CSE4227 Digital Image Processing

Chapter 10 – **Image Segmentation Part II (Region based)**

Dr. Kazi A Kalpoma
Professor, Department of CSE
Ahsanullah University of Science & Technology (AUST)

Contact: kalpoma@aust.edu

Today's Contents

- Region Based Segmentation
  - What is a Region
  - Region Growing, and
  - Region Splitting & Merging
- Thresholding Based Segmentation
  - GLOBAL
  - ADAPTIVE
  - LOCAL

- Chapter 10 from R.C. Gonzalez and R.E. Woods, Digital Image Processing (3rd Edition), Prentice Hall, 2008 [**Section 10.1, 10.2 (excluding 10.2.7)**]

Segmentation Algorithms

Segmentation can be performed based on one of the following two basic properties of intensity values:

- **Similarity**
  - Partitioning an image into regions that are similar according to a set of predefines criteria.
- **Discontinuity**
  - Detecting boundaries of regions based on local discontinuity in intensity.

Region based segmentation

- In edge detection we segment an image by identifying the boundaries of the objects.
- These boundaries are the locations where the intensity is changed.
- In the **region based segmentation** approach, we will identify regions occupied by the objects.
- We will group pixels which are **similar in some region property.**

What is a Region?

- Basic definition :- A group of connected pixels with similar properties.

- Important in interpreting an image because they may correspond to objects in a scene.
  - For that an image must be partitioned into regions that correspond to objects or parts of an object.

Edge Pixels

20	0	0	0	0	0	0	0
20	0	0	0	0	0	0	0
20	0	143	143	143	143	0	0
20	0	143	200	200	143	0	0
20	0	143	200	200	143	0	0
20	0	143	143	143	143	0	0
20	0	0	0	0	0	0	0
20	0	0	0	0	0	0	0

8x8 cropped Image

Edge based vs Region based Segmentation

20	0	0	0	0	0	10	10
20	0	0	0	0	0	10	10
20	0	143	143	143	143	10	10
20	0	143	200	200	143	10	10
20	0	143	200	200	143	10	10
20	0	143	143	143	143	10	10
20	0	0	0	0	0	10	10
20	0	0	0	0	0	10	10

8x8 cropped Image

Edge based vs Region based Segmentation

20	0	0	0	0	0	**10**	**10**
20	0	0	0	0	0	**10**	**10**
20	0	**143**	**143**	**143**	**143**	**10**	**10**
20	0	**143**	**200**	**200**	**143**	**10**	**10**
20	0	**143**	**200**	**200**	**143**	**10**	**10**
20	0	**143**	**143**	**143**	**143**	**10**	**10**
20	0	0	0	0	0	**10**	**10**
20	0	0	0	0	0	**10**	**10**

Region-Based vs. Edge-Based

Region-Based	Edge-Based
Closed boundaries	Boundaries formed not necessarily closed
Multi-spectral images improve segmentation	No significant improvement for multi-spectral images
Computation based on similarity	Computation based on difference

Region Based Segmentation Method

- Segments the image into various regions having similar characteristics.

- The two basic techniques based on this method are:
  - i) Region Growing, and
  - ii) Region Splitting & Merging.

Region-based segmentation

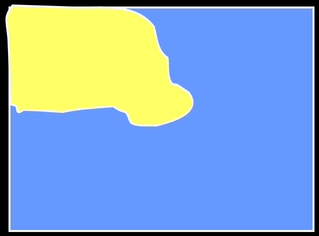
Region growing *(Bottom-up approach)*

1. Find starting points.
2. Include neighboring pixels with similar feature (gray level, texture, color, etc.).
3. Continue until all pixels have been associated with one of the starting points.

Problems

Non trivial to find good starting points, difficult to automate and needs good criteria for similarity.

Region Growing



Region growing techniques start with one pixel of a potential region and try to grow it by adding adjacent pixels till the pixels being compared are too disimilar.

- The first pixel selection can be just the first unlabeled pixel in the image or a set of seed pixels can be chosen from the image.

- Usually a statistical test is used to decide which pixels can be added to a region.

Region-Based Segmentation

Example: Region Growing based on 8-connectivity

$f(x, y)$: input image array

$S(x, y)$: seed array containing 1s (seeds) and 0s

$Q(x, y)$: predicate

Region Growing Algorithm

- Start with seed points $S(x,y)$ and grow to larger regions satisfying a predicate.
- Needs a stopping rule.
- **Algorithm**
  - Find all connected components in $S(x,y)$ and erode them to 1 pixel.
  - Form image $f_q(x,y)=1$ if $f(x,y)$ satisfies the predicate.
  - Form image $g(x,y)=1$ for all pixels in $f_q(x,y)$ that are 8-connected with to any seed point in $S(x,y)$.
  - Label each connected component in $g(x,y)$ with a different label.

