Fundamentals of Image Segmentation.

Image Segmentation is the process by which a digital image is partitioned into various subgroups (of pixels) called Image Objects, which can reduce the complexity of the image, and thus analyzing the image becomes simpler.

Similarity Detection (Region Approach)

This fundamental approach relies on detecting similar pixels in an image – based on a threshold, region growing, region spreading, and region merging so does classification, which detects similarity based on a pre-defined (known) set of features.

Discontinuity Detection (Boundary Approach)

This is a stark opposite of the similarity detection approach where the algorithm rather searches for discontinuity. Image Segmentation Algorithms like Edge Detection, Point Detection, Line Detection follow this approach — detected based on various metrics of discontinuity like intensity, etc.

Thresholding - Basics of Intensity thresholding and Global Thresholding.

Image thresholding is a technique employed to facilitate easy image segmentation for various image processing tasks.

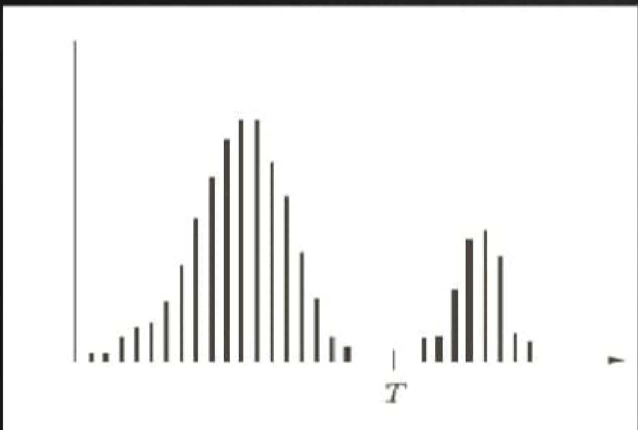
Simple thresholding technique (Binary Thresholding)

In a simple thresholding technique, a standard threshold value is set and each pixel value is compared with the threshold value. If the pixel value is less than the mentioned threshold value then the value is set to 0 or else it is set to the maximum value.

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

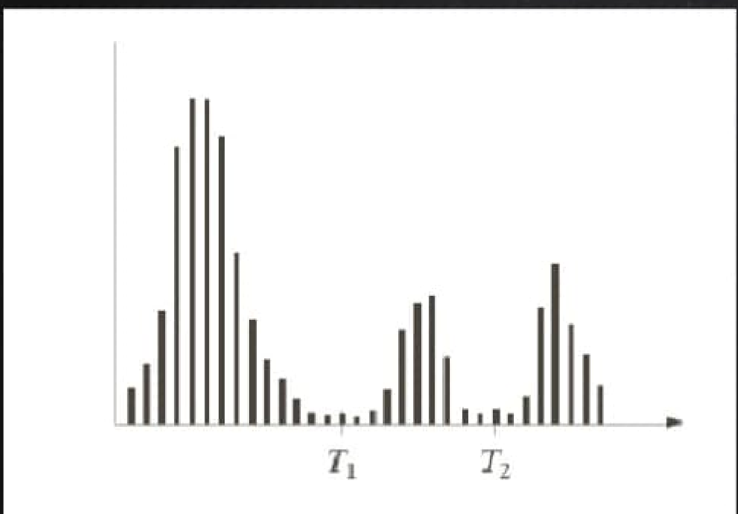
$$g(x,y) = \begin{cases} 1, & \text{if } f(x,y) > T \\ 0, & \text{if } f(x,y) \leq T \end{cases},$$

where 1 is object and 0 is background



Multiple thresholding:

