CSE3018 : Content Based Image and Video Retrieval

# Lab - 9

Naga Harshith Bezawada
19BCE1547

**IMAGE 1**

**CODE :**

```
clc
clear all
close all
img = imread('TPTest1.png');
imshow(img);

img = imbinarize(im2gray(img), graythresh(img));

[B, L] = bwboundaries(img);
figure;
imshow(img);
hold on;
for k=1:length(B)
boundary = B{k};
plot(boundary(:,2), boundary(:,1), 'g', 'LineWidth', 2);
end

[L, N] = bwlabel(img);
d = sprintf("N Value returned by bwlable = %d",N);
disp(d);
RGB = label2rgb(L, 'hsv', [.5 .5 .5], 'shuffle');
figure;
imshow(RGB);
hold on;
for k=1:length(B)
boundary = B{k};
plot(boundary(:,2), boundary(:,1), 'w', 'LineWidth', 2);
```

```
text(boundary(1,2)-11, boundary(1,1)+11, num2str(k), 'Color', 'y',
...
'FontSize', 14, 'FontWeight', 'bold');
end


stats = regionprops(L, 'all');
temp = zeros(1, N);
for k=1:N

temp(k) = 4*pi*stats(k,1).Area / (stats(k,1).Perimeter)^2;
stats(k,1).ThinnessRatio = temp(k);


temp(k) = (stats(k, 1).BoundingBox(3)) / (stats(k,1).BoundingBox(4));
stats(k,1).AspectRatio = temp(k);
end



areas = zeros(1,N);
for k=1:N
areas(k) = stats(k).Area;
end
TR = zeros(1,N);
for k=1:N
TR(k) = stats(k).ThinnessRatio;
end
figure();
hold on;
cmap = colormap(lines(16));
for k=1:N
scatter(areas(k), TR(k), [], cmap(k,:), 'filled'), ylabel('Thinness
Ratio'), xlabel('Area');
hold on;
end

for i=1:4
     d = sprintf("SHAPE - %d",i);
```

```
      disp(d);
      d = sprintf(" Area %d : %f ",i,stats(i).Area);
      disp(d);
      d = sprintf(" Centroid %d : (%f,%f)
",i,stats(i).Centroid(1),stats(i).Centroid(2));
      disp(d);
      d = sprintf(" Orientation %d : %f ",i,stats(i).Orientation);
      disp(d);
      d = sprintf(" Euler Number %d : %f ",i,stats(i).EulerNumber);
      disp(d);
      d = sprintf(" Eccentricity %d : %f ",i,stats(i).Eccentricity);
      disp(d);
      d = sprintf(" Aspect Ratio %d : %f ",i,stats(i).AspectRatio);
      disp(d);
      d = sprintf(" Perimenter %d : %f ",i,stats(i).Perimeter);
      disp(d);
      d = sprintf(" Thinness Ratio %d : %f
",i,stats(i).ThinnessRatio);
      disp(d);
end
```

**OUTPUT :**

**Figure 1 - Image**

**Figure 2 - bwboundaries**

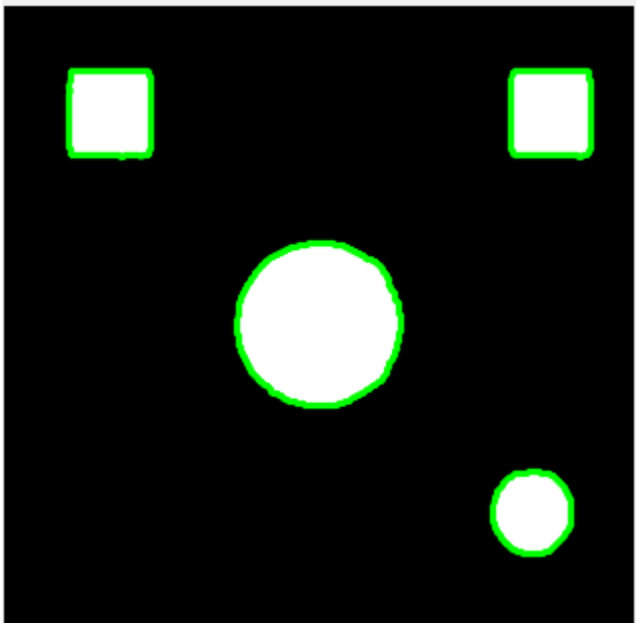


**Figure 3 - label2rgb**

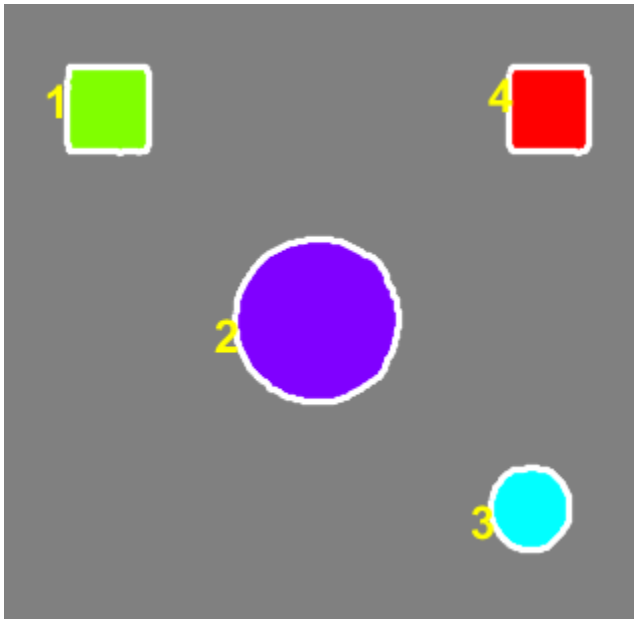
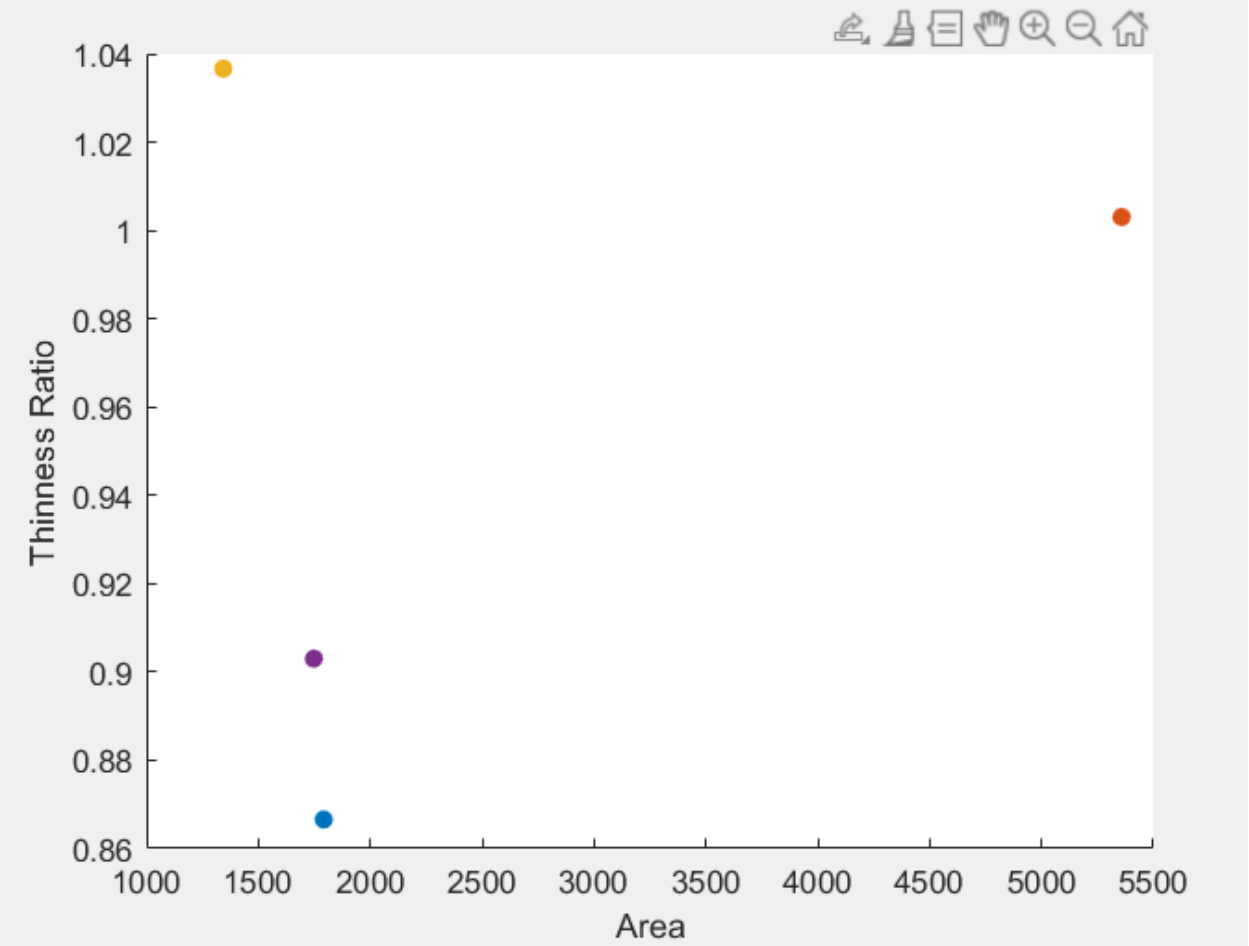


**Figure 4 - Graph area vs thinness ratio**



**TABLE**

| Object | Area | Centroid | Orientation | Euler Number | Eccentricity | Aspect Ratio | Perimeter | Thinness Ratio |
|---|---|---|---|---|---|---|---|---|
| **Top Left Square** | 1792.0 | (53.57, 54.06) | 88.8323 | 1.00 | 0.233 | 0.9545 | 161.20 | 0.866 |
| **Big Circle** | 5358.00 | (158.09, 159.52) | 20.106 | 1.00 | 0.101 | 1.012 | 259.088 | 1.003 |
| **Small Circle** | 1343.00 | (264.63, 253.42 | 88.05 | 1.00 | 0.299 | 0.952 | 127.59 | 1.036 |
| **Top Right Square** | 1748.00 | (274.00, 54.03) | -87.866 | 1.00 | 0.308 | 0.931 | 155.965 | 0.903 |

## IMAGE 2

## CODE :

```
clc
clear all
close all
img = imread('TPTest2.png');
imshow(img);

img = imbinarize(im2gray(img), graythresh(img));

[B, L] = bwboundaries(img);
figure;
imshow(img);
hold on;
for k=1:length(B)
boundary = B{k};
plot(boundary(:,2), boundary(:,1), 'g', 'LineWidth', 2);
end
```

```
[L, N] = bwlabel(img);
d = sprintf("N Value returned by bwlable = %d",N);
disp(d);
RGB = label2rgb(L, 'hsv', [.5 .5 .5], 'shuffle');
figure;
imshow(RGB);
hold on;
for k=1:length(B)
boundary = B{k};
plot(boundary(:,2), boundary(:,1), 'w', 'LineWidth', 2);
text(boundary(1,2)-11, boundary(1,1)+11, num2str(k), 'Color', 'y',
...
'FontSize', 14, 'FontWeight', 'bold');
end


stats = regionprops(L, 'all');
temp = zeros(1, N);
for k=1:N

temp(k) = 4*pi*stats(k,1).Area / (stats(k,1).Perimeter)^2;
stats(k,1).ThinnessRatio = temp(k);


temp(k) = (stats(k, 1).BoundingBox(3)) / (stats(k,1).BoundingBox(4));
stats(k,1).AspectRatio = temp(k);
end



areas = zeros(1,N);
for k=1:N
areas(k) = stats(k).Area;
end
TR = zeros(1,N);
for k=1:N
TR(k) = stats(k).ThinnessRatio;
end
```

```
figure();
hold on;
cmap = colormap(lines(16));
for k=1:N
scatter(areas(k), TR(k), [], cmap(k,:), 'filled'), ylabel('Thinness
Ratio'), xlabel('Area');
hold on;
end

for i=1:10
     d = sprintf("SHAPE - %d",i);
     disp(d);
     d = sprintf(" Area %d : %f ",i,stats(i).Area);
     disp(d);
     d = sprintf(" Centroid %d : (%f,%f)
",i,stats(i).Centroid(1),stats(i).Centroid(2));
     disp(d);
     d = sprintf(" Orientation %d : %f ",i,stats(i).Orientation);
     disp(d);
     d = sprintf(" Euler Number %d : %f ",i,stats(i).EulerNumber);
     disp(d);
     d = sprintf(" Eccentricity %d : %f ",i,stats(i).Eccentricity);
     disp(d);
     d = sprintf(" Asoect Ratio %d : %f ",i,stats(i).AspectRatio);
     disp(d);
     d = sprintf(" Perimenter %d : %f ",i,stats(i).Perimeter);
     disp(d);
     d = sprintf(" Thinness Ratio %d : %f
",i,stats(i).ThinnessRatio);
     disp(d);
end
```

**OUTPUT :**

**Figure 1 - Image**

**Figure 2 - bwboundaries**

**Figure 3 - label2rgb**

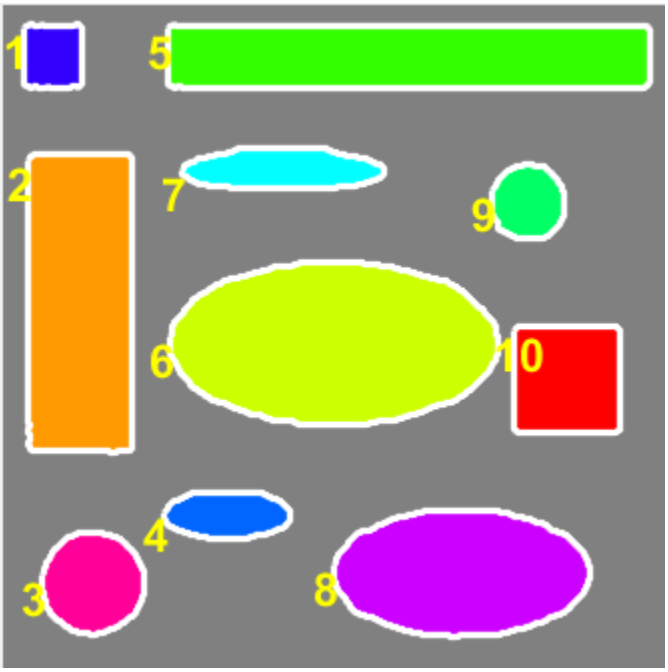
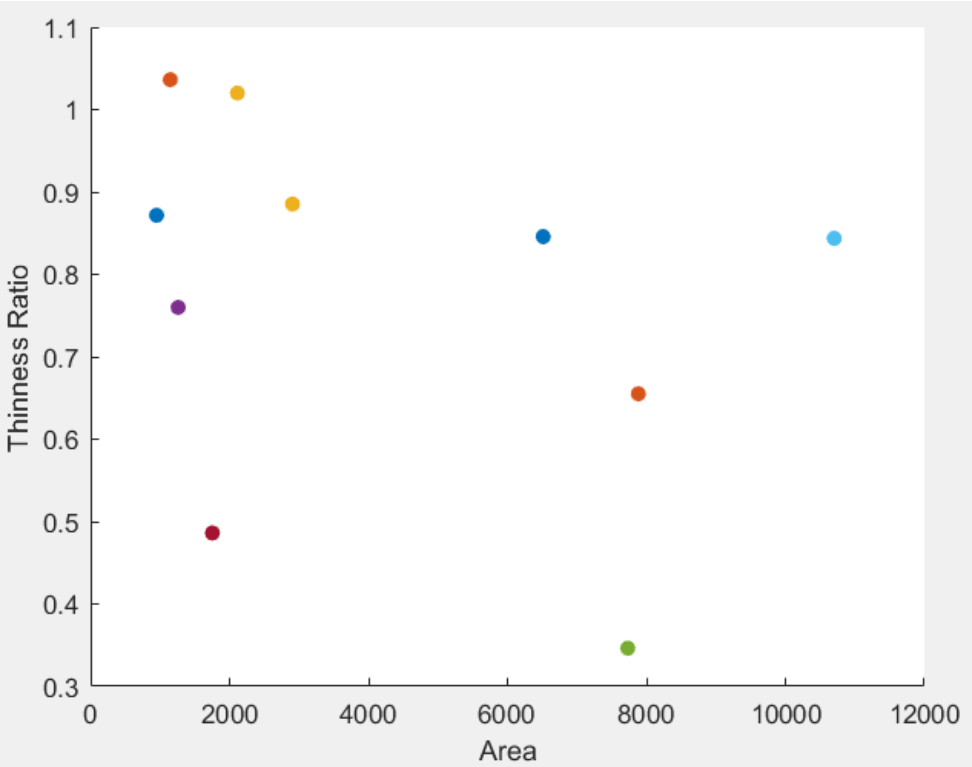



**Figure 4 - Graph area vs thinness ratio**




**TABLE**

| Object | Area | Centroid | Orientation | Euler Number | Eccentricity | Aspect Ratio | Perimeter | Thinness Ratio |
|---|---|---|---|---|---|---|---|---|
| **1** | 950.00 | (25.49, 26.23) | -89.5 | 1.00 | 0.34 | 0.90 | 117.02 | 0.86 |
| **2** | 7883.00 | (39.04, 149.00) | -89 | 1.00 | 0.93 | 0.35 | 388.36 | 0.6 |
| **3** | 2113.00 | ( 45.51, 289.10) | 20.87 | 1.00 | 0.15 | 1.00 | 161.33 | 1.02 |
| **4** | 1260.00 | (112.66, 255.18) | -0.08 | 1.00 | 0.92 | 2.66 | 144.34 | 0.75 |
| **5** | 7731.00 | (203.48, 26.48) | -0.008 | 1.00 | 0.99 | 7.33 | 529.68 | 0.34 |
| **6** | 10702.0 | (135.96, 169.52) | -0.017 | 1.00 | 0.86 | 2.01 | 399.24 | 0.84 |
| **7** | 1753.00 | (140.29, | -0.054 | 1.00 | 0.97 | 4.85 | 212.86 | 0.48 |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | | 82.56) | | | | | | |
| **8** | 6514.00 | (229.65, 284.11) | 0.08 | 1.00 | 0.86 | 2.01 | 311.09 | 0.84 |
| **9** | 1147.00 | (263.10, 99.09) | -85.01 | 1.00 | 0.19 | 0.97 | 117.93 | 1.03 |
| **10** | 2905.00 | (282.50, 187.50) | -43.89 | 1.00 | 0.043 | 1.00 | 203.06 | 0.88 |

**QUESTIONS**

1. **What is the value of N returned by bwlabel? Does it make sense to you?**

( for image 1 ) `N Value returned by bwlable = 4`

( for image 2 ) `N Value returned by bwlable = 10`

As we can see in the command window the value of N returned is 4, this is because our image 1 has 4 distinct shapes thus it is correct.

For image 2, we can see 10 distinct shapes and the N value returned is 10 thus it is correct as well.

2. **Do the results obtained for the extracted features correspond to your expectations? Explain.**

Area represents the number of pixels inside the region. Centroid represents the center of the image. The eccentricity is the ratio of the distance between the foci of the ellipse and its major axis length. Eccentricity value of 0 means the region is actually a circle, while a value

of 1 means it is a line segment. Euler Number is the number of objects in the region minus the number of holes in those objects, thus it is 1.00 for all closed images. Orientation is the angle at which the image is present, a negative value is an angle generated in the opposite direction. Perimeter is the distance of the boundary  region of the image. Thinness ratio is the measure of roundness of the binary object, regular objects have a higher thinness ratio than irregular objects. The aspect ratio of an image is the ratio of its width to its height.

3. **Which of the extracted features have the best discriminative power to help tell squares from circles? Explain.**

The Thinness Ratio describes the relation between a polygon's perimeter to its area using geometric attributes of a circle as a basis for comparison. It can be used to differentiate between squares and circles. When the object is a circle the thinness ratio is always 1

4. **Which of the extracted features have the worst discriminative power to help tell squares from circles? Explain.**

Area, Centroid, Orientation and perimeter are the worst descriptors to help tell squares from circles as all these features can change based on size or how the object is positioned in the image. They have nothing to do with the shape of the object and thus cannot be used to distinguish between a circle and a square.

5. **Which of the extracted features are ST invariant, that is, robust to changes in size and translation? Explain.**

Thinness Ratio and Eccentricity are ST Invariant. Thinness Ratio  describes the relation between a polygon's perimeter to its area using geometric attributes of a circle as a basis for comparison, so as area decreases even perimeter decreases thus the ratio remains the same .  The eccentricity is the ratio of the distance between the foci of the ellipse and its major axis length.

6. **If you had to use only one feature to distinguish squares from circles, in a ST-invariant way, which feature would you use? Why?**

Thinness feature is the best way to distinguish squares from circles as it is the relation between a polygon's perimeter to its area using geometric attributes of a circle as a basis for comparison. Its value is 1 for circles and less than 1 for other polygons.