NATIONAL AND KAPODISTRIAN UNIVERSITY OF ATHENS

School of Science

Information Technologies in Medicine and Biology

Direction: *Bioinformatics*

Image Processing and Analysis

Postgraduate Student: *Begetis Nikolaos*

Professor: *Sangriotis Manolis*

Deadline Date: 26*/04/2013*

Assignment 5

Task 1

In this assignment we were asked to implement a program which will use three functions for the isolation and admeasure of the cyclic disks that exist in three images given. The assignment is separated in three tasks and we have been given one image for each task. In our main program we let the user select which of the three images he wants the program to resolve and admeasure its circles. Then, according to the user selection we execute one of the three functions, the best for each purpose.

So, if user selects the first image *(Image_1)*, then the circle disks of the image given are admeasured, by using the first of our implemented functions. Else if the user selects the second image *(Image_2)*, then the dark circles of Image_2 are isolated and admeasured by our second implemented function. Finally, if user selects the third image *(Image_3)*, then the program separates the large circles from the minor and isolates them differently before admeasuring them, by using the third of our implemented functions.

Task1 using `function []=func_1(im)`

For the first task we used the func_1(im) function which takes as an argument the image to be processed. For isolating the white circles from the white lines we used *erosion* by giving as a structure element the *'disk'* with diameter *5*, in which we concluded after some iterations for finding the best value. Then, we used the *bwlabel* function to calculate the number of the circles and we concluded that in *Image_1* exist 30 circles.

The input and the resulted figure and number of circles are shown in Figure 1, Figure 2 and Figure 3.

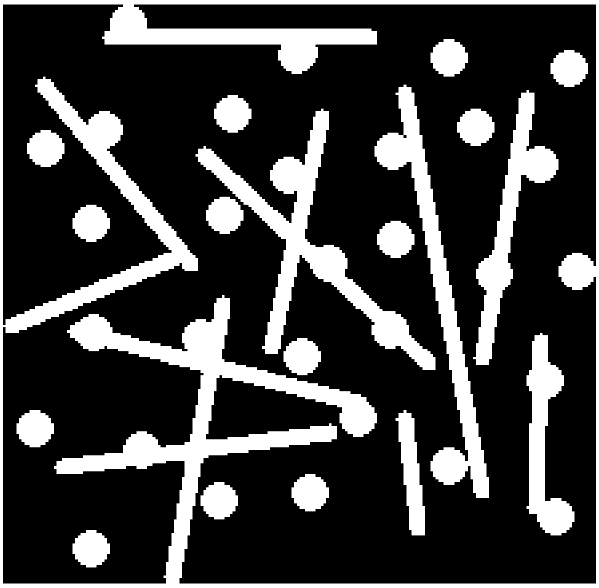


Figure 1: Input Image_1

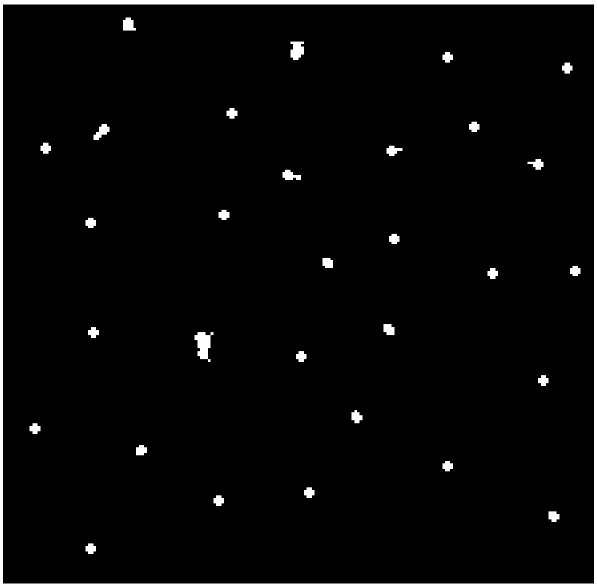
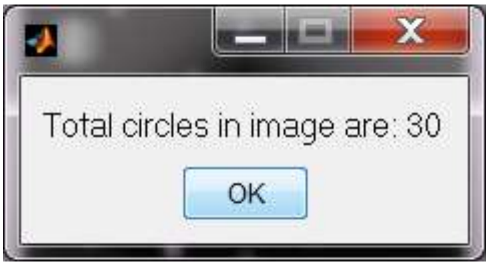


Figure 2: Output Image_1



Total circles in image are: 30

Figure 3: Result of calculating Image_1 circles

Task2 using function []=func_2(im)

For the second task we used the func_2(im) function which takes as an argument the image to be processed. For isolating the black circles and enumerate them we used *erosion* by giving as a structure element the *'disk'* with diameter *10*, in which we concluded after some iterations for finding the best value. But before that, we were needed to make the cyclic disks white and the background black. To do that, we found the threshold in which the image differs the circles from the background and with observation we applied a threshold with value 100 and turned the black circles in white and drew the background black. Then, we used the *bw* and *bwlabel* functions to calculate the number of the circles and we concluded that in *Image_1* exist 22 circles.

