Cairo University

Faculty of Engineering

Systems and Biomedical Engineering

Computer Vision

Task 4 Report

Submitted to:

Dr. Ahmed M. Badawi

TA/ Eng Laila Abbas.

TA/ Eng. Peter Salah.

Prepared by:

Team 8.

Name	Section	Bench Number
Arwa Essam	1	12
Sohaila Mahmoud	1	45
Shrouk Shawky	1	46
Maryam Megahed	2	34

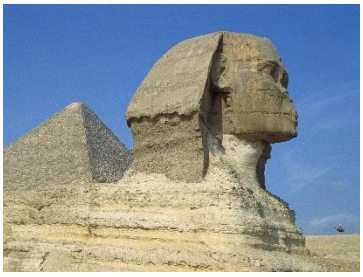
Functions Implemented

1. Optimal Thersholding:

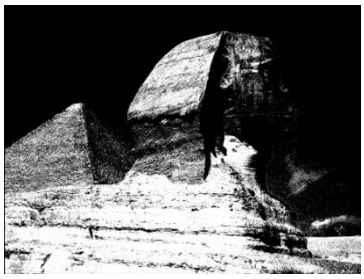
Optimal thresholding is a method used to automatically determine the threshold value to separate an image into foreground and background regions.

Here's how the function works:

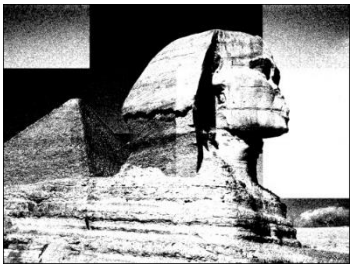
1. 1.The function calculates the mean intensity value of the four corner pixels (top left, top right, bottom left, bottom right) to get an initial estimate for the background mean.
2. then calculates the mean intensity value of all the pixels except for the corner pixels to get an initial estimate for the object mean.
3. The initial threshold value is calculated as the average of the background and object mean intensity values.
4. The OptimalThreshold() function is called with the initial threshold value to obtain a new threshold value.
5. A while loop is used to repeatedly call OptimalThreshold() with the previous threshold value until the threshold value no longer changes.



Original image



optimal global



optimal local

2. Otsu Thresholding:

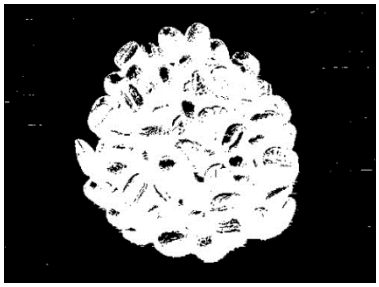
Otsu thresholding is a widely used image thresholding method that automatically determines the threshold value for image segmentatio

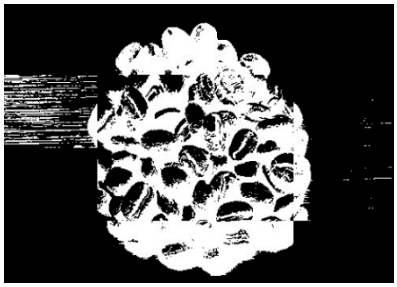
Here's how the function works:

1. first compute the histogram of the input image.
2. normalize the histogram so that it represents a probability distribution.
3. compute the cumulative sum and cumulative mean of the normalized histogram.
4. compute the global mean of the image.
5. compute the between-class variance for each possible threshold value.

6.find the threshold value that maximizes the between-class variance, which is the Otsu threshold value


Original image


ostu global


ostu local

3. Spectral Thresholding:

Spectral thresholding is a method of image segmentation that involves identifying different modes or clusters in an image's intensity histogram and assigning each cluster to a different segment or class

Here's how the function works:

1. It takes the input grayscale image and converts it to grayscale if it's not already.
2. It calculates the histogram of the image.
3. It calculates the probability density function (PDF) and cumulative distribution function (CDF) of the histogram.
4. It initializes the optimal low and high threshold values and the maximum variance to zero.
5. It calculates the global mean of the image intensity values.
6. It iterates over all possible low and high threshold values, and for each combination, it calculates the variance between the three classes of pixel intensities.
7. If the variance is greater than the maximum variance found so far, it updates the maximum variance and the optimal low and high threshold values.
8. It applies double thresholding to the input image using the optimal low and high threshold values found in the previous step.


