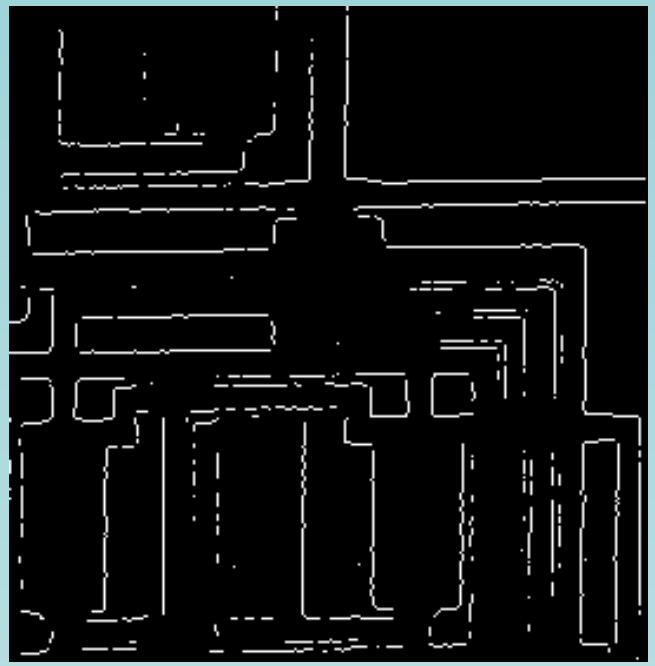
CSE6706:
Advanced Digital Image Processing

Dr. Md. Monirul Islam

CSE-BUET



Edge Linking

CSE-BUET

Edge Linking

- Local Processing
- Regional Processing
- Global Processing
  - Hough transform
  - Graph theoretic approach

CSE-BUET

Local Processing

- Analyze neighbors of each edge pixel
- Join two pixels if they are similar

CSE-BUET

Local Processing

- Analyze neighbors of each edge pixel
  - Check neighbors of area 3X3 or 5X5
- Join two pixels if they are similar

CSE-BUET

Local Processing

- Analyze neighbors of each edge pixel
  - Check neighbors of area 3X3 or 5X5
- Join two pixels if they are similar
  - Strength of response of the gradient

CSE-BUET

Local Processing

- Analyze neighbors of each edge pixel
  - Check neighbors of area 3X3 or 5X5
- Join two pixels if they are similar
  - Strength of response of the gradient

$$\left|\nabla f(x,y) - \nabla f(x_0,y_0)\right| \leq E$$

CSE-BUET

Local Processing

- Analyze neighbors of each edge pixel
  - Check neighbors of area 3X3 or 5X5
- Join two pixels if they are similar
  - Strength of response of the gradient

$$\left|\nabla f(x,y) - \nabla f(x_0,y_0)\right| \le E$$

  - The direction of the gradient vector

$$\left|\alpha(x,y) - \alpha(x_0,y_0)\right| < A$$

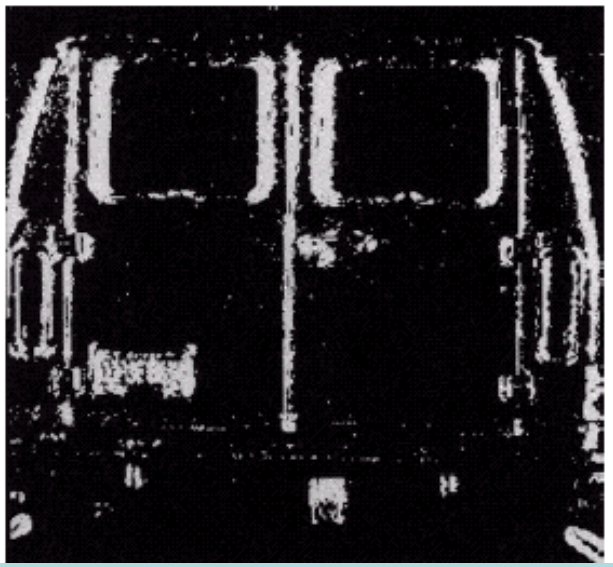
CSE-BUET

Example of Local Processing

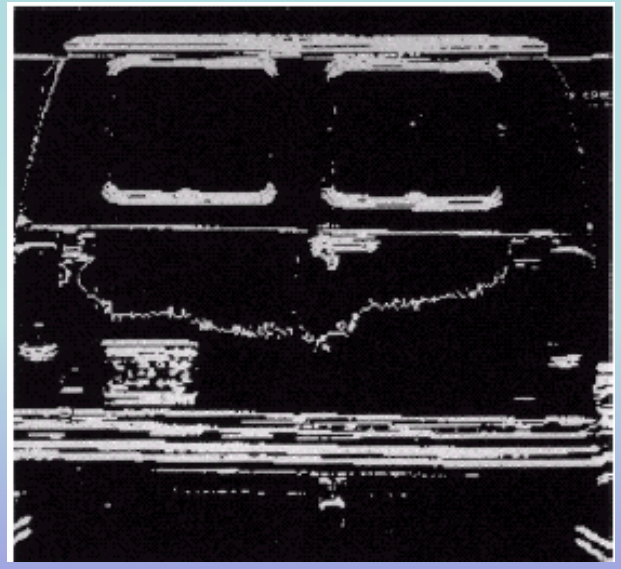
- Objective: *detect the license plate*

CSE-BUET

Example of Local Processing



By Vertical
Sobel Operator

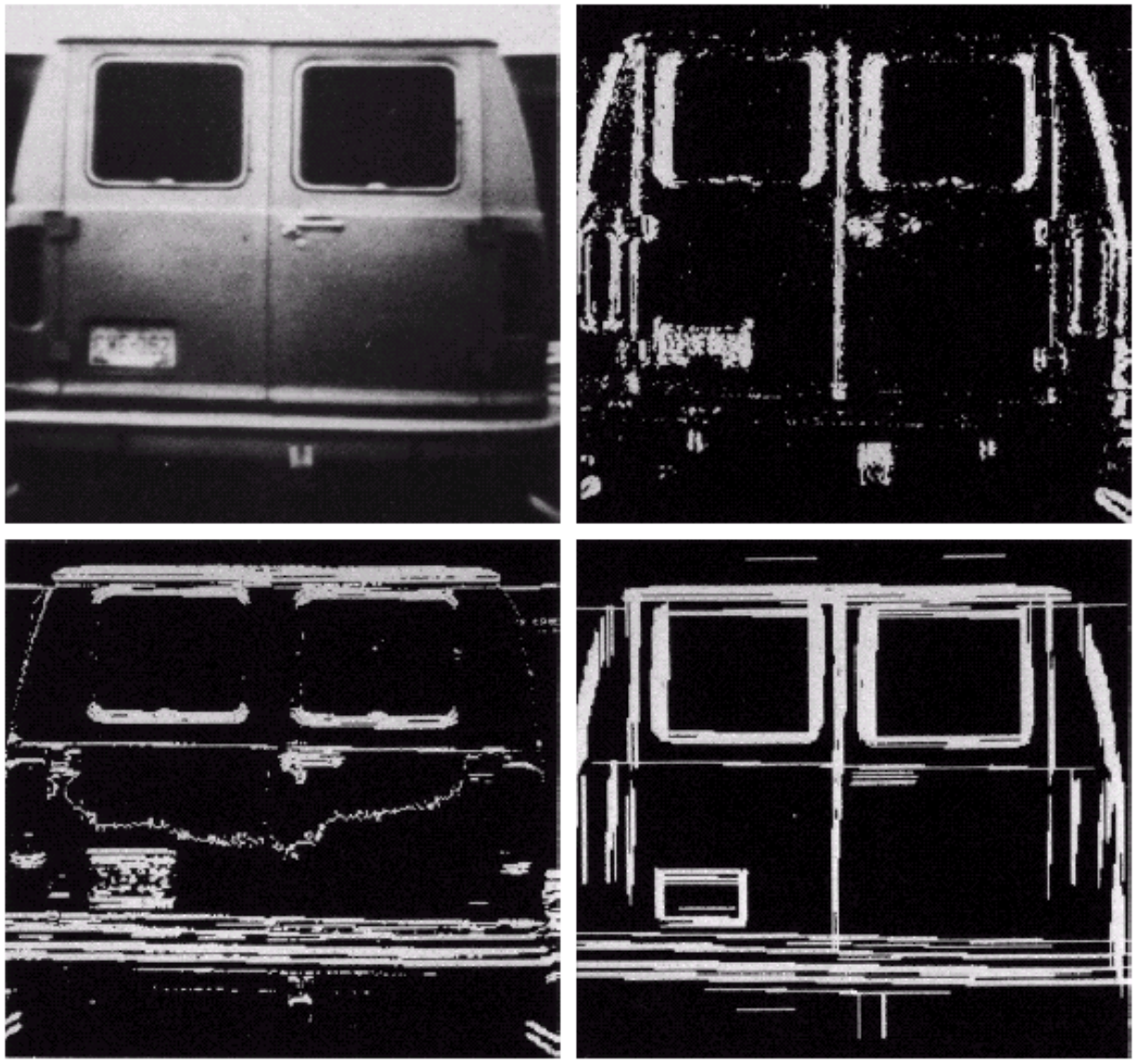


By Horizontal
Sobel Operator

- *Disconnection* and *breaks* in edges

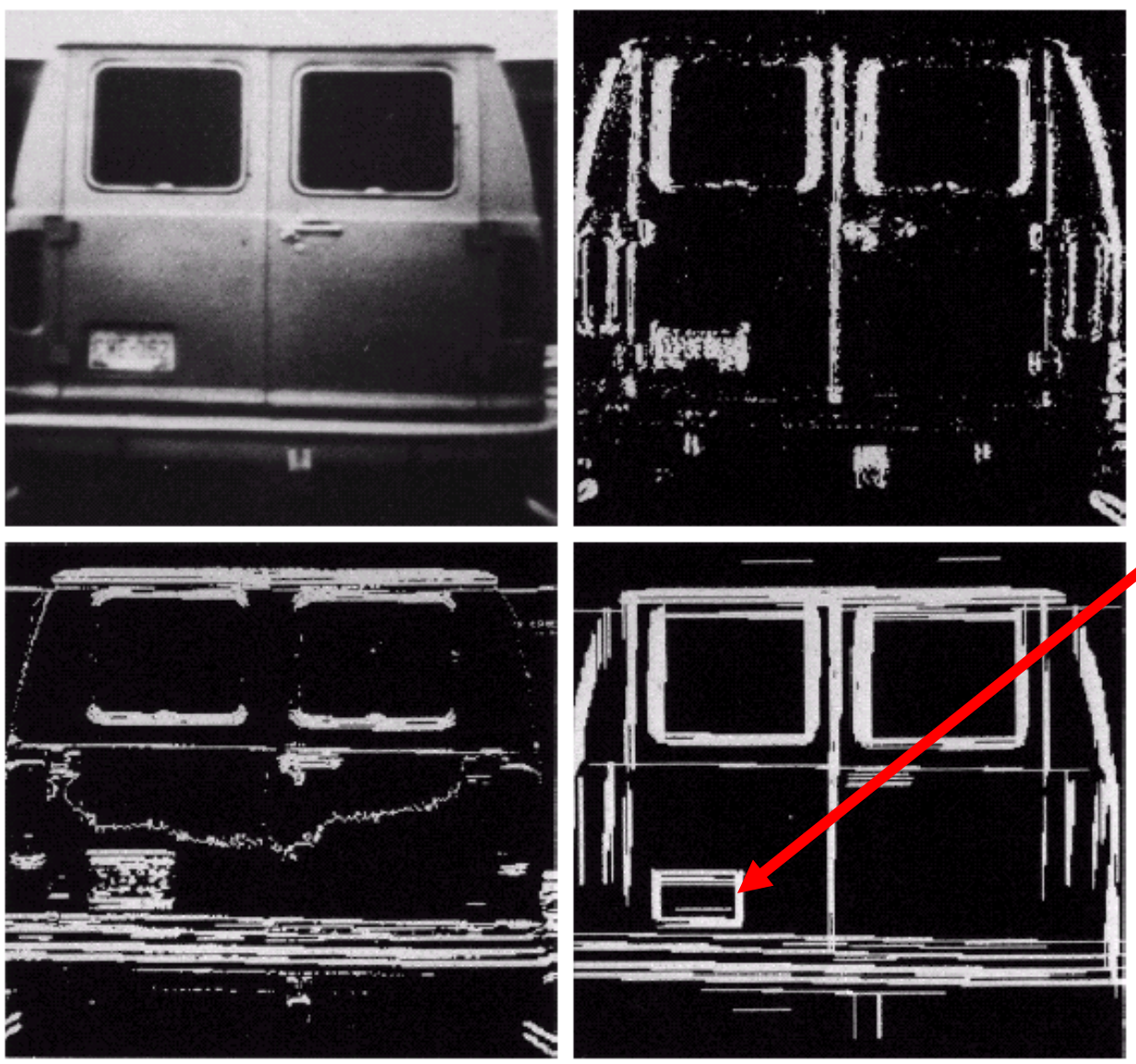
CSE-BUET

Example of Local Processing



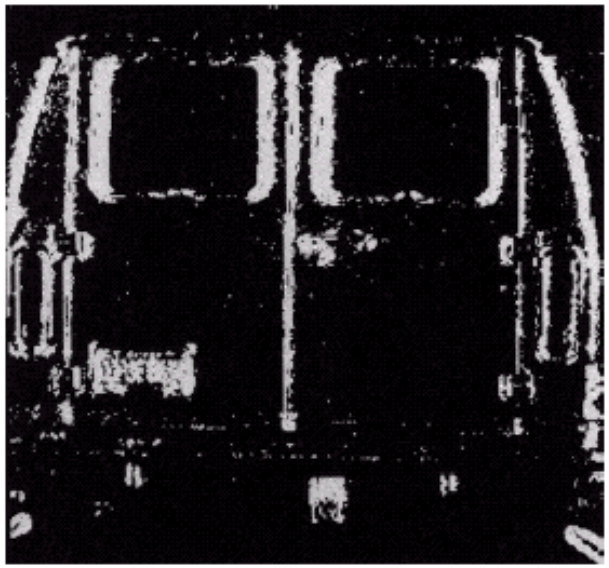
- Sequential scanning: row by row and col by col with
  - $E = 25$
  - $A = 15º$

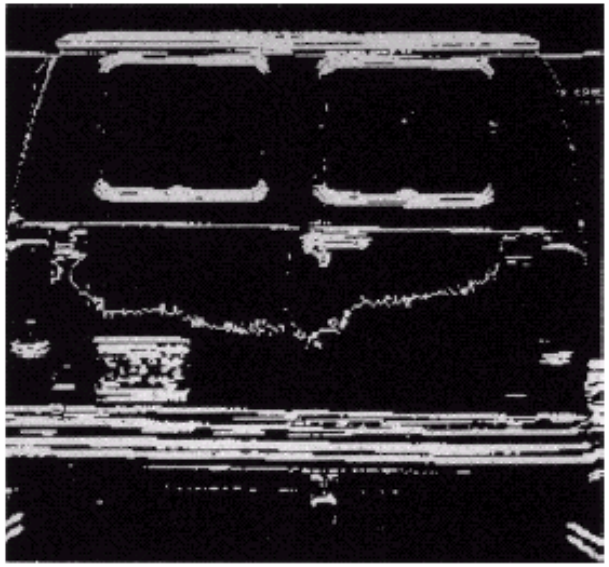
Example of Local Processing

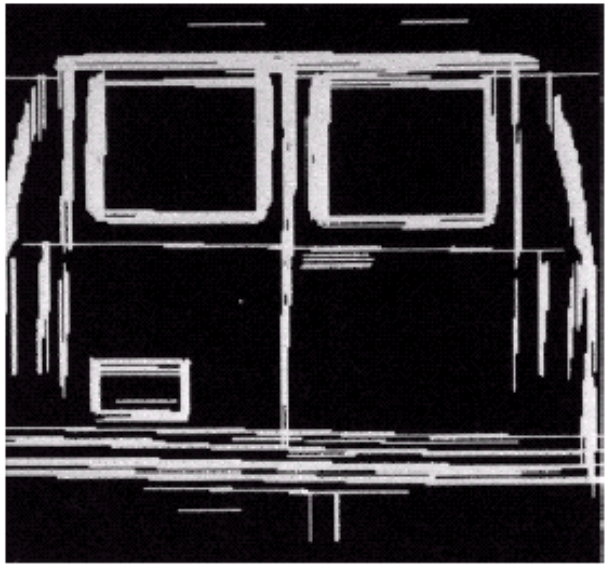


- Note the small rectangle in bottom-left corner

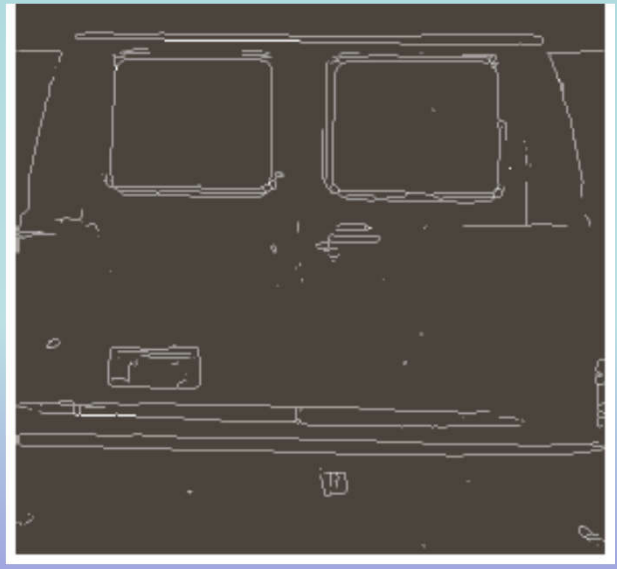
Example of Local Processing







After Morphological Thinning

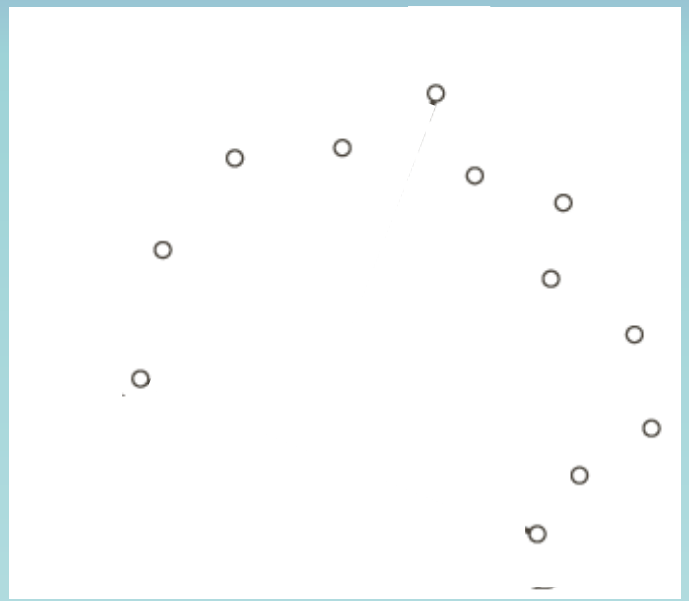


Regional Processing

- Useful when prior info about region location is available
- Example:
  - Polynomial approximation
  - Curve fitting