Example

Suppose that we have the image given below.

(a) Use the region growing idea to segment the object. The seed for the object is the center of the image. Region is grown in horizontal and vertical directions, and when the difference between two pixel values is less than or equal to 5.

Table 1: Show the result of Part (a) on this figure.

10	10	10	10	10	10	10
10	10	10	69	70	10	10
59	10	60	64	59	56	60
10	59	10	<u>60</u>	70	10	62
10	60	59	65	67	10	65
10	10	10	10	10	10	10
10	10	10	10	10	10	10

Example

Suppose that we have the image given below.

(a) Use the region growing idea to segment the object. The seed for the object is the center of the image. Region is grown in horizontal and vertical directions, and when the difference between two pixel values is less than or equal to 5.

Table 1: Show the result of Part (a) on this figure.

10	10	10	10	10	10	10
10	10	10	69	70	10	10
59	10	60	64	59	56	60
10	59	10	60	70	10	62
10	60	59	65	67	10	65
10	10	10	10	10	10	10
10	10	10	10	10	10	10

4-connectivity

Example

Suppose that we have the image given below.

(a) Use the region growing idea to segment the object. The seed for the object is the center of the image. Region is grown in horizontal and vertical directions, and when the difference between two pixel values is less than or equal to 5.

(b) **What will be the segmentation if region is grown in horizontal, vertical, and diagonal directions?**

Table 2: Show the result of Part (b) on this figure.

10	10	10	10	10	10	10
10	10	10	69	70	10	10
59	10	60	64	59	56	60
10	59	10	60	70	10	62
10	60	59	65	67	10	65
10	10	10	10	10	10	10
10	10	10	10	10	10	10

Example

Suppose that we have the image given below.

(a) Use the region growing idea to segment the object. The seed for the object is the center of the image. Region is grown in horizontal and vertical directions, and when the difference between two pixel values is less than or equal to 5.

(b) **What will be the segmentation if region is grown in horizontal, vertical, and diagonal directions?**

Table 2: Show the result of Part (b) on this figure.

10	10	10	10	10	10	10
10	10	10	69	70	10	10
59	10	60	64	59	56	60
10	59	10	<u>60</u>	70	10	62
10	60	59	65	67	10	65
10	10	10	10	10	10	10
10	10	10	10	10	10	10

8-connectivity

Region Splitting and Merging

- Let R represent the entire spatial region occupied by an image. Image segmentation is a process that partitions R into *n* sub-regions, R_1, R_2, …, R_n, such that

(a) $\bigcup_{i=1}^{n} R_i = R.$

(b) R_i is a connected set. $i = 1,\ 2,\ \dots,\ n.$

(c) $R_i \cap R_j = \Phi.$

(d) $\square\,(R_i) = \text{TRUE for } i = 1,\ 2,\ \dots,\ n.$ Logical predicate

(e) $\square\,(R_i \cup R_j) = \text{FALSE for any adjacent regions}$

R_i and R_j.

