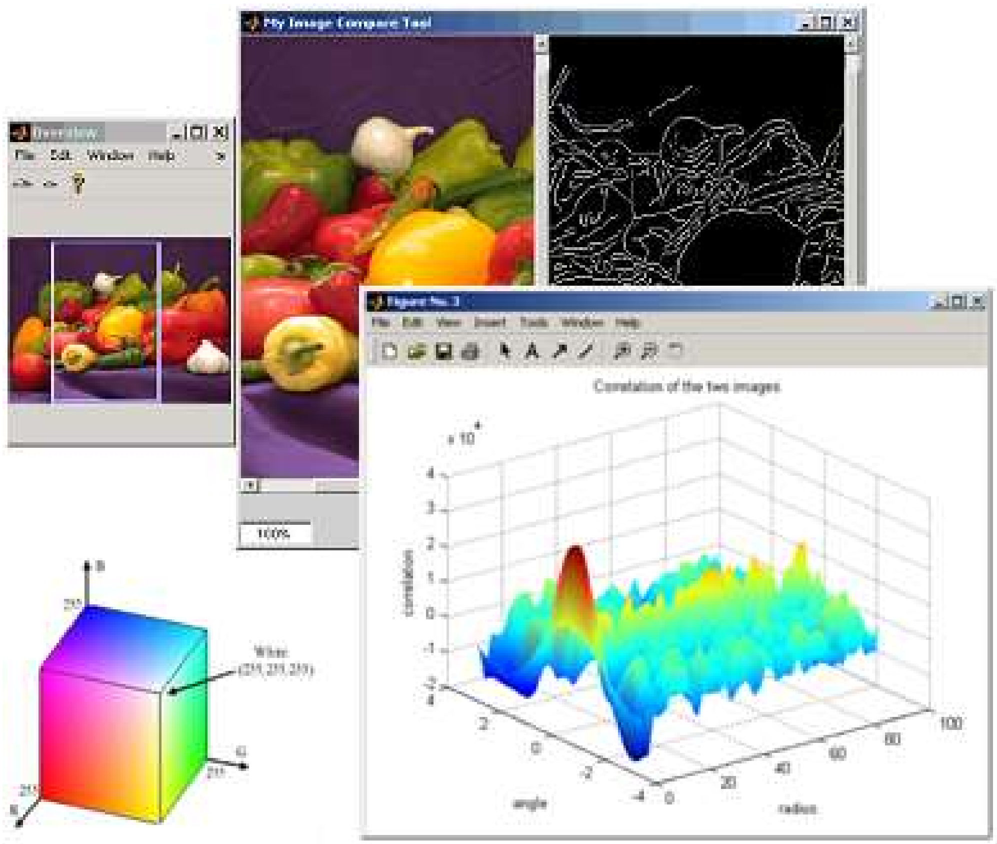
Digital Image Processing



Dr. Ajay Kumar Mahato

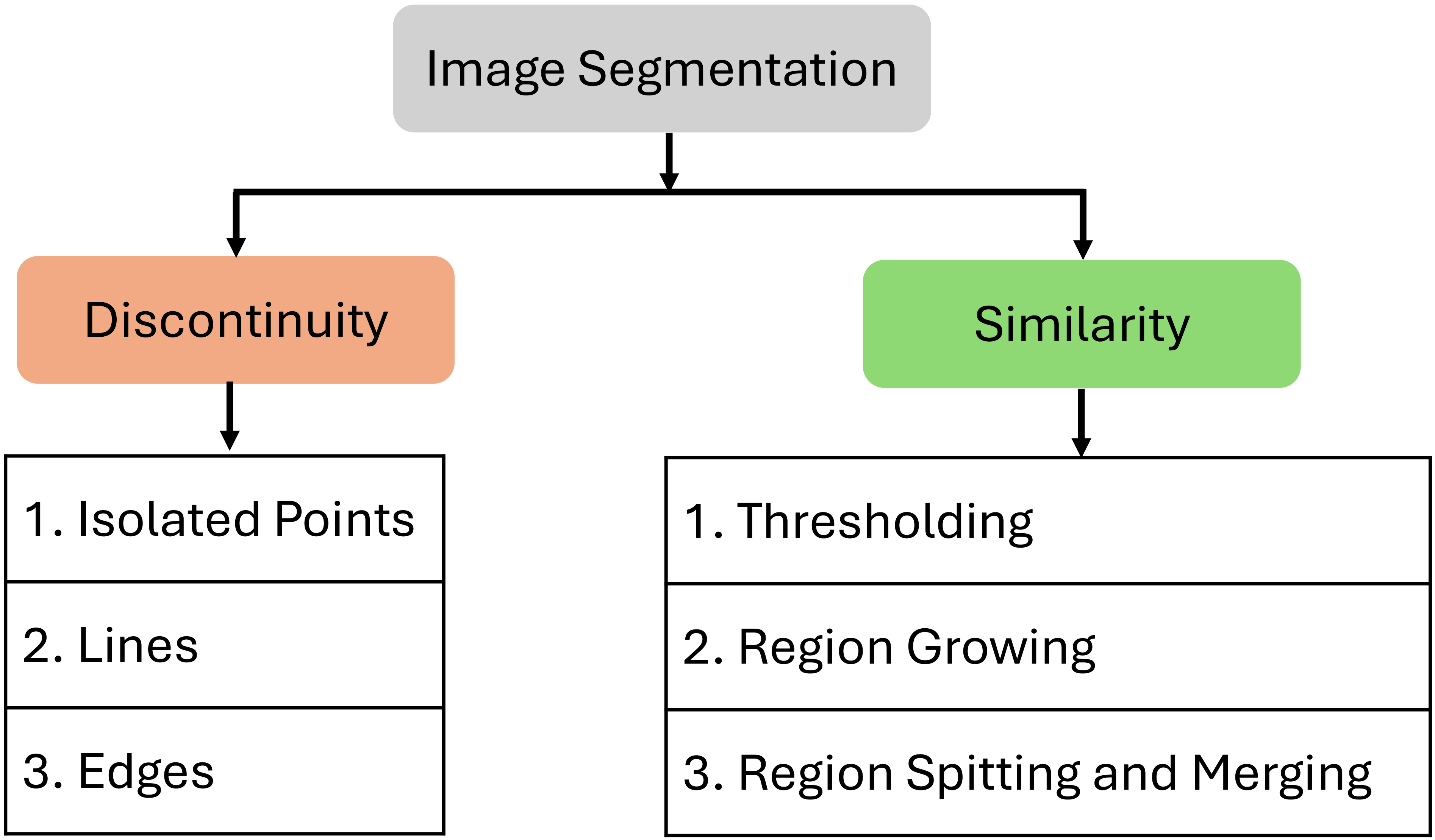
Assistant Professor
ECE Department
Gla University Mathura

DIGITAL IMAGE PROCESSING

LECTURE -19

Image Segmentation - 3

Image Segmentation Technique



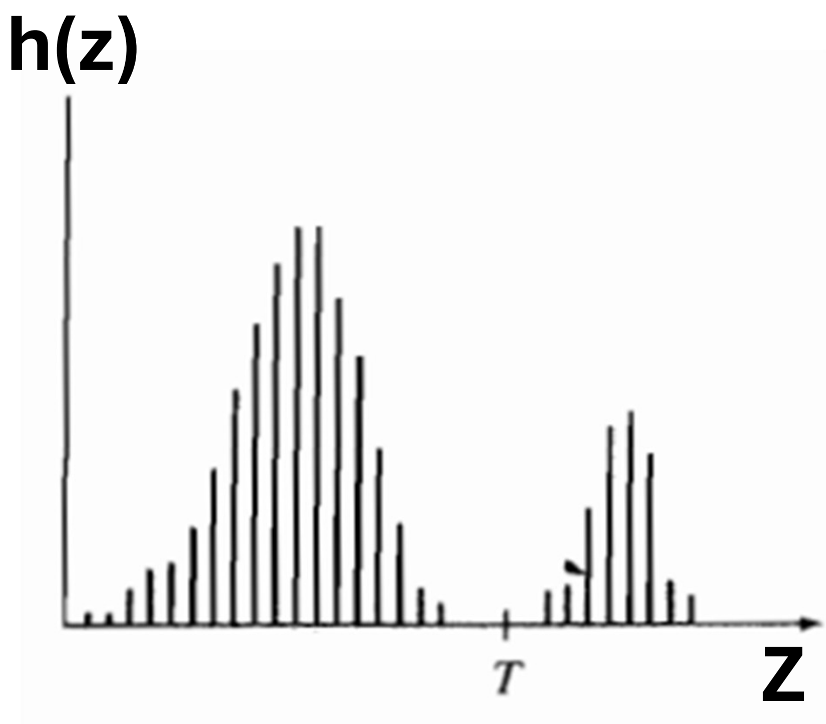
Similarity Based

Image Segmentation Technique

Thresholding

Thresholding

Case 1



- **If $f(x, y) > T$**

  It belongs to the object

- **If $f(x, y) \leq T$**

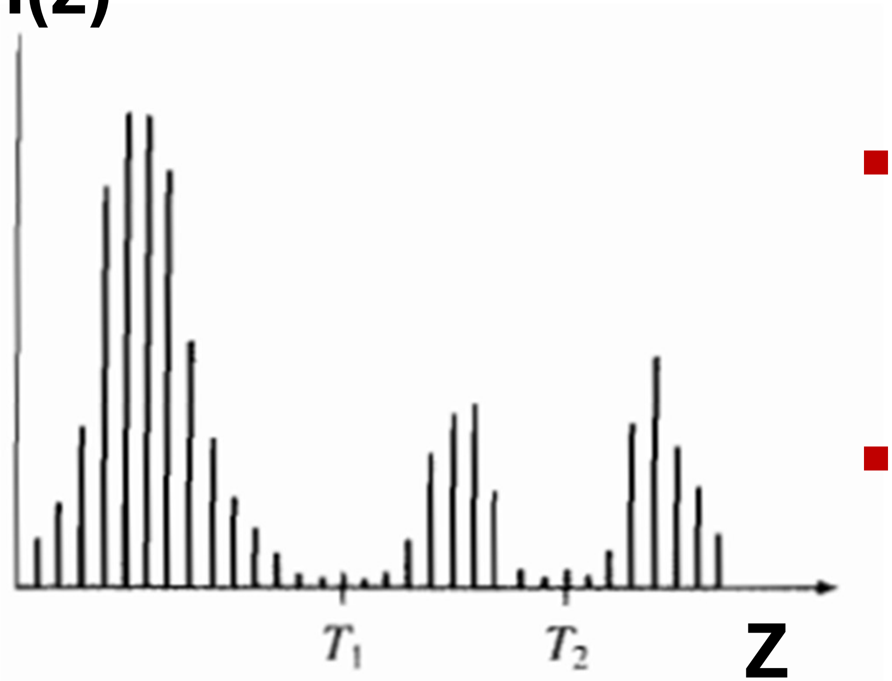
  It belongs to the background

T: be the threshold in the valley region

Thresholding

Case 2:

h(z)



- **If $f(x, y) > T_2$**
  It belongs to the object O_2
- **If $T_1 < f(x, y) \leq T_2$**
  It belongs to the Object O_1
- **$f(x, y) \leq T_1$**
  It belongs to the Background

T_1: be the threshold in the valley region V_1

T_2: be the threshold in the valley region V_2

Thresholding

- T = function of {(x, y), p(x, y), f(x, y)}
    - **(x, y):** be the pixel location
    - **f(x, y):** be the pixel intensity at (x, y)
    - **p(x, y):** be the local property in a neighbourhood centered at (x, y)

- Depending on the T, thresholding can be of three types:
    - Global threshold
    - Local Threshold
    - Adaptive threshold

Thresholding

- Depending on the T, thresholding can be of three types:
  - **T{f(x, y)}:** Global threshold
  - **T{f(x, y), p(x, y) }:** Local Threshold
  - **T{f(x, y), p(x, y), (x, y)}:** Adaptive or Dynamic threshold.
- Thresholding is the transformation of an input image f to an output (segmented/Thresholded) binary image g(x, y):
  - **Case1:** Object is brighter, and background is dark

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

  - **Case 2**: Object is Dark, and background is Brighter

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

Global Thresholding

Global Thresholding

- When the intensity distribution if objects and background pixels are sufficiently distinct, it is possible to use a single (global) threshold applicable over the entire image.
  - Based on the histogram of an image
  - Partition the image histogram using a single global threshold
  - The success of this technique strongly depends on how well the histogram can be partitioned
- In most of the application, there is usually enough variability between images that, even if global thresholding is a suitable approach, an algorithm capable of estimating the threshold value for each image is required.

Global Thresholding

❑ Thresholding is the transformation of an input image f to an output (segmented/Thresholded) binary image g(x, y):

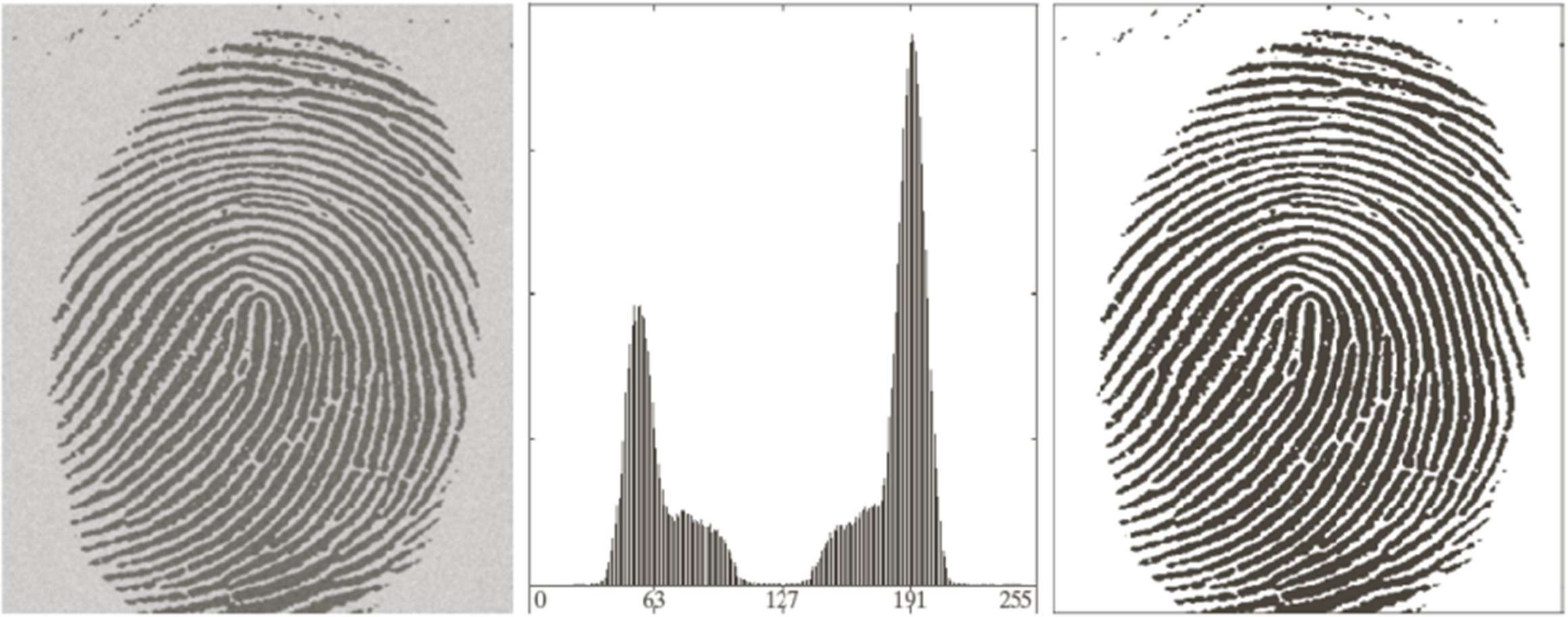
- **Case1:** Object is brighter, and background is dark

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

- **Case 2**: Object is Dark, and background is Brighter

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

Global Thresholding



a b c

FIGURE 10.38 (a) Noisy fingerprint. (b) Histogram. (c) Segmented result using a global threshold (the border was added for clarity). (Original courtesy of the National Institute of Standards and Technology.)

Global Thresholding

- **Algorithm:**

1. Select an initial value of T
2. Based on the value of T, do the image segmentation or the image is divided into two groups $\mathbf{G_1}$ and $\mathbf{G_2}$.
3. Pixels intensity values in Group $\mathbf{G_1}$ are similar and Pixel intensity values in Group $\mathbf{G_2}$ are similar. However, $\mathbf{G_1}$ and $\mathbf{G_2}$ will have different pixel intensity value.
4. Compute the average (mean) intensity values $\mathbf{m_1}$ and $\mathbf{m_2}$ for the pixels in Groups $\mathbf{G_1}$ and $\mathbf{G_2}$ respectively.
5. Compute a new threshold value midway between $\mathbf{m_1}$ and $\mathbf{m_2}$

$$T = \frac{m_1 + m_2}{2}$$

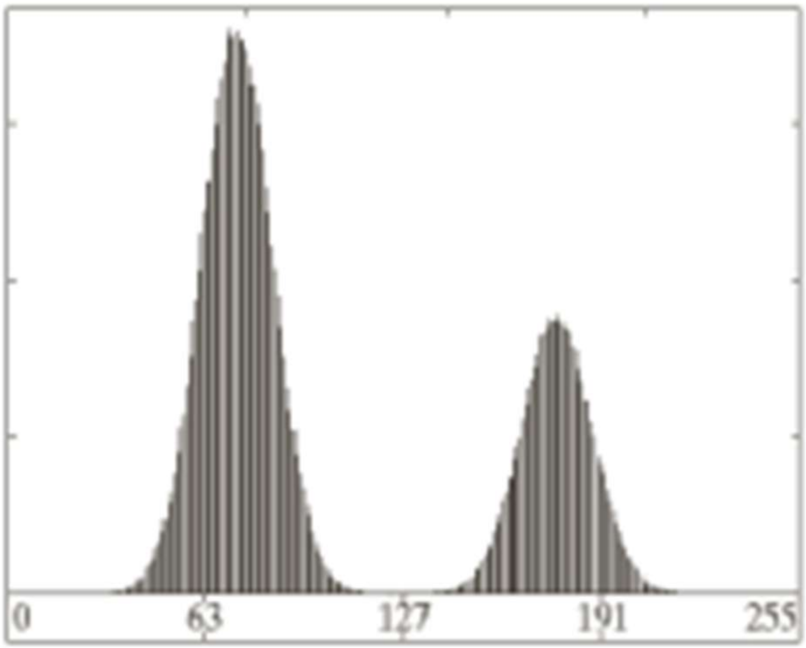
6. Go to step 2 and repeat the process.
7. If $|T_i - T_{i+1}| \leq T'$ then stop.