The input and the resulted figures and number of circles are shown in Figure 4, Figure 5, Figure 6, Figure 7 and Figure 8.

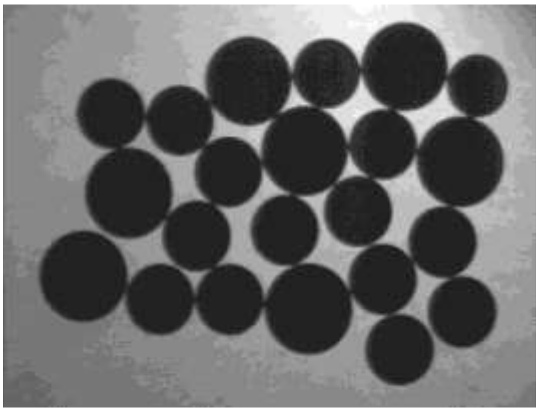


Figure 4: Input Image_2

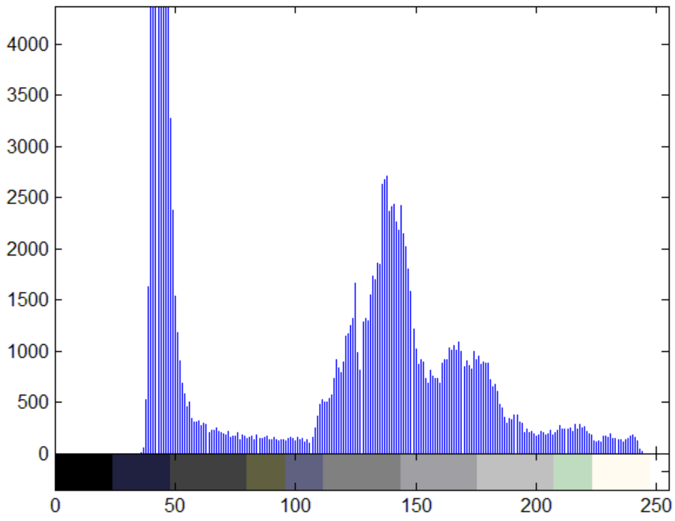


Figure 5: Input Image_2 histogram

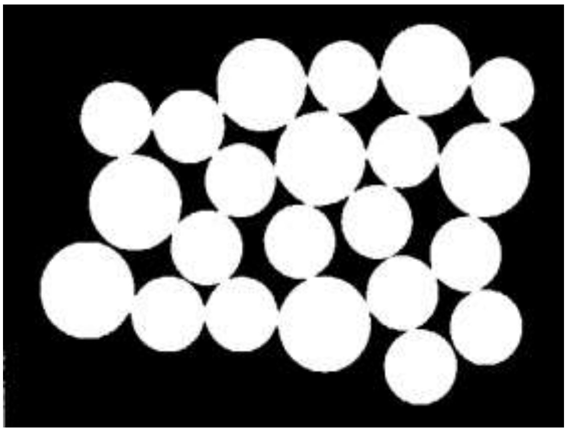


Figure 6: Output Image_2 before erosion

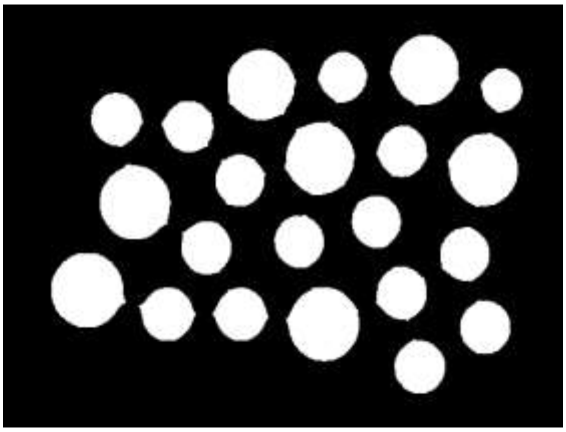
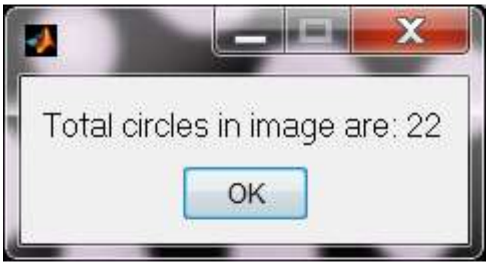