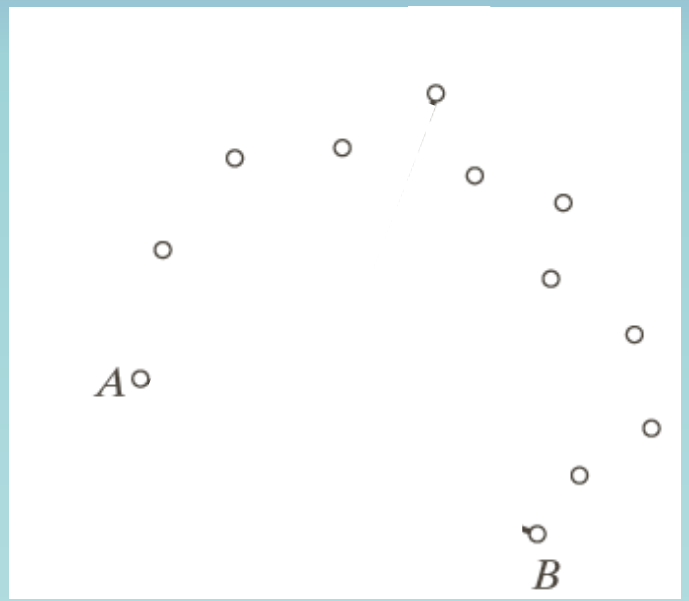
CSE-BUET

Regional Processing



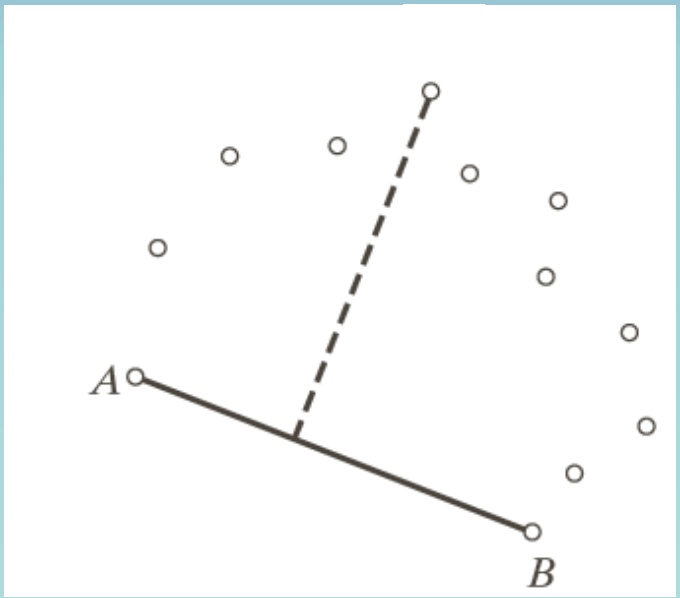
CSE-BUET

Regional Processing



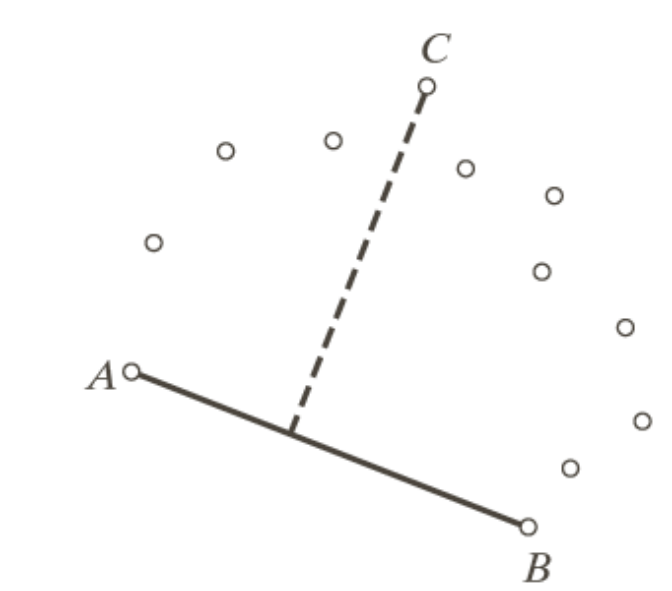
CSE-BUET

Regional Processing



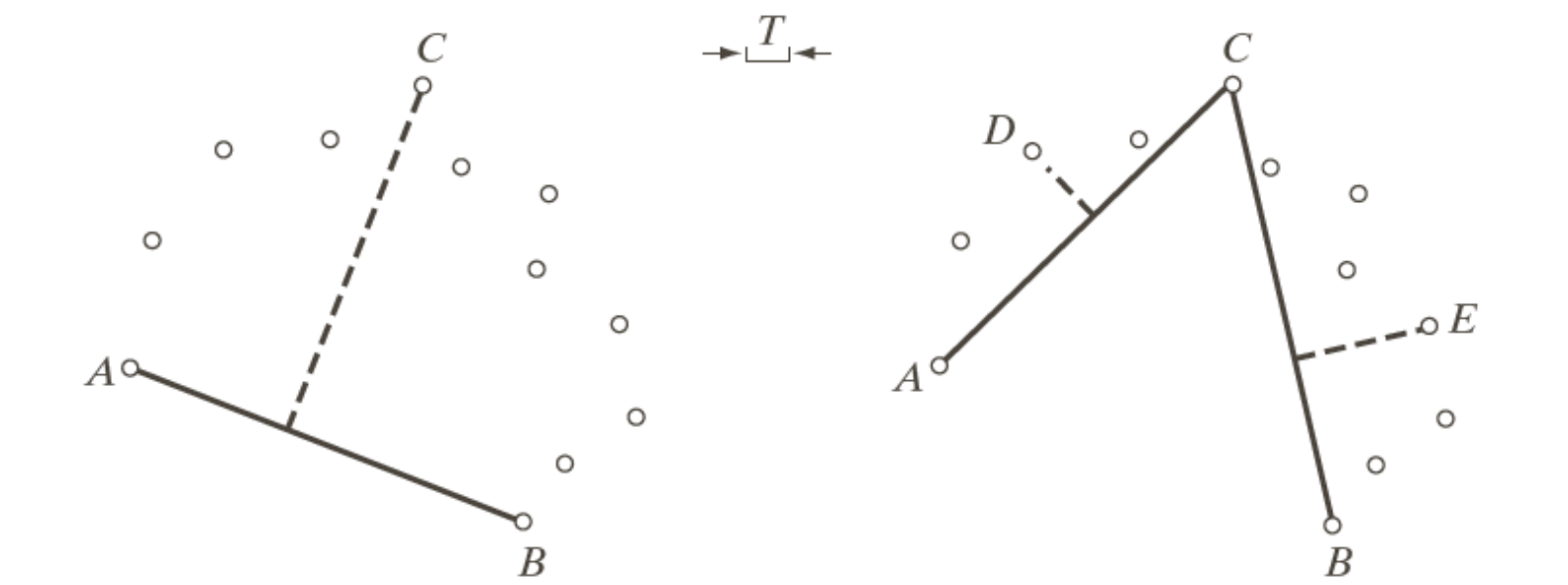
CSE-BUET

Regional Processing



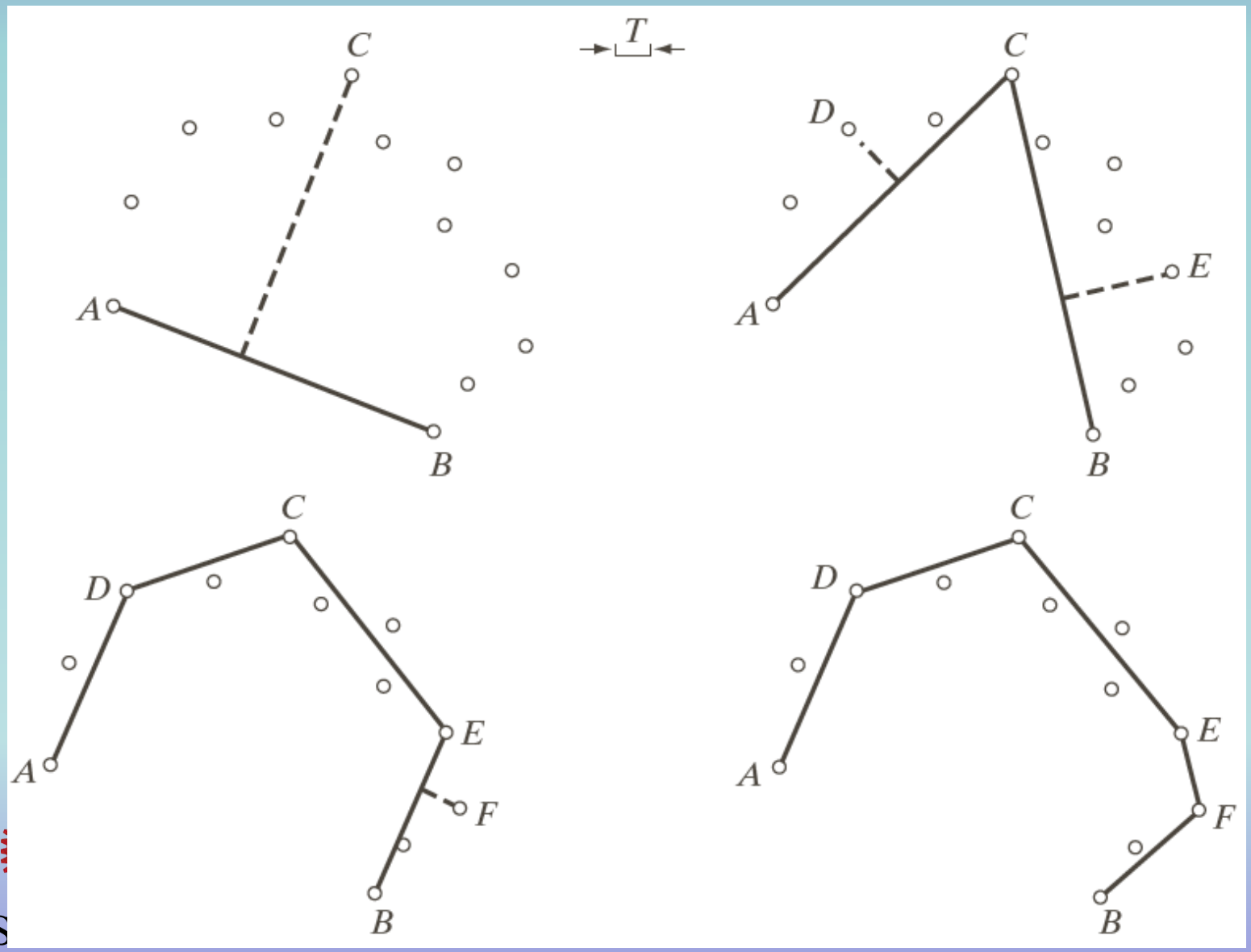
CSE-BUET

Regional Processing



CSE-BUET

Regional Processing

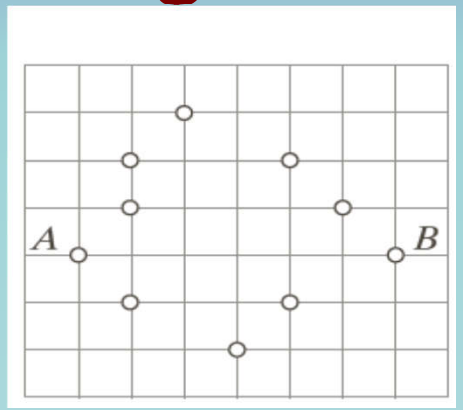


Issues in Regional Processing

- Finding the initial two points
- Order of the points
- Closed path or open path?

CSE-BUET

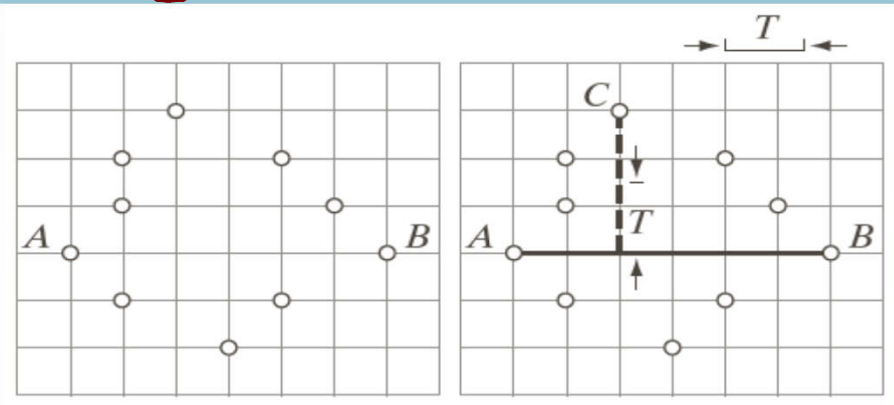
Regional Processing with Closed Path



CLOSED	OPEN	Curve segment processed	Vertex generated
B	*B*, *A*	—	*A*, *B*

CSE-BUET

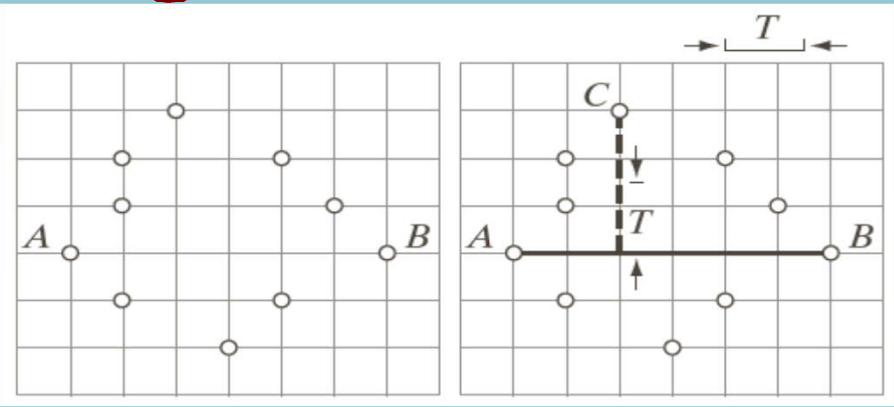
Regional Processing with Closed Path



CLOSED	OPEN	Curve segment processed	Vertex generated
B	*B, A*	—	*A, B*
B	*B, A*	(*BA*)	*C*

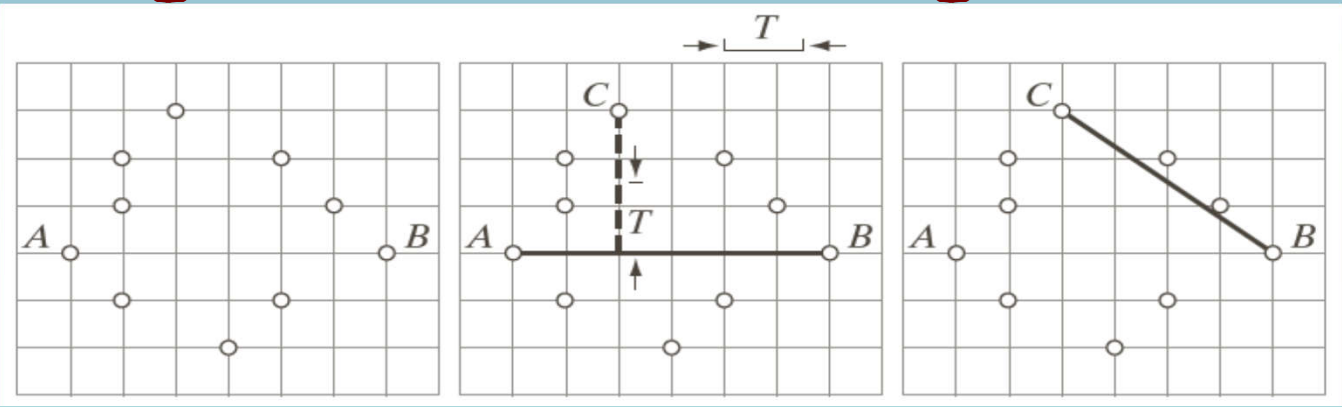
CSE-BUET

Regional Processing with Closed Path



CLOSED	OPEN	Curve segment processed	Vertex generated
B	B, A	—	A, B
B	B, A	(BA)	C
B	B, A, C		

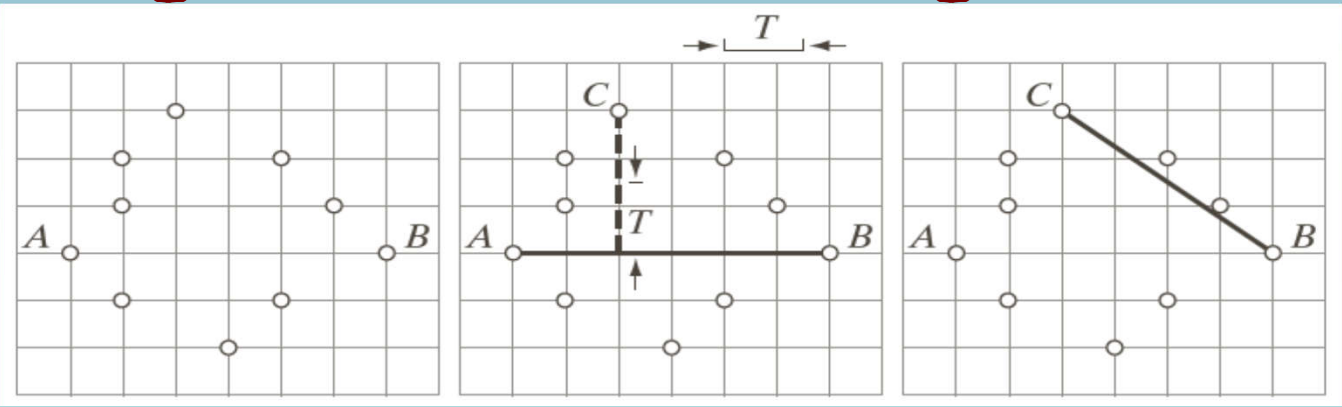
Regional Processing with Closed Path



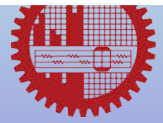
CLOSED	OPEN	Curve segment processed	Vertex generated
B	B, A	—	A, B
B	B, A	(BA)	C
B	B, A, C	(BC)	—

CSE-BUET

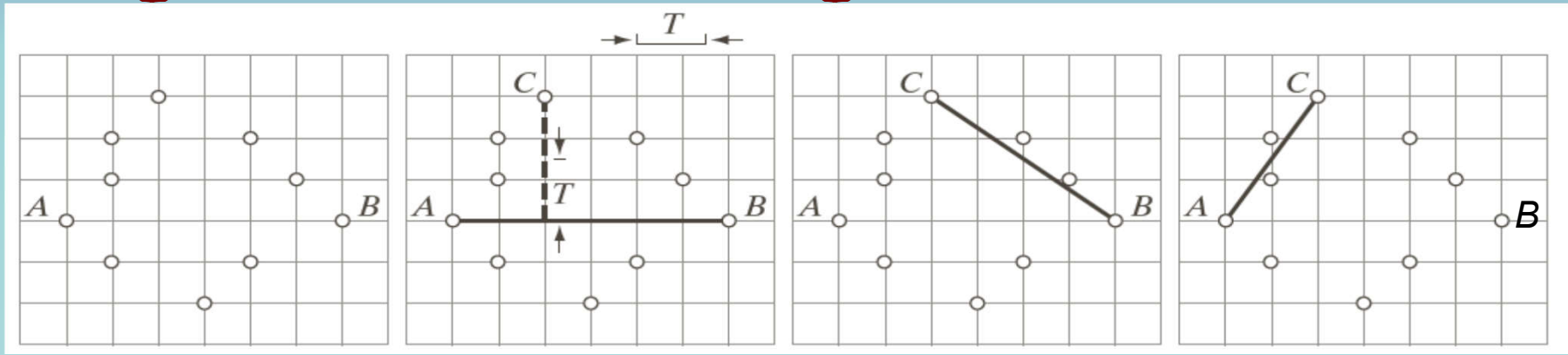
Regional Processing with Closed Path



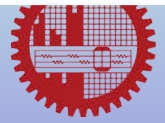
CLOSED	OPEN	Curve segment processed	Vertex generated
B	B, A	—	A, B
B	B, A	(BA)	C
B	B, A, C	(BC)	—
B, C	B, A		



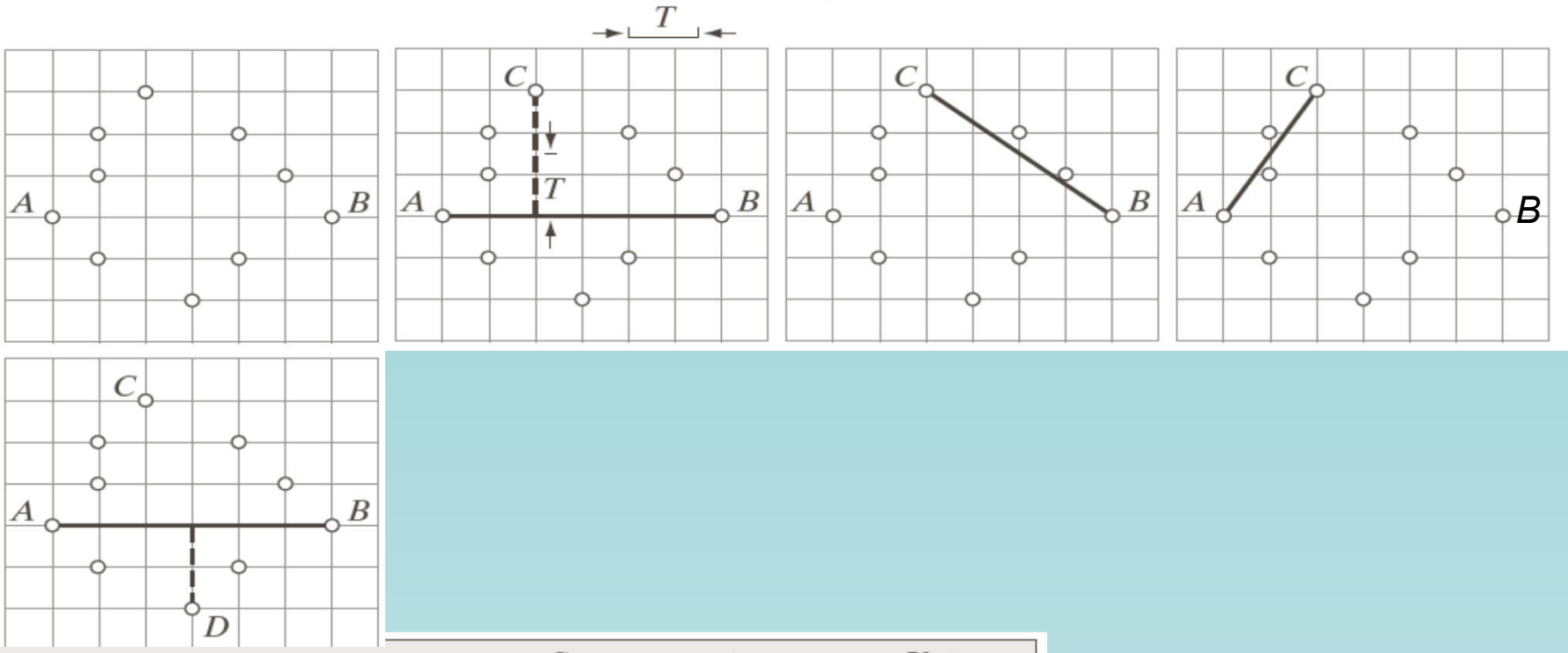
Regional Processing with Closed Path



CLOSED	OPEN	Curve segment processed	Vertex generated
B	B, A	—	A, B
B	B, A	(BA)	C
B	B, A, C	(BC)	—
B, C	B, A	(CA)	—

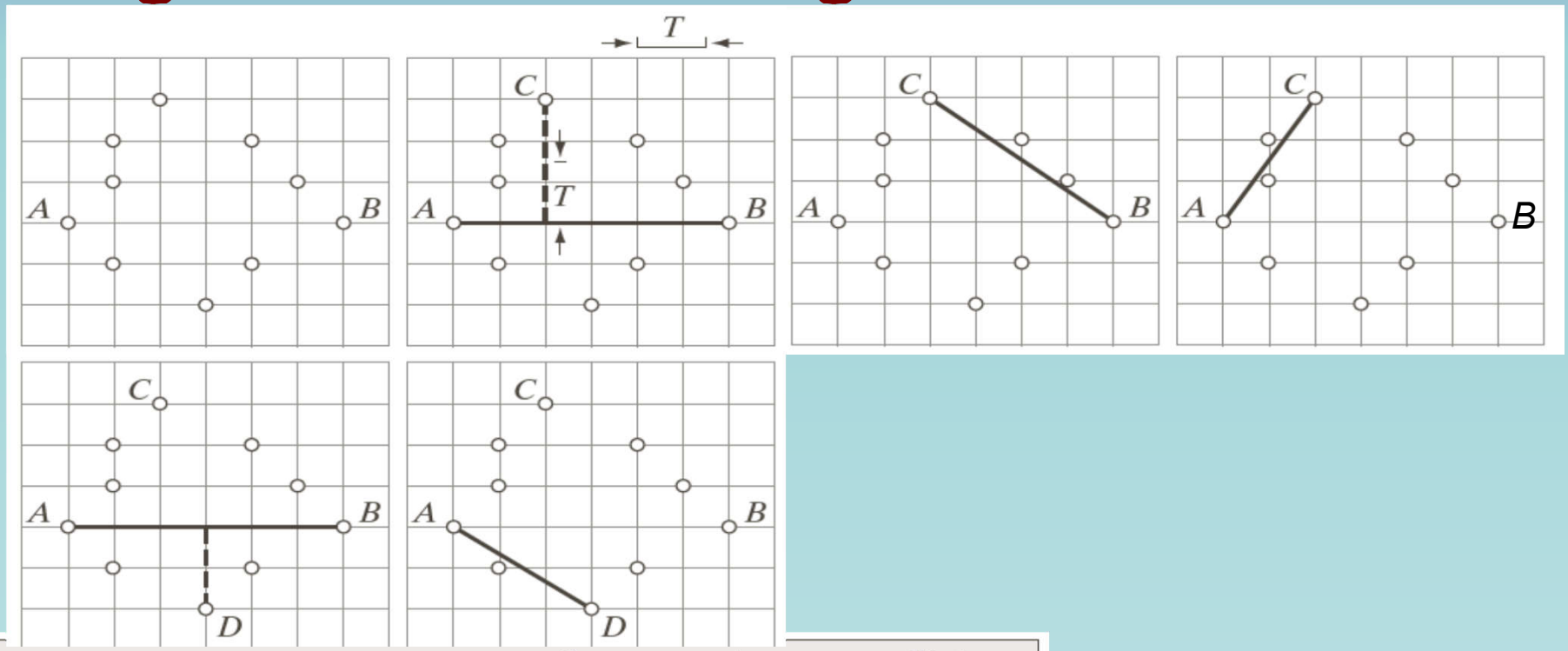


Regional Processing with Closed Path



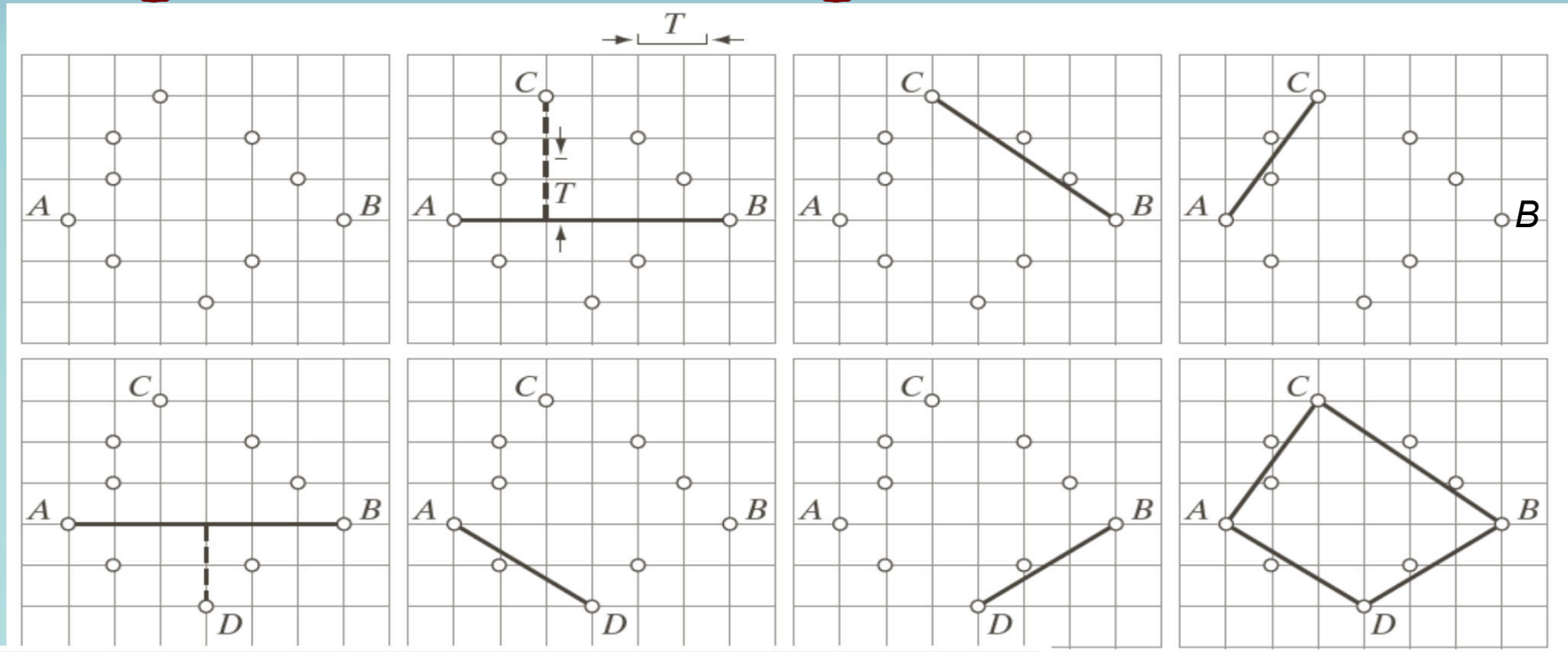
CLOSED	OPEN	Curve segment processed	Vertex generated
B	B, A	—	A, B
B	B, A	(BA)	C
B	B, A, C	(BC)	—
B, C	B, A	(CA)	—
B, C, A	B	(AB)	D

Regional Processing with Closed Path



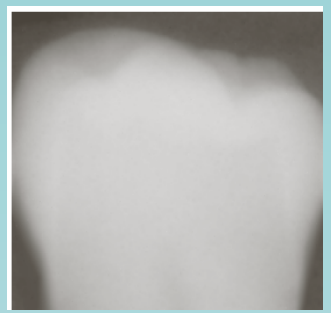
CLOSED	OPEN	Curve segment processed	Vertex generated
B	B, A	—	A, B
B	B, A	(BA)	C
B	B, A, C	(BC)	—
B, C	B, A	(CA)	—
B, C, A	B	(AB)	D
B, C, A	B, D	(AD)	—

Regional Processing with Closed Path



CLOSED	OPEN	Curve segment processed	Vertex generated
B	B, A	–	A, B
B	B, A	(BA)	C
B	B, A, C	(BC)	–
B, C	B, A	(CA)	–
B, C, A	B	(AB)	D
B, C, A	B, D	(AD)	–
B, C, A, D	B	(DB)	–
B, C, A, D, B	Empty	–	–

Example of Regional Processing



Dental Xray

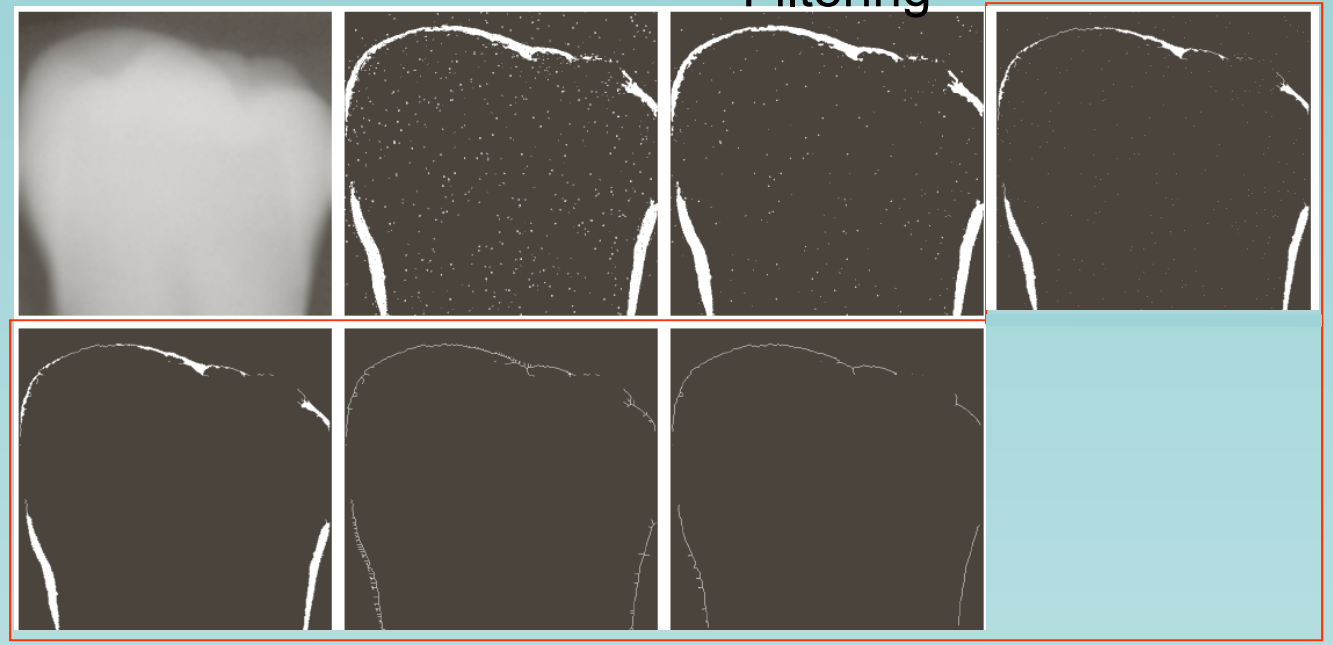
CSE-BUET

Example of Regional Processing

Dental Xray Gradient Majority Filtering

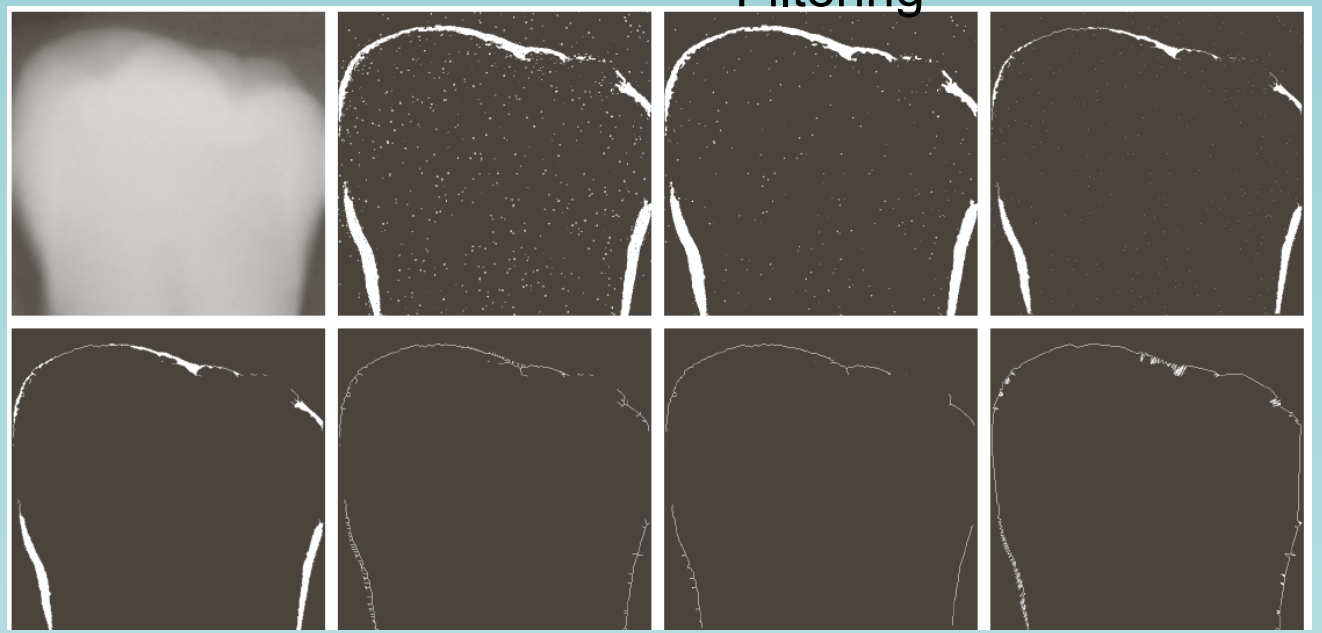
CSE-BUET

Example of Regional Processing



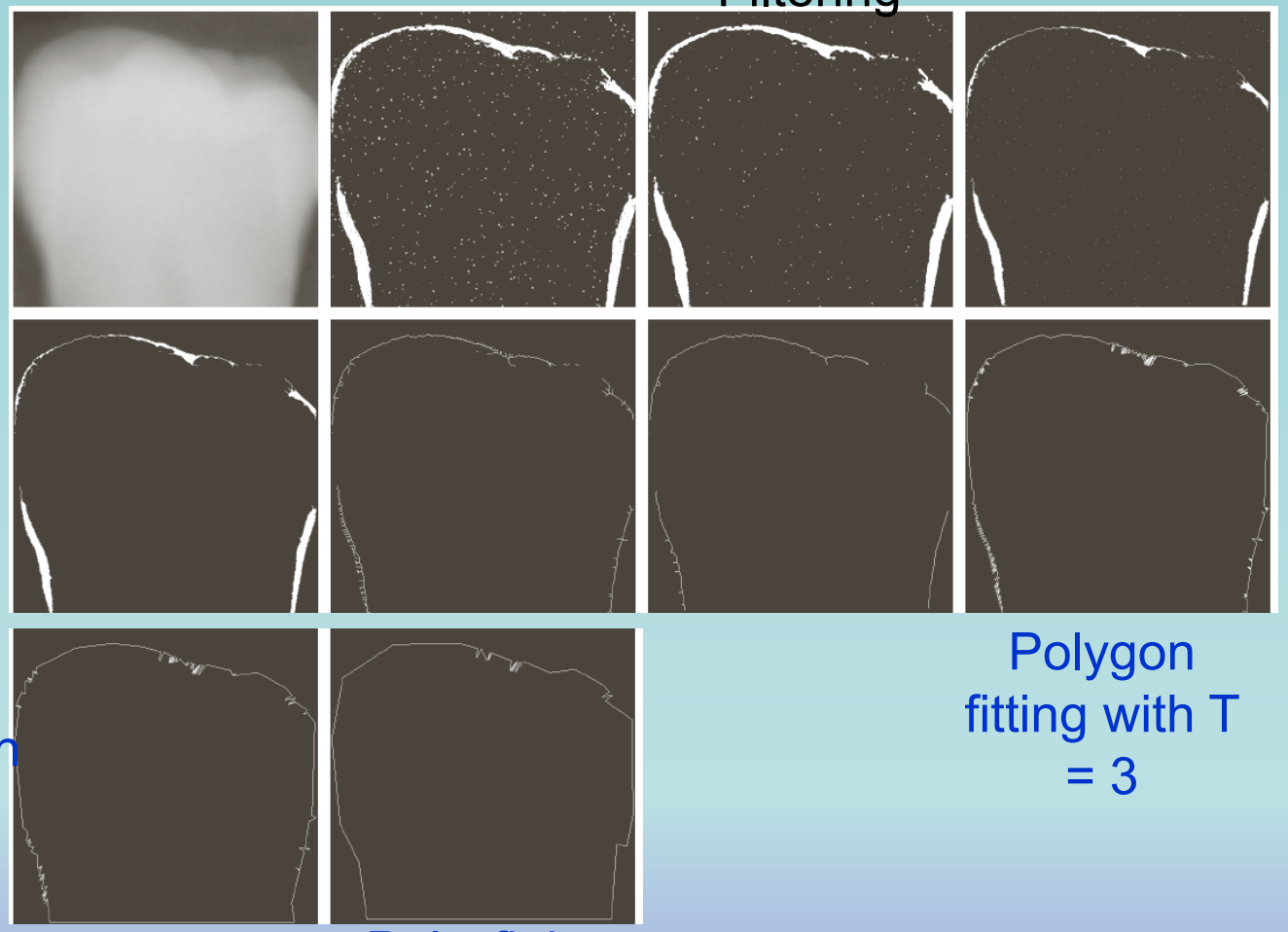
CSE-BUET

Example of Regional Processing



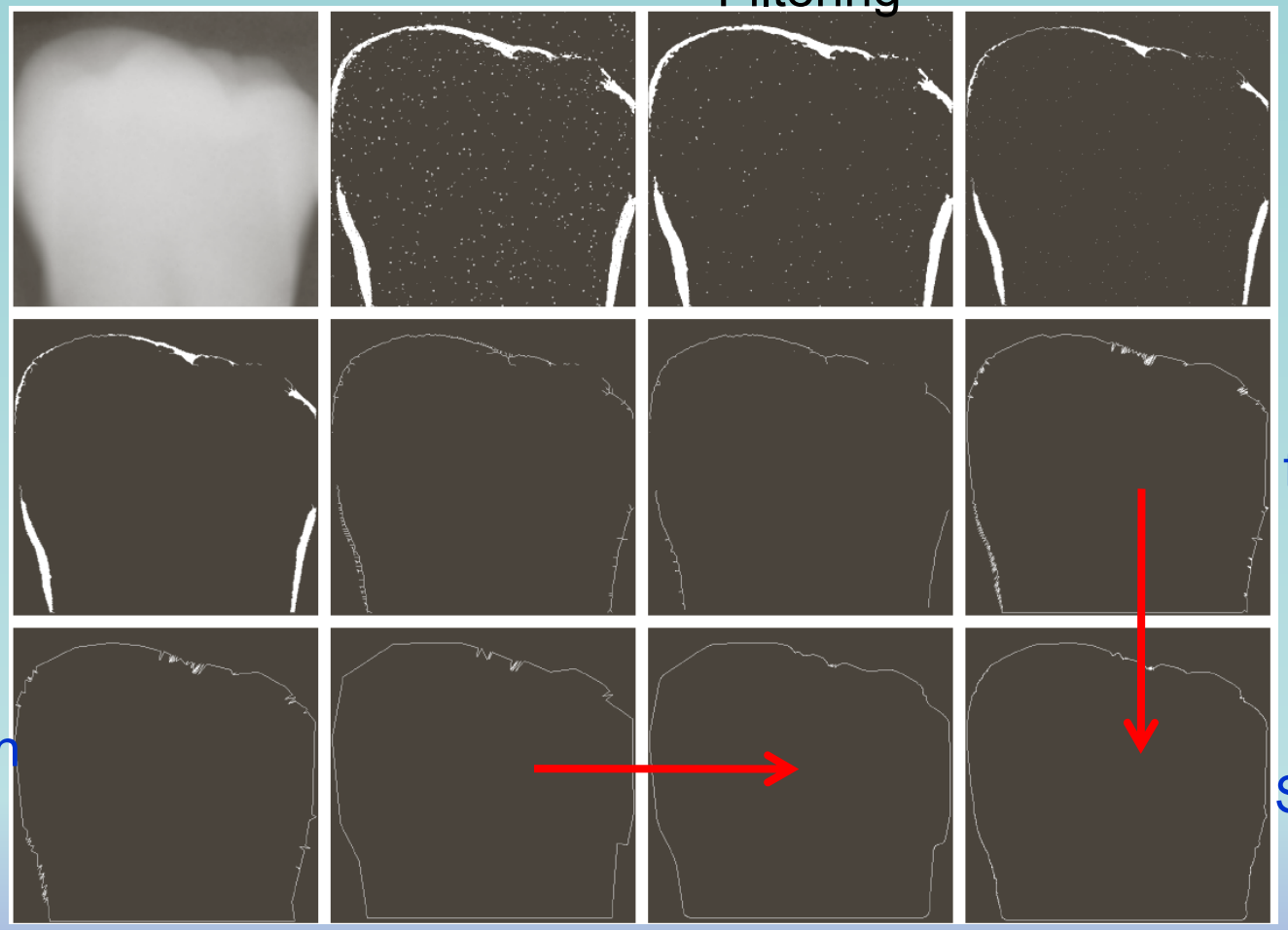
CSE-BUET

Example of Regional Processing



CSE-BUET

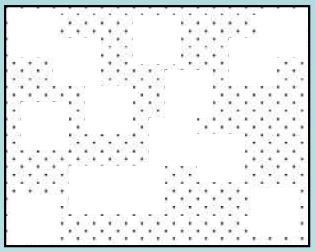
Example of Regional Processing



CSE-BUET

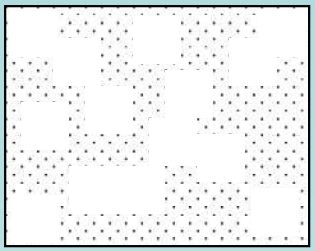
Global Processing via Hough Transform

- Let, *n* pixels are defined as edge-points
- Which points are co-linear?



Global Processing via Hough Transform

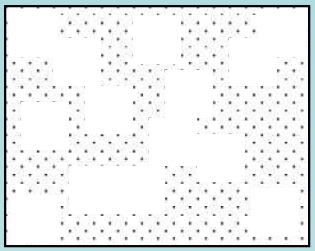
- Let, *n* pixels are defined as edge-points
- Which points are co-linear?
- Solution:
  - Find all possible lines
  - Then, find subsets of *n* points co-linear with each line



CSE-BUET

Global Processing via Hough Transform

- Let, *n* pixels are defined as edge-points
- Which points are co-linear?
- Complexity is too high:
  - Find all possible lines $^{n}C_{2} = n(n-1)/2 \approx n^{2}$
  - Then, find subsets of *n* points co-linear with each line $n \times {}^{n}C_{2} \approx n^{3}$

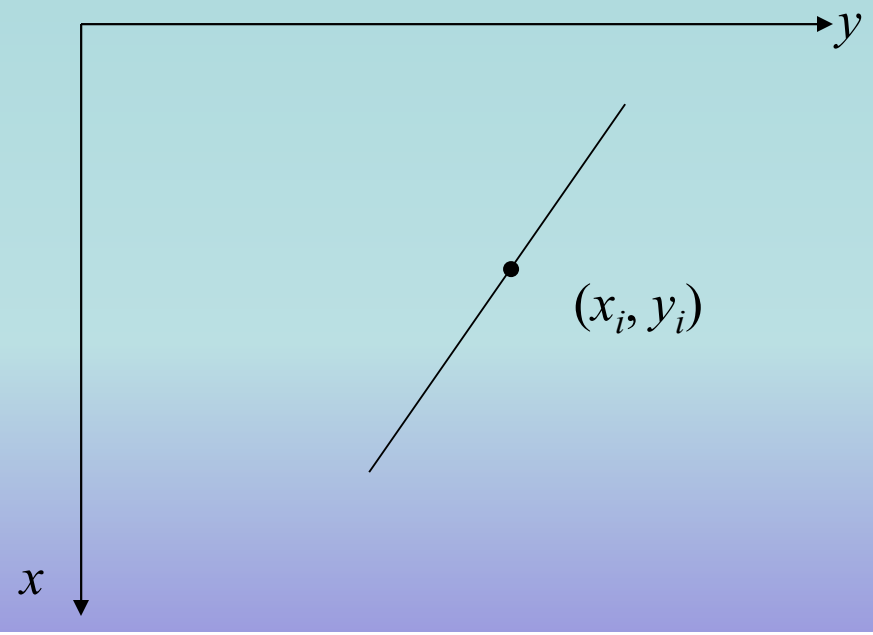


CSE-BUET

Hough Transform

- Developed by Hough

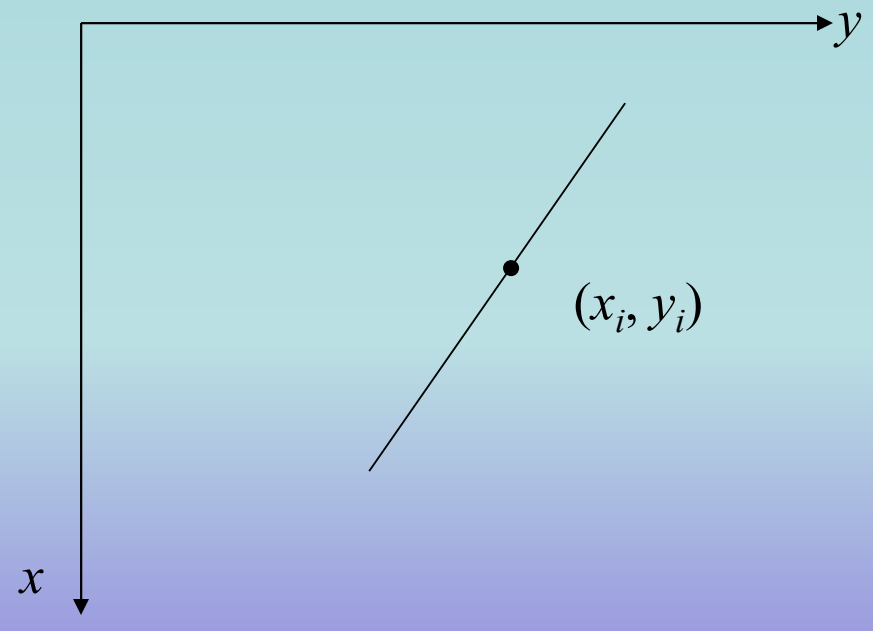
- Let a point (x_i, y_i)
- The equation of the line passing through (x_i, y_i) is $y_i = ax_i + b$



Hough Transform

- Developed by Hough

- Let a point (x_i, y_i)
- The equation of the line passing through (x_i, y_i) is $y_i = ax_i + b$
- How many lines are there?

