogonal
→ Orthonormal

## Fourier Transformation

$[\ldots]_{N \times N}$.

$v = A.u$ → I/P vector

$v$ → Transformed vector

$A$ → unitary Matrix

$$u = A^{-1} v = A^{*T} u.$$

$$u(k,l) = \sum_{m,n=0}^{N-1} \sum a_{k,l}(m,n)\, u(m,n).$$

$a_{k,l}$ →
- matrix of NxN
- each element is matrix of size NxN

## Fourier transform formula

$$\mathcal{F}\{f(x)\} = F(u) = \int_{-\infty}^{\infty} f(x)\, e^{-j2\pi ux}\, dx$$

$$F[u,v] = \int_{-\infty}^{\infty}\int f(x,y)\, e^{-j2\pi(ux+vy)}\, dx\, dy$$

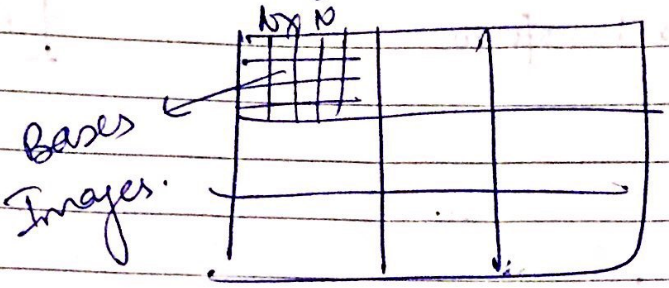
## 2-D Discrete Fourier transform

$$F(u,v) = \frac{1}{NN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x,y)\, e^{-j2\pi\left(\frac{ux}{M} + \frac{vy}{N}\right)}$$

$$f(x,y) = \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F(u,v)\, e^{-j2\pi\left(\frac{ux}{M}+\frac{vy}{N}\right)}$$

$$F(u,v) = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x,y)\, e^{-j}$$

low filter & high filter pass




1. Multiply the I/P image by $(-1)^{x+y}$, to centre the transform.
2. Compute D.F.T $\left(f(u,v)\right)$ of the image from 1.
3. Multiply $f(u,v)$ via filter function $h(u,v)$.
4. Compute the inverse DFT of the resulting 3
5. Obtain the real part of result in 4.
6. Multiply the result by $(-1)^{x+y}$.