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

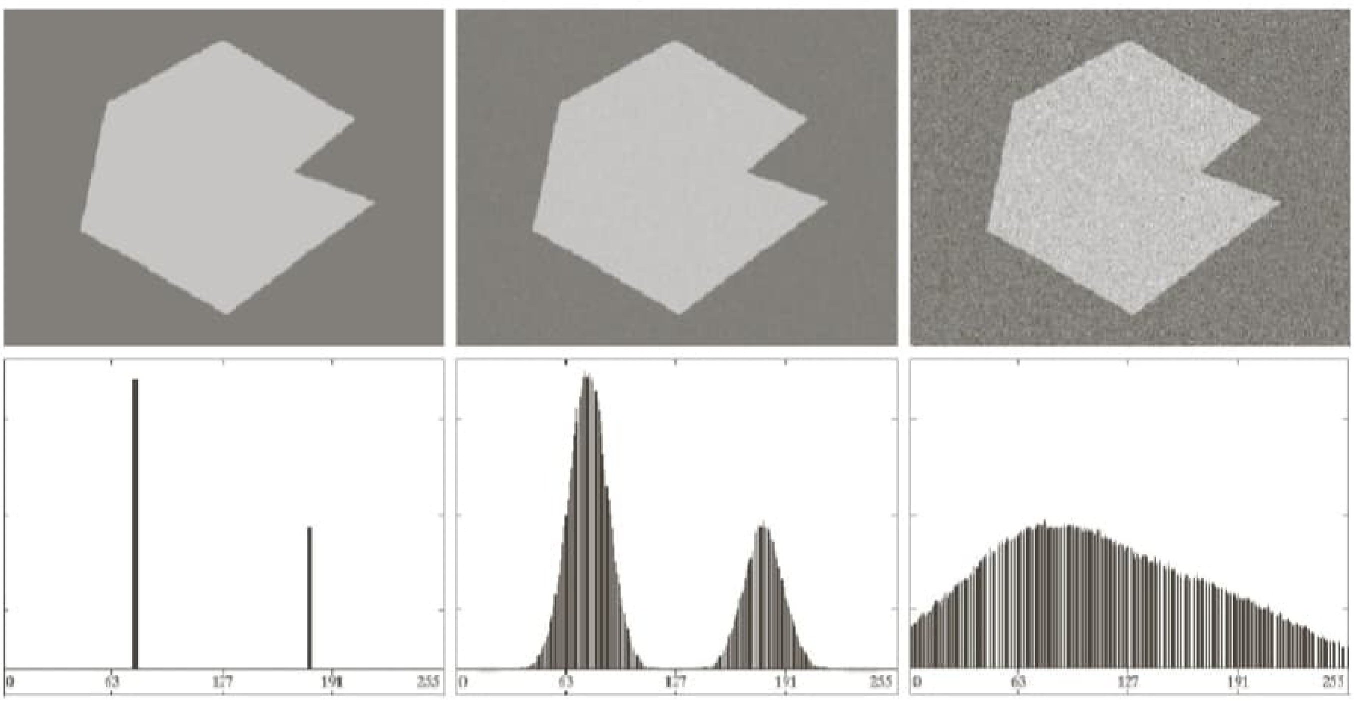


Global thresholding: T is constant and applicable over the whole image.

Variable/ Local thresholding: T changes over an image. T at a point (x,y) is a function of the neighborhood of (x,y).

Dynamic / Adaptic thresholding: T changes over an image. T at any point(x,y) is a function of spatial coordinate (x,y)

The role of noise in image thresholding



Basics of Global Thresholding

When the intensity distributions of objects and background pixels are sufficiently distinct, it is possible to use a single global threshold applicable over the entire image.

We have algorithms for estimating automatically the threshold value for each image.

Iterative algorithm

Otsu's method

Iterative algorithm for automatic estimation of threshold T:

(1) Select an initial estimate for T

(2) Segment image using $T \longrightarrow$ Group G_1 (values $> T$)
Group G_2 (values $\leq T$)

(3) Compute average intensity values for G_1, $G_2 \longrightarrow m_1, m_2$

(4) Compute a new threshold value $T = \frac{1}{2}(m_1 + m_2)$

(5) Repeat (2) through (4) until the difference in T in successive iterations is smaller than ΔT

Average intensity is good initial estimate for T

Fundamentals of Image Segmentation.

Image Segmentation is the process by which a digital image is partitioned into various subgroups (of pixels) called Image Objects, which can reduce the complexity of the image, and thus analyzing the image becomes simpler.

Similarity Detection (Region Approach)

This fundamental approach relies on detecting similar pixels in an image – based on a threshold, region growing, region spreading, and region merging so does classification, which detects similarity based on a pre-defined (known) set of features.

Discontinuity Detection (Boundary Approach)

This is a stark opposite of the similarity detection approach where the algorithm rather searches for discontinuity. Image Segmentation Algorithms like Edge Detection, Point Detection, Line Detection follow this approach —
detected based on various metrics of discontinuity like intensity, etc

Region-based Segmentation Approach - Region Growing, Region Splitting and Merging

A region can be classified as a group of connected pixels exhibiting similar properties. The similarity between pixels can be in terms of intensity, color, etc. In this type of segmentation, some predefined rules must be obeyed by a pixel to be classified into similar pixel regions. Region-based segmentation methods are preferred over edge-based segmentation methods in case of a noisy image. Region-Based techniques are further classified into 2 types based on the approaches they follow.

Region growing method

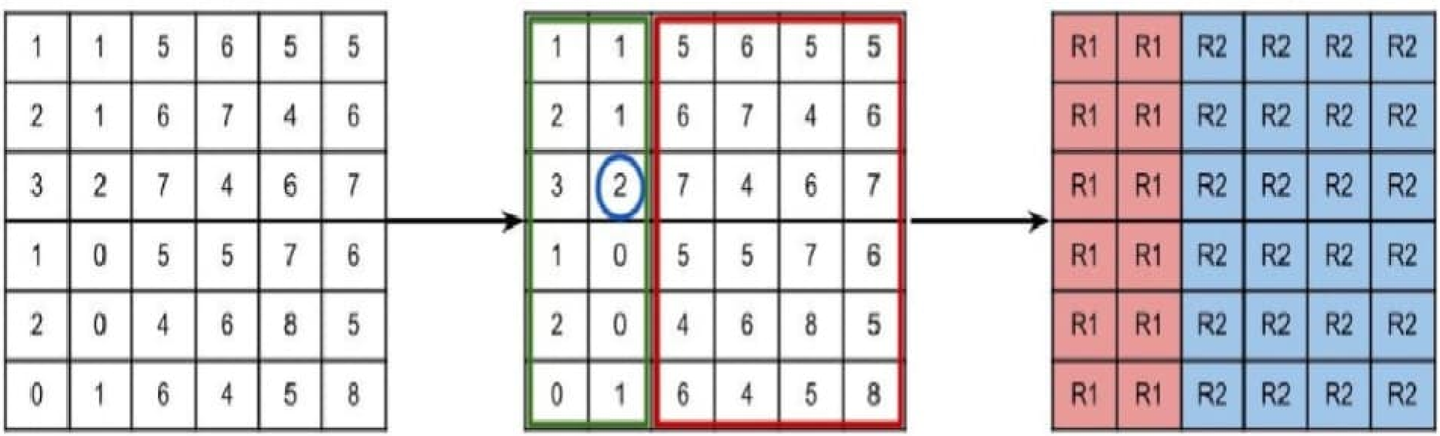
Region splitting and merging method

Region Growing Technique

In the case of the Region growing method, we start with some pixel as the seed pixel and then check the adjacent pixels.
If the adjacent pixels abide by the predefined rules, then that pixel is added to the region of the seed pixel and the following process continues till there is no similarity left. This method follows the bottom-up approach.

In case of a region growing, the preferred rule can be set as **a threshold.**

For example: Consider a seed pixel of 2 in the given image and a threshold value of 3, if a pixel has a value less than 3 then it will be considered inside the seed pixel region. Otherwise, it will be considered in another region.
Hence 2 regions are formed in the following image based on a threshold value of 3.



Original Image

Region growing process with 2 as the seed pixel.

Splitting image into two regions based on a threshold.

Region Splitting and Merging Technique

In Region splitting, the whole image is first taken as a single region. If the region does not follow the predefined rules, then it is further divided into multiple regions (usually 4 quadrants) and then the predefined rules are carried out on those regions in order to decide whether to further subdivide or to classify that as a region. The following process continues till there is no further division of regions required i.e every region follows the predefined rules.

In Region merging technique, we consider every pixel as an individual region. We select a region as the seed region to check if adjacent regions are similarly based on predefined rules. If they are similar, we merge them into a single region and move ahead in order to build the segmented regions of the whole image.

Usually, first region splitting is done on an image so as to split an image into maximum regions, and then these regions are merged in order to form a good segmented image of the original .

Apply region splitting on the following image. Assume the threshold value be<=4.

	5	6	6	6	7	7	6	6	
R1	6	7	6	7	5	5	4	7	R2
	6	6	4	4	3	2	5	6	
	5	4	5	4	2	3	4	6	
	0	3	2	3	3	2	4	7	
R4	0	0	0	0	2	2	5	6	R3
	1	1	0	1	0	3	4	4	
	1	0	1	0	2	3	5	4	

Higher value-Lower value>4 then split

7-0=7>4 ,split into 4 quadrants

Higher value-Lower value>4 then split

R1

7-4=3<=4 ,NO split

R1 R2

R4 R3

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

Higher value-Lower value>4 then split

R2

7-2=5>4 ,split

R21 R22

R1 R2

R24 R23

R4 R3

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

R21
7-5=2<=4 ,NO split

R22
7-4=3<=4 ,NO split

R23
6-4=2<=4 ,NO split

R24
3-2=1<=4 ,NO split

Higher value-Lower value>4 then split

R3

7-0=7>4 ,split

R21 R22

R1 R2 R24 R23 R31 R32 R4 R33 R34

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

R31
3-2=1<=4 ,NO split

R32
6-4=3<=4 ,NO split

R33
5-4=1<=4 ,NO split

R34
3-0=3<=4 ,NO split

Higher value-Lower value>4 then split

R4

3-0=3<=4 ,No split

R21 R22

R1 R2 R24 R23 R31 R32 R4 R33 R34

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

Merging

Higher value-Lower value>4 then split

Compare Higher value-Lower value>4 then split in both direction

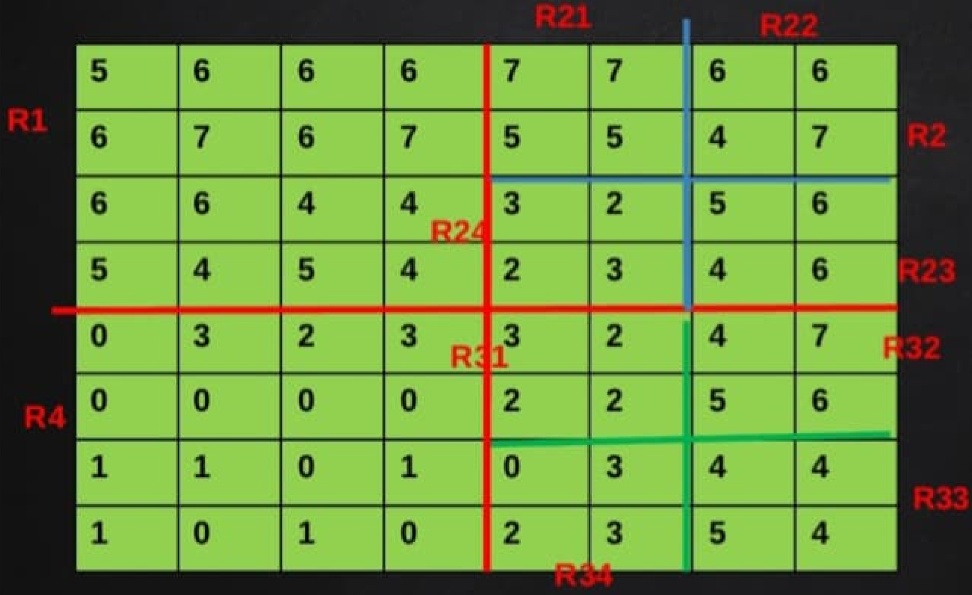
R1 and R21 (MAX-7,MIN-5)
7-5=2<=4,merge
R21 and R1 (MAX-7,MIN-4)

7-4=3<=4 ,MERGE

R1- R21 and R22 (MAX-7,MIN-4)

7-4=3<=4 ,MERGE

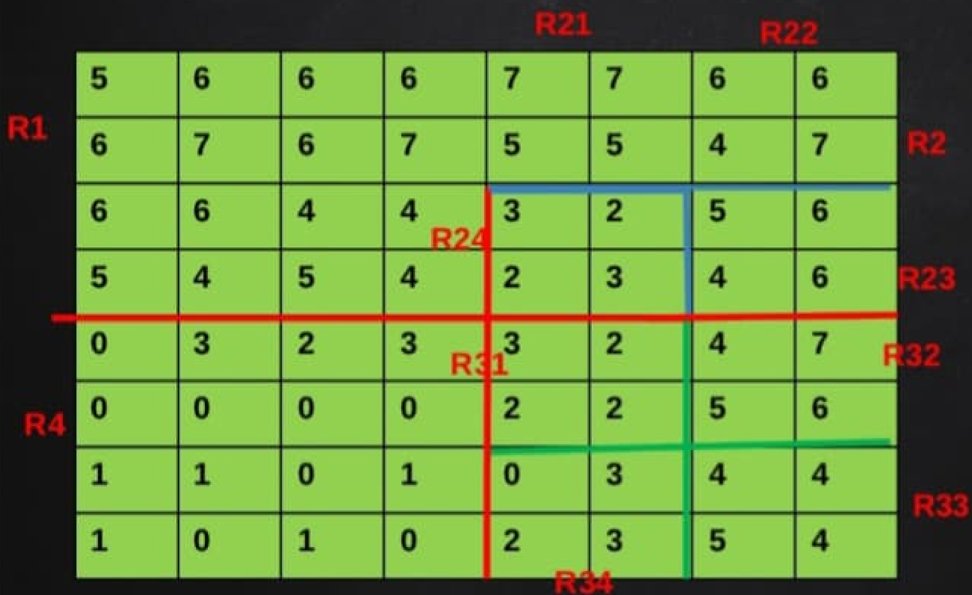
R1- R21 and R22 (MAX-7,MIN-4)
7-4=3<=4 ,MERGE



Merging

Higher value-Lower value>4 then split

Compare Higher value-Lower value>4 then split in both direction



Merging

Higher value-Lower value>4 then split

Compare Higher value-Lower value>4 then split in both direction

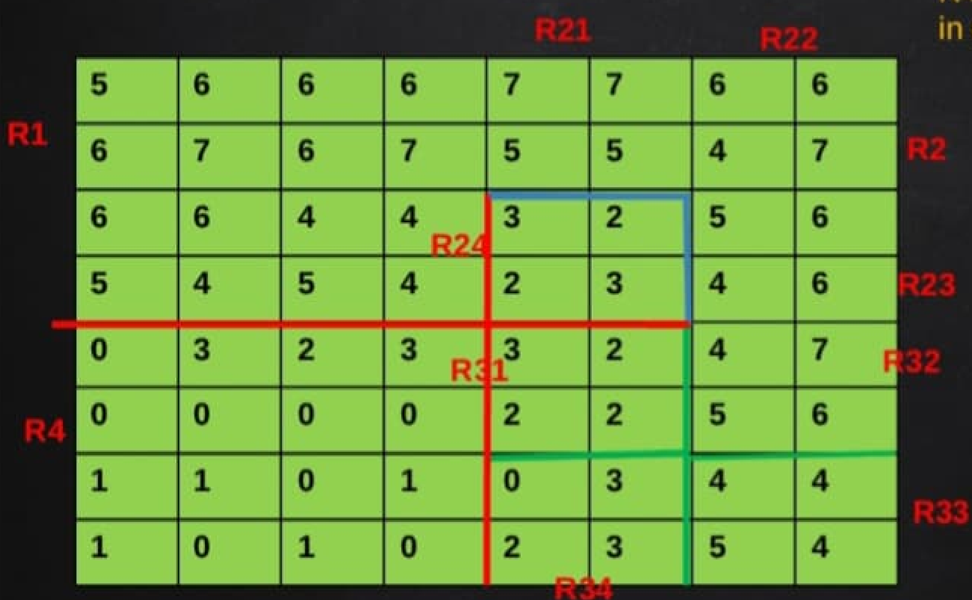
5	6	6	6	7	7	6	6
6	7	6	7	5	5	4	7
6	6	4	4	3	2	5	6
5	4	5	4	2	3	4	6
0	3	2	3	3	2	4	7
0	0	0	0	2	2	5	6
1	1	0	1	0	3	4	4
1	0	1	0	2	3	5	4

Merging

Higher value-Lower value>4 then split

Compare Higher value-Lower value>4 then split in both direction

R4, R31, R34, R24 Satisfy merging condition in either direction, so form a region.



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

Merging

Higher value-Lower value>4 then split

Repeat the same with all other regions and finally we get 2 regions