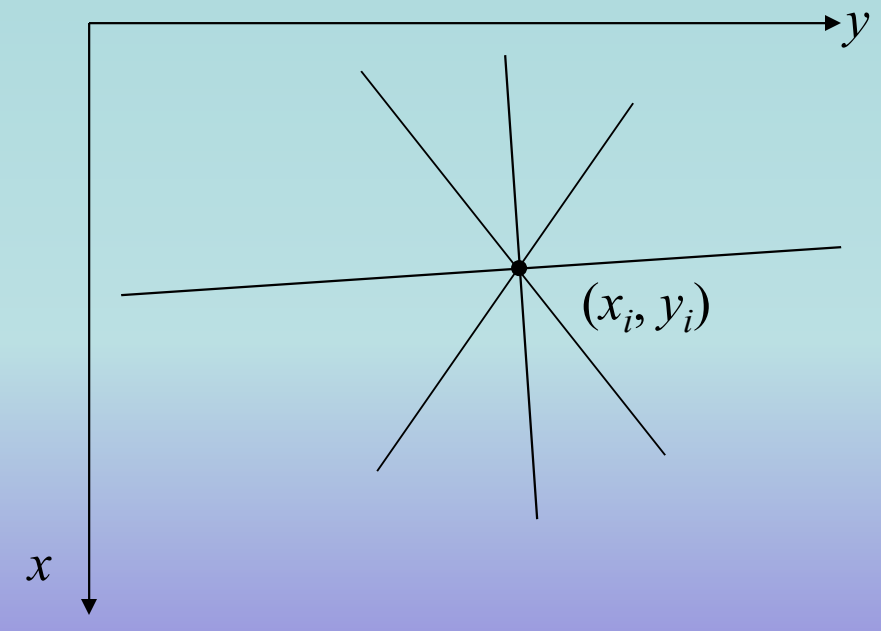


Hough Transform

- Developed by Hough

- Let a point (x_i, y_i)
- The equation of the line passing through (x_i, y_i) is $y_i = ax_i + b$
- How many lines are there?

- Infinitely many lines!



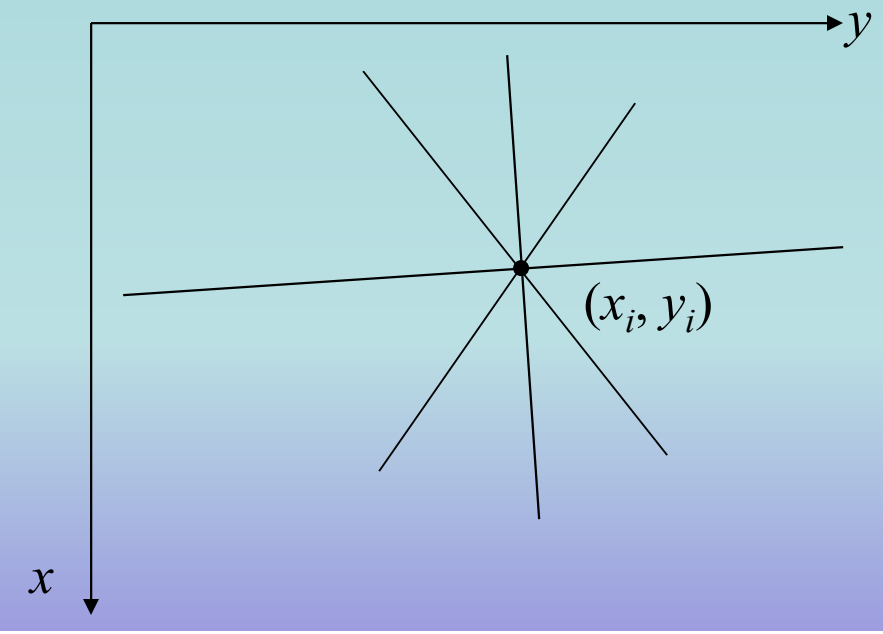
CSE-BUET

Hough Transform

- Developed by Hough

- Let a point (x_i, y_i)
- The equation of the line passing through (x_i, y_i) is $y_i = ax_i + b$
- How many lines are there?

- Infinitely many lines!
- Changing *a*, *b* gives a new line

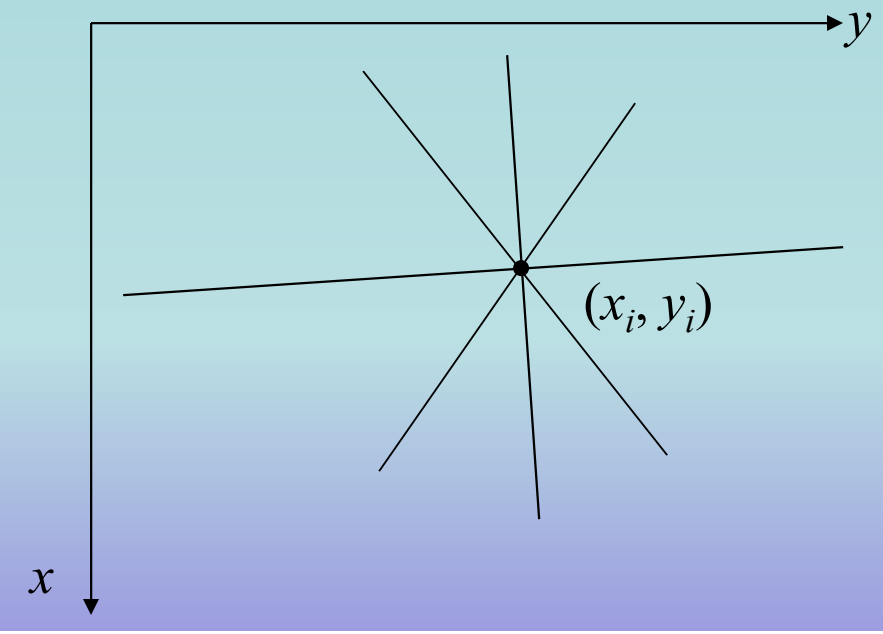


Hough Transform

- Developed by Hough

- Let a point (x_i, y_i)
- The equation of the line passing through (x_i, y_i) is $y_i = ax_i + b$
- But, if the equation is changed to

$$b = -x_i a + y_i$$



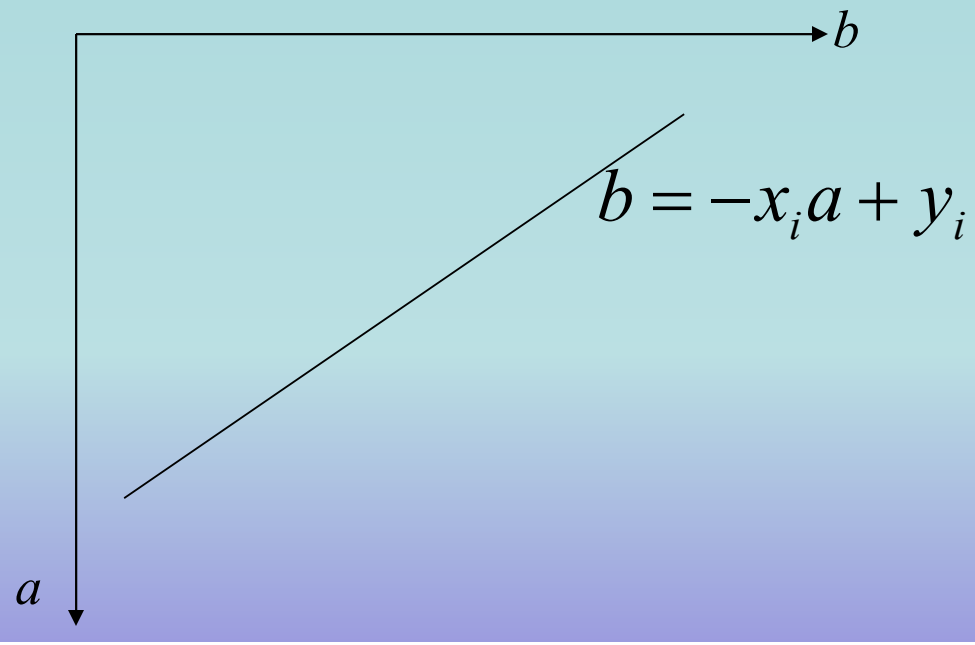
Hough Transform

- Developed by Hough

- Let a point (x_i, y_i)
- The equation of the line passing through (x_i, y_i) is $y_i = ax_i + b$
- But, if the equation is changed to

$$b = -x_i a + y_i$$

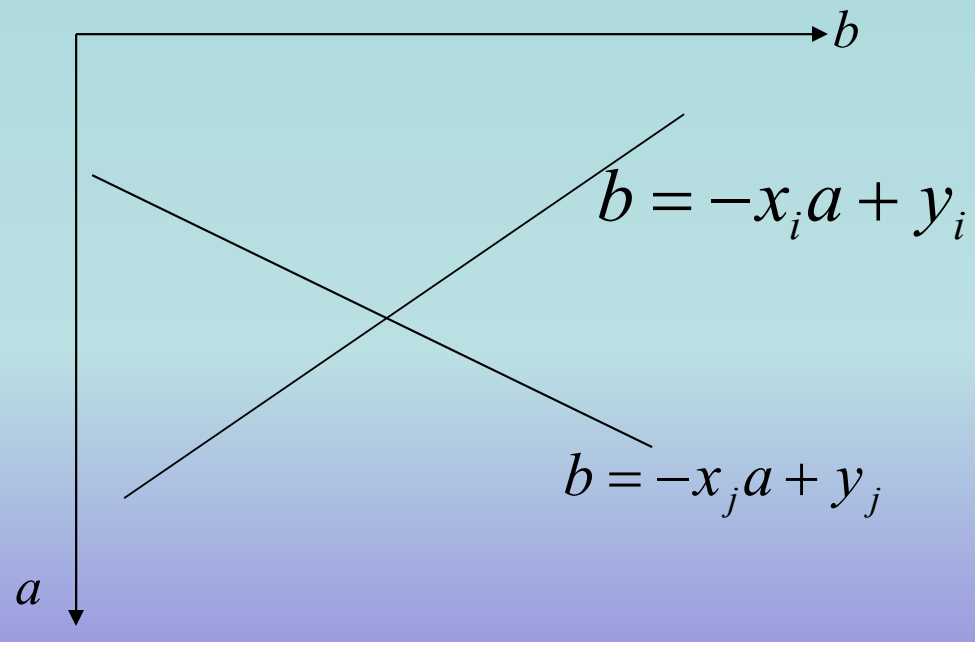
This is a *single* line in a, b plane



Hough Transform

- If we have another point (x_j, y_j), its equation is

$$b = -x_j a + y_j$$

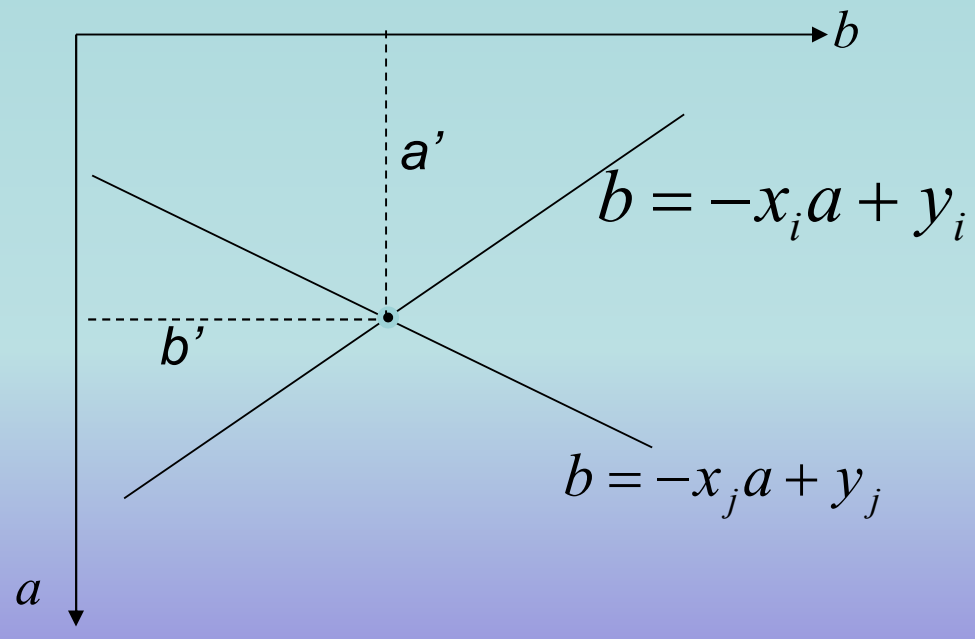


CSE-BUET

Hough Transform

- If we have another point (x_j, y_j), its equation is

$$b = -x_j a + y_j$$



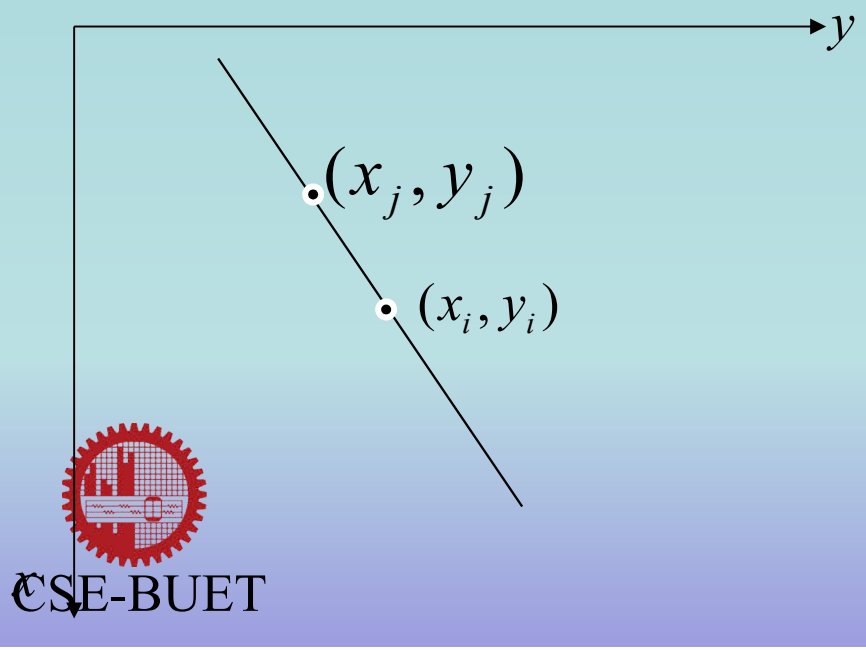
CSE-BUET

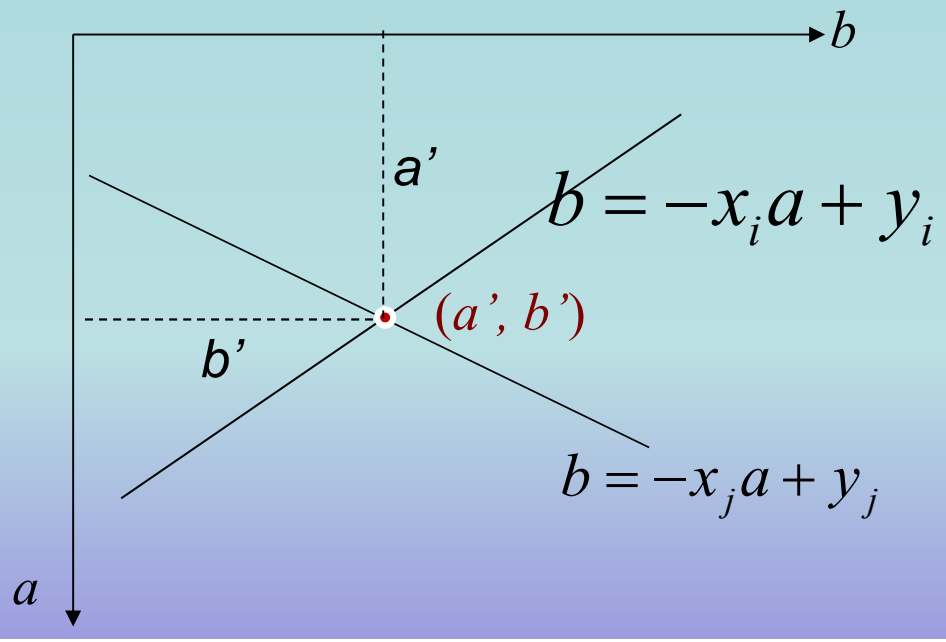
Hough Transform

- If we have another point (x_j, y_j), its equation is

$$b = -x_j a + y_j$$

- *a'* and *b'* are slope and intercept of the line in x, y plane



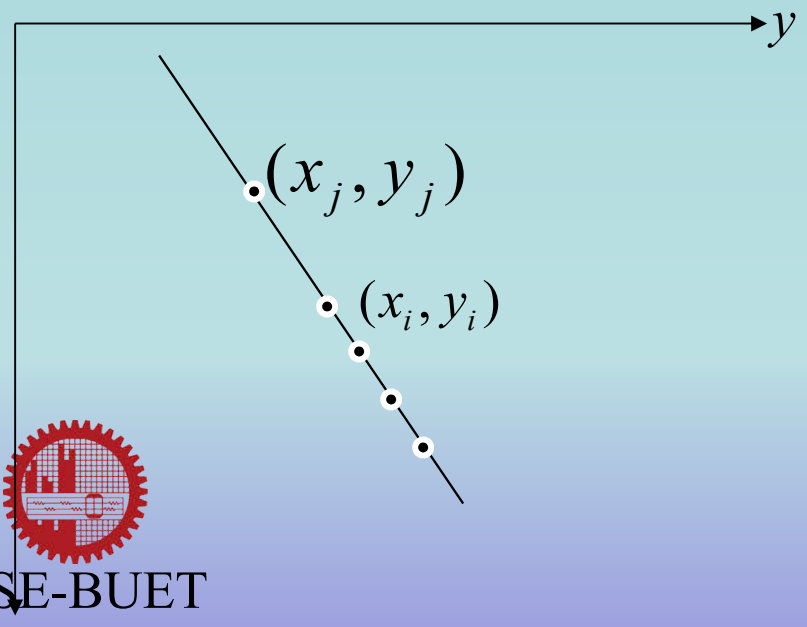


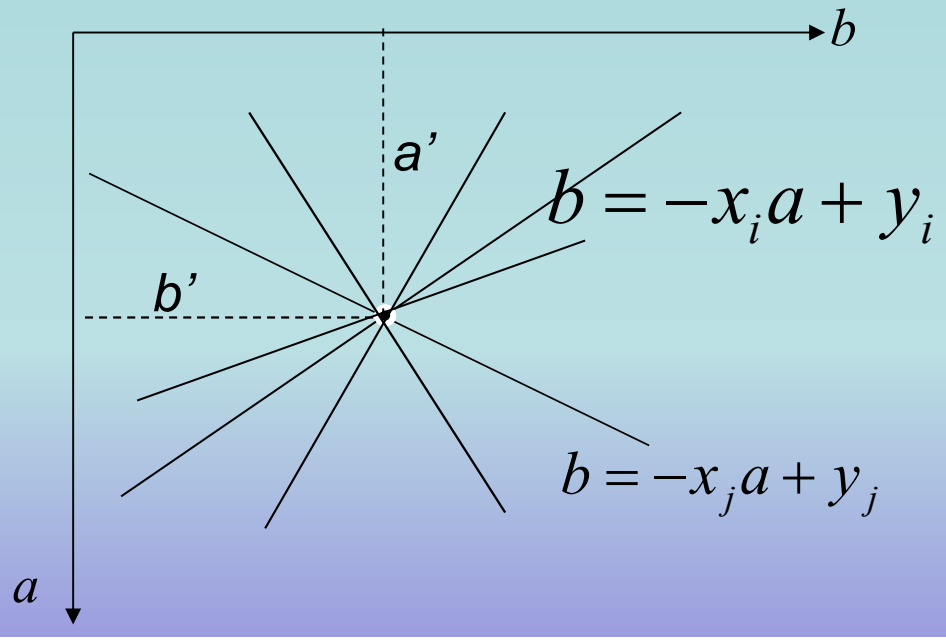
Hough Transform

- If we have another point (x_j, y_j), its equation is

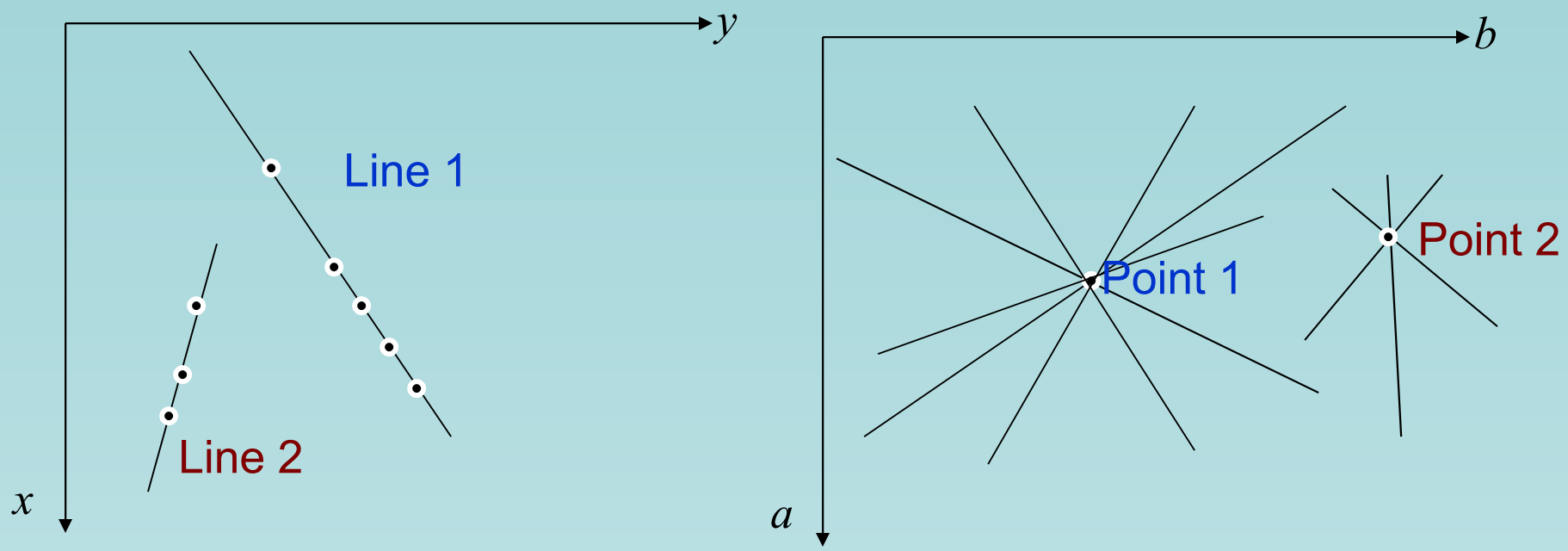
$$b = -x_j a + y_j$$

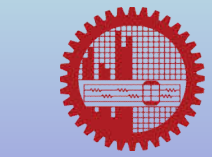
- *a'* and *b'* are slope and intercept of the line in x, y plane