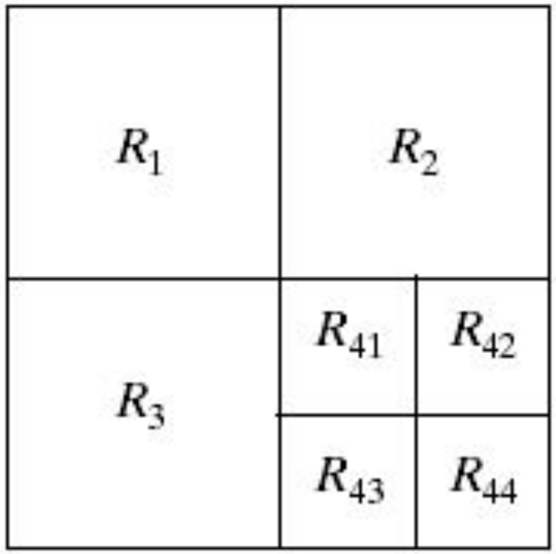
Region-Based Segmentation

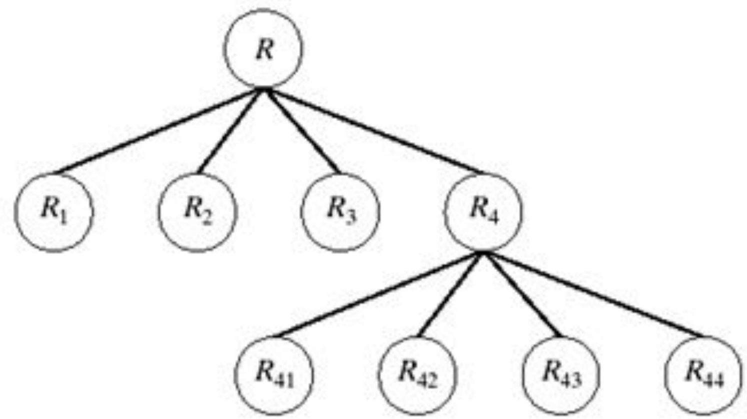
- Region Splitting
  - Region Growing: Starts from a set of seed points.
  - Region Splitting: Starts with the whole image as a single region and subdivide the regions that do not satisfy a condition.
  - Image = One Region R
  - Select a predicate P (gray values etc.)
  - Successively divide each region into smaller and smaller quadrant regions so that:

$$P(R_i) = true$$

Region-Based Segmentation

- Region Splitting





Problem? Adjacent regions could be same

Solution? Allow Merge

Region-Based Segmentation

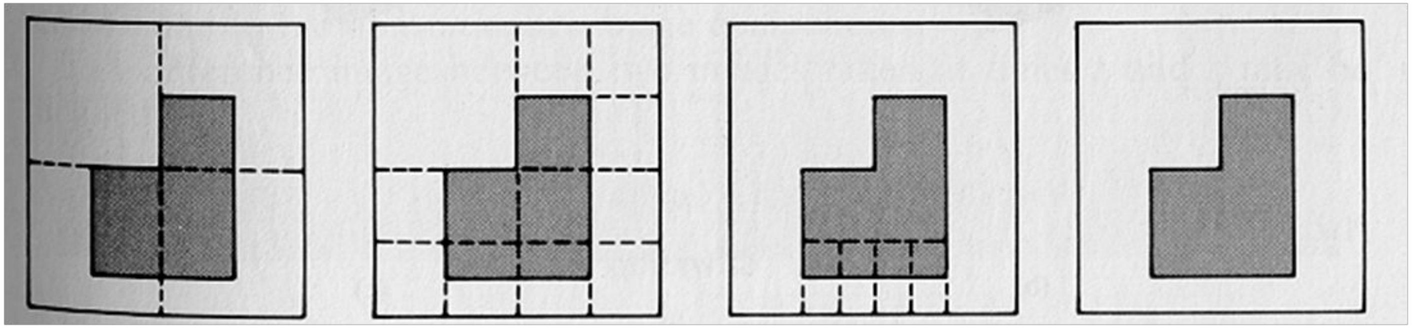
- Region Merging
  - Region merging is the opposite of region splitting.
  - Merge adjacent regions R_i and R_j for which:

$$P(R_i \cup R_j) = True$$

- Region Splitting/Merging
  - Stop when no further split or merge is possible

Region-Based Segmentation

- Example



1. Split into four disjointed quadrants any region R_i where $P(R_i)$=False
2. Merge any adjacent regions R_j and R_k for which $P(R_j \cup R_k)$=True
3. Stop when no further merging or splitting is possible

Region Splitting and Merging Example

Select a threshold first…say T <= 4.

5	6	6	6	7	7	6	6
6	7	6	7	5	5	4	7
6	6	4	4	3	2	5	6
5	4	5	4	2	3	4	6
0	3	2	3	3	2	4	7
0	0	0	0	2	2	5	6
1	1	0	1	0	3	4	4
1	0	1	0	2	3	5	4

Please see the video recording to understand the example.

Region Splitting Example

Split it if T < (max-min), continue until no more spit is possible

5	6	6	6	7	7	6	6
6	7	6	7	5	5	4	7
6	6	4	4	3	2	5	6
5	4	5	4	2	3	4	6
0	3	2	3	3	2	4	7
0	0	0	0	2	2	5	6
1	1	0	1	0	3	4	4
1	0	1	0	2	3	5	4

Please see the video recording to understand the example.

Region Splitting Example

Split it if T < (max-min), continue until no more spit is possible

5	6	6	6	7	7	6	6
6	7	6	7	5	5	4	7
6	6	4	4	3	2	5	6
5	4	5	4	2	3	4	6
0	3	2	3	3	2	4	7
0	0	0	0	2	2	5	6
1	1	0	1	0	3	4	4
1	0	1	0	2	3	5	4

Please see the video recording to understand the example.

Region Merging Example

Merge the adjacent regions if **T** is greater than **(max_i-min_j) and (max_j-min_i)** of both regions

5	6	6	6	7	7	6	6
6	7	6	7	5	5	4	7
6	6	4	4	3	2	5	6
5	4	5	4	2	3	4	6
0	3	2	3	3	2	4	7
0	0	0	0	2	2	5	6
1	1	0	1	0	3	4	4
1	0	1	0	2	3	5	4

Please see the video recording to understand the example.