original image

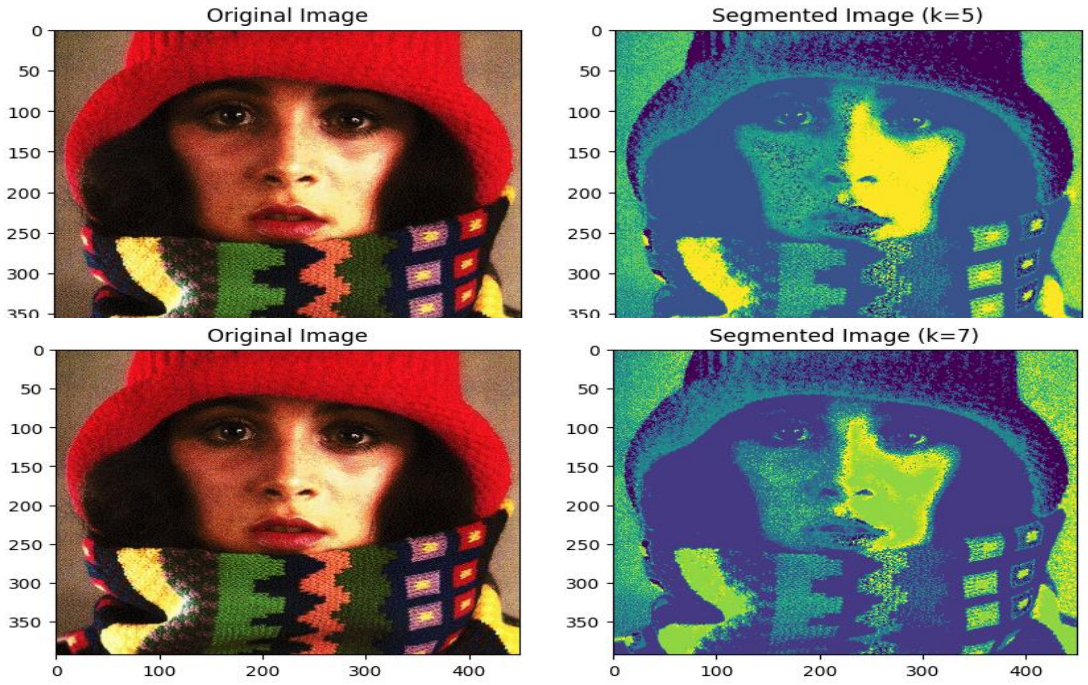

spectral global


spectral local

4. K-means Segmentation:

K-means segmentation is a clustering algorithm that partitions an image into K clusters, where K is a user-defined number.in this function user set the max number of iteration he want and also can set threshold.

In this algorithm, each pixel in the image is treated as a data point and the goal is to group similar pixels into clusters. The algorithm starts by randomly assigning K cluster centers, which are typically chosen from the image pixels. Then, it iteratively performs the following two steps: (1) Assign each pixel to the nearest cluster center, based on its color or intensity value; (2) Recalculate the cluster centers as the mean of all the pixels assigned to each cluster. These two steps are repeated until convergence, which occurs when the cluster centers no longer change significantly between iterations.

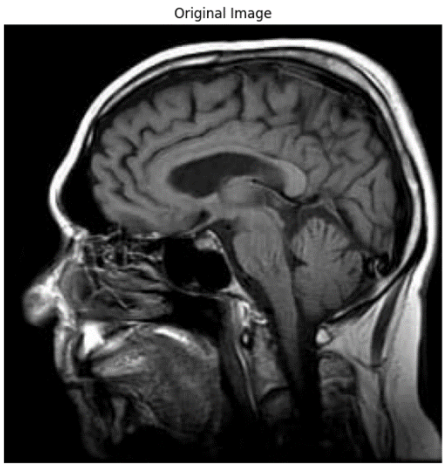


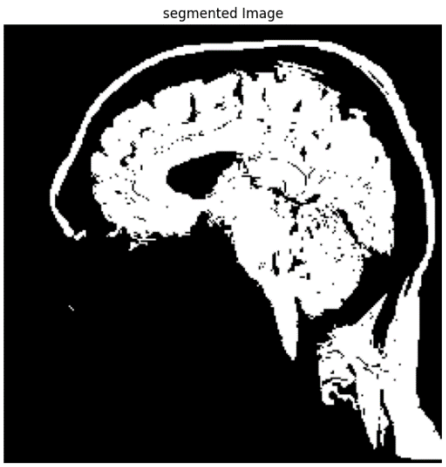
5. Region Growing Segmentation:

Region growing is a segmentation algorithm that groups pixels in an image based on their similarity to each other. It starts by selecting a seed pixel or group of pixels and then examines its neighbors to determine if they meet a certain similarity criterion. If the neighbors are similar, they are added to the growing region, and the algorithm continues examining their neighbors in a similar manner. This process continues until there are no more pixels that meet the similarity criterion.

We have implemented two functions one that asks the user to set seed points and its thresholds value to try and see when he changes their values how the output image change.
The other one set the seed point random by a function.