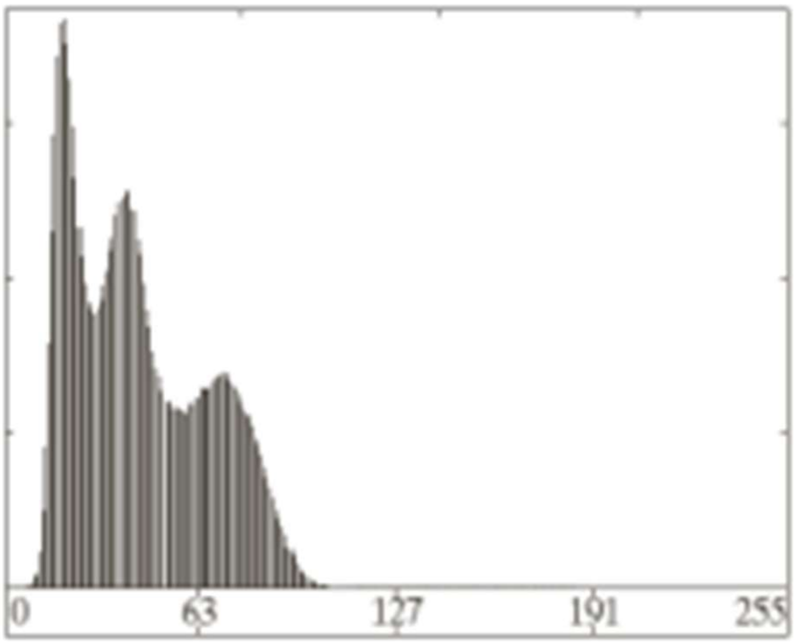
Problems with Global Thresholding

Problems in Global Thresholding

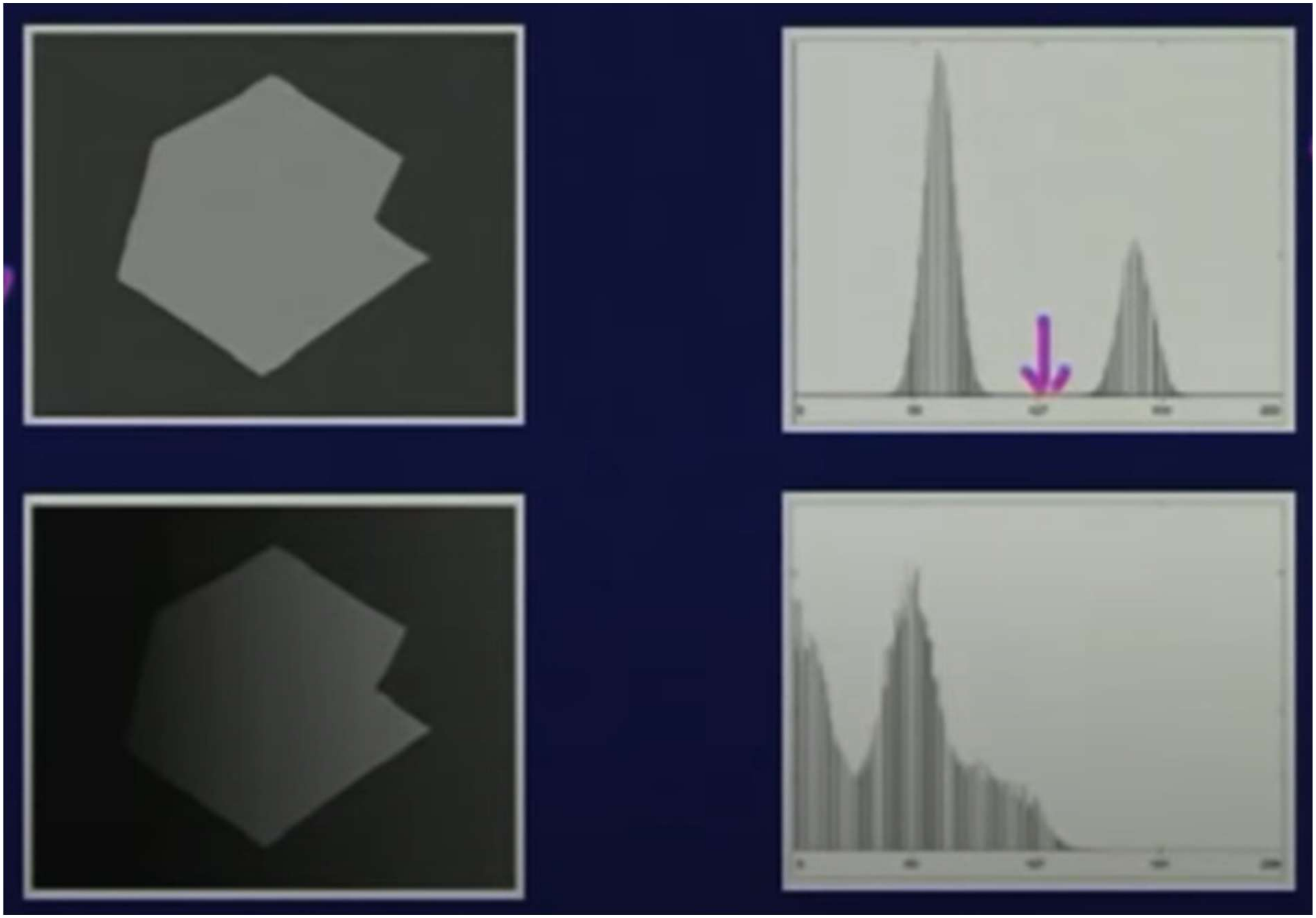
- **Problems with Global Thresholding:**