Region Merging Example…

Merge the adjacent regions if **T** is greater than **(max_i-min_j) and (max_j-min_i)** of both regions

5	6	6	6	7	7	6	6
6	7	6	7	5	5	4	7
6	6	4	4	3	2	5	6
5	4	5	4	2	3	4	6
0	3	2	3	3	2	4	7
0	0	0	0	2	2	5	6
1	1	0	1	0	3	4	4
1	0	1	0	2	3	5	4

Please see the video recording to understand the example.

Region Merging Example

Merge the adjacent regions if **T** is greater than **(max_i-min_j) and (max_j-min_i)** of both regions

5	6	6	6	7	7	6	6
6	7	6	7	5	5	4	7
6	6	4	4	3	2	5	6
5	4	5	4	2	3	4	6
0	3	2	3	3	2	4	7
0	0	0	0	2	2	5	6
1	1	0	1	0	3	4	4
1	0	1	0	2	3	5	4

Please see the video recording to understand the example.

Region Merging Example

Merge the adjacent regions if **T** is greater than **(max_i-min_j) and (max_j-min_i)** of both regions

5	6	6	6	7	7	6	6
6	7	6	7	5	5	4	7
6	6	4	4	3	2	5	6
5	4	5	4	2	3	4	6
0	3	2	3	3	2	4	7
0	0	0	0	2	2	5	6
1	1	0	1	0	3	4	4
1	0	1	0	2	3	5	4

Please see the video recording to understand the example.

Region Merging Example

Merge the adjacent regions if **T** is greater than **(max_i-min_j) and (max_j-min_i)** of both regions

5	6	6	6	7	7	6	6
6	7	6	7	5	5	4	7
6	6	4	4	3	2	5	6
5	4	5	4	2	3	4	6
0	3	2	3	3	2	4	7
0	0	0	0	2	2	5	6
1	1	0	1	0	3	4	4
1	0	1	0	2	3	5	4

Please see the video recording to understand the example.

Region Merging Example

Merge the adjacent regions if **T** is greater than **(max_i-min_j) and (max_j-min_i)** of both regions

5	6	6	6	7	7	6	6
6	7	6	7	5	5	4	7
6	6	4	4	3	2	5	6
5	4	5	4	2	3	4	6
0	3	2	3	3	2	4	7
0	0	0	0	2	2	5	6
1	1	0	1	0	3	4	4
1	0	1	0	2	3	5	4

Please see the video recording to understand the example.

Region Merging Example

Merge the adjacent regions if **T** is greater than **(max_i-min_j) and (max_j-min_i)** of both regions

5	6	6	6	7	7	6	6
6	7	6	7	5	5	4	7
6	6	4	4	3	2	5	6
5	4	5	4	2	3	4	6
0	3	2	3	3	2	4	7
0	0	0	0	2	2	5	6
1	1	0	1	0	3	4	4
1	0	1	0	2	3	5	4

Please see the video recording to understand the example.

Region Merging Example

Merge the adjacent regions if **T** is greater than **(max_i-min_j) and (max_j-min_i)** of both regions

5	6	6	6	7	7	6	6
6	7	6	7	5	5	4	7
6	6	4	4	3	2	5	6
5	4	5	4	2	3	4	6
0	3	2	3	3	2	4	7
0	0	0	0	2	2	5	6
1	1	0	1	0	3	4	4
1	0	1	0	2	3	5	4

Please see the video recording to understand the example.

Region Merging Example

Merge the adjacent regions if **T** is greater than **(max_i-min_j) and (max_j-min_i)** of both regions

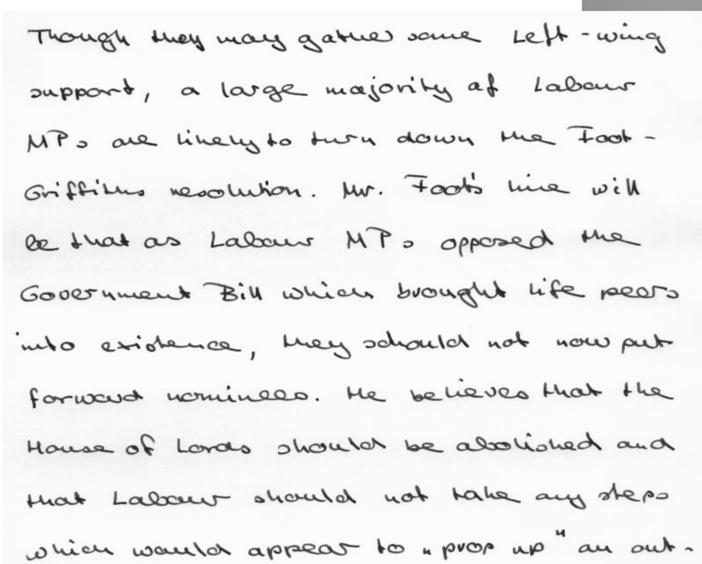
5	6	6	6	7	7	6	6
6	7	6	7	5	5	4	7
6	6	4	4	3	2	5	6
5	4	5	4	2	3	4	6
0	3	2	3	3	2	4	7
0	0	0	0	2	2	5	6
1	1	0	1	0	3	4	4
1	0	1	0	2	3	5	4

Please see the video recording to understand the example.