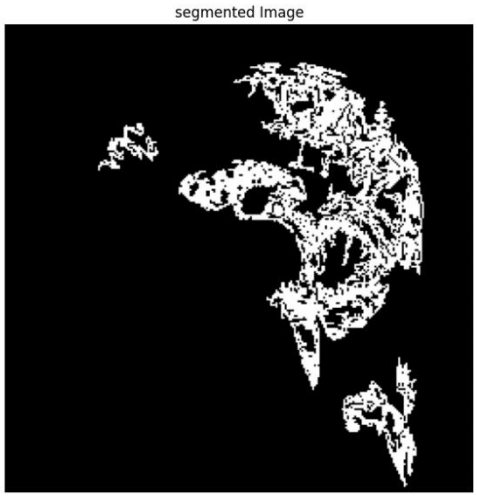
- With Random Seed:





- Whit Another Random Seed:

- With set seed points and threshold:
  seeds = [(70, 70), (100, 150), (200, 200)] and threshold = (10, 55, 105)

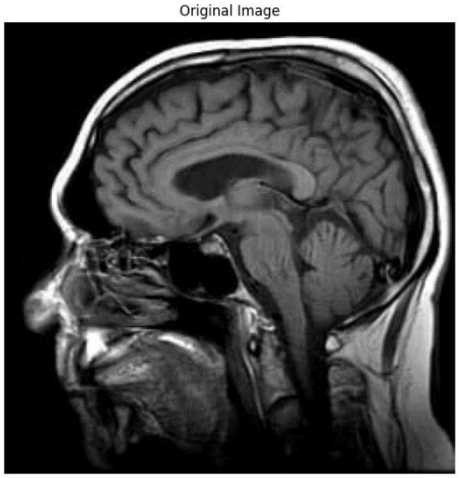


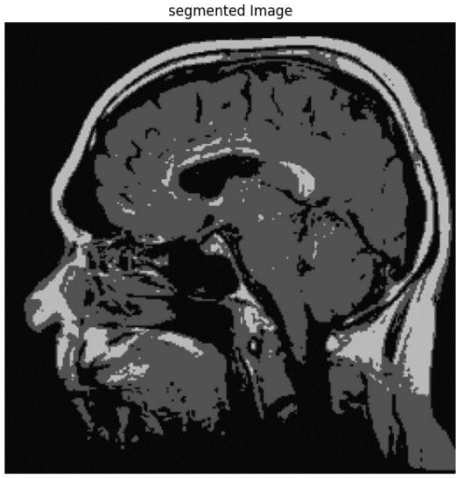
- With another set seed points and threshold:
  seeds = [(70, 70), (100, 150), (200, 200)] and threshold = (30, 55, 105)

6. Agglomerative Segmentation:

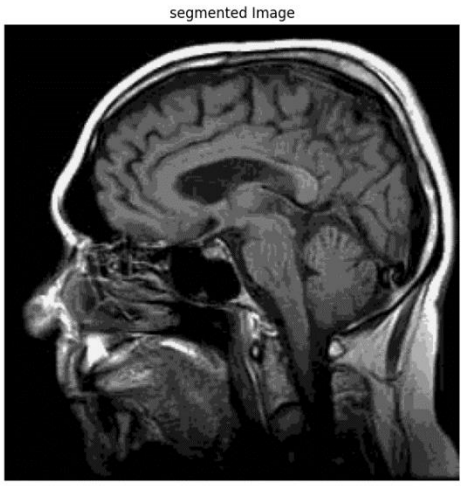
Agglomerative segmentation is a technique that groups similar pixels into clusters based on their proximity to each other. The algorithm starts by considering each pixel as a separate cluster and then merges the closest clusters iteratively until a desired number of segments is achieved.

- With number of clusters equal 5

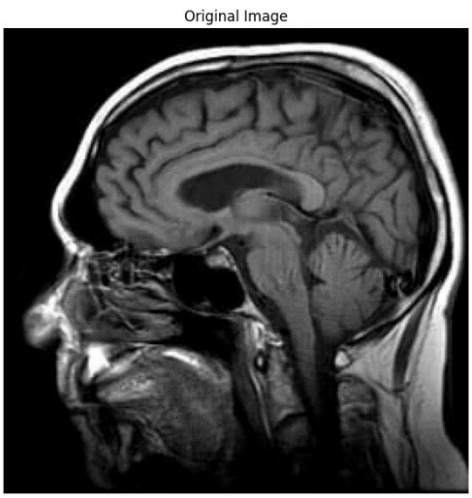


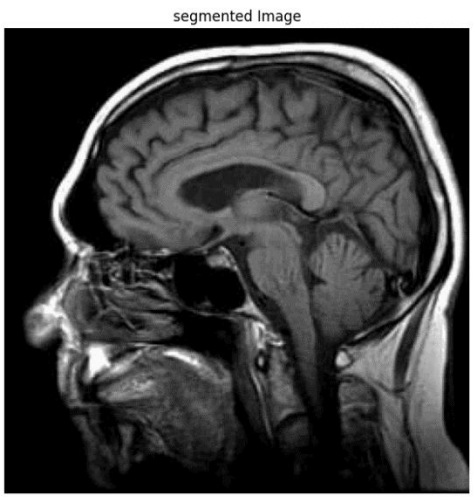


- With number of clusters equal 10



- With number of clusters equal 25:

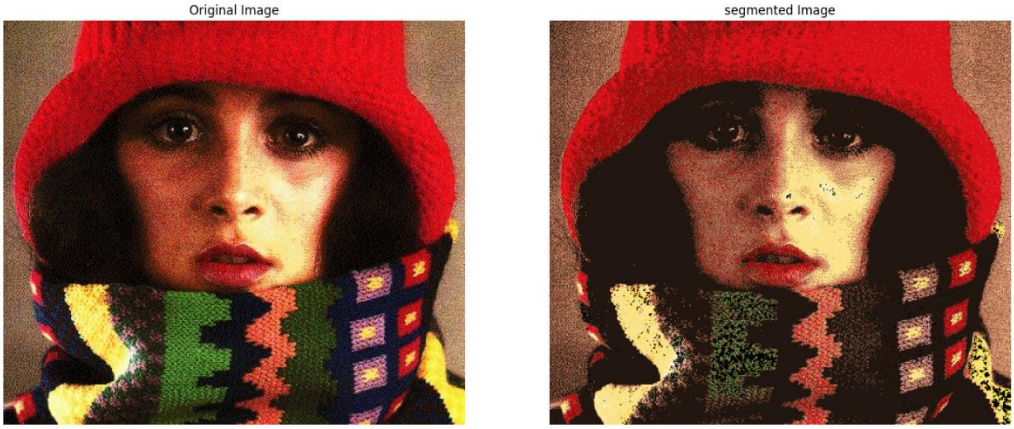




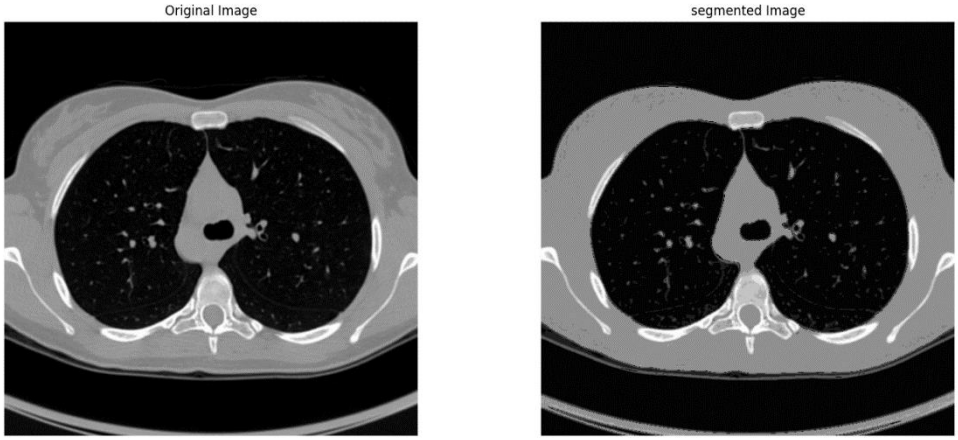
Mean Shift Segmentation

Mean shift segmentation is a clustering-based algorithm that is used to group similar pixels in an image together into distinct regions. The algorithm works by first selecting a window around each pixel in the image. The size of the window is determined by the user, which controls the sensitivity of the algorithm to changes in pixel intensity.

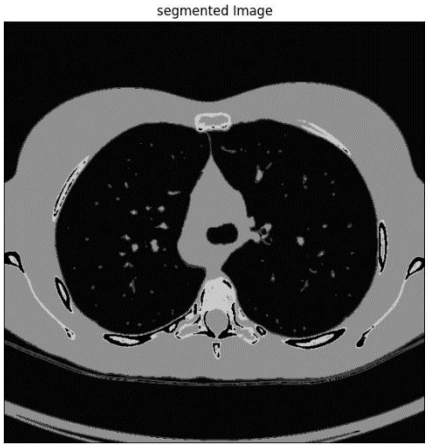
- Mean Shift with window size 60:



- Mean Shift with window size 30:



- Mean Shift with window size 60:



- Mean Shift with window size 90: