Image Segmentation -Region Growing

Aim:- To segment given image using region growing technique.

Theory:-

Edges and thresholds sometimes do not give good results for segmentation.

Region-based segmentation is based on the connectivity of similar pixels in a region.

- Each region must be uniform.
- Connectivity of the pixels within the region is very important.

There are two main approaches to region-based segmentation: region growing and region splitting.

Region-Growing based segmentation
Homogeneity of regions is used as the main segmentation criterion in region growing.
The criteria for homogeneity:
• gray level
• color
• texture
• shape
• model
The basic purpose of region growing is to segment an entire image R into smaller sub-images, Ri, i=1,2,….,N. which satisfy the following conditions:

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

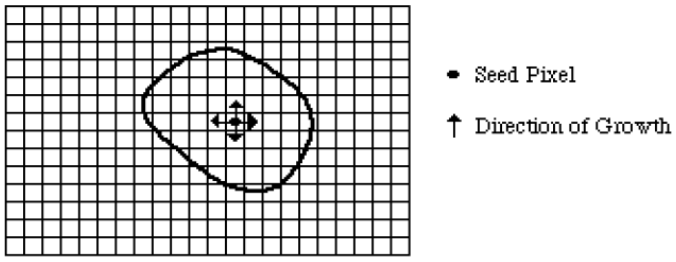
(b) R_i is a connected region, $i = 1,2,...,n$

(c) $R_i \cap R_j = \phi$ for all i and $j, i \neq j$

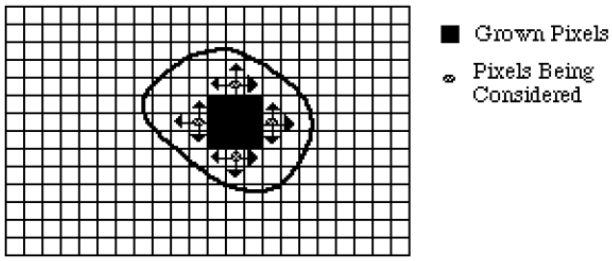
(d) $P(R_i) = \text{TRUE}$ for $i = 1,2,...,n$

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

where $P(R_k)$: a logical predicate defined over the points in set R_k
For example: $P(R_k)$=TRUE if all pixels in R_k have the same gray level.



(a) Start of Growing a Region



(b) Growing Process After a Few Iterations

Region growing approach is the opposite of the split and merge approach:
• An initial set of small areas is iteratively merged according to similarity constraints.
• Start by choosing an arbitrary *seed pixel* and compare it with neighboring pixels (see Fig).
• Region is *grown* from the seed pixel by adding in neighboring pixels that are similar, increasing the size of the region.
• When the growth of one region stops we simply choose another seed pixel which does not yet belong to any region and start again.
• This whole process is continued until all pixels belong to some region.
Region growing methods often give very good segmentations that correspond well to the observed edges. However starting with a particular seed pixel and letting this region grow completely before trying other seeds biases the segmentation in favour of the regions which are segmented first.
This can have several undesirable effects:
• Current region dominates the growth process -- ambiguities around edges of adjacent regions may not be resolved correctly.
• Different choices of seeds may give different segmentation results.
• Problems can occur if the (arbitrarily chosen) seed point lies on an edge.
To counter the above problems, *simultaneous region growing* techniques have been developed.
• Similarities of neighboring regions are taken into account in the growing process.
• No single region is allowed to completely dominate the proceedings.
• A number of regions are allowed to grow at the same time.
• Similar regions will gradually coalesce into expanding regions.
• Control of these methods may be quite complicated but efficient methods have been developed.
• Easy and efficient to implement on parallel computers.

Implementation Instructions:-

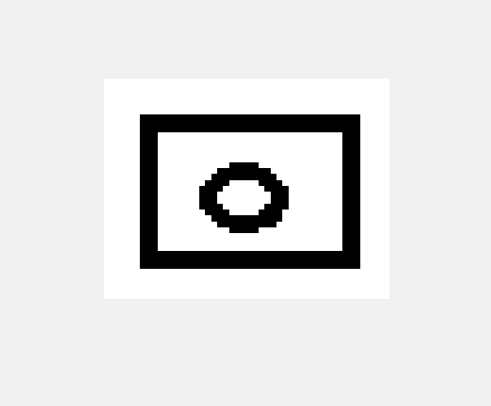
1) Read the given binary image.
2) Select a seed pixel using mouse cursor. (use ginput matlab function)
3) Create new image of same size as original image & assign value 1 to co-ordinates of seed pixel in new image.
4) Keep assigning value 1 to all connected pixels if predicate is true.
5) If predicate is false for all connected pixels of above region then select the next seed pixel from remaining part of the image & perform above operation with value 0.
6) Repeat the above steps for entire image & display the segmented image.