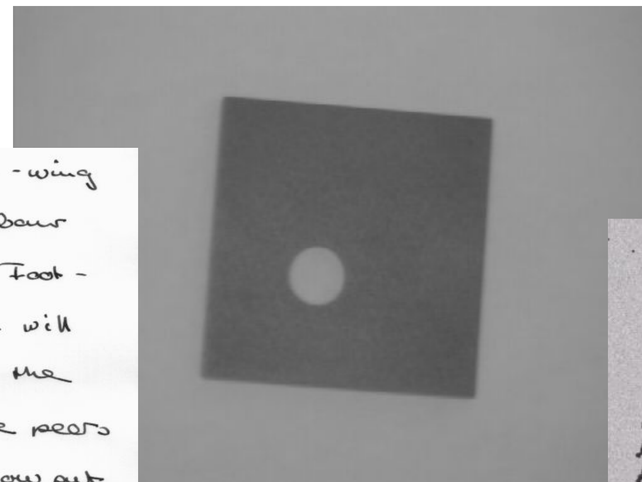
https://www.youtube.com/watch?v=hKYBNdSspIo

Thresholding

- Segmentation into two classes/groups
  - Foreground (Objects)
  - Background



Though they may gather some Left-wing
support, a large majority of Labour
MPs are likely to turn down the Foot-
Griffiths resolution. Mr. Foot's line will
be that as Labour MPs opposed the
Government Bill which brought life peers
into existence, they should not now put
forward nominees. He believes that the
House of Lords should be abolished and
that Labour should not take any steps
which would appear to "prop up" an out-



Thresholding

- Suppose that an image, f(x,y), is composed of light objects on a dark background, and the following figure is the histogram of the image.

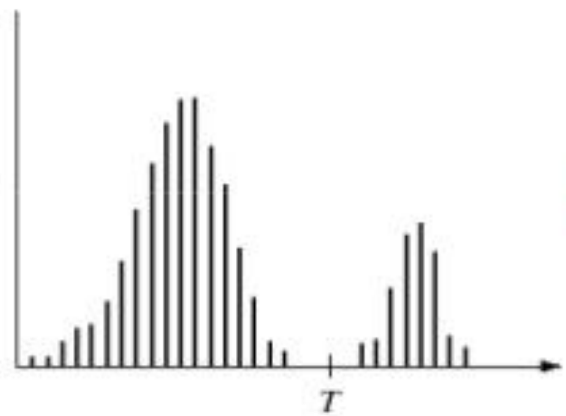


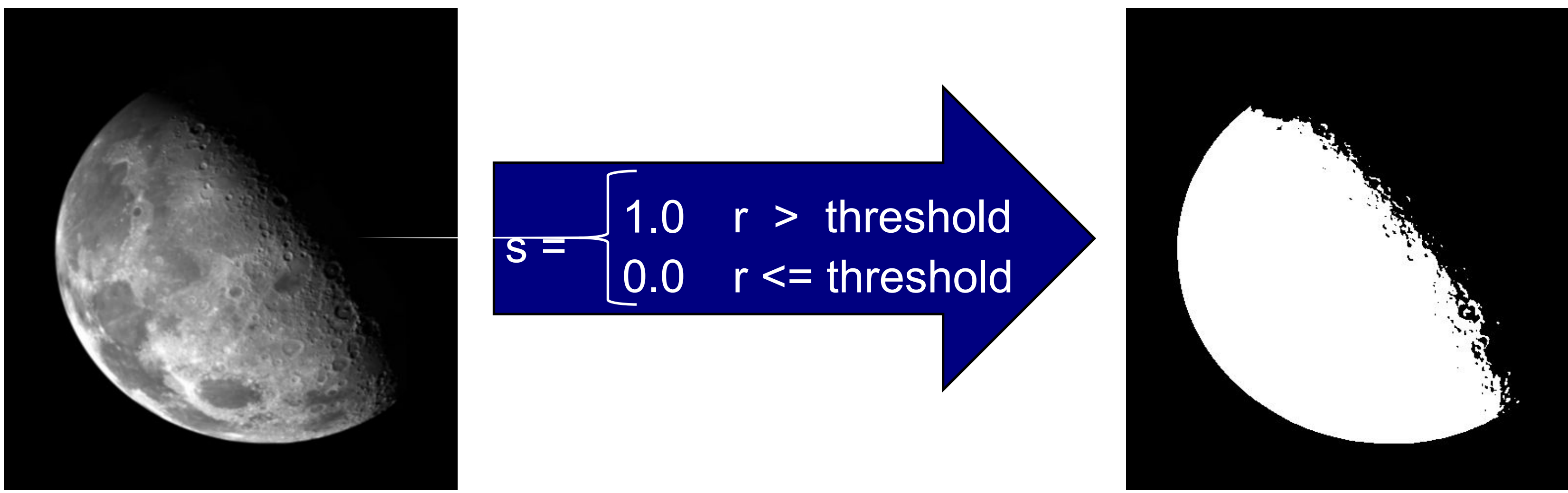
image with dark background
and a light object

