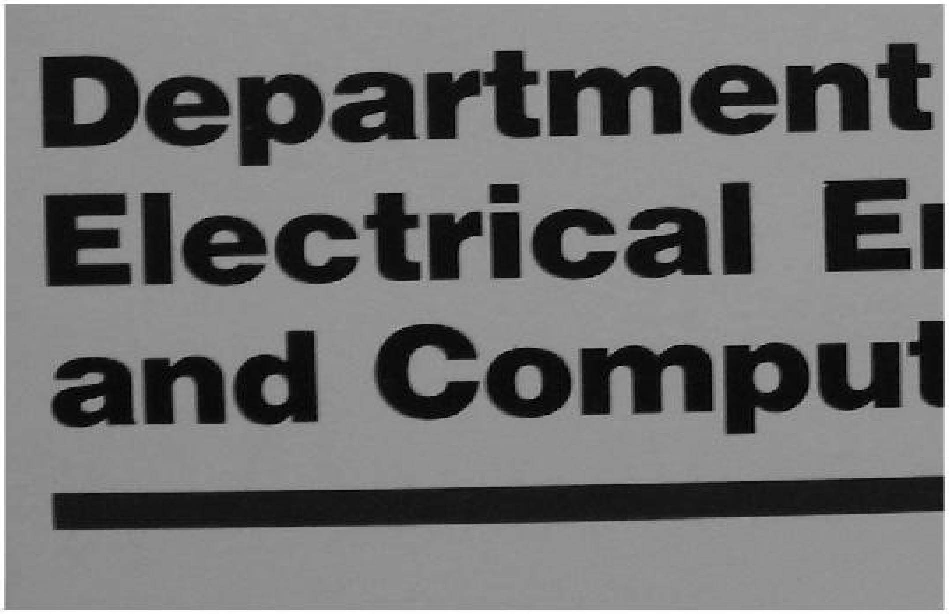
Purvang Lapsiwala

UIN 662689378

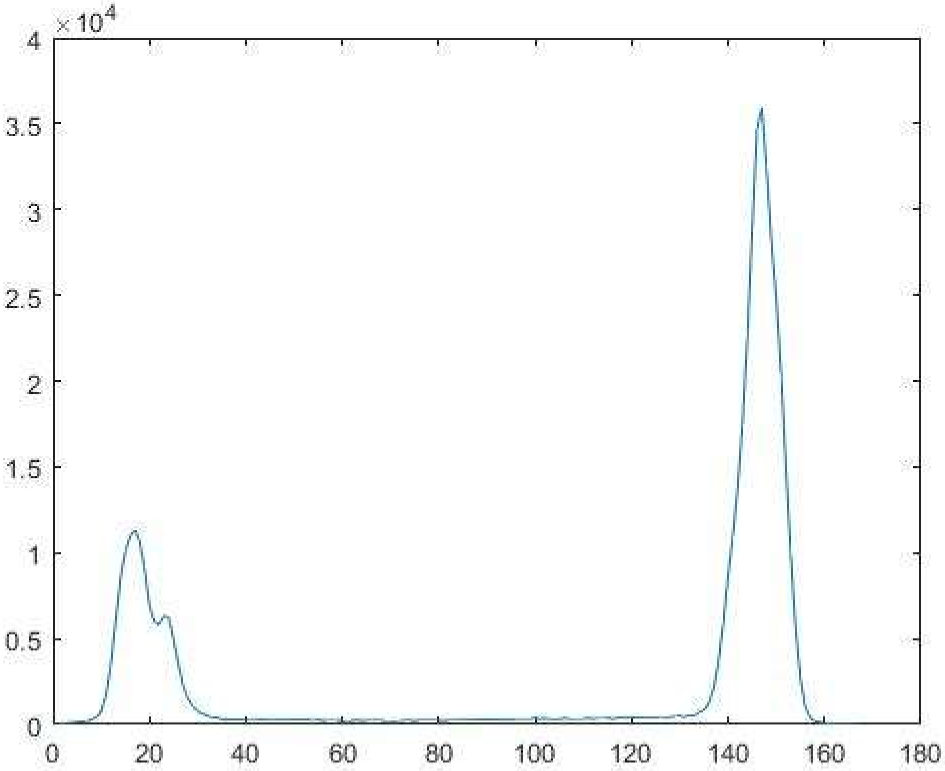
HOMEWORK- 3

Answer - 1

(1) Image Display



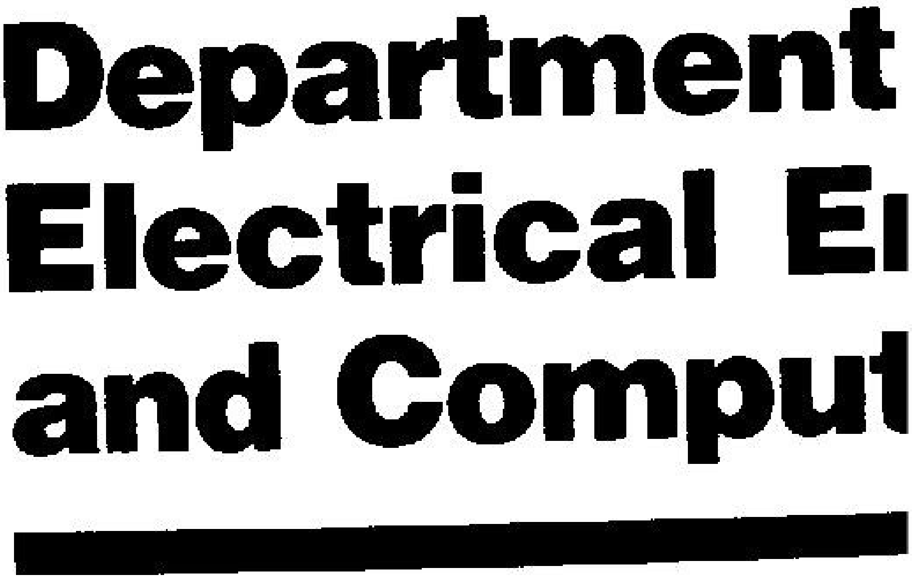
(2) Histogram of Image Pixel Intensities



Histogram basically shows all pixel intensities of given image in form of plot. It gives information about number of pixels having particular intensity value.

(3) Thresholding:

Thresholding is done to convert image from gray scale to 2 color image. Nonlinear median filter is used as threshold criteria. The mean value among all pixel intensities is taken as the threshold criteria of the image. Any pixel that is less than this value is taken as a 0 and any value greater than the threshold is taken as 1. This gives us black and white image also known as Thresholding operation.