A) Binary Image

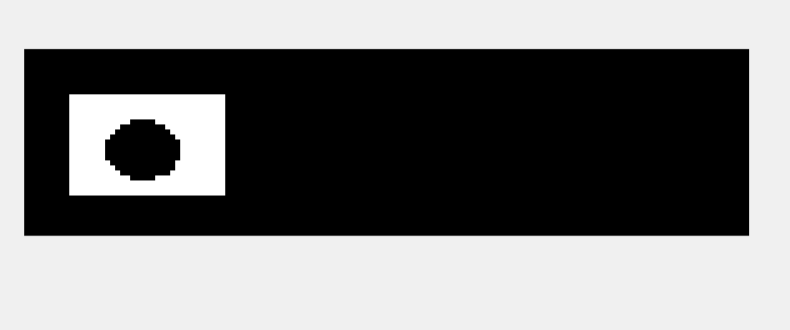
Code

```
%Experiment 6 - Region Growing
%Krisha Lakhani - 60001200097
clc;
clear all;
close all;
a=imread('test2.bmp');
% a=rgb2gray(a);
[r c]=size(a);
g=zeros(r,c);
figure(1);
imshow(a);
[Y,X]=ginput(1);
temp=a(round(X),round(Y));
g(round(X),round(Y))=1;
for x=2:1:r-1
    for y=2:1:c-1
        for p=2:1:r-1
            for q=2:1:c-1
             if(g(p,q)==1)
                    if a(p,q)==a(p,q+1)
                        g(p,q+1)=1;
                    end
                    if a(p,q)==a(p,q-1)
                        g(p,q-1)=1;
                    end
                    if a(p,q)==a(p-1,q)
                        g(p-1,q)=1;
                    end
                    if a(p,q)==a(p+1,q)
                        g(p+1,q)=1;
                    end
                end
            end
        end
    end
end
figure(2);
imshow(g);
```

Input



Output



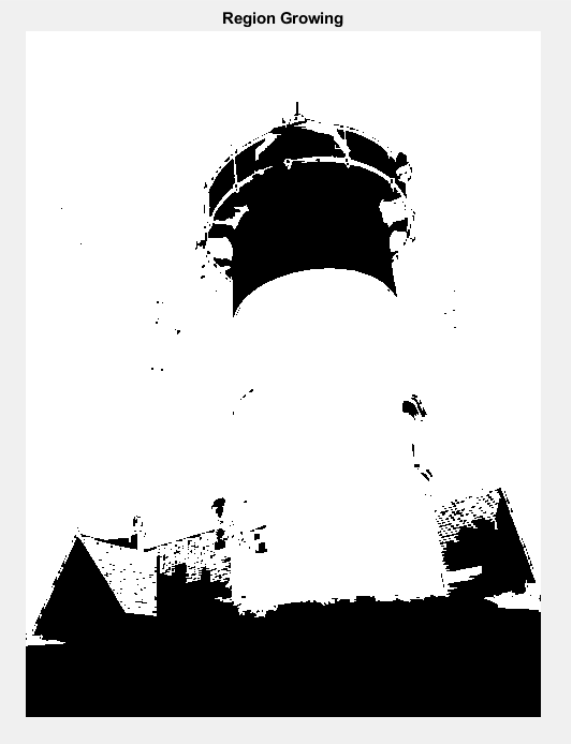
B) Greyscale Image

Code

```
%Experiment 6 - Image Segmentation (converting coloured to gray)
%Krisha Lakhani - 60001200097
clc
clear all;
image = imread('lighthouse.png');
image = rgb2gray(image);
figure(1);
imshow(image);
title("Original Image");
seed = ginput(1);
threshold = 5;
segmented_image = false(size(image));
[rows, cols] = size(image);
queue = zeros(rows * cols, 2);
queueSize = 0;
segmented_image(seed(1), seed(2)) = true;
queueSize = queueSize + 1;
queue(queueSize, :) = seed;
neighborhood = [-1, 0; 1, 0; 0, -1; 0, 1];
while queueSize > 0
    currentPixel = queue(queueSize, :);
    queueSize = queueSize - 1;
    for i = 1:size(neighborhood, 1)
        neighbor = currentPixel + neighborhood(i, :);
        if neighbor(1) >= 1 && neighbor(1) <= rows && neighbor(2) >= 1 && neighbor(2) <= cols
            if ~segmented_image(neighbor(1), neighbor(2)) && abs(image(neighbor(1), neighbor(2)) - image(currentPixel(1), currentPixel(2))) <= threshold
                segmented_image(neighbor(1), neighbor(2)) = true;
                queueSize = queueSize + 1;
                queue(queueSize, :) = neighbor;
            end
        end
    end
end
figure(2);
imshow(segmented_image);
title("Region Growing");
```

Input

Output



Conclusion:

The experiment of Image Segmentation (Region Growing) has been performed successfully by taking an input image and selecting a seed pixel using user input with the help of inbuilt MATLAB function (ginput) and a segmented image is generated.