- Then, the objects can be extracted by comparing pixel values with a threshold T.

Thresholding

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

Objects & Background



Thresholding

- GLOBAL
- ADAPTIVE
- LOCAL

Thresholding

- One way to extract the objects from the background is to select a threshold ***T*** that separates object from background.
- Any point *(x,y)* for which ***f(x,y) > T*** is called an object point; otherwise the point is called a background point.

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

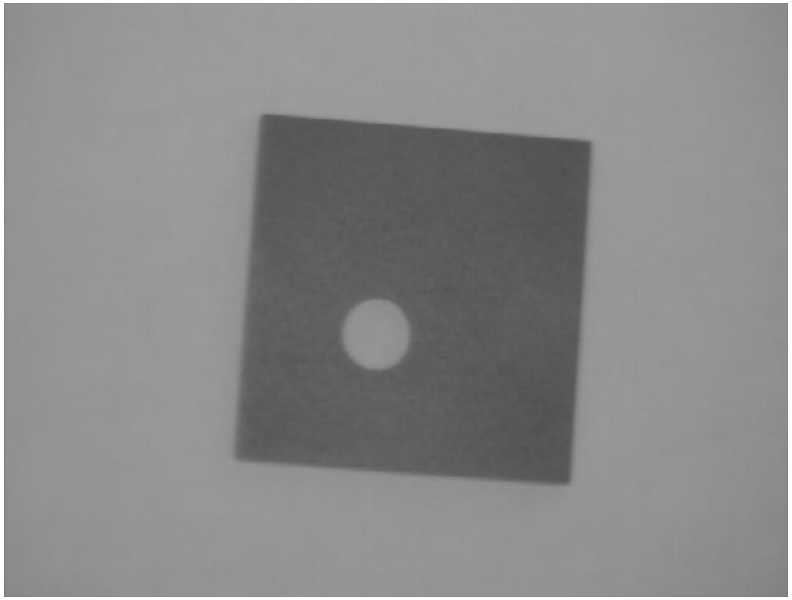
- When ***T*** is a constant applicable over an entire image, then the above process is called as *Global thresholding*.

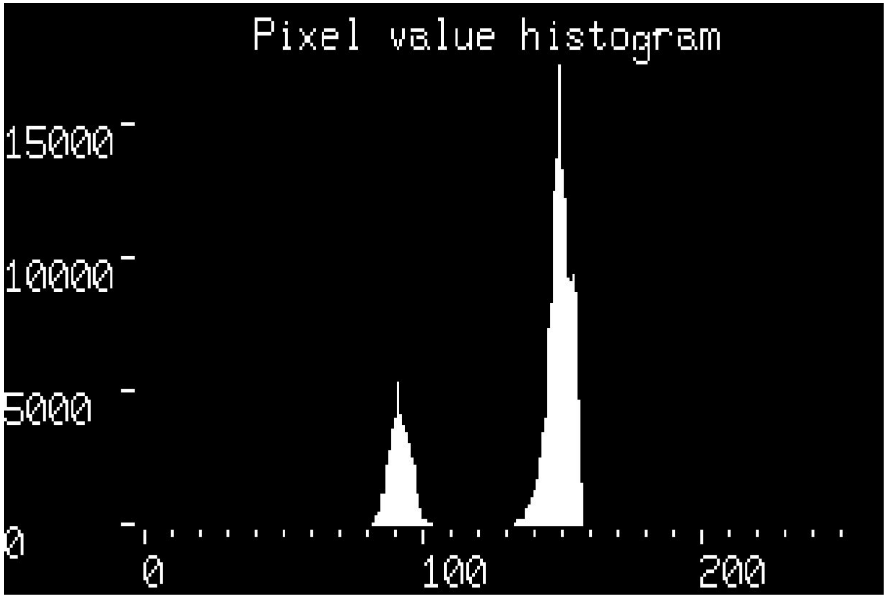
Thresholding

- When the value of ***T*** changes over an image
- Then that process is referred as *Variable thresholding*.
- Sometimes it is also termed as *local* or *regional thresholding.*
- Where, the value of ***T*** at any point (x,y) in an image depends on properties of a *neighborhood of* (x,y).
- If ***T*** depends on the spatial coordinates (x,y) themselves, then variable thresholding is often referred to as *dynamic* or *adaptive thresholding*.

Global Thresholding

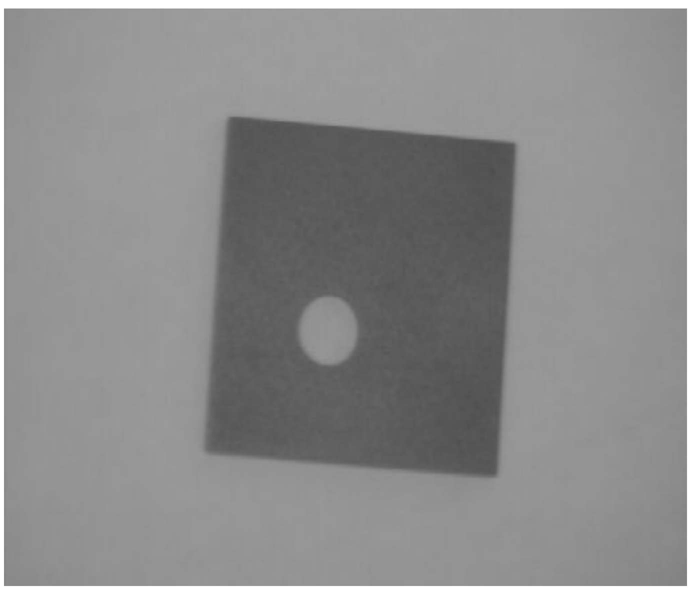
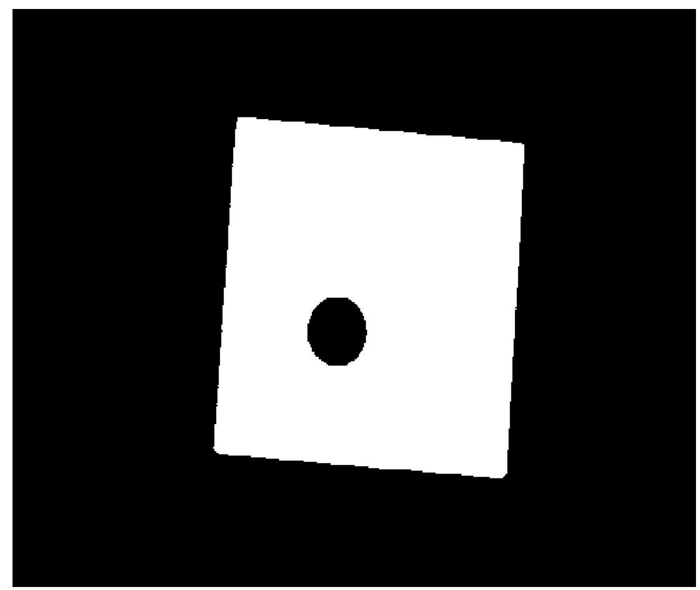
- Single threshold value for entire image
- Fixed ?
- Automatic
  - Intensity histogram





Global Thresholding

- Single threshold value for entire image
- Fixed ?
- Automatic
  - Intensity histogram