Figure 7: Output Image_2



Total circles in image are: 22

Figure 8: Result of calculating Image_2 circles

Task3 using `function []=func_3(im)`

For the third task we used the func_3(im) function which takes as an argument the image to be processed. For isolating the black circles and enumerate them we used *erosion* by giving as a structure element the *'disk'* with diameter *5*, in which we concluded after some iterations for finding the best value. But before that, we were needed to make the cyclic disks white and the background black. To do that, we reversed the Image_3 color by using the complement of the image. Then, we used *erode* to distinguish the big circles from the small by eliminating the small ones and find both the big and the small ones. At the end we used the *bw* and *bwlabel* functions to calculate the number of the circles in group of circles and we concluded that in *Image_3* exist 23 small circles and 6 big circles.

The input and the resulted figures and number of circles are shown in Figure 9, Figure 10, Figure 11 and Figure 12. At the end of this report follows the appendix with the code implemented for this assignment.

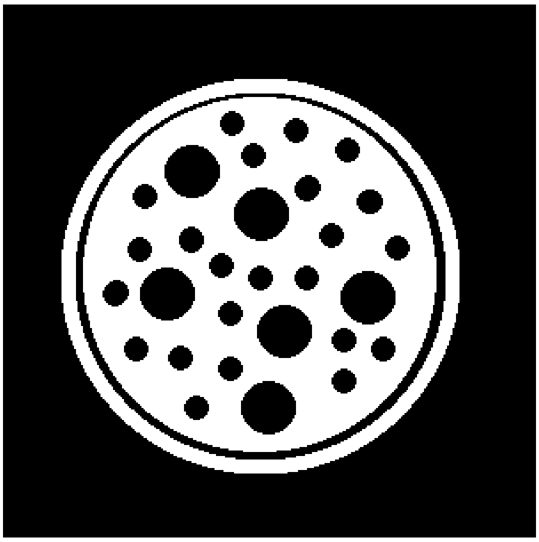


Figure 9: Input Image_3

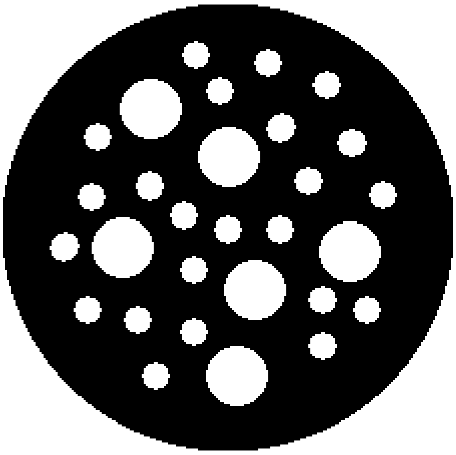


Figure 10: Output complement of input Image_3

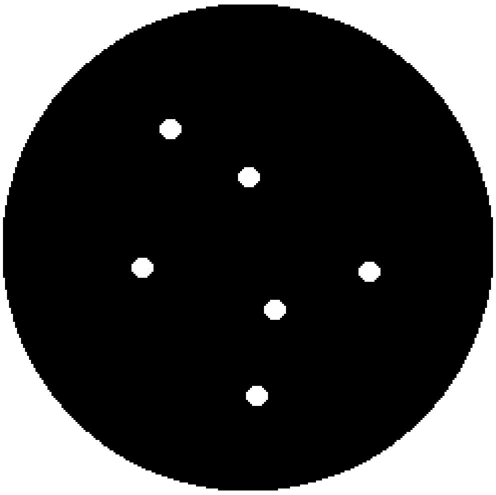
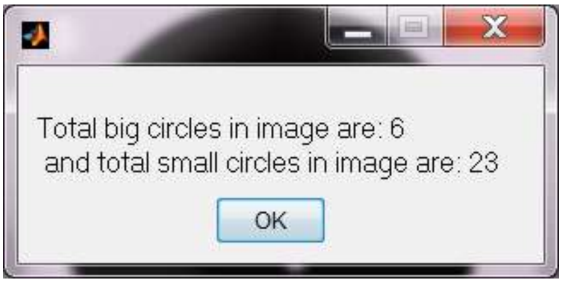