  1. **Effect of illumination:**



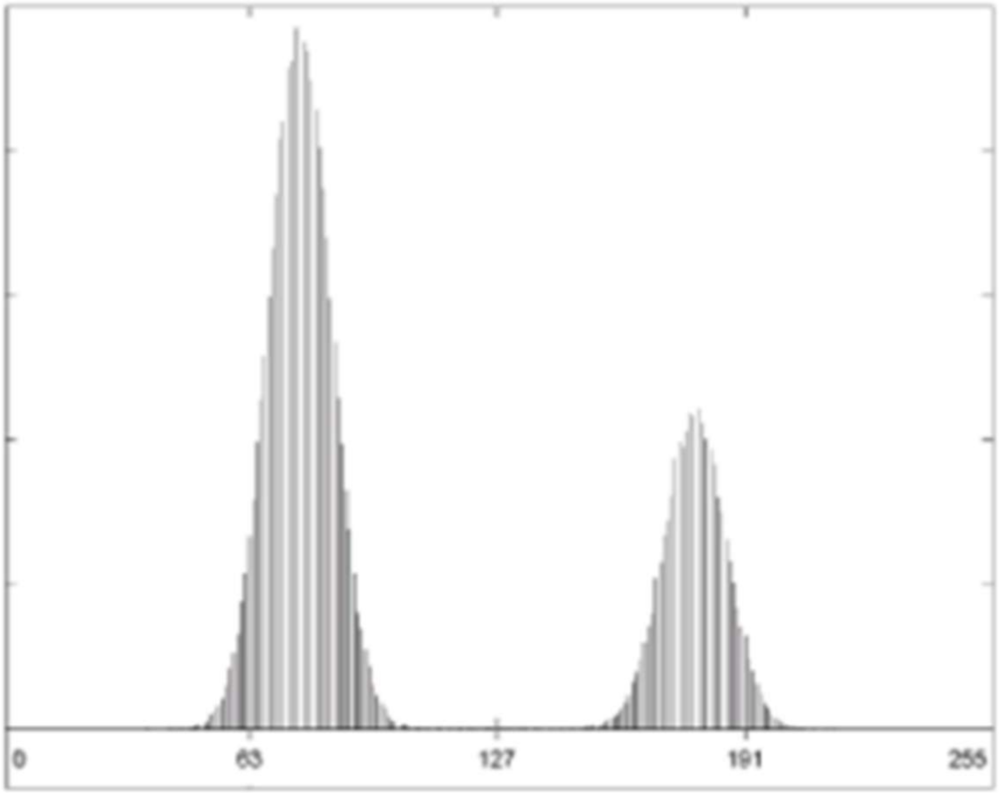


Problems in Global Thresholding

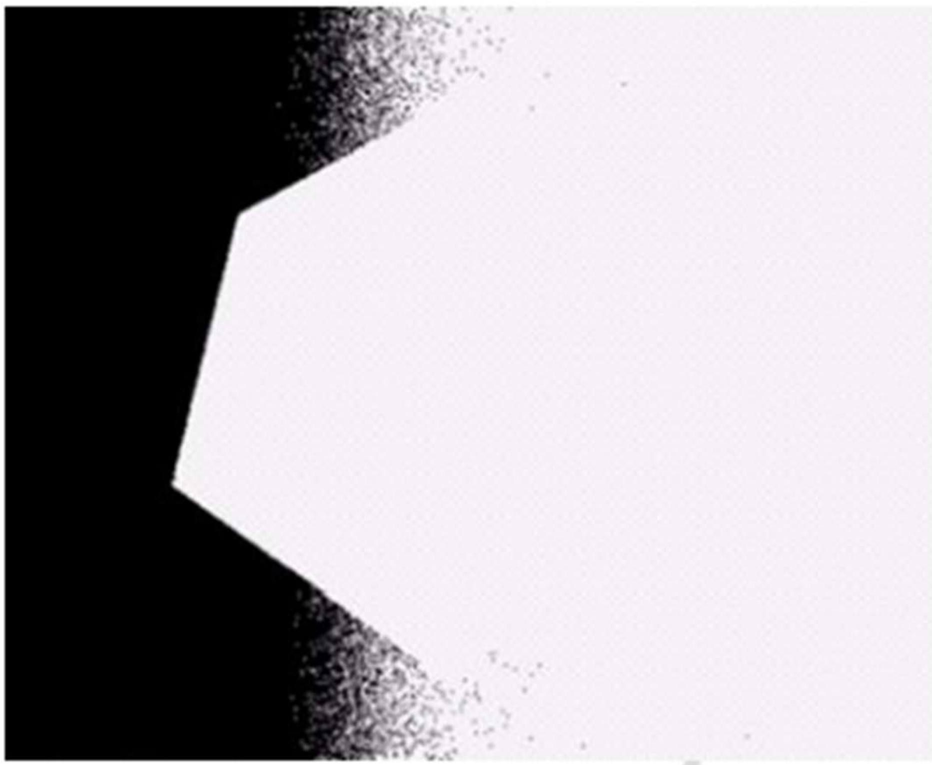


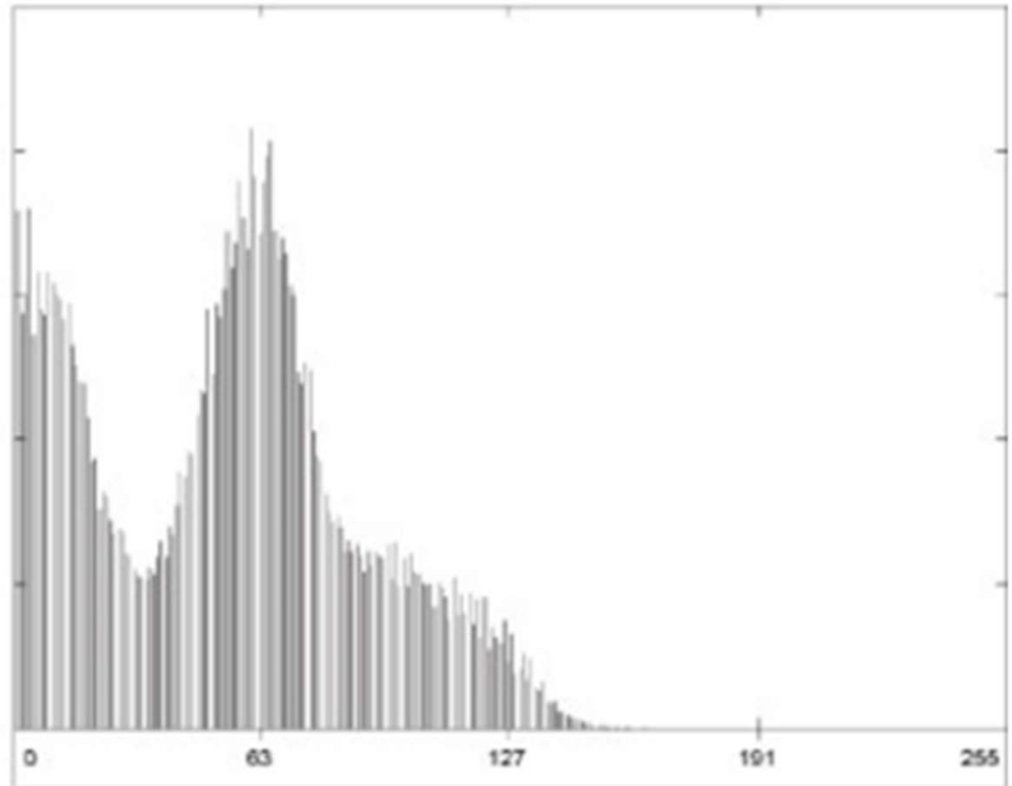
Source: Digital Image Processing, IIT Kharagpur Prof. P. K. Biswas

Problems in Global Thresholding



Problems in Global Thresholding



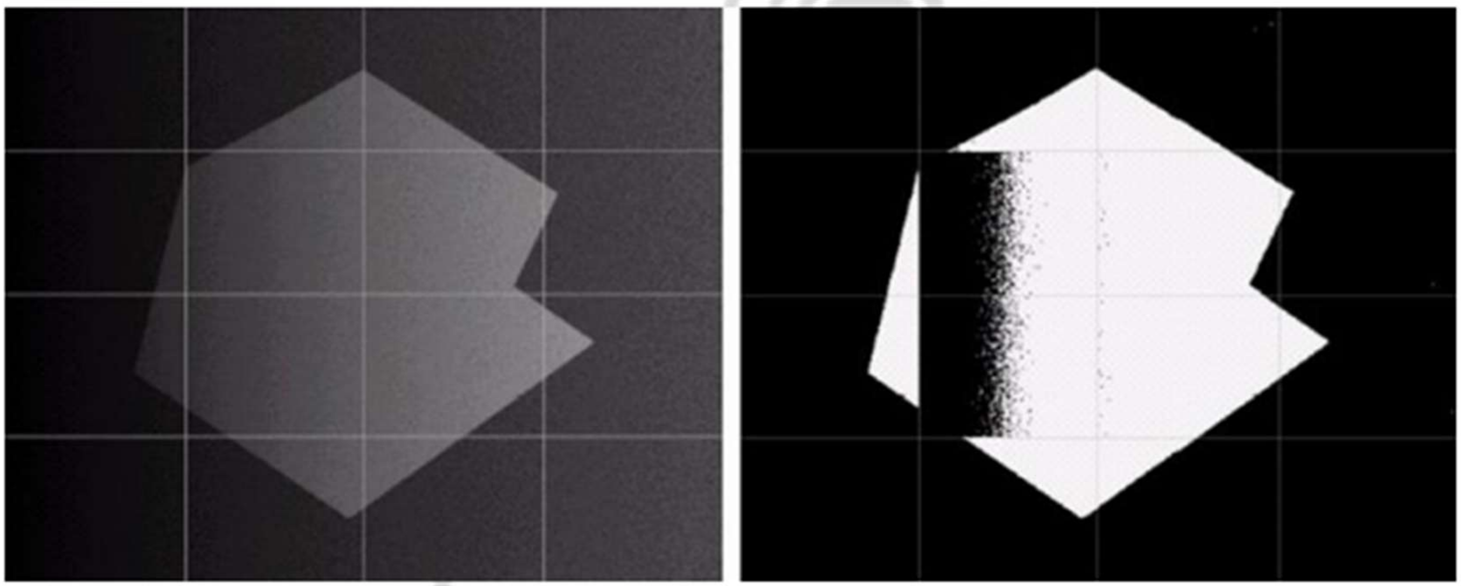


So, to overcome the limitation of Global Thresholding, we need to use Adaptive thresholding