Hough Transform

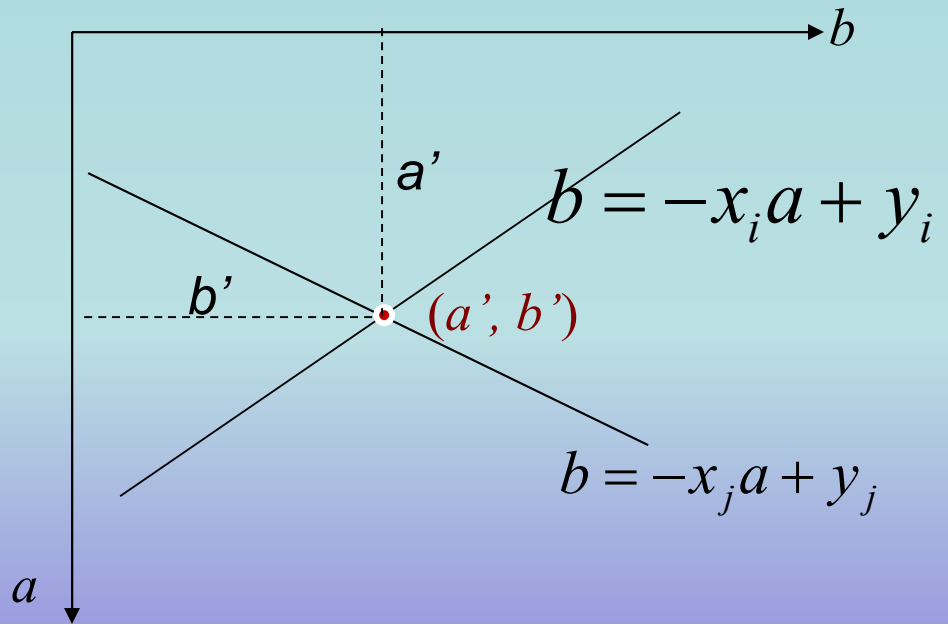




CSE-BUET

Hough Transform

- If we have *n* points, find *n* equations in *a*, *b* plane
- Find points (*a*', *b*')
- Concentrations in *a,b* plane is co-linearity in *x, y* plane



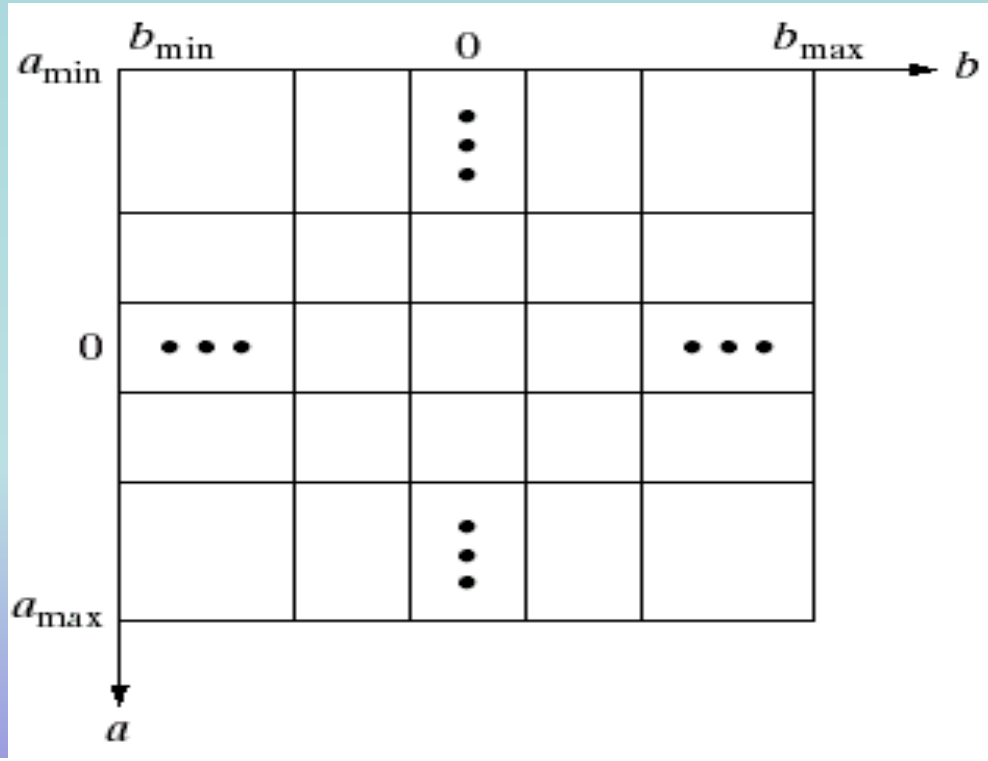
CSE-BUET

Hough Transform

- If we have *n* points, find equations in *a*, *b* plane
- Find points (*a*', *b*')
- Concentrations in *a,b* plane is co-linearity in *x, y* plane

- An accumulator is used to measure the concentration
- For each point (*x,y*)
  - select a value of *a, let is* a_p
  - find b_q

$$b_q = -xa_p + y$$

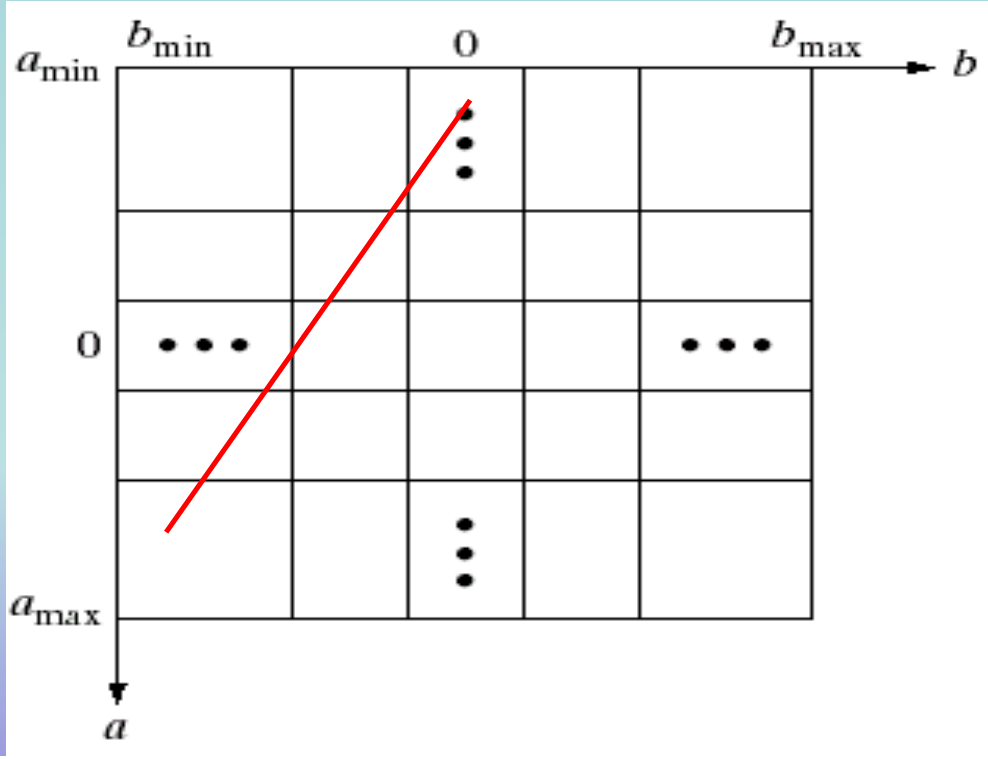


Hough Transform

- If we have *n* points, find equations in *a*, *b* plane
- Find points (*a*', *b*')
- Concentrations in *a,b* plane is co-linearity in *x, y* plane

- An accumulator is used to measure the concentration
- For each point (*x,y*)
  - select a value of *a, let is* a_p
  - find b_q

$$b_q = -xa_p + y$$



CSE-BUET

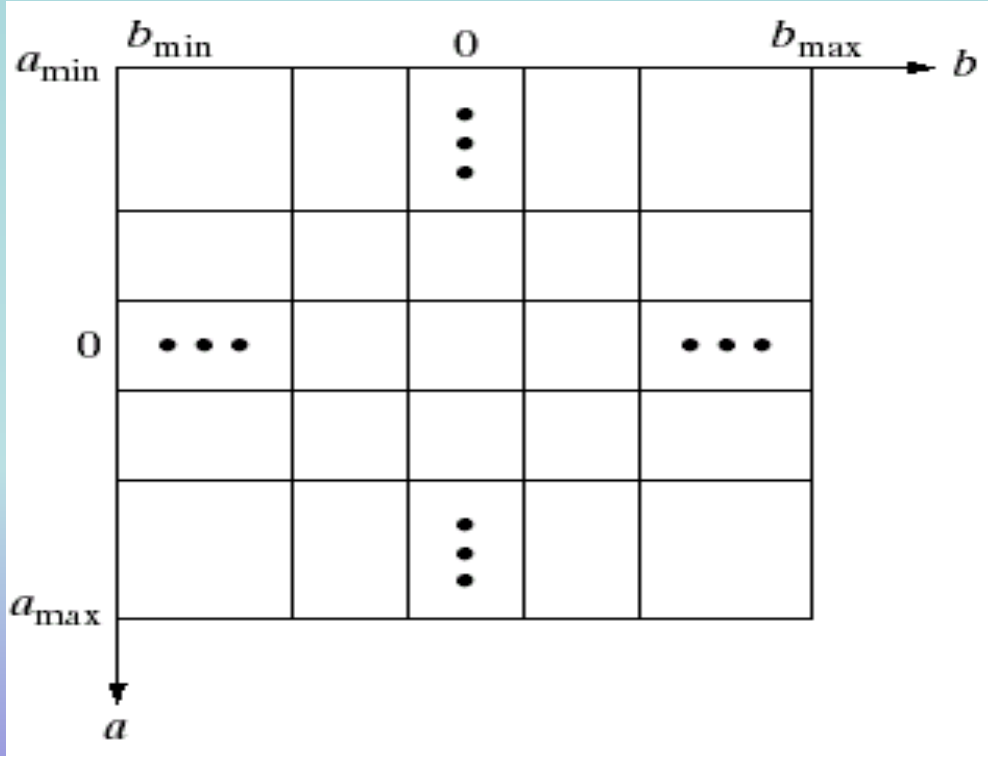
Hough Transform

- If we have *n* points, find equations in *a*, *b* plane
- Find points (*a*', *b*')
- Concentrations in *a,b* plane is co-linearity in *x, y* plane

- select a value of *a, let is* a_p
- find b_q

$$b_q = -xa_p + y$$

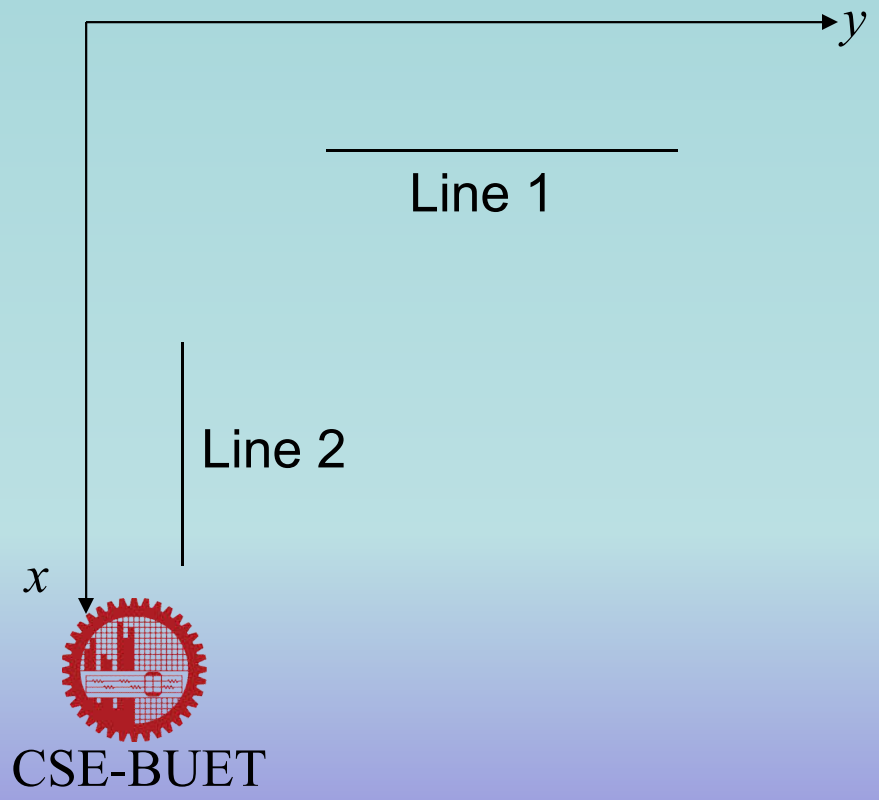
- Increase *A*(*p. q*) by 1

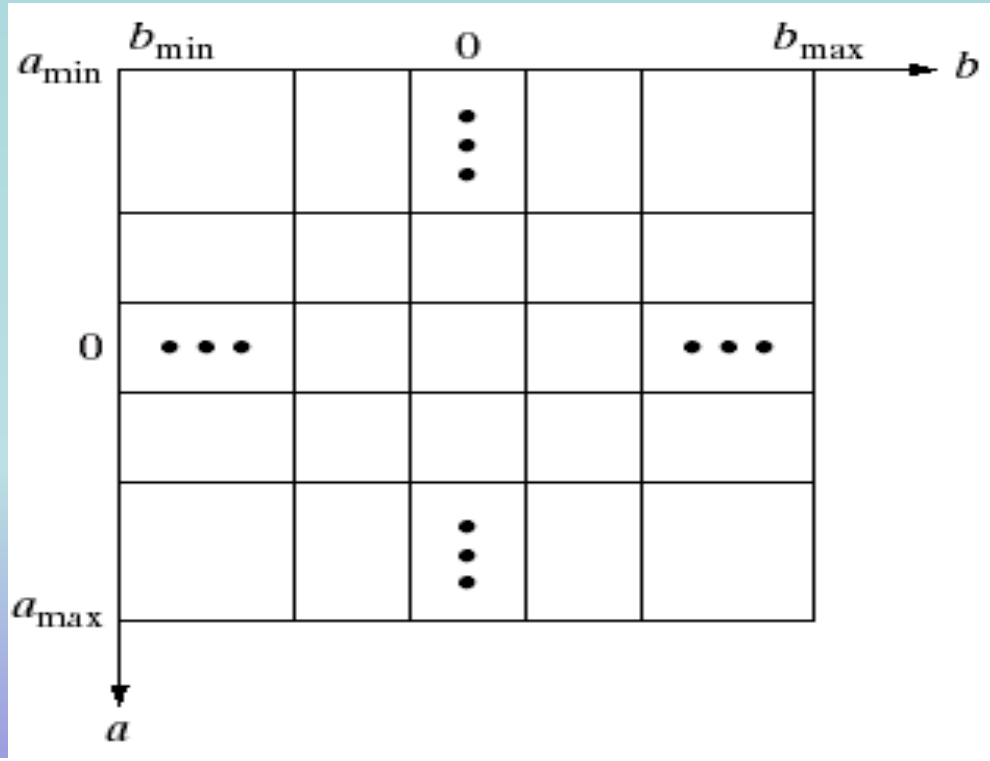


CSE-BUET

Hough Transform: Problem

- What will be the maximum value of a?