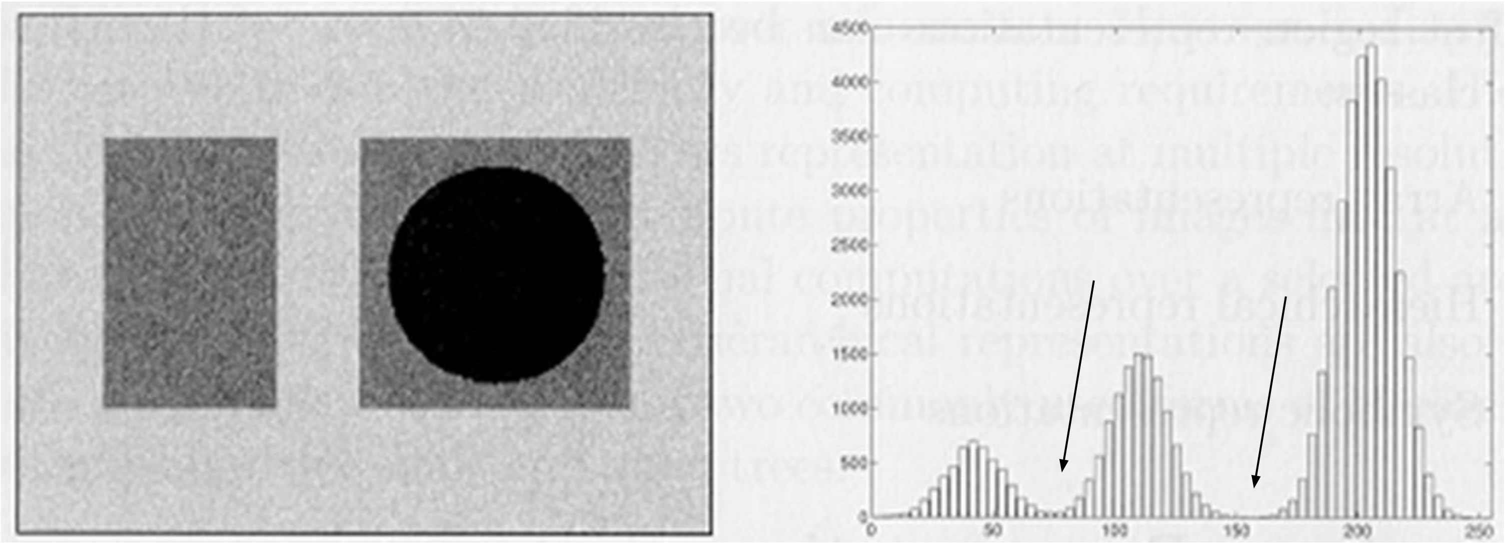
Global Thresholding

Based on visual inspection of histogram

1. Select an initial estimate for **T**.
2. Segment the image using **T**. This will produce two groups of pixels: G_1 consisting of all pixels with gray level values > **T** and G_2 consisting of pixels with gray level values $\leq$ **T**
3. Compute the average gray level values μ_1 and μ_2 for the pixels in regions G_1 and G_2
4. Compute a new threshold value
5. $\mathbf{T} = 0.5\,(\mu_1 + \mu_2)$
6. Repeat steps 2 through 4 until the difference between the values of **T** in successive iterations is smaller than a predefined parameter **ΔT**.

https://www.youtube.com/watch?v=f1SaYzOthCM

Global Thresholding?



Multilevel Thresholding can be a solution?

Multilevel thresholding

- A point (x,y) belongs
  - to an object class if $T_1 < f(x,y) \leq T_2$
  - to another object class if $f(x,y) > T_2$
  - to background if $f(x,y) \leq T_1$

$$g(x,y) = \begin{cases} a & \text{if } f(x,y) > T_2 \\ b & \text{if } T_1 \leq f(x,y) \leq T_2 \\ c & \text{if } f(x,y) \leq T_1 \end{cases}$$

Thresholding

Segmentation problems requiring multiple thresholds are best solved using region growing methods

Thresholding can be viewed as

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

where $f(x,y)$ is gray-level at (x,y) and $p(x,y)$ denotes some local property, for example average gray level in neighbourhood

Local Adaptive Thresholding

- The Local Adaptive Thresholding chooses different threshold values for every pixel in the image based on an analysis of its neighboring pixels.
  - Threshold: function of neighboring pixels

$$T = mean$$

$$T = median$$

$$T = \frac{\max + \min}{2}$$

Local Adaptive Thresholding

- Niblack Algorithm

$$T = m + k \times s$$

$$m = \text{mean}$$

$$s = \text{standard deviations}$$

$$k = \text{ Niblack constant}$$

- Neighborhood size???

Basic Adaptive Thresholding

- subdivide original image into small areas.
- utilize a different threshold to segment each subimages.
- since the threshold used for each pixel depends on the location of the pixel in terms of the subimages, this type of thresholding is adaptive.

Thresholding

A thresholded image $g(x, y)$ is defined as

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

where 1 is object and 0 is background

When $T = T[f(x, y)]$, threshold is **global**

When $T = T[p(x, y), f(x, y)]$, threshold is **local**

When $T = T[x, y, p(x, y), f(x, y)]$, threshold is **dynamic** or **adaptive**

https://youtu.be/nPyb2BVBeB8

K-Means Clustering

1. Chose the number (K) of clusters and randomly select the centroids of each cluster.
2. For each data point:
   - Calculate the distance from the data point to each cluster.
   - Assign the data point to the closest cluster.
3. Recompute the centroid of each cluster.
4. Repeat steps 2 and 3 until there is no further change in the assignment of data points (or in the centroids).