Adaptive Thresholding

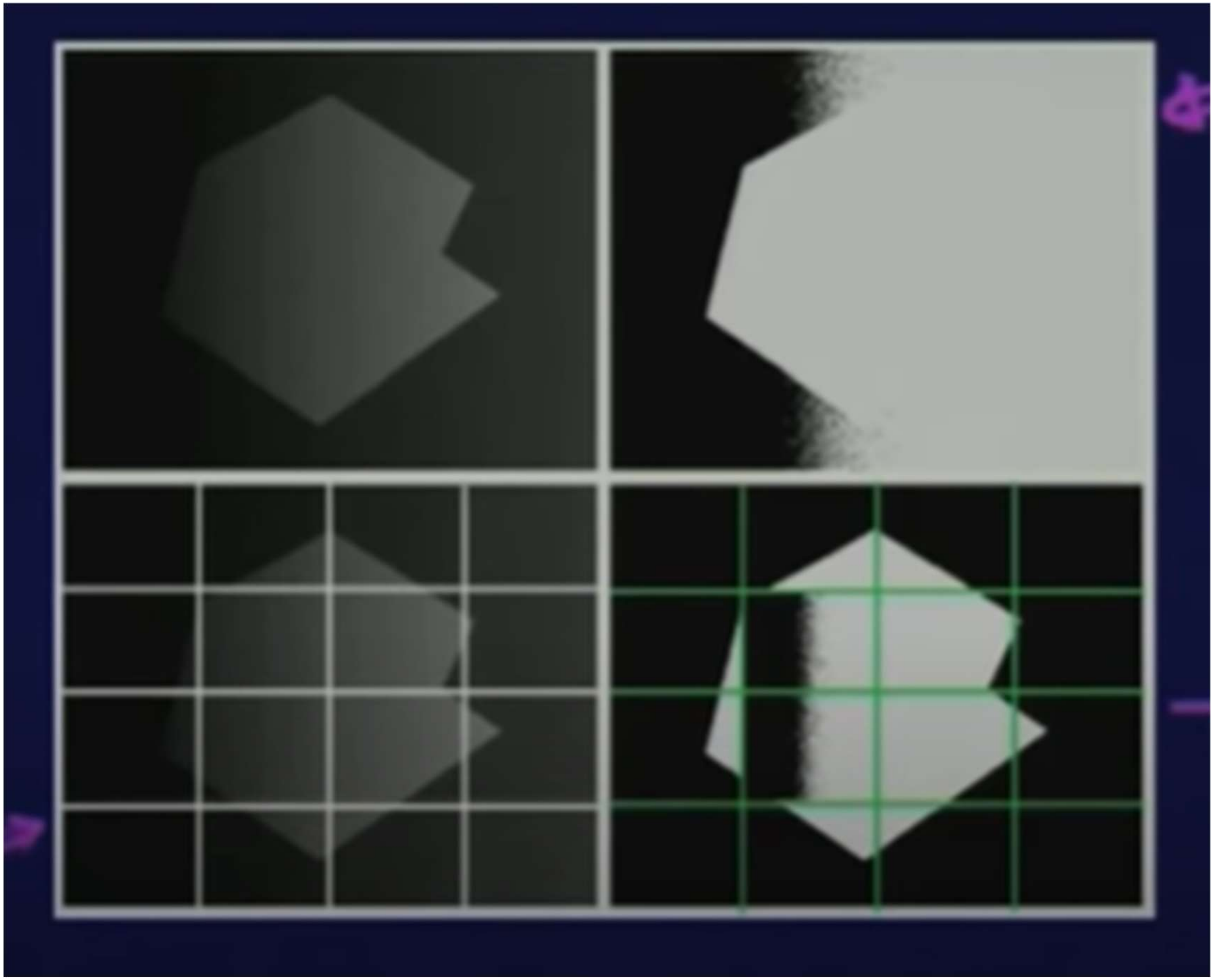
Adaptive Thresholding

- Divide the image into smaller sub-segments and then determine the threshold value for each sub-segment and then the final output is the union of thresholded results of all sub-segments.



- In this case, the sub-segmented output is relatively better than the one value thresholded output.
- Since this threshold value is position/location dependent. So, this kind of thresholding is called as Adaptive or Dynamic thresholding.

Adaptive Thresholding



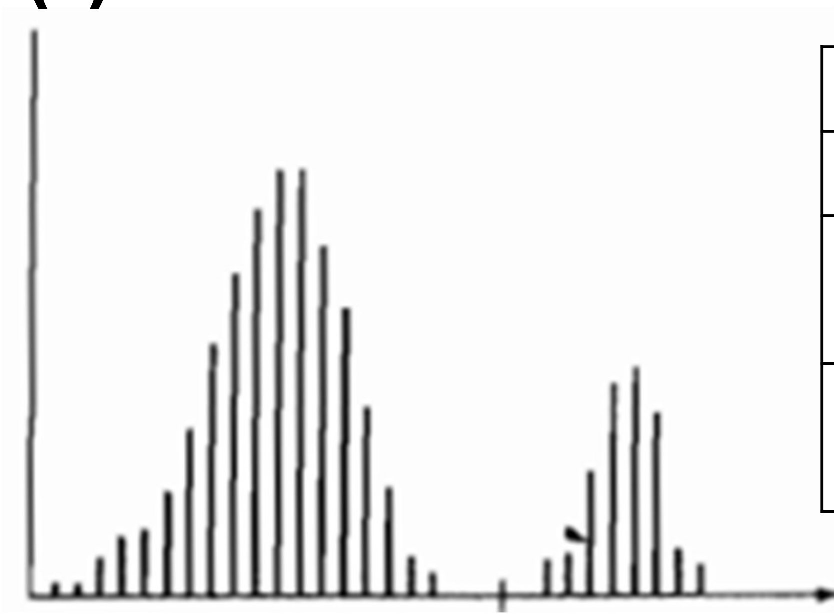
Source: Digital Image Processing, IIT Kharagpur Prof. P. K. Biswas

Optimal Thresholding

Optimal Thresholding

- None of the thresholding operation give the information about the error or accuracy.
- By applying some statistical algorithm, we design thresholding operation, where the error would be minimum is called as **“Optimal Thresholding”.**

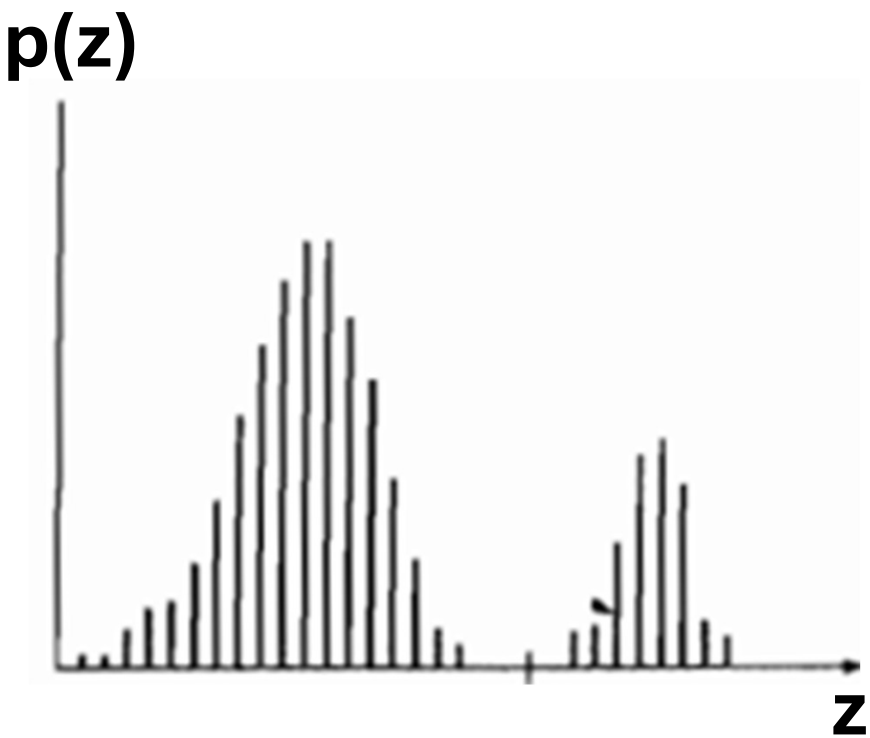
p(z)



z

Z :	Intensity variable (random in nature)
p(z) :	Probability density function of z
p1(z):	Probability density function of pixels belongs to the background
p2(z):	Probability density function of pixels belongs to the object

Optimal Thresholding



Z :	Intensity variable (random in nature)
p(z) :	Probability density function of z
p_1(z):	Probability density function of pixels belongs to the background
p_2(z):	Probability density function of pixels belongs to the object

P(z) can be represented as combination of p1(z) and p2(z) $= P_1 p_1(z) + P_2 p_2(z)$

Where,

P_1: be the probability that the pixel belongs to the background,

P_2: be the probability that the pixel belongs to the object.

$$P_1 + P_2 = 1$$

Optimal Thresholding

- Assuming:

  E1(T): a background particular is considered as object pixel

  E2(T): an object pixel is considered as a background pixel

- Overall error probability can be written as:

$$E(T) = P_2 E_1(T) + P_1 E_2(T)$$

- For minimization:

$$\frac{\partial E(T)}{\partial T} = 0$$

- On simplification,

$$P_1 p_1(T) = P_2 p_2(T)$$

Optimal Thresholding

- Solution of this

$$P(z) = \frac{P_1}{\sqrt{2\pi}\sigma_1} e^{-\left(\frac{(z-\mu_1)^2}{2\sigma_1^2}\right)} + \frac{P_2}{\sqrt{2\pi}\sigma_2} e^{-\left(\frac{(z-\mu_2)^2}{2\sigma_2^2}\right)}$$

Where,

- **μ_1:** average intensity value of the background pixel
- **μ_2:** average intensity value of the object pixel
- **σ_1:** are the standard deviation of intensity values in the background
- **σ_2:** are the standard deviation of intensity values in the object region

Optimal Thresholding

- $AT^2 + BT + C = 0$

Where,

$$A = \sigma_1^2 - \sigma_2^2$$
$$B = 2(\mu_1\sigma_2^2 - \mu_2\sigma_1^2)$$
$$C = \sigma_1^2\mu_2^2 - \sigma_2^2\mu_1^2 + 2\sigma_1^2\sigma_2^2 \ln\left(\frac{\sigma_2 P_1}{\sigma_1 P_2}\right)$$

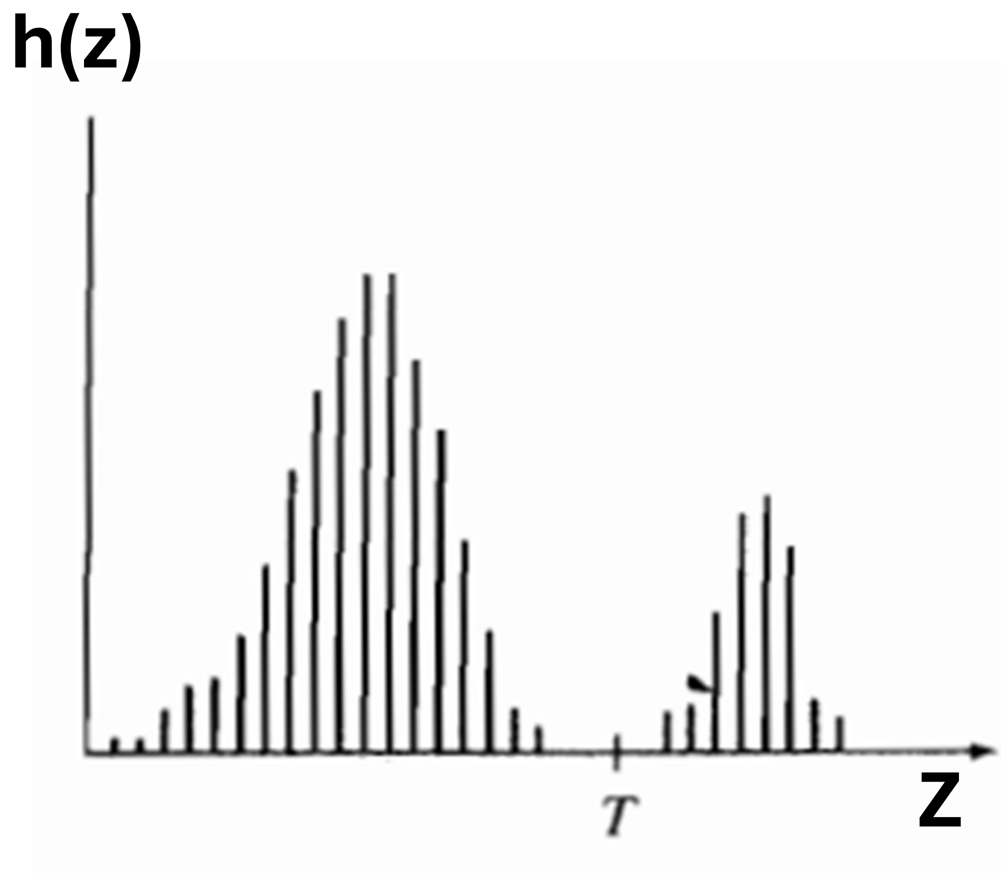
- Assuming, $\sigma_1^2 = \sigma_2^2 = \sigma^2$; Optimal value of threshold ($T_{optimal}$) which offer minimum value of average segmentation error is given as follows:

$$\boldsymbol{T_{optimal} = \left(\frac{\mu_1 + \mu_2}{2}\right) + \frac{\sigma^2}{\mu_1 - \mu_2}\ln\left(\frac{P_2}{P_1}\right)}$$

Local Thresholding

Local Thresholding

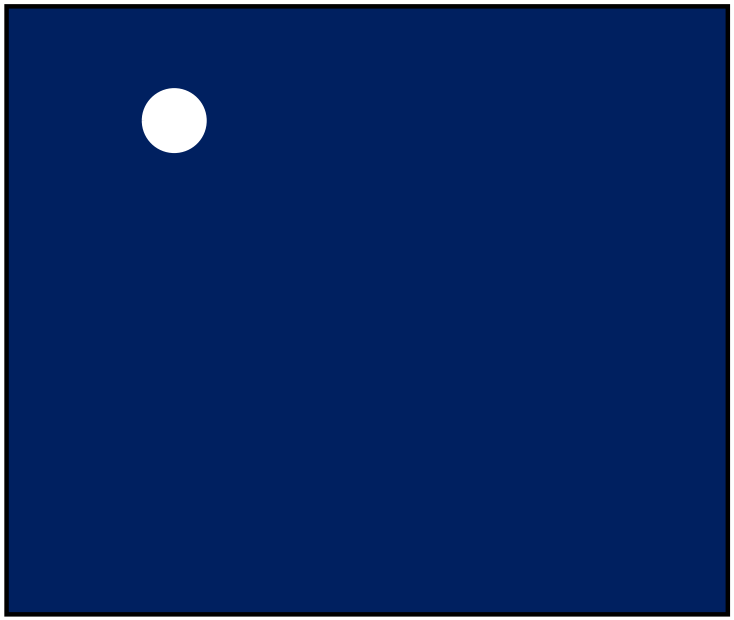
- Selection of threshold value is simple if the histogram is bimodal histogram



- By using this threshold value; we can do the segmentation of pixel in background or in object.

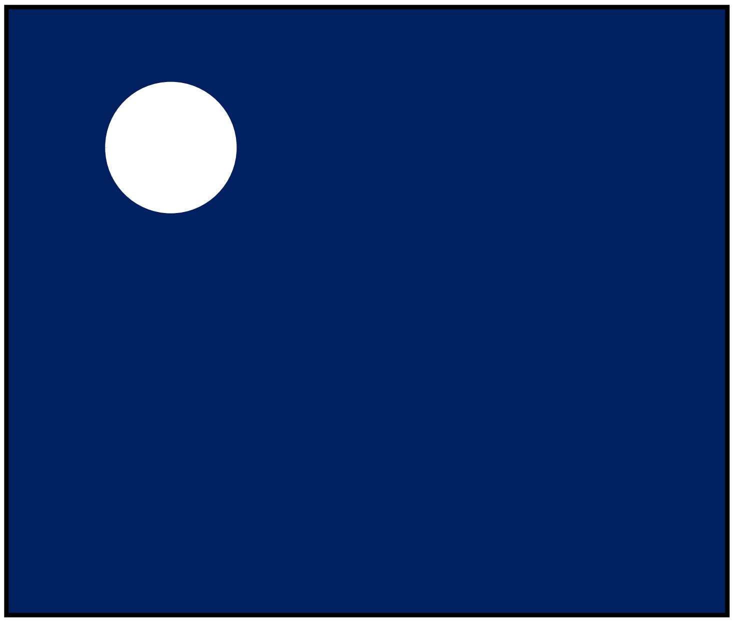
Local Thresholding

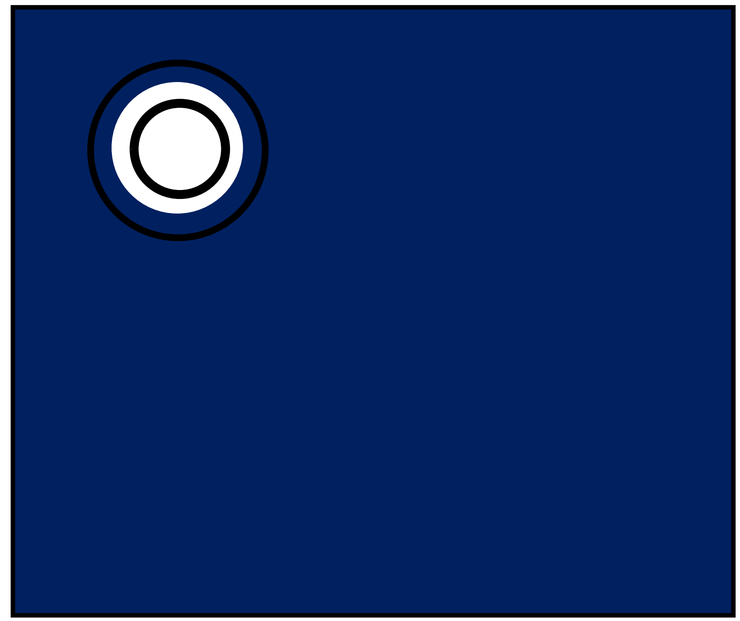
- Selection of threshold it becomes very easy of the histogram is symmetric i.e. area under the peaks are similar



- To overcome this problem, we need to consider pixels around the boundary in very narrow strip. This method is called as **local thresholding.**