CSE-BUET

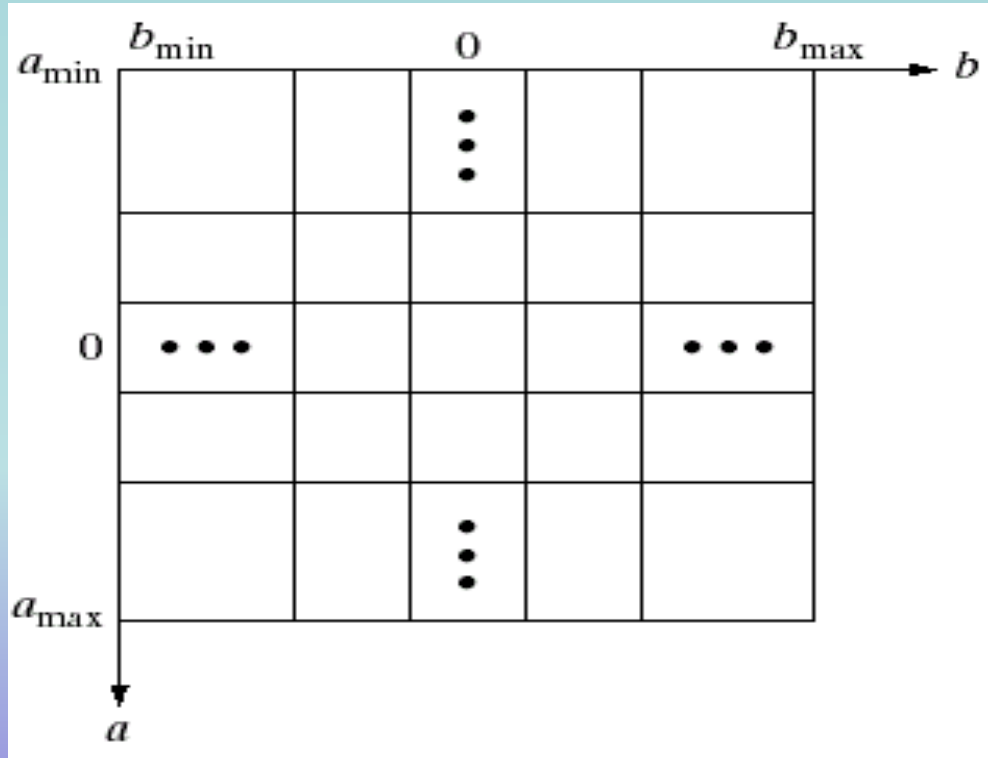
Hough Transform: An Alternate Approach

use

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

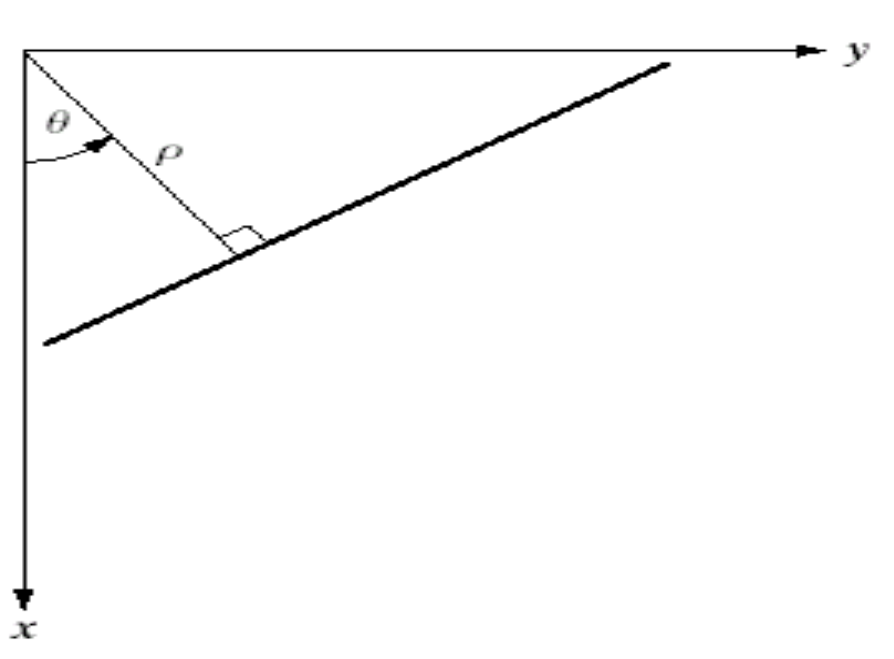
instead of

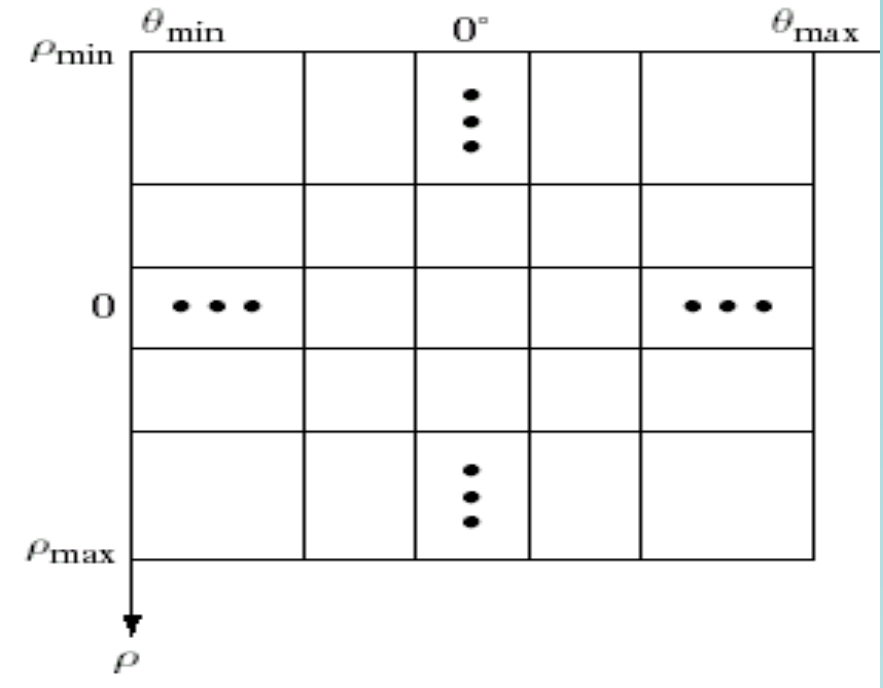
$$b = -xa + y$$



CSE-BUET

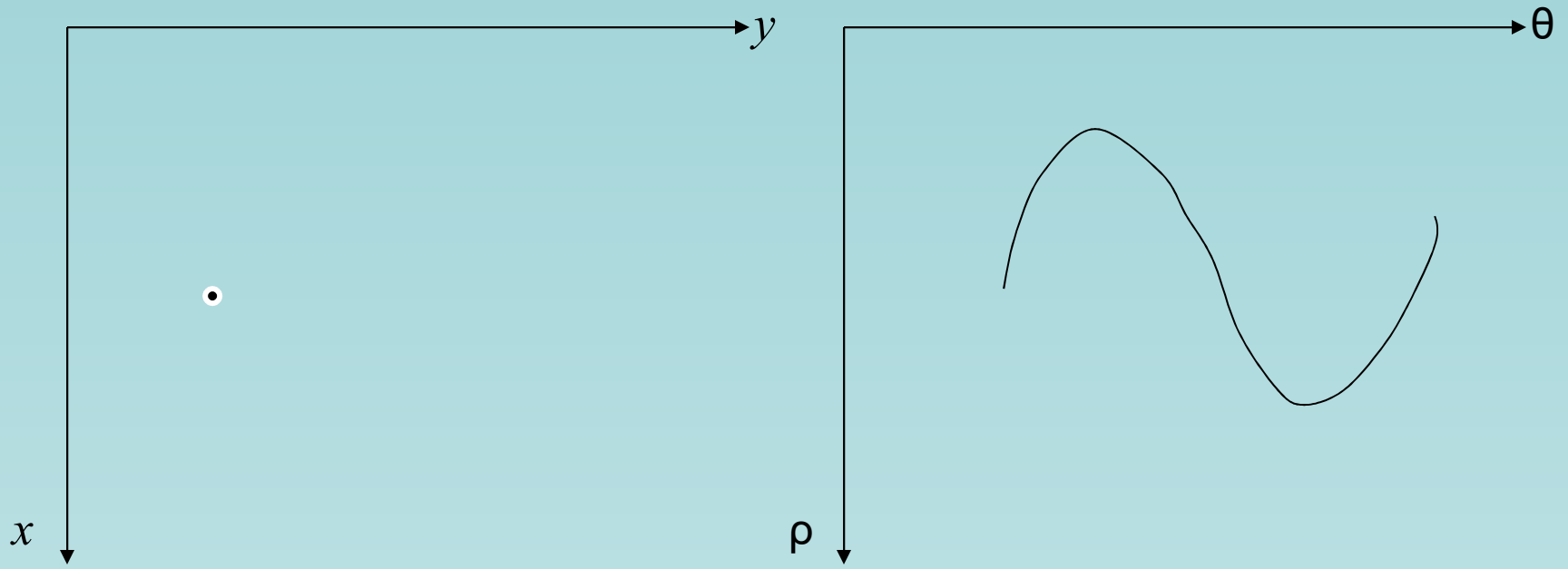
Hough Transform : An Alternate Approach





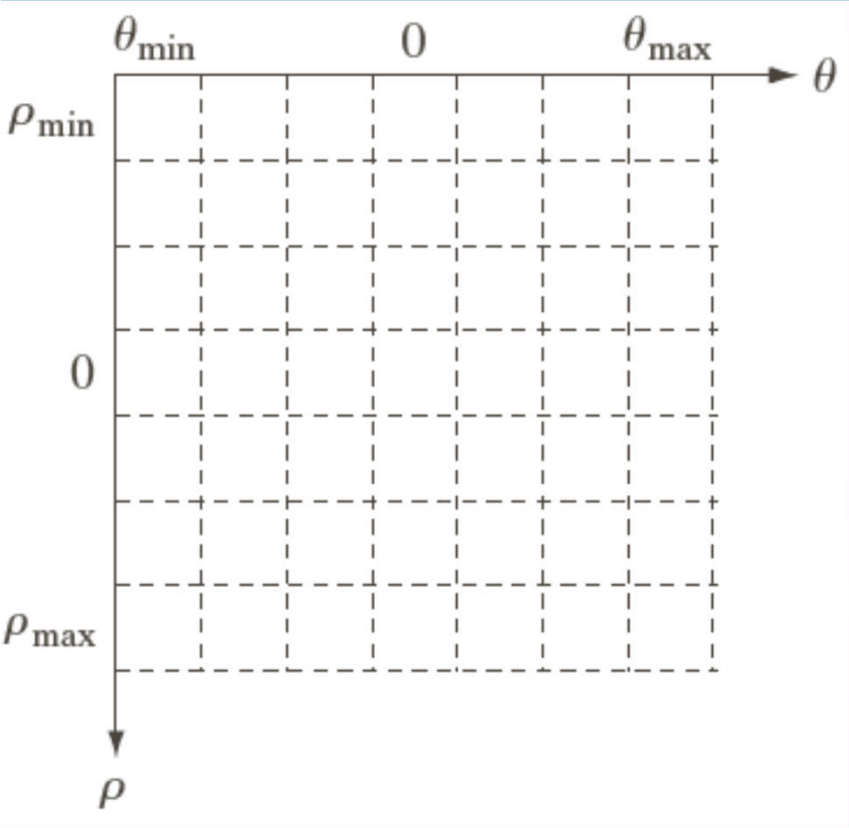
- θ varies between -90 and +90

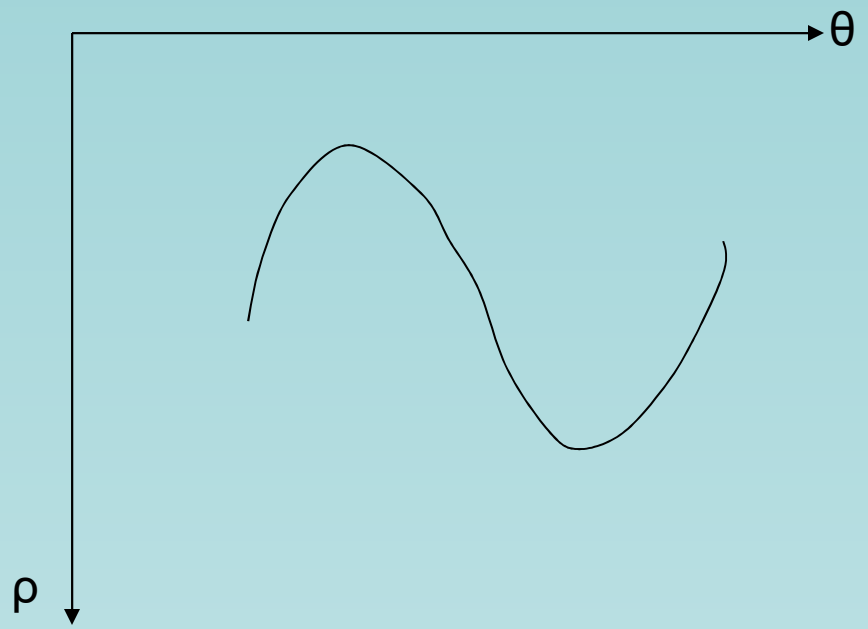
Hough Transform : An Alternate Approach



- Each point in x, y plane corresponds to a sinusoidal curve in ρ, θ plane

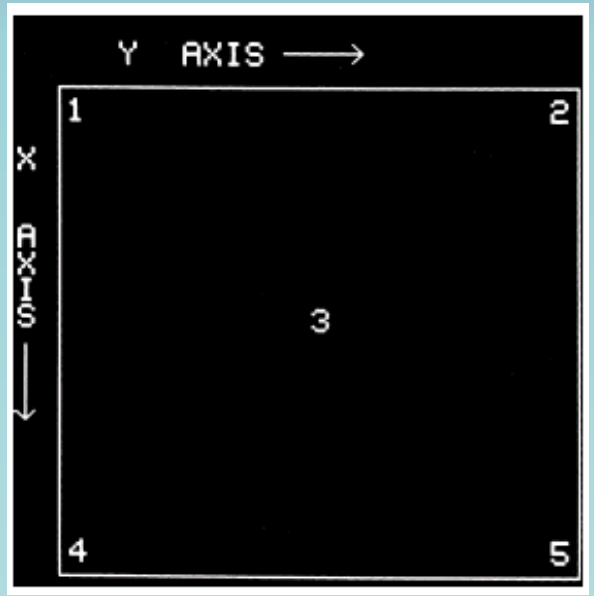
Hough Transform : An Alternate Approach





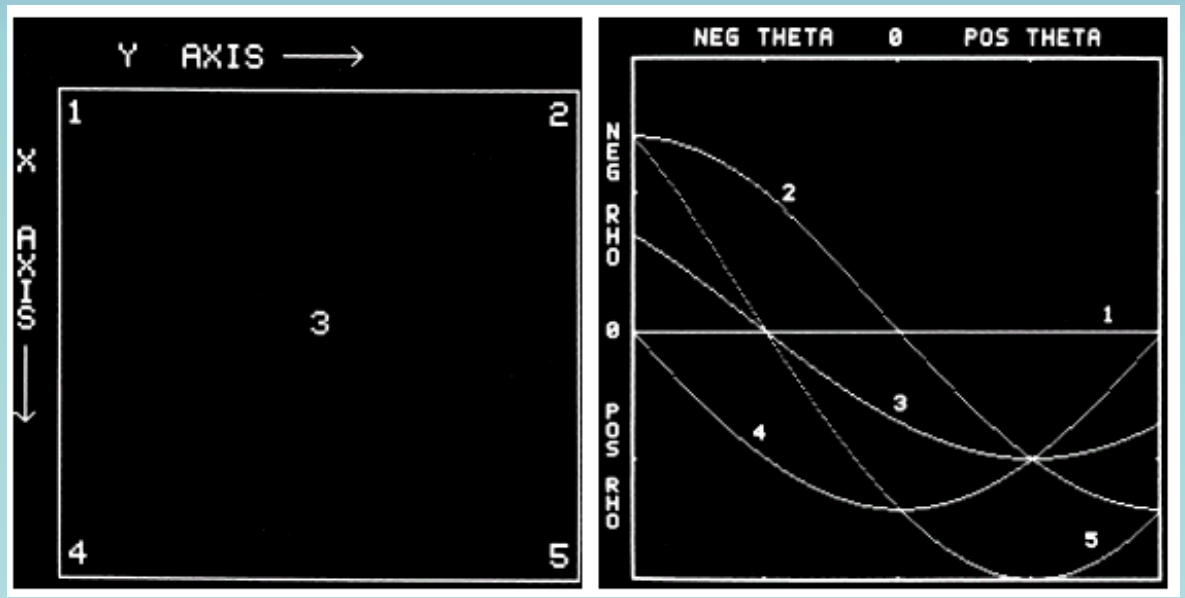
CSE-BUET

Example of Hough Transform



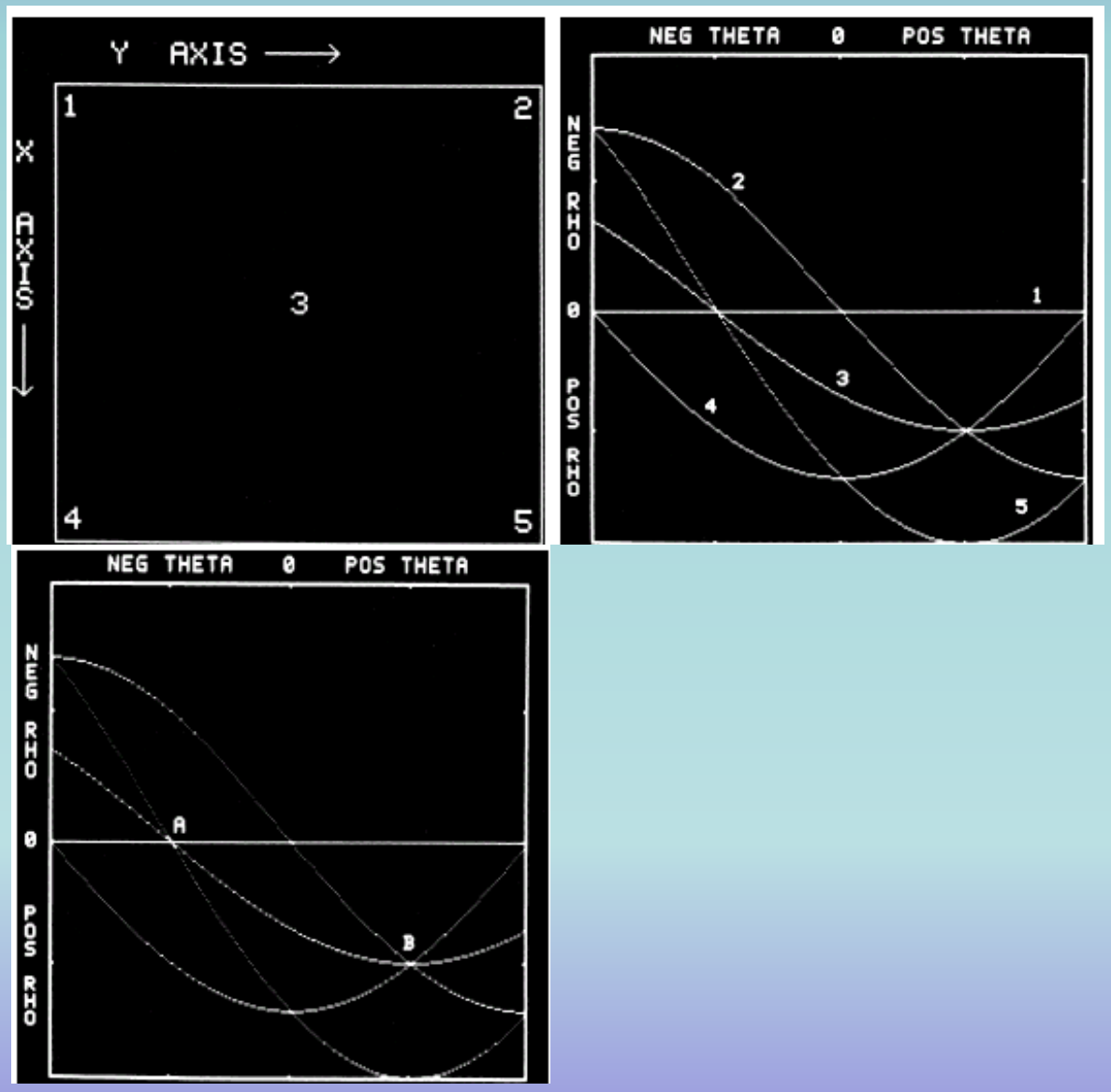
CSE-BUET

Example of Hough Transform



CSE-BUET

Example of Hough Transform



CSE-BUET

Hough Transform: With Other Shape

- Hough transform can be used to detect other shapes as well
- In general,

$$g(\mathbf{v}, \mathbf{c}) = 0$$

Hough Transform: With Other Shape

- Hough transform can be used to detect other shapes as well
- For example, to detect pixels on circle, use,

$$(x-c_1)^2+(y-c_2)^2=c_3^2$$

Hough Transform: With Other Shape

- Hough transform can be used to detect other shapes as well
- For example, to detect pixels on circle, use,

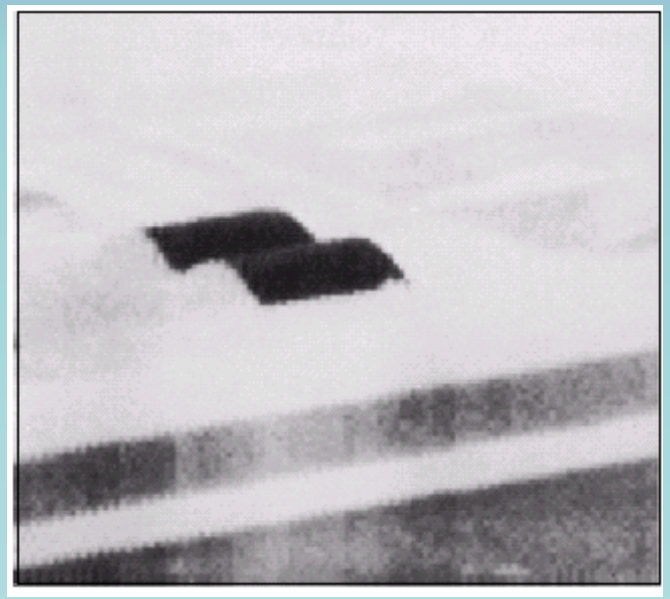
$$(x - c_1)^2 + (y - c_2)^2 = c_3^2$$

- Three parameters!
- The accumulator, A, will be a 3D cubelike space
- So fill in $A(i, j, k)$

Steps to Use Hough Transform

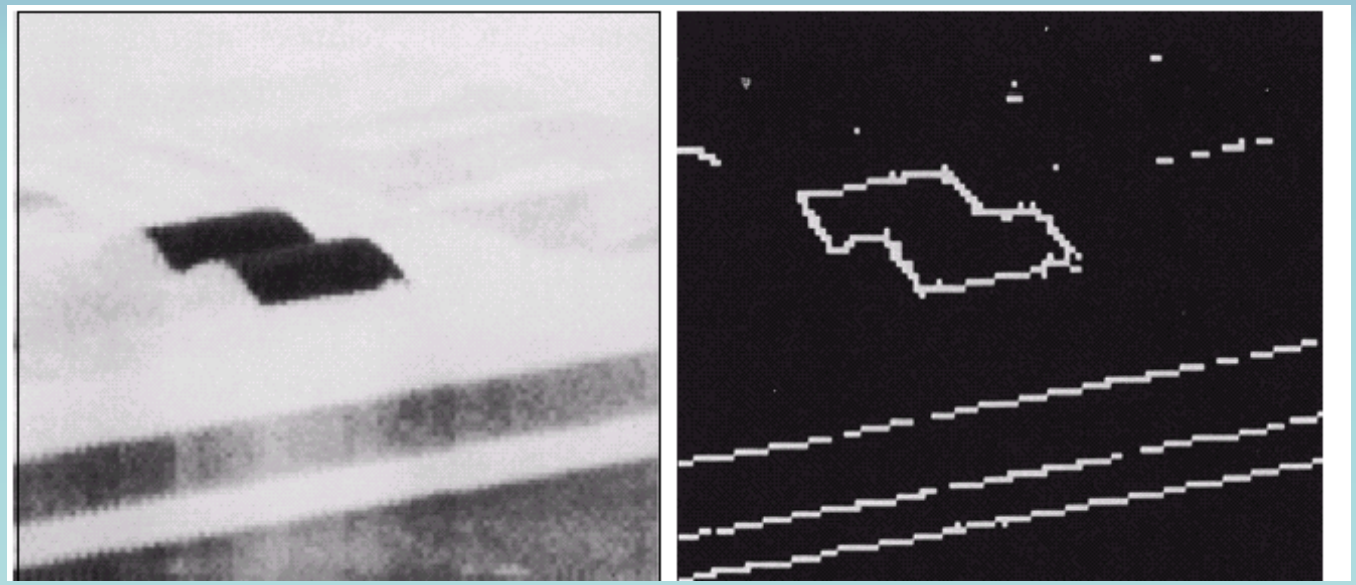
- Compute gradient image
- Threshold the gradient image to get binary image
- Specify subdivision in ρ, θ plane
- Fill in the accumulator cell
- Examine the accumulator cells for high pixel concentrations
- Examine the relationship between pixels in a chosen cell

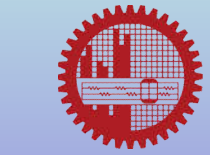
Example of Hough Transform



CSE-BUET

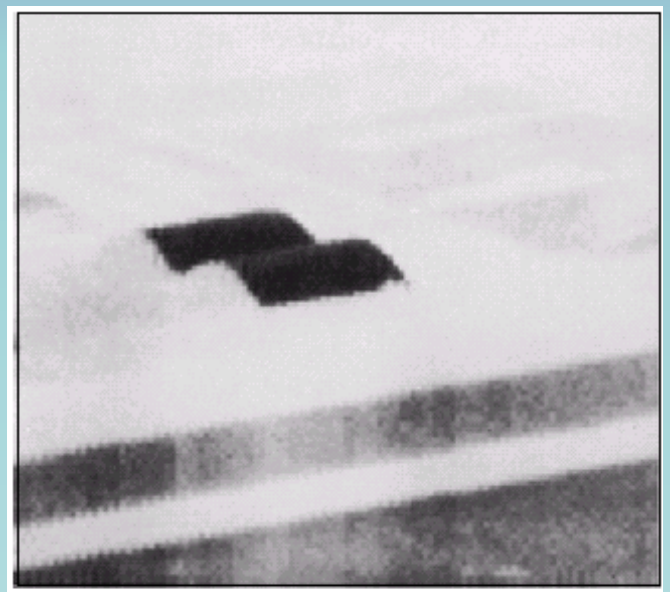
Example of Hough Transform

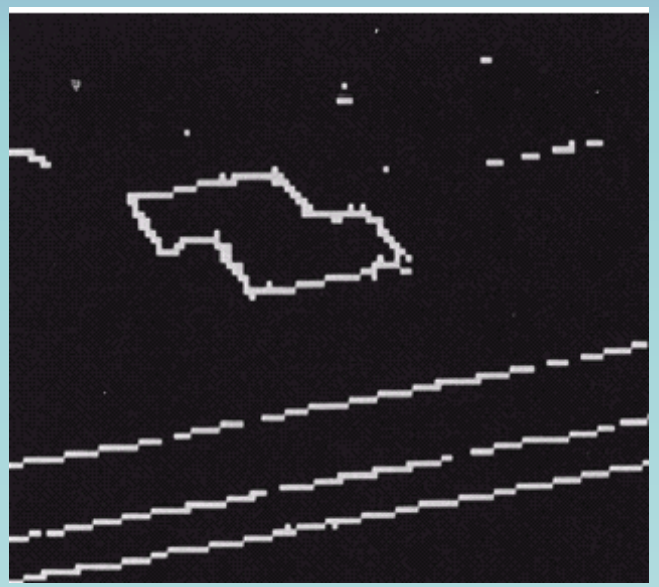


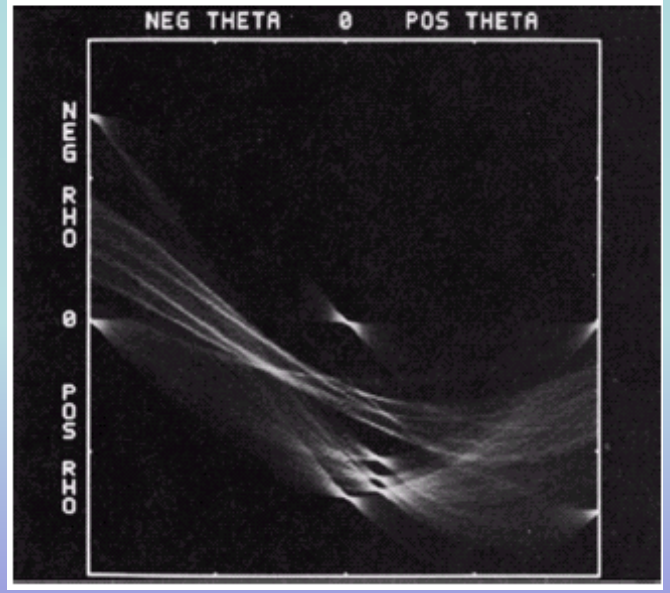


CSE-BUET

Example of Hough Transform

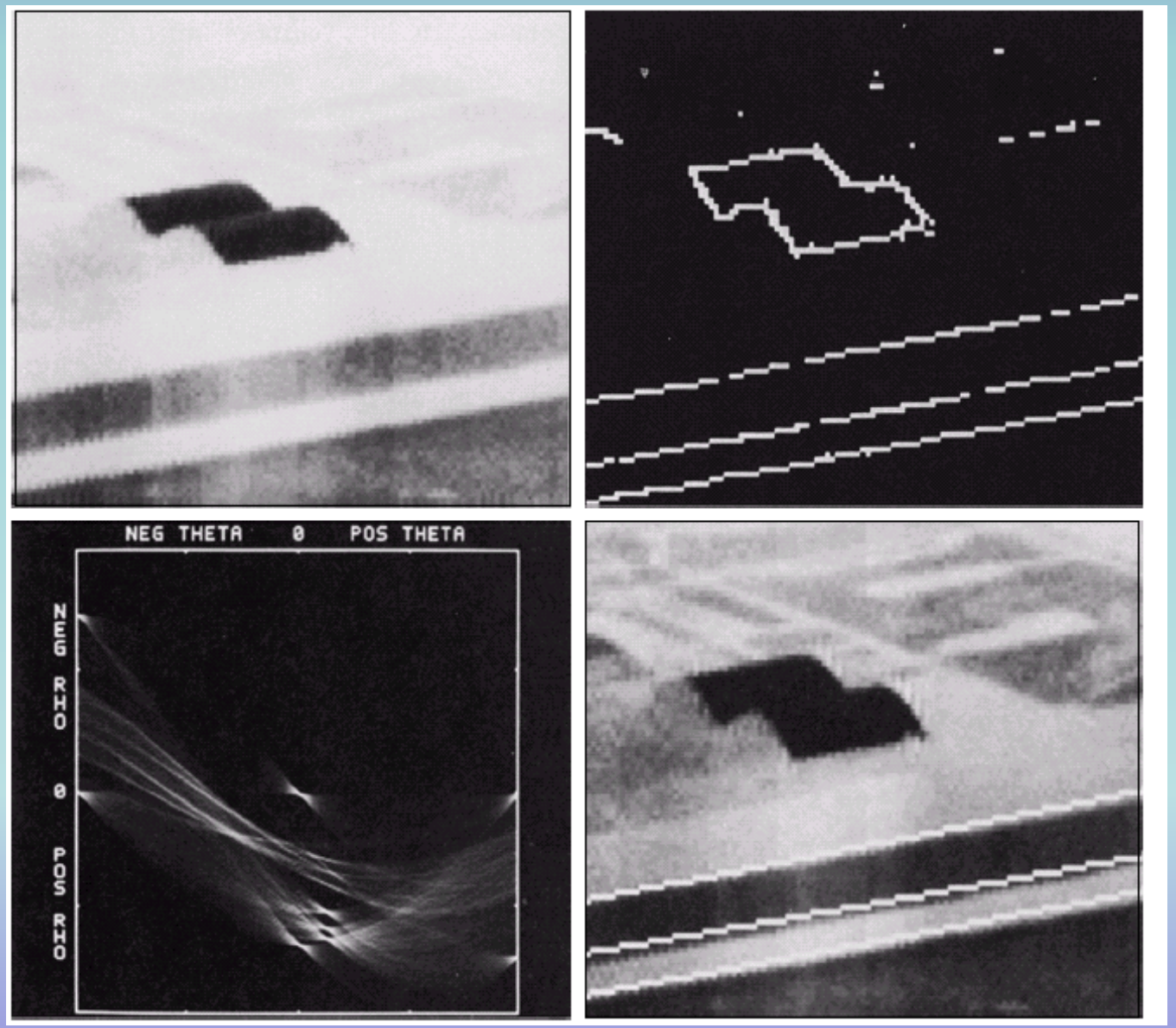


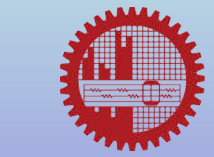




CSE-BUET

Example of Hough Transform





CSE-BUET

Edge Linking

- Local Processing
- Regional Processing
- Global Processing
  - Hough transform
  - Graph theoretic approach

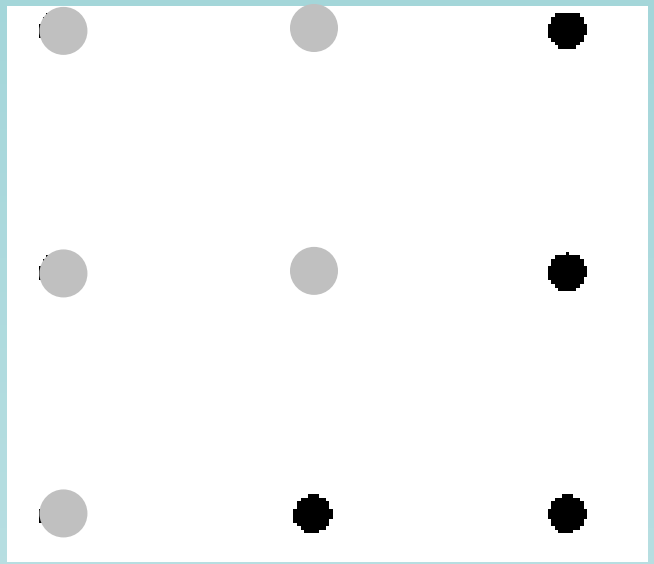
Global Processing via Graphical Approach

- Definition:
- $G = (N, U)$
- (n_i, n_j): arch
- $c(n_i, n_j)$ = cost of arc (n_i, n_j)
- A sequence $n_1, n_2, n_3, \ldots, n_k$, is a path with cost

$$c = \sum_{i=2}^{k} c(n_{i-1}, n_i)$$

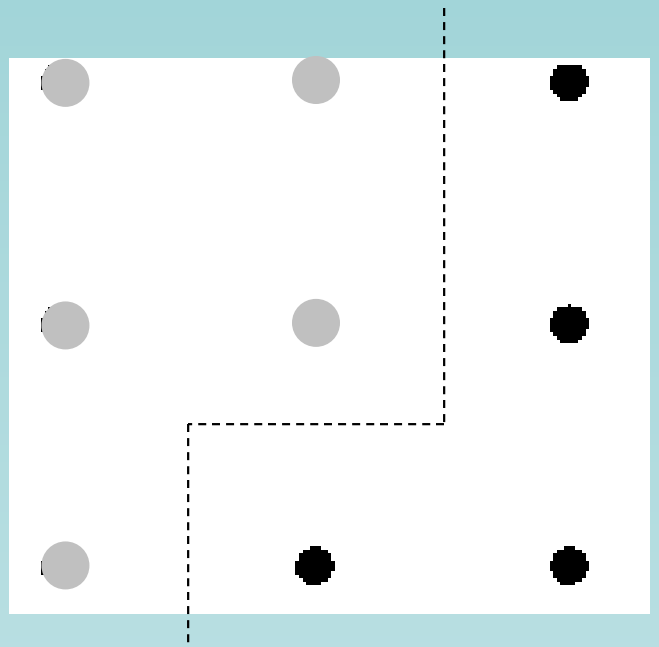
CSE-BUET

Global Processing via Graphical Approach

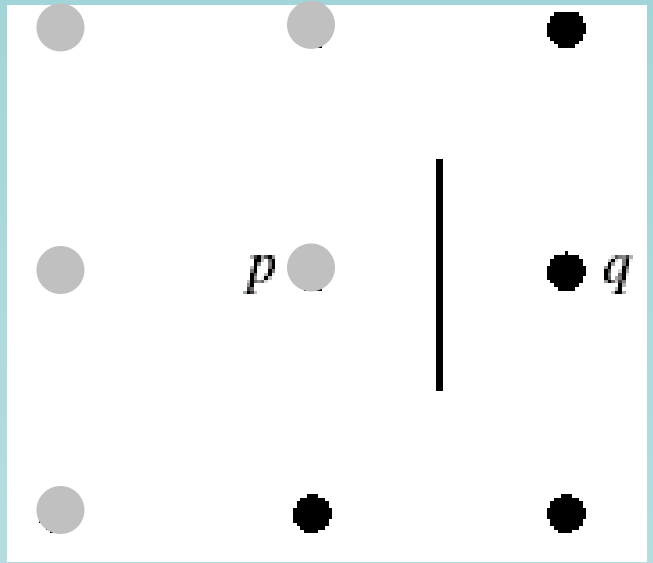


CSE-BUET

Global Processing via Graphical Approach



Global Processing via Graphical Approach

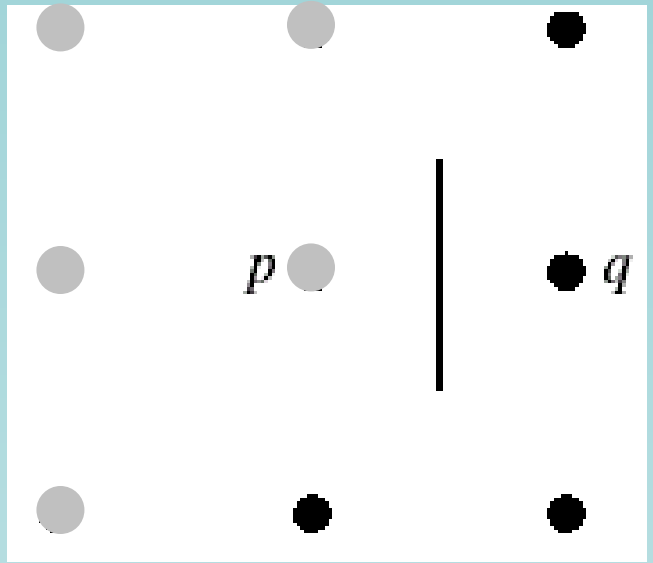


Edge element btn p and q

Can be represented as the pair ((x_p, y_p) , (x_q, y_q))

CSE-BUET

Global Processing via Graphical Approach

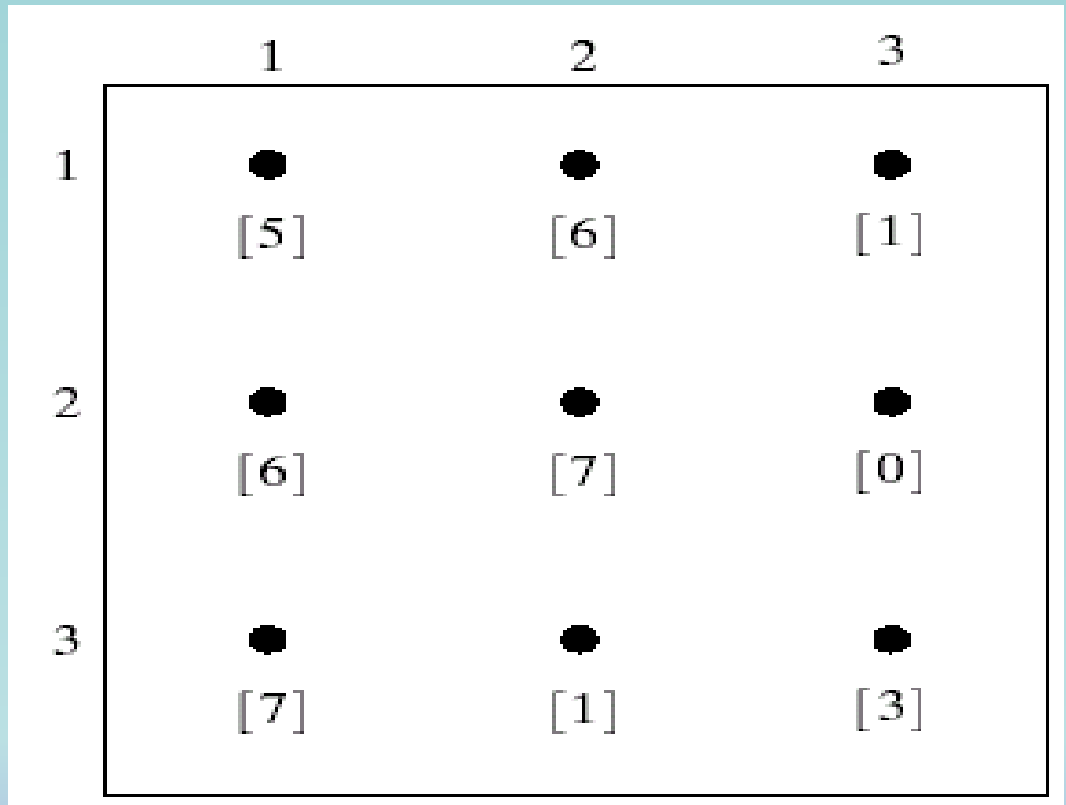


Edge element btn p and q

An edge is a sequence of edge elements

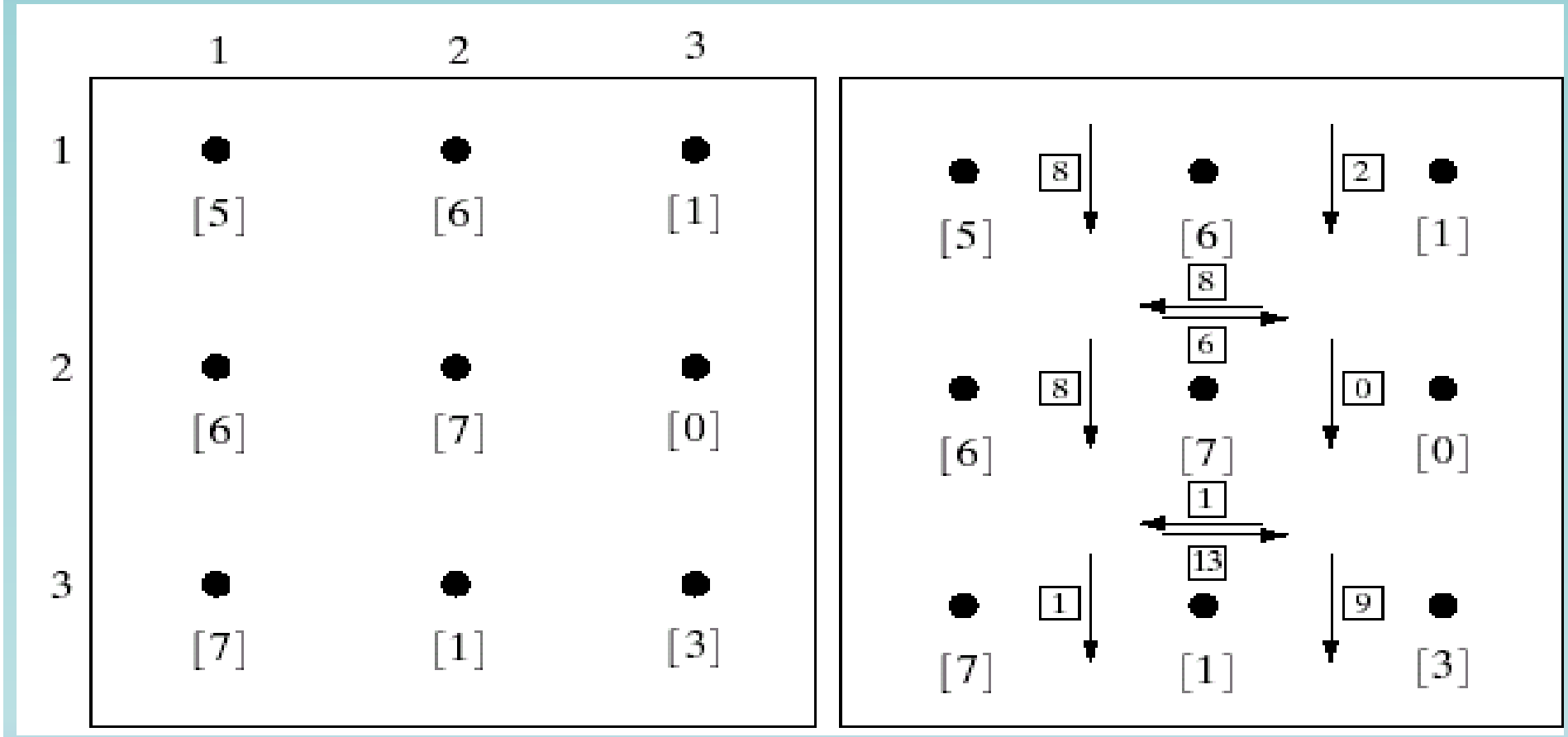
CSE-BUET

Global Processing via Graphical Approach



CSE-BUET

Global Processing via Graphical Approach

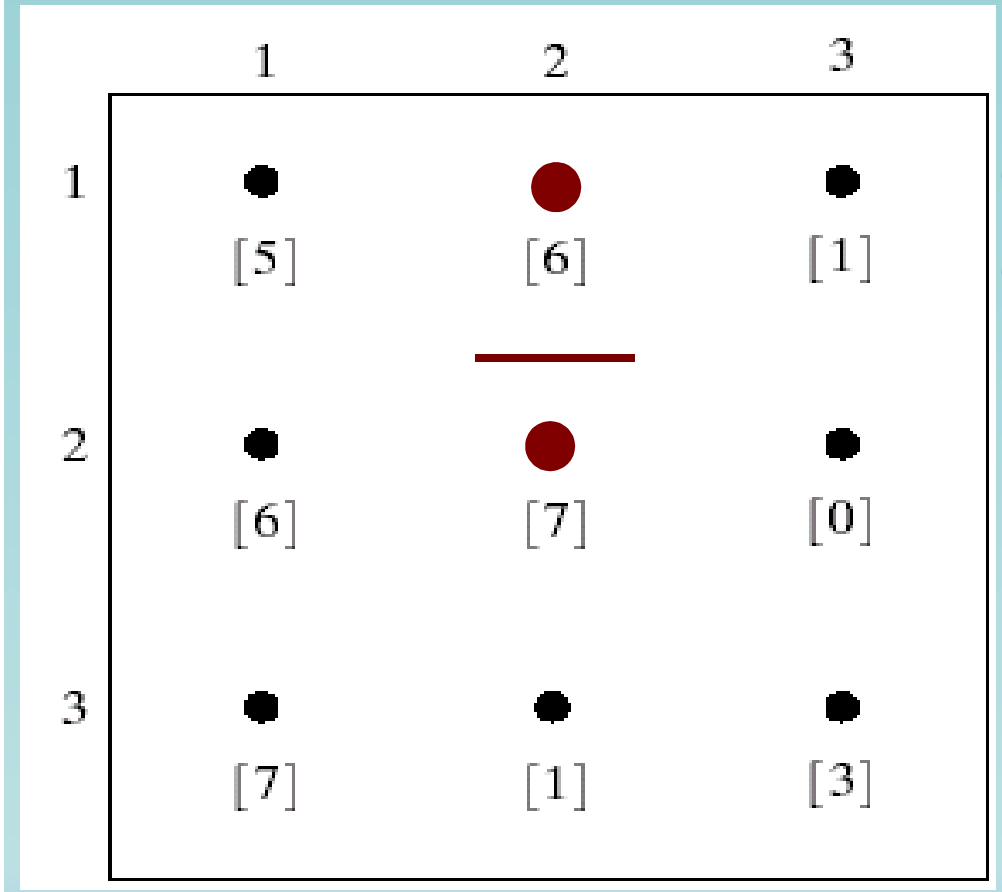


Each edge element is associated with a cost

$$c(p,q) = H - [f(p) - f(q)]$$

CSE-BUET

Global Processing via Graphical Approach



Cost of edge (1,2)(2,2) = ?

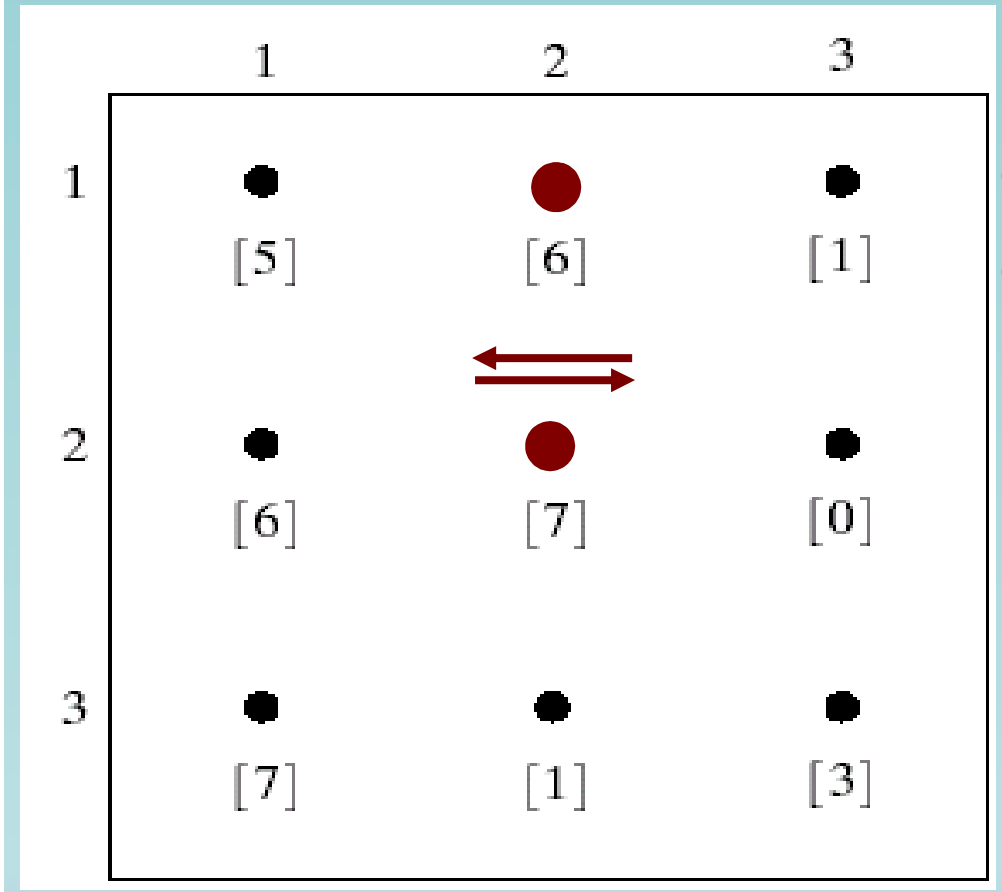
Cost of edge (2,2)(1,2) = ?

Each edge element is associated with a cost

$$c(p,q) = H - [f(p) - f(q)]$$

CSE-BUET

Global Processing via Graphical Approach



Cost of edge (1,2)(2,2) = ?

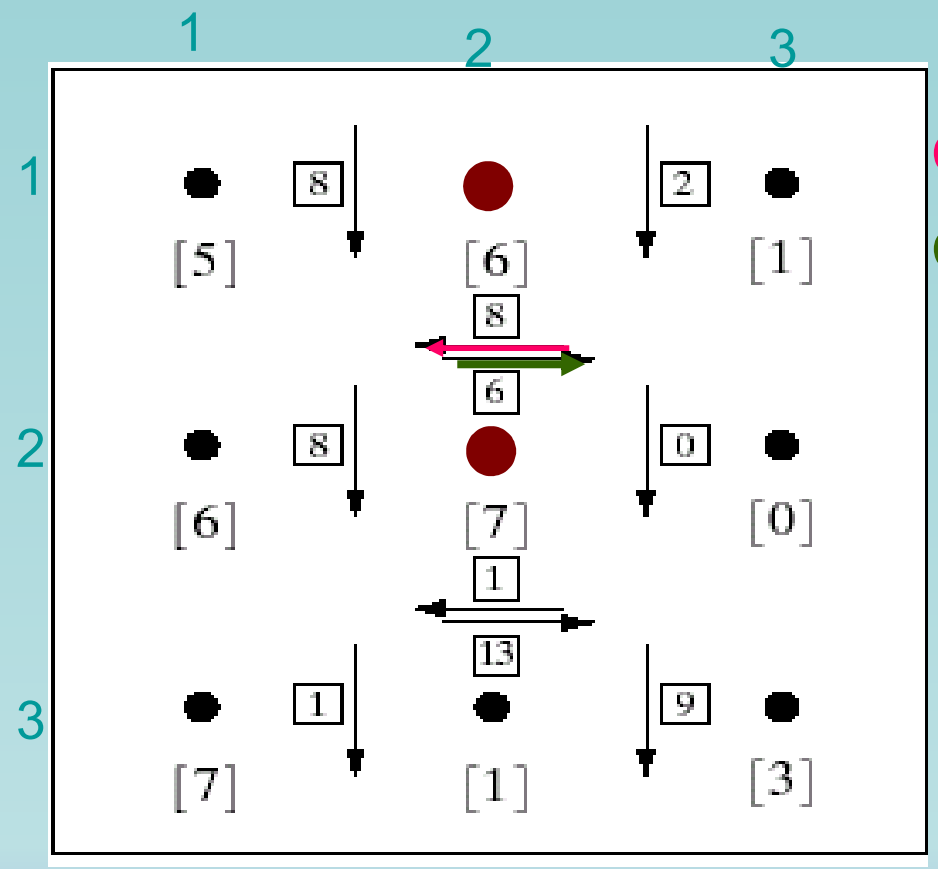
Cost of edge (2,2)(1,2) = ?

CSE-BUET

Each edge element is associated with a cost

$$c(p,q) = H - [f(p) - f(q)]$$

Global Processing via Graphical Approach



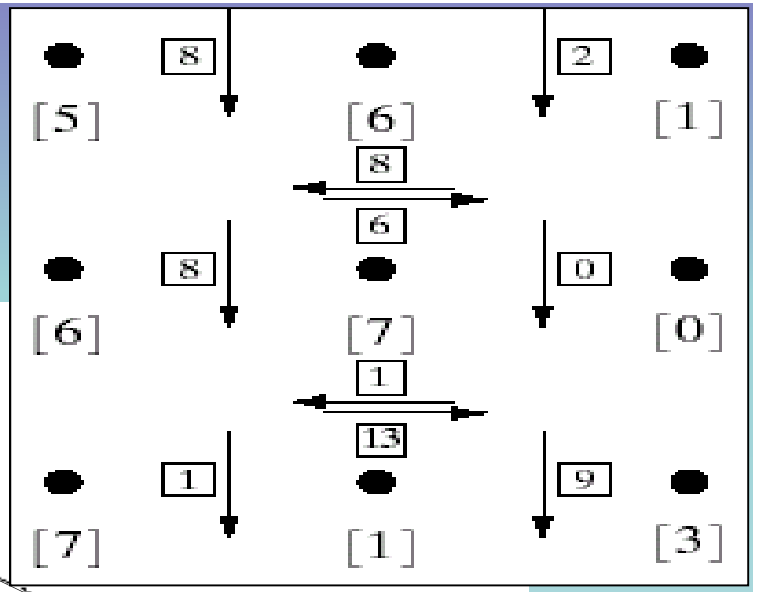
Cost of edge (1,2)(2,2) = 8: To left

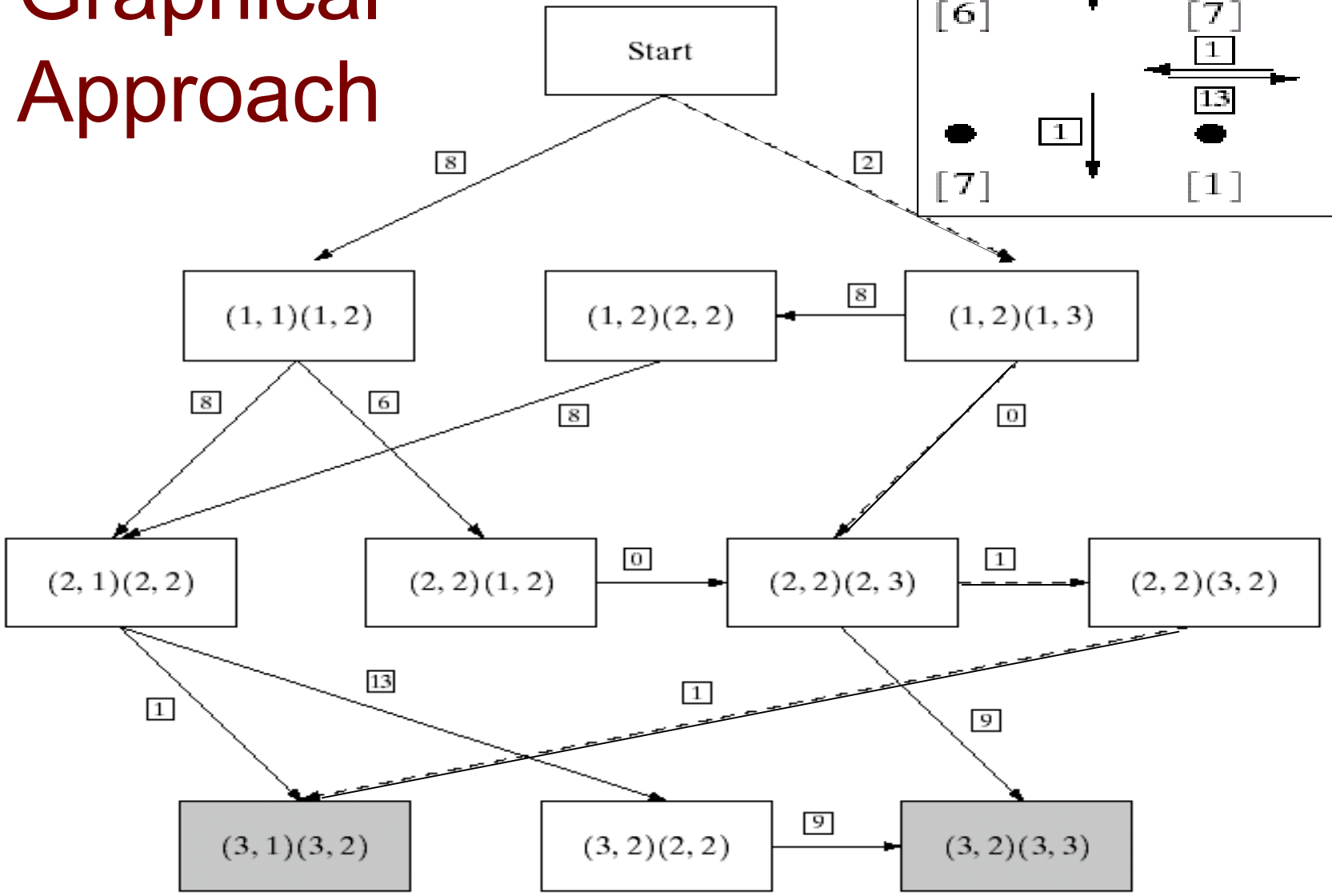
Cost of edge (2,2)(1,2) = 6: To right

Each edge element is associated with a cost

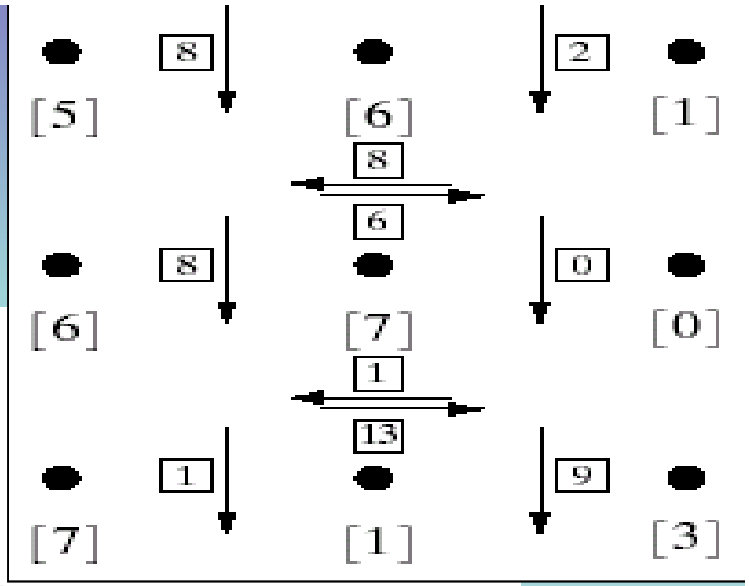
$$c(p,q) = H - [f(p) - f(q)]$$

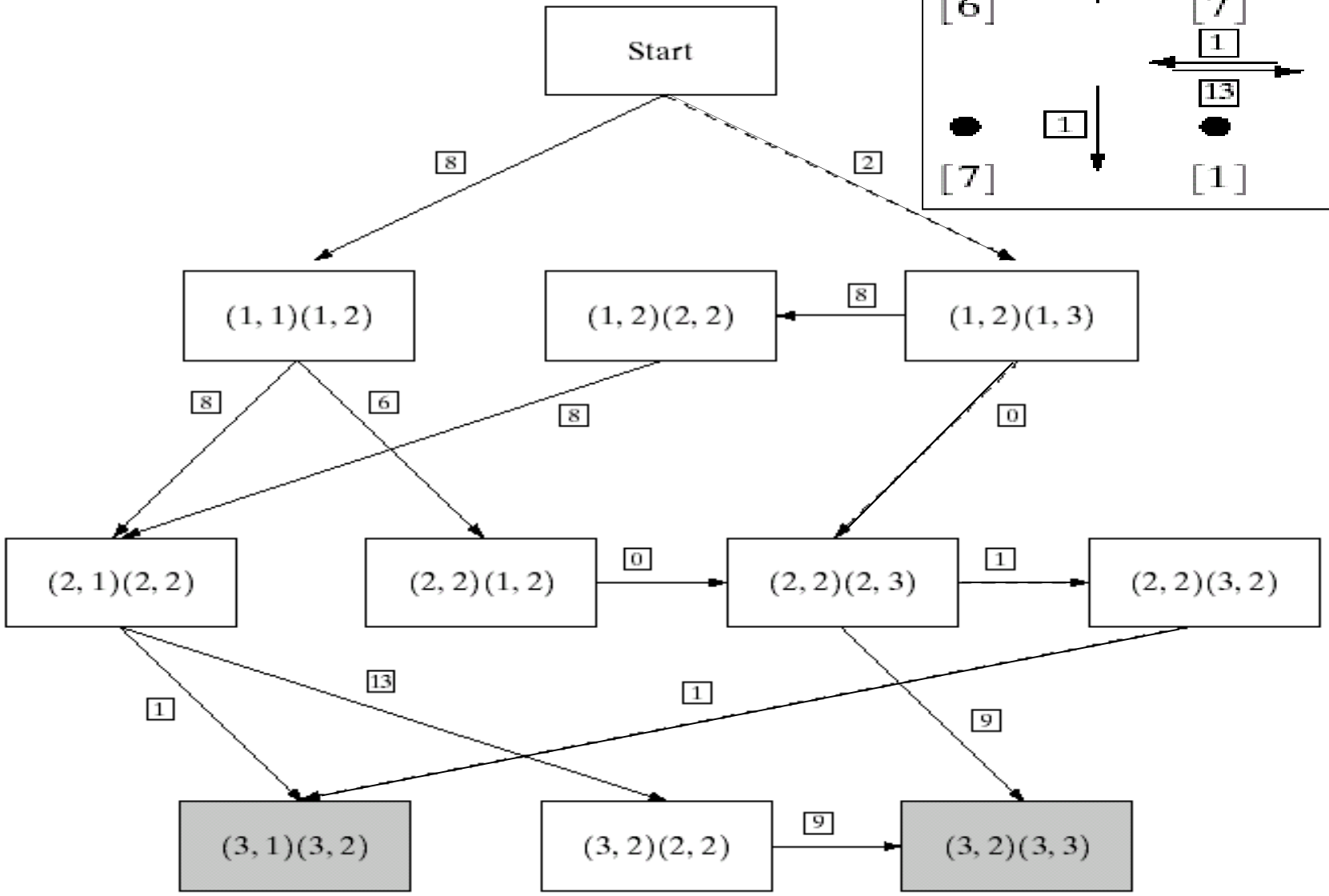
Global Processing via Graphical Approach





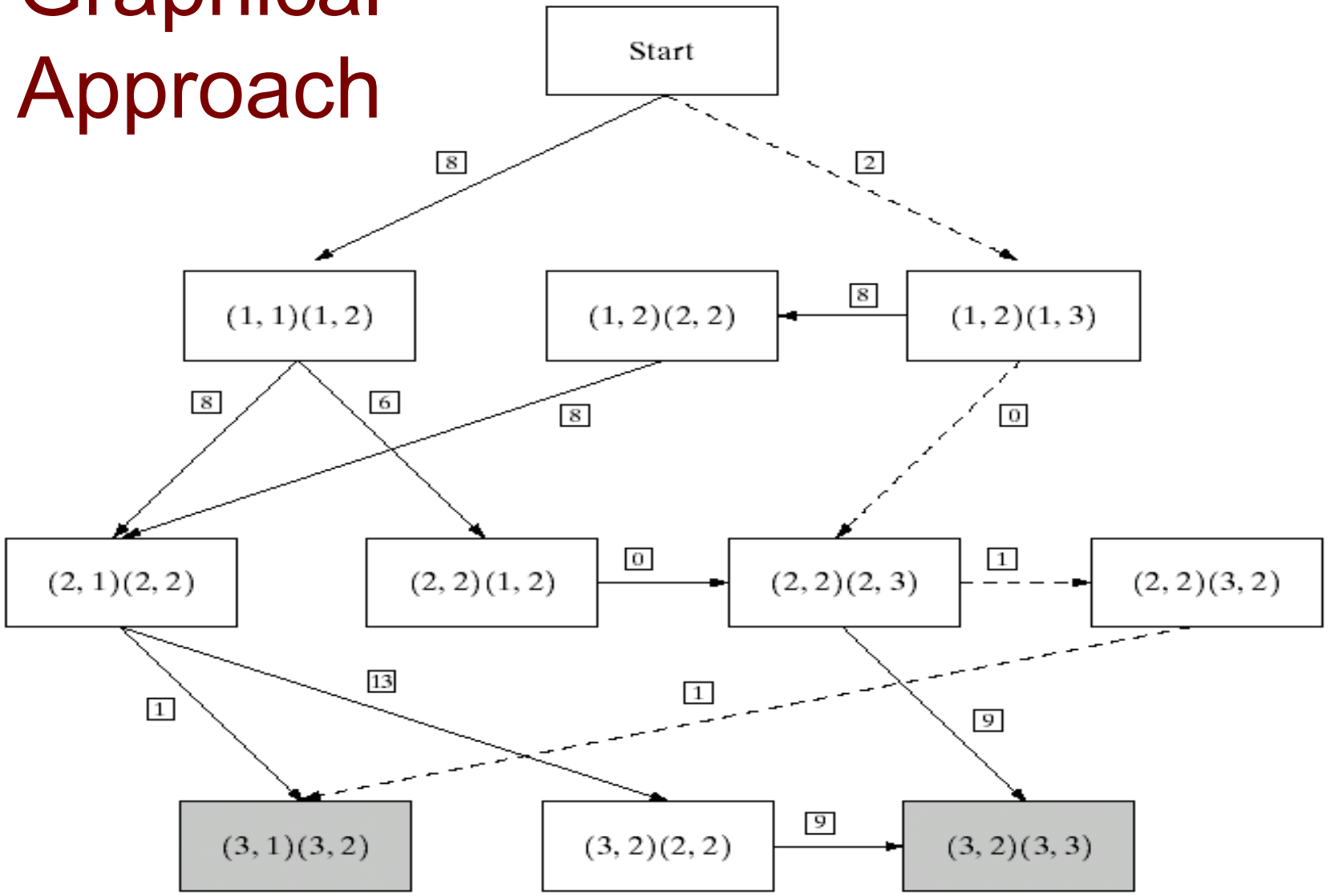
Assume: edge starts at top row and ends at bottom row only



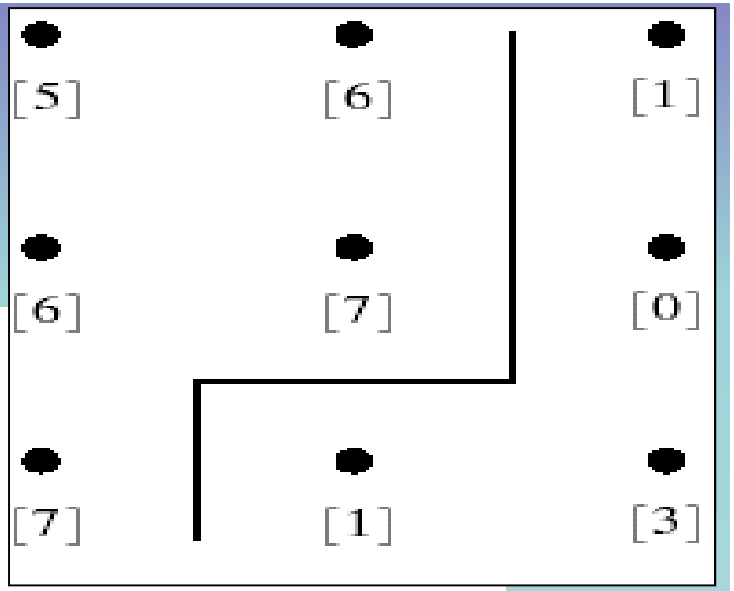


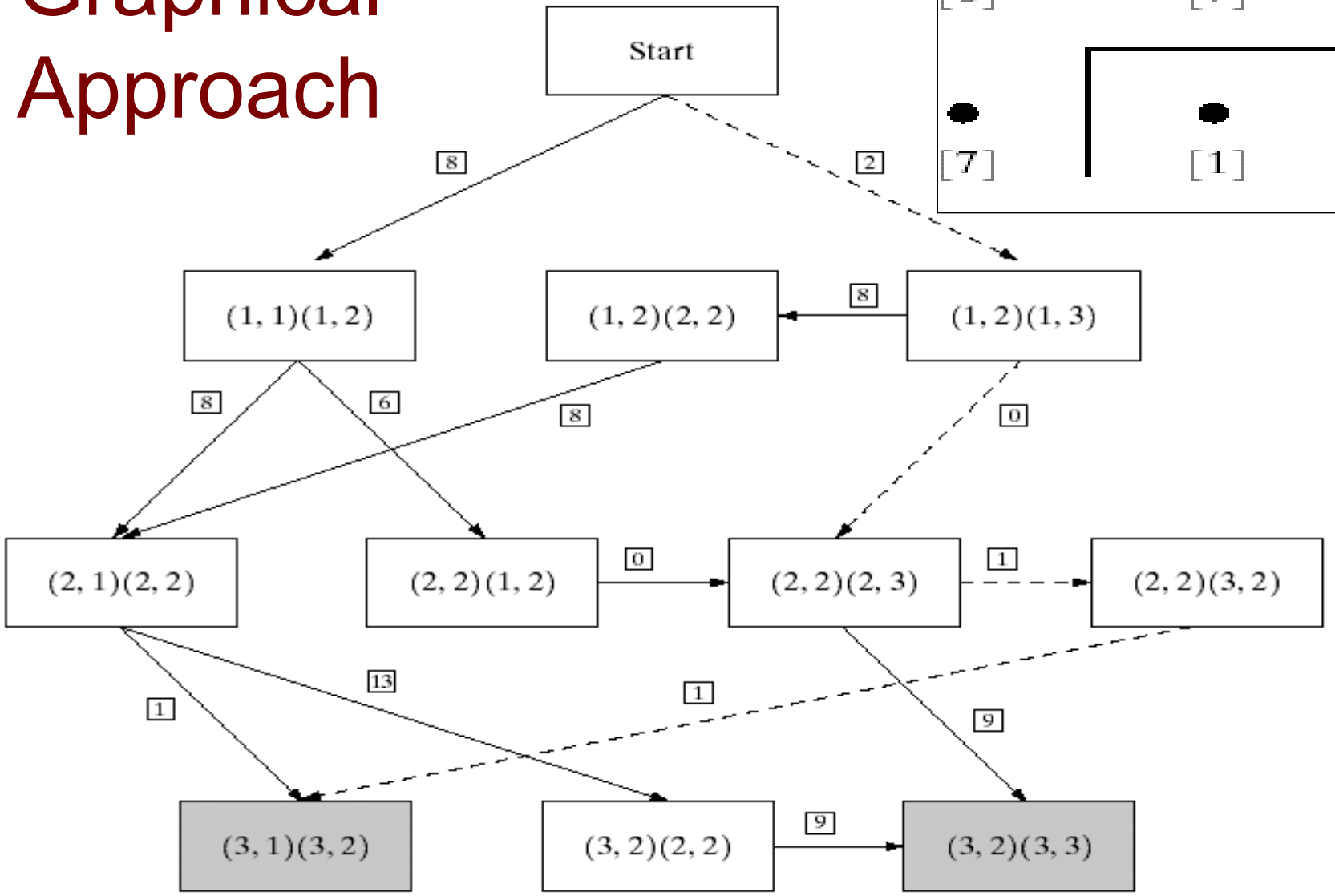
Global Processing via Graphical Approach

Lowest-cost path shown in dotted line



Global Processing via Graphical Approach





Heuristic Approach to Find a Path

- Optimal sol$^{n.}$ is costly for image segmentation
- Heuristic is applied
- Any algorithm can be used
  - Uniform, iterative deepening, A*, etc

$$r(n) = g(n) + h(n)$$

CSE-BUET