“Makhana Size and Defect Detection”

A Report

Submitted as special assignment

of

2EIDE59 IMAGE PROCESSING AND APPLICATIONS

By
(PRAKHAR AGARWAL)
(21bei040))

Under the Guidance of
Prof. Harsh Kapadia

ELECTRONICS AND

INSTRUMENTATION ENGINEERING

INSTITUTE OF TECHNOLOGY

NIRMA UNIVERSITY

Ahmedabad 382 481

APRIL 2024

INTRODUCTION

Problem Statement: -

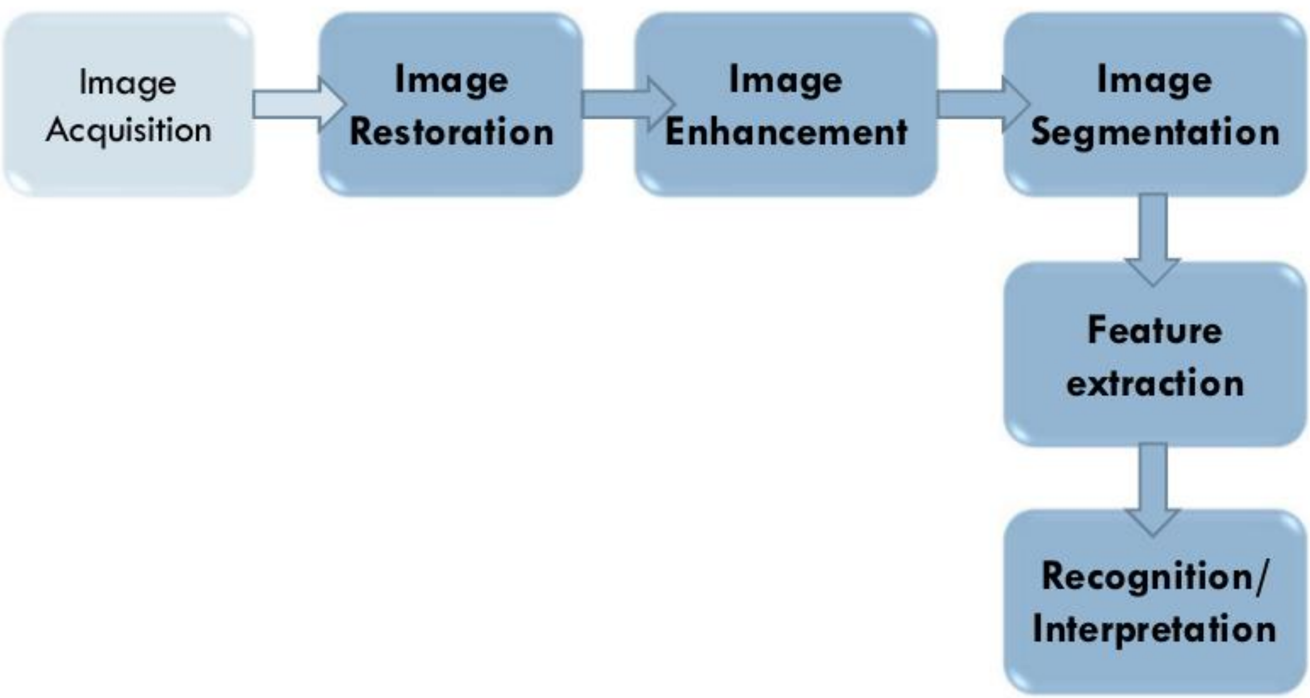
Makhana also known as Foxnut comes in various sizes so size Identification of makhana and also finding circularity of Makhana is difficult task.
Makhana if not processed has brown covers on top so identification of that is also a difficult task.

Here we have used various image processing algorithms to identity the makhana size, Circularity and the makhana with brown cover on it.

The algorithms used are Thresholding, Dilating, Smoothing filters, and various size Filters.

The Software used for this Project is Matlab.

BLOCK DIAGRAM



IMAGES

We start the project by Clicking the images of makhana under black background. We then Uploaded the Images into the Matlab Online Application . We then Applied various Operations.




The Algorithm

We first read the Image.

```
%% Read in Image
RGB = imread("20240422_100749.jpg");
% Display Image
figure(1);
clf
imshow(RGB)
```

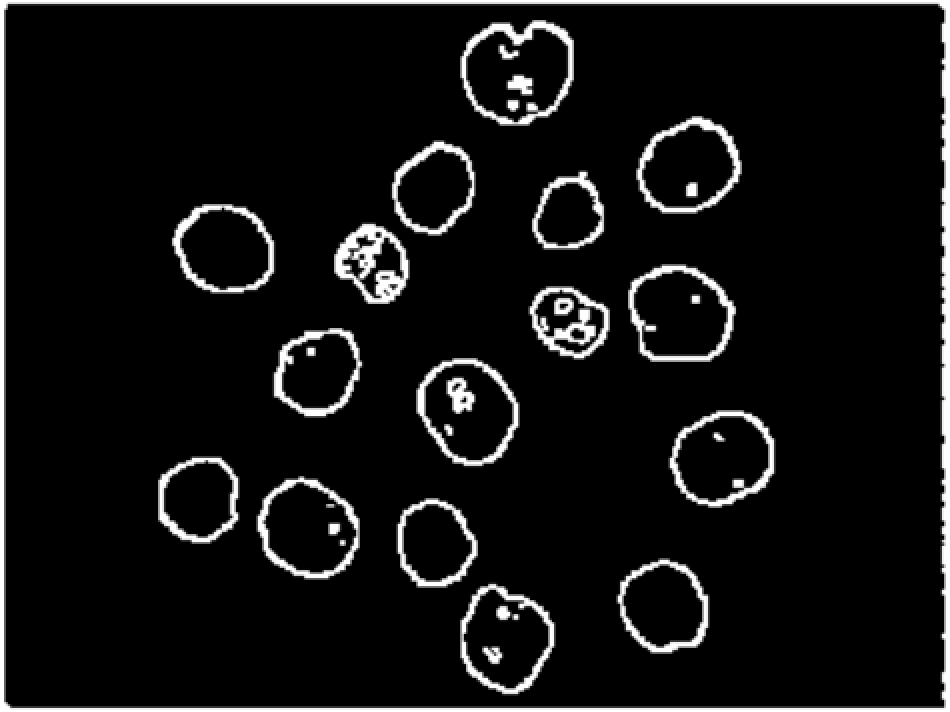
Convert RGB to Grayscale

```
%% Convert Truecolor (RGB) Image into Grayscale Image
I = im2gray(RGB);
% Use Median Filter
I = medfilt2(I,[30 30]);
figure(2);
clf
imshow(I)
```

Calculate the Gradient Image and Apply a Threshold

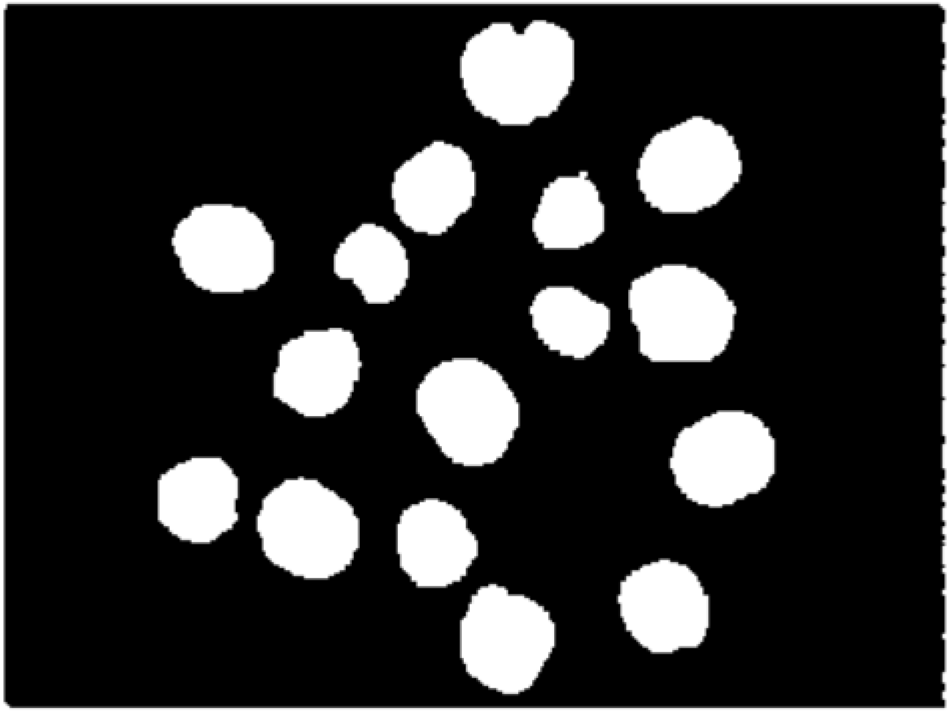
```
% Use edge and Sobel operator to calculate the threshold value. Then  Tune the threshold value and
use edge again to obtain a binary mask that contains the segmented cell
[~,threshold] = edge(I,'sobel');
fudgeFactor = 1;
BWs = edge(I,'sobel',threshold * fudgeFactor);
% Display resulting binary gradient mask
figure(3);
clf
imshow(BWs)
```

This is used to identify the scars.



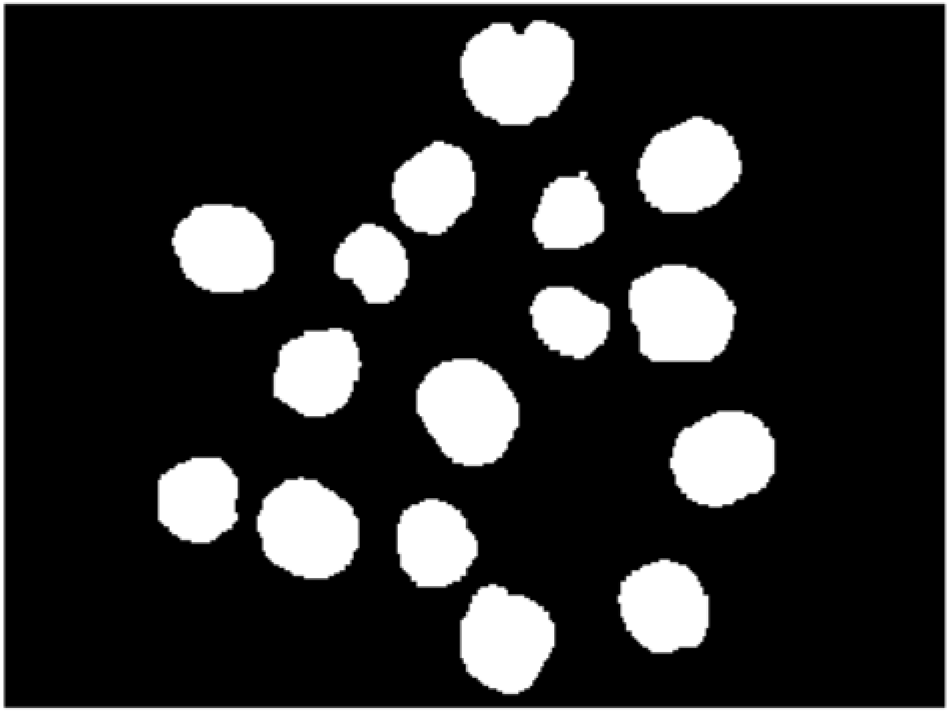
Dilate the Image.

```
% Create two perpendicular linear structuring elements by using strel function.
se90 = strel('line',3,90);
se0 = strel('line',3,0);
% Dilate the binary gradient mask using the vertical structuring element followed by the
horizontal structuring element. The imdilate function dilates the image.
BWsdil = imdilate(BWs,ones(20,20));
imshow(BWsdil)
% Fill Interior Gaps
% Fill remaining holes using the imfill function
BWdfill = imfill(BWsdil,'holes');
figure(4);
clf
imshow(BWdfill)
```



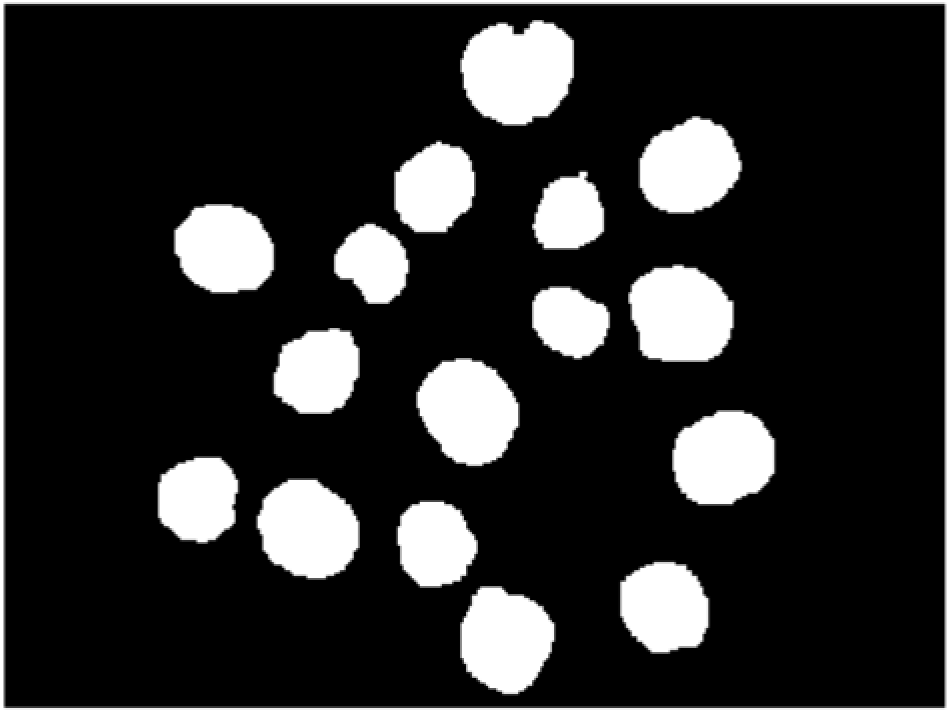
Remove Connected Objects

```
% Remove any objects that are connected to the border of the image using
% imclearborder function. We use 4 as connectivity to remove 2D diagonal
% connections
BWnobord = imclearborder(BWdfill,4);
figure(5);
clf
imshow(BWnobord)
```



Smooth the Object

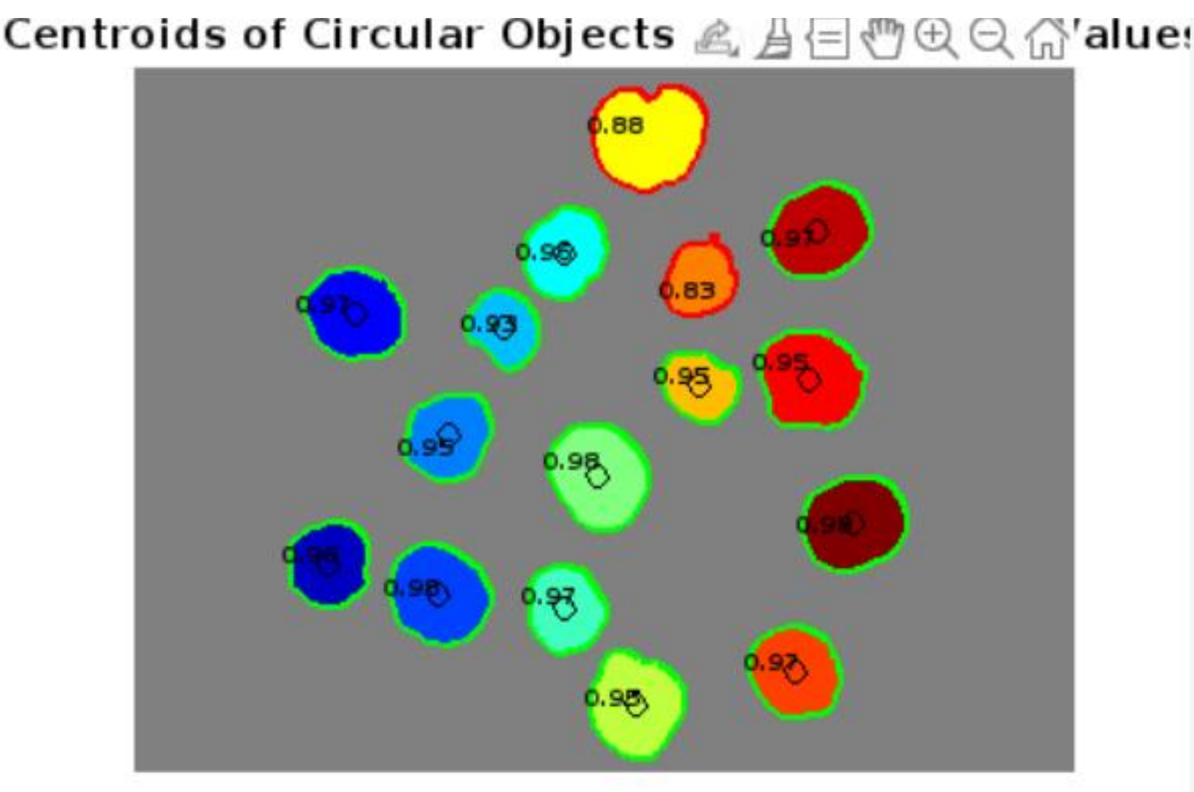
```
% We create the diamond structuring element using the strel function in
% order to make the object look natual/smooth
seD = strel('diamond',1);
BWfinal = imerode(BWnobord,seD);
BWfinal = imerode(BWfinal,seD);
figure(6);
clf
imshow(BWfinal)
```



Visualize the Segmentation

```
% Labelloverlay function allows us to display the mask over the original
% image
figure(7);
clf
imshow(labeloverlay(I,BWfinal))
stats = regionprops('table',BWfinal, 'Area','EquivDiameter','Perimeter');
% storing the values in other variables
x = stats.Area;
%using area to find the diameter
r = sqrt(x/(4*pi))*0.2645;
[B,L] = bwboundaries(BWfinal,"noholes");

imshow(label2rgb(L,@jet,[.5 .5 .5]))
hold on
stats = regionprops(L,"Circularity","Centroid");
threshold = 0.90;
for k = 1:length(B)
  boundary = B{k};
  circ_value = stats(k).Circularity;
  circ_string = sprintf("%2.2f",circ_value);
  if circ_value > threshold
    centroid = stats(k).Centroid;
    plot(boundary(:,2),boundary(:,1),"g",LineWidth=2)
  else
       plot(boundary(:,2),boundary(:,1),"r",LineWidth=2)
  end
end
title("Objects with Boundaries in White")
```

```
for k = 1:length(B)

  % Obtain (X,Y) boundary coordinates corresponding to label "k"
  boundary = B{k};

  % Obtain the circularity corresponding to label "k"
  circ_value = stats(k).Circularity;

  % Display the results
  circ_string = sprintf("%2.2f",circ_value);

  % Mark objects above the threshold with a black circle
  if circ_value > threshold
    centroid = stats(k).Centroid;
    plot(centroid(1),centroid(2),"ko");

  end

  text(boundary(1,2)-35,boundary(1,1)+13,circ_string,Color="black",...
       FontSize=6,FontWeight="bold")

end
title("Centroids of Circular Objects and Circularity Values")
```

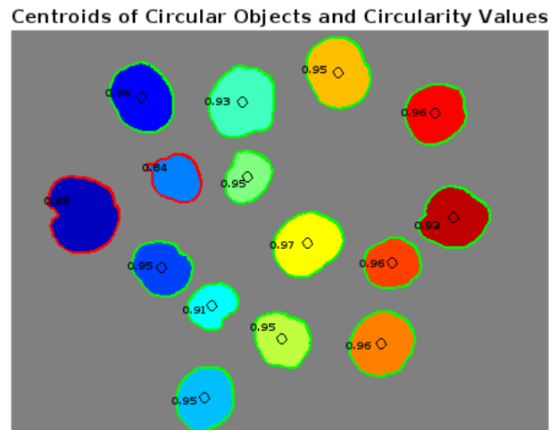


	I	
1	23.5528	
2	26.6624	
3	27.9713	
4	24.2117	
5	20.2948	
6	23.8807	
7	23.1275	
8	28.4218	
9	27.6517	
0	30.5680	
1	20.4927	
2	20.4113	
3	25.5335	
4	28.6629	
5	27.2472	
6	27.9457	
7		

RAIDUS OF VARIOUS MAKHANAS

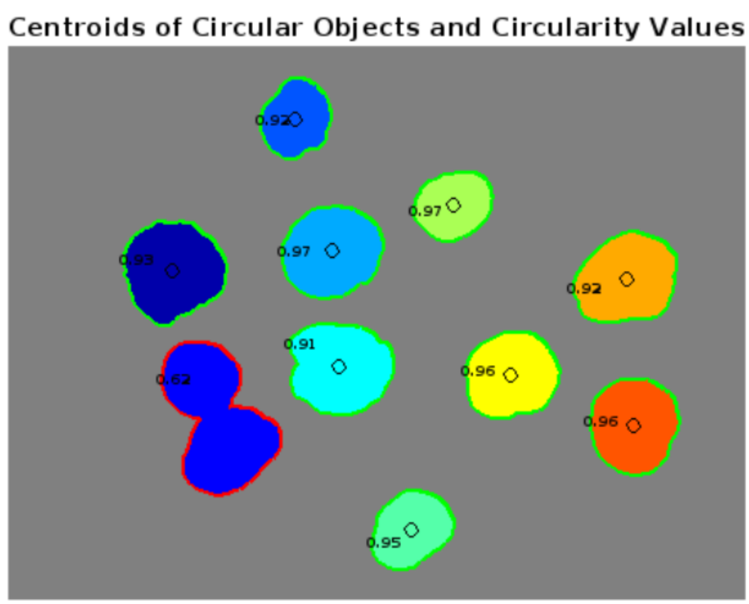
DIFFERENT IMAGES

1.



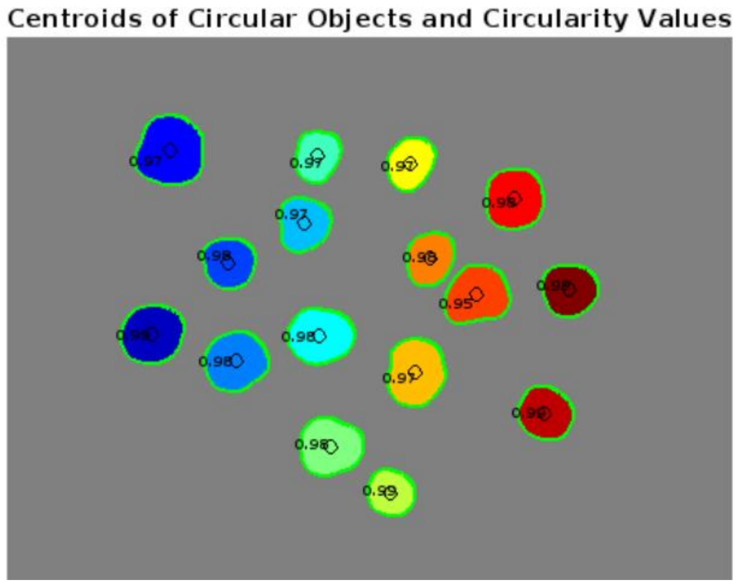
OUTPUT: - The 2 makhana are rejected

2.



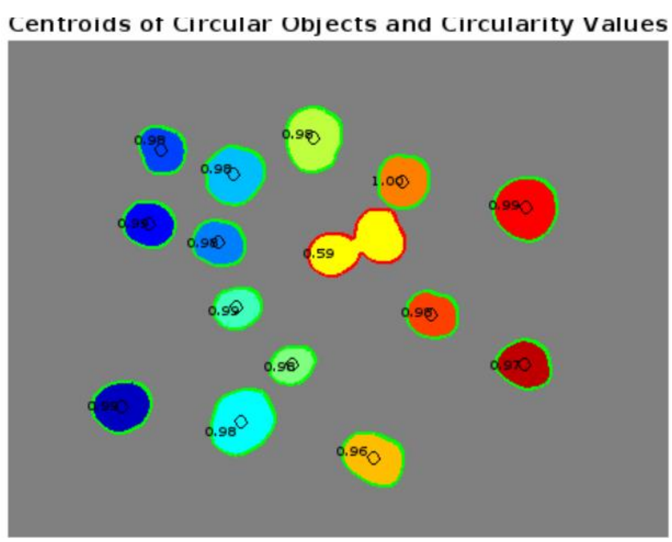
OUTPUT: - This is the fail case where the 2 Makhana are connected in Im

3.



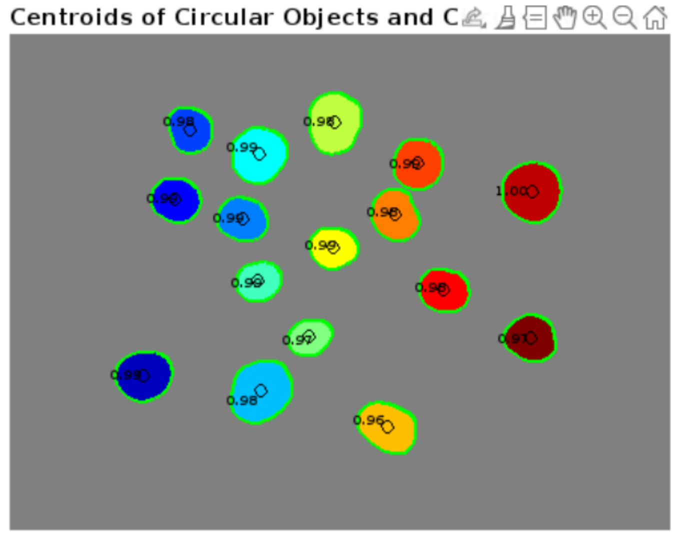
OUTPUT: Here all the Makhana passes the test.

4.



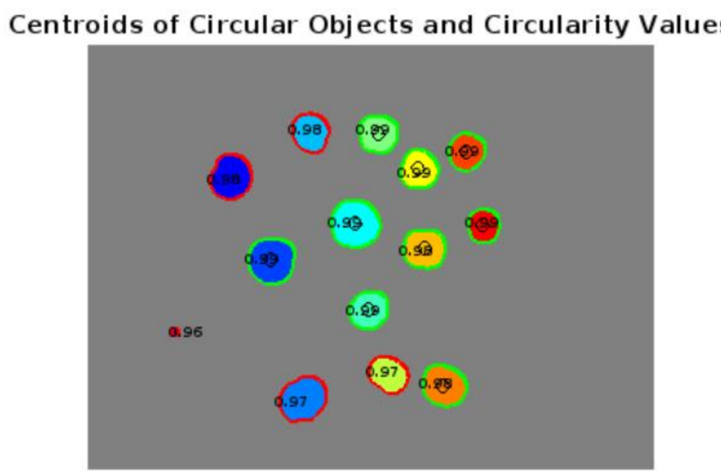
OUTPUT: - Here the the two makhana are connected

5.



OUTPUT:
Here all makhana are gives Pass.

6.



OUTPUT:- Here we have taken threshold as 98 and 4 Makhana are given red boundry

CONCLUSION:

We have applied various image processing techniques and found the radius(Size), Circularity and Spots in the Makhana.

The Makhana that passes the threshold are given green boundary while the Makhana that fail the test are identified with red boundary.