Local Thresholding





Local Thresholding

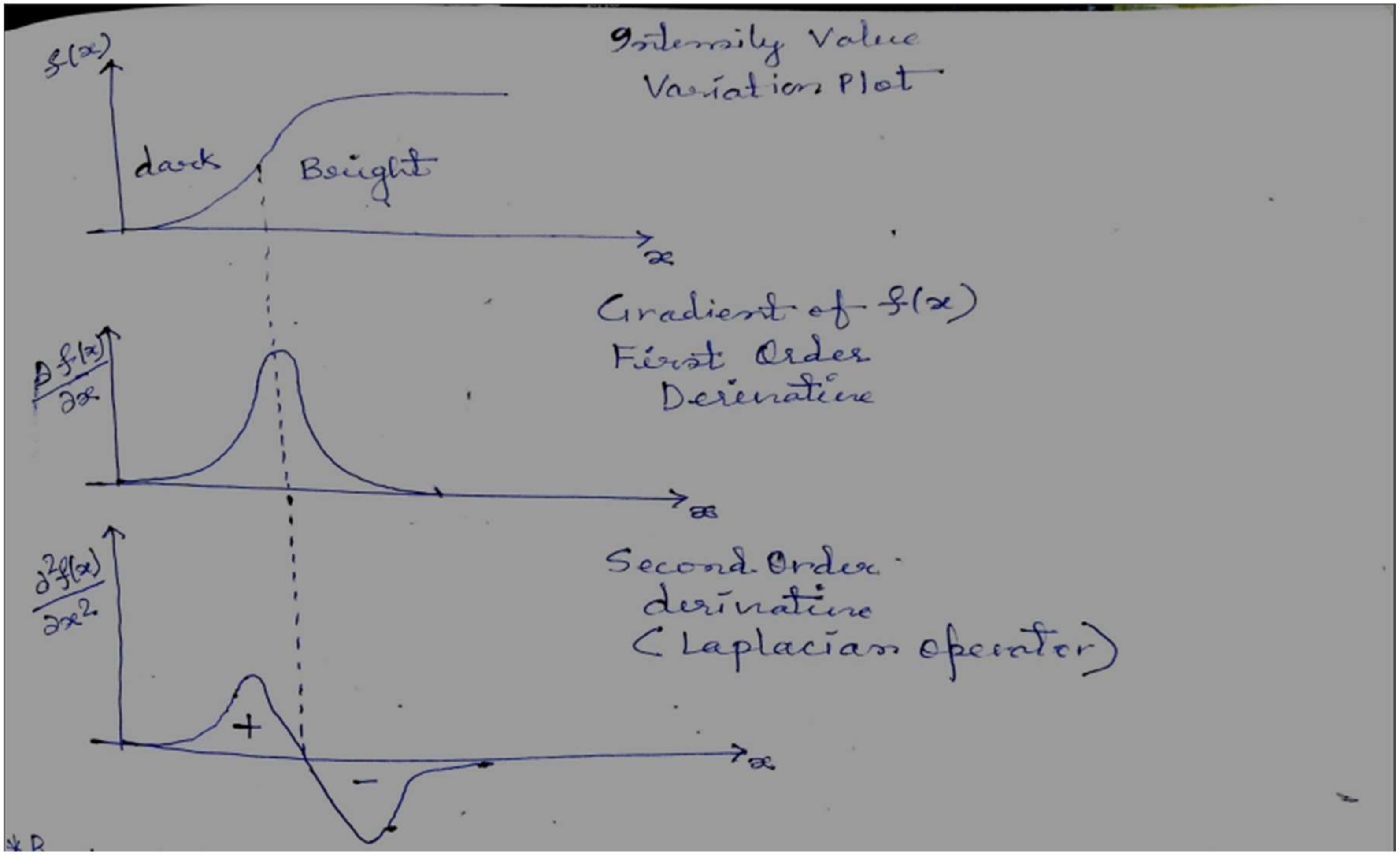
❑Advantages of local thresholding:

- Since we are considering only the boundary pixel. So, the histogram will be symmetric, and it is not dependent on the relative size of the object and the background.

- The probability of pixel belonging to the object and the probability of pixel belonging to the background in this narrow strip is almost equal.

- **Challenge:** The programmer should know the boundary of an object. In many practical scenario, boundary of an object is not known and using image segmentation we need to determine the boundary of an object.

Local Thresholding



By knowing the Laplacian operator output(Second order derivative), we can make decision; whether the pixel lying on the dark side of the edge, or it is lying on the bright side of the edge.

Local Thresholding

$S(x,y) =$	**0**	If **($\nabla f < T$),** no edge is detected. Since at the edge ∇f will have very high value. **Assume a value S(x,y) = 0;** representing the pixel does not belong to the boundary between object and background
	+	If **($\nabla f \geq T$) and ($\nabla^2 f \geq 0$),** representing it is an edge point or it is a point near the edge and the point belongs to the dark side. **Assume a value S(x,y) = 128;** representing the pixel belongs to the object region
	-	If **($\nabla f \geq T$) and ($\nabla^2 f < 0$),** representing it is an edge point or it is a point near the edge and the point belongs to the brighter side. **Assume a value S(x,y) = 255;** representing the pixel belongs to the background region.

Region Growing

Region Growing

❑ Region:

- A group of connected pixels with similar properties
- For correct interpretation, image must be partitioned into regions that correspond to objects or parts of an object.

$$R \Rightarrow \{R1, R2, R3, \dots \dots Rn\}$$

❑ Criteria/Conditions for Region Segmentation:

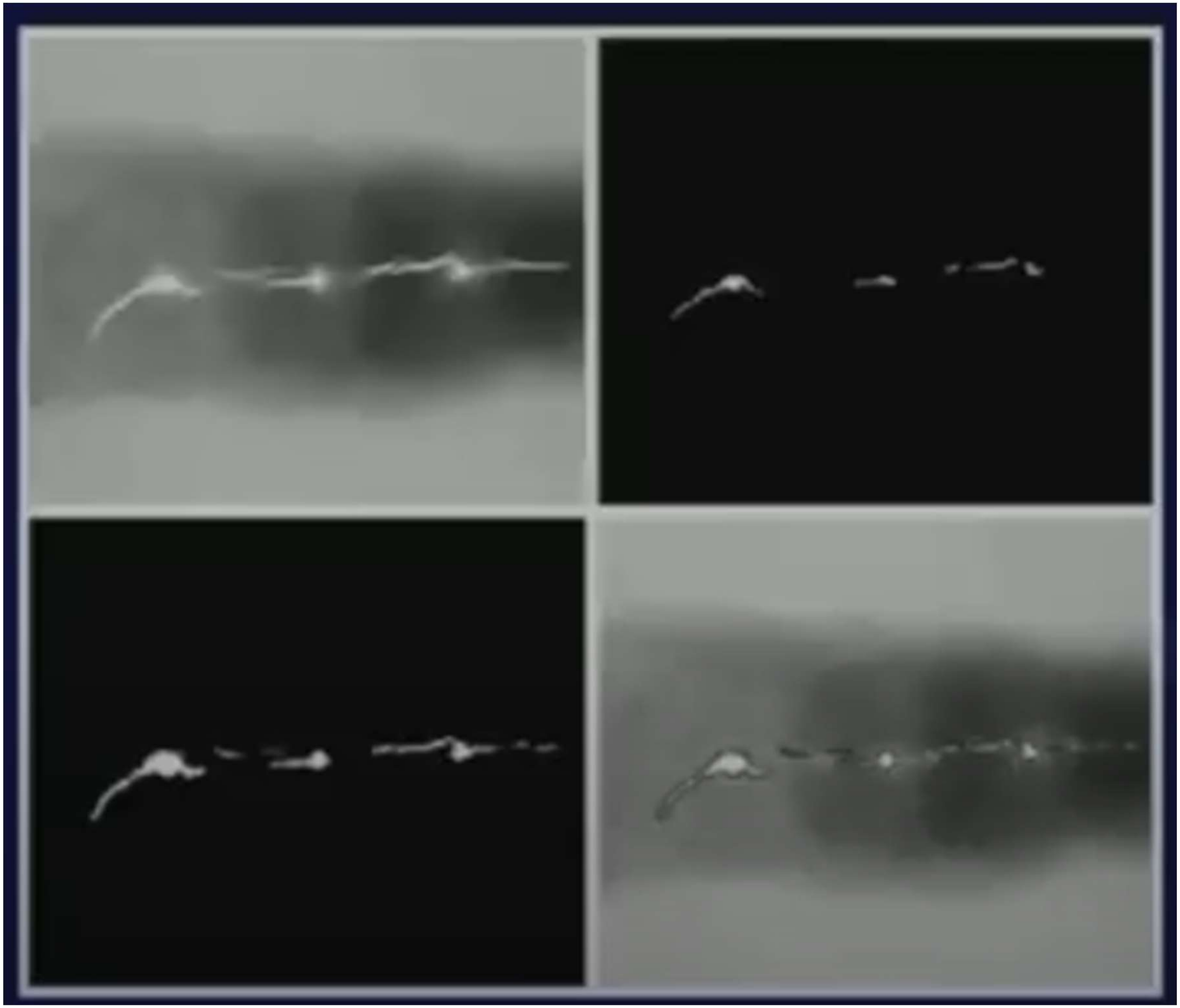
- Every pixel should be a part of one of the subsegment
- Region R_i should be connected
- $R_i \cap R_j = \varphi$ (NULL) for $i \neq j$
- $P(R_i)$ = TRUE where P is the logical predicate defined over the points in Set Ri in this partition Ri
- $P(R_i \cup R_j)$ = FALSE for $i \neq j$

Region Growing

- **Region Growing:** The region growing as the name implies that it is a procedure which groups the pixels on subgroups into larger region based on predefined criteria for the growth, and these predefined criteria is predicate.

- **Step by step Procedure for Region Growing:**
  - Choose the seed pixels (1 for every segment)
  - Check the neighboring pixels and add them to the region if they are similar to the seed
  - Repeat previous step for each of the newly added pixels, stop if no more pixels can be added

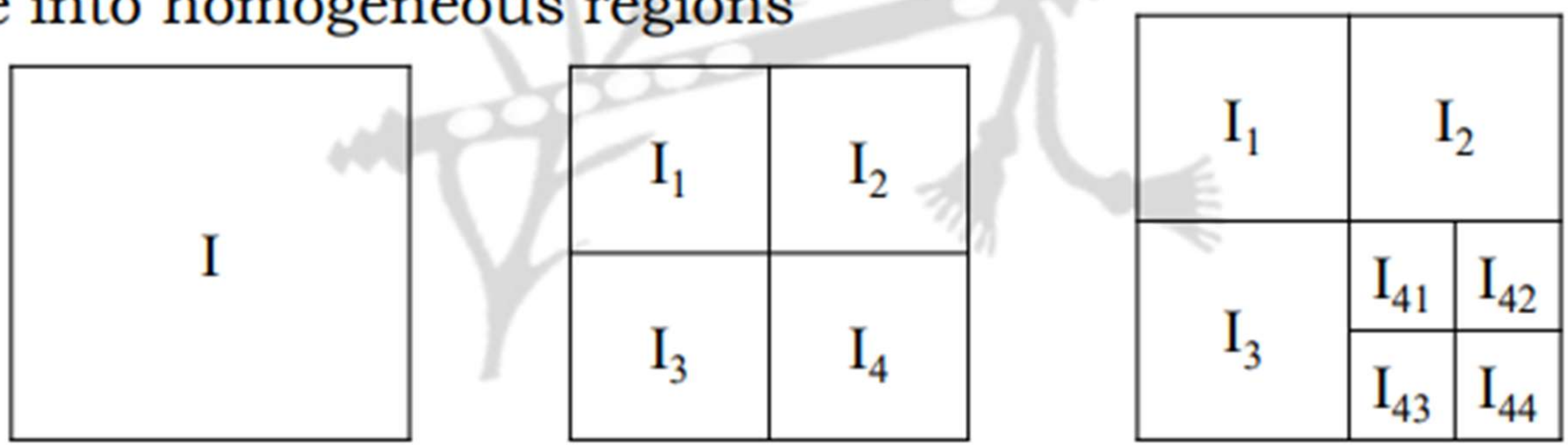
Region Growing



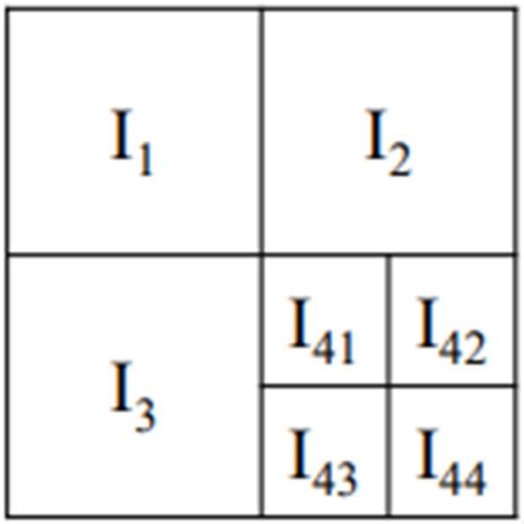
Region Splitting and Merging

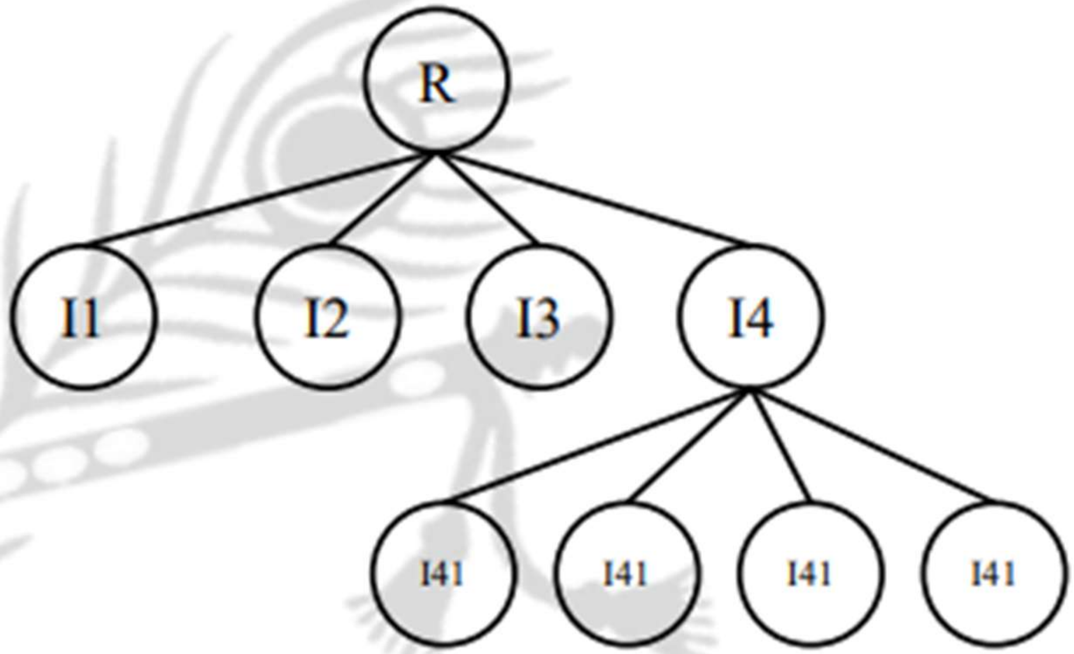
Region Splitting

- Split starts from the assumption that the entire image is homogeneous
- If this is not true (by the homogeneity criterion), then split image into four sub images
- This splitting procedure is repeated recursively until we split the image into homogeneous regions

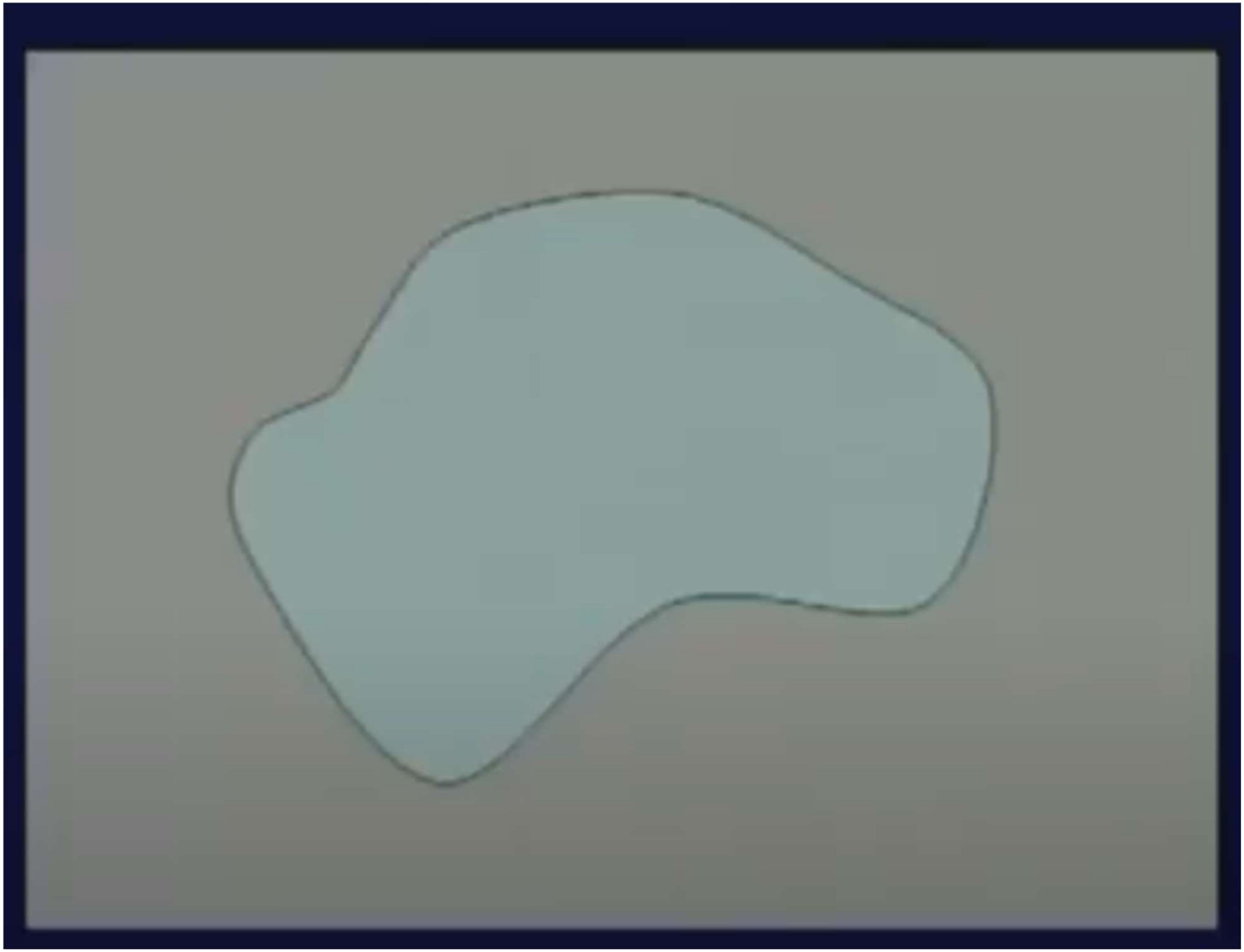


Region Splitting





Region Splitting & Merging



Region Splitting

❑Advantage

- Created regions are adjacent and homogenous

❑Disadvantage

- Over splitting, since no merge is performed – More clusters

❑Improvement

- Split and Merge

Region Splitting and Merging

- After splitting

- Merging phase
  - If 2 adjacent regions are homogenous, they are merged

- Repeat merging step, until no further merging is possible

Thank You