Figure 11: Output Image_3 after erosion for keeping the big circles

.



```
Total big circles in image are: 6
 and total small circles in image are: 23
```

Figure 12: Result of calculating Image_3 circles

Appendix

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%
%% Author: Begetis Nikolaos - Postgraduate Student - Bioinformatics,
ITMB
%% Supervisor: Sangriotis Manolis - Professor - ITMB
%% Course: Image Processing and Analysis
%% Filename: cirlces_nbegetis.m
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%

% A program which uses three functions, according to the user
selection. At
% first, if user selects the first choice, then the circle disks of
the
% image given are admeasured. Else if the user selects the second
choice,
% the dark circles of Image_2 are isolated and admeasured. And
finally, if
% user selects the third choice, then the program separates the large
% circles from the minor and isolates them differently and at last
% admeasures them.

function []=circles_nbegetis()
%% first part of the program
    clear *;clc;
    close all;

    im1=imread('./Image_1.tif');
    im2=imread('./Image_2.tif');
    im3=imread('./Image_3.tif');

    % Plot images
    colormap('gray');
    subplot(1, 3, 1); imagesc(im1); xlabel('Image_1');
    axis equal; axis([1 size(im1, 2) 1 size(im1, 1)]);

    subplot(1, 3, 2); imagesc(im2); xlabel('Image_2');
    axis equal; axis([1 size(im2, 2) 1 size(im2, 1)]);

    subplot(1, 3, 3); imagesc(im3); xlabel('Image_3');
    axis equal; axis([1 size(im3, 2) 1 size(im3, 1)]);


    choice=menu('Choose the image of cyclic disk you want to
admeasure', 'Image_1', 'Image_2', 'Image_3');

    switch choice
        case 1
            im=imread('./Image_1.tif');
            func_1(im);
        case 2
            im=imread('./Image_2.tif');
            func_2(im);
         case 3
            im=imread('./Image_3.tif');
            func_3(im);
    end
end


%% function 1: the circle disks of the Image_1 are admeasured
function []=func_1(im)
    se=strel('disk',5,0);
    imshow(im);
    im=imerode(im,se);
    figure;
    imshow(im);
    imwrite(im,'Image_1_circles.bmp');
    [L,num]=bwlabel(im);
    msgbox(sprintf('Total circles in image are: %d',num))
end


%% function 2: the dark circles of Image_2 are isolated and
admeasured
function []=func_2(im)
    figure;
    imshow(im)
    figure;
    imhist(im)
    imwrite(im,'Image_2_hist.bmp');
    % Threshold thresh=100 is used to distinguish the circles from
the
    % background
    thresh=100;

    % The threshold procedure
    bw=ones(size(im))*255;
    for line=1:size(bw,1)
        for clmn=1:size(bw,2)
            if double(im(line,clmn)>thresh)
                bw(line,clmn)=0;
            end
        end
    end
    figure;imshow(bw,[])
    imwrite(bw,'Image_2_circles.bmp');

    % Selection of element structure.
    se=strel('disk',10,0);
    bw=imerode(bw,se);
    figure;imshow(bw,[]);
    imwrite(bw,'Image_2_circles_final.bmp');
    [L,num]=bwlabel(bw);
    msgbox(sprintf('Total circles in image are: %d',num))
end


%% function 3: the program separates the large circles from the minor
and
% isolates them differently and at last admeasures them.
function []=func_3(im)
    figure;
    imshow(im)
    % remove cyclic black line between the two white regions
    se=strel('disk',5,0);
    im=imclose(im,se);

    % find the complement of the image
    im=imcomplement(im);
    figure;
    imshow(im);
    imwrite(im,'Image_3_complement.bmp');
    [L,num1]=bwlabel(im);    %
num1=big_circles_num+small_circles_num+1

    % erode to remove small cycles
    se=strel('disk',10,0);
    im=imerode(im,se);
    figure;imshow(im)
    imwrite(im,'Image_3_big_circles.bmp');
    [L,num2]=bwlabel(im);    % num1=big_circles_num+1

    big_circles_num=num2-1
    small_circles_num=num1-big_circles_num-1
    msgbox(sprintf('Total big circles in image are: %d \n and total
small circles in image are: %d',big_circles_num,small_circles_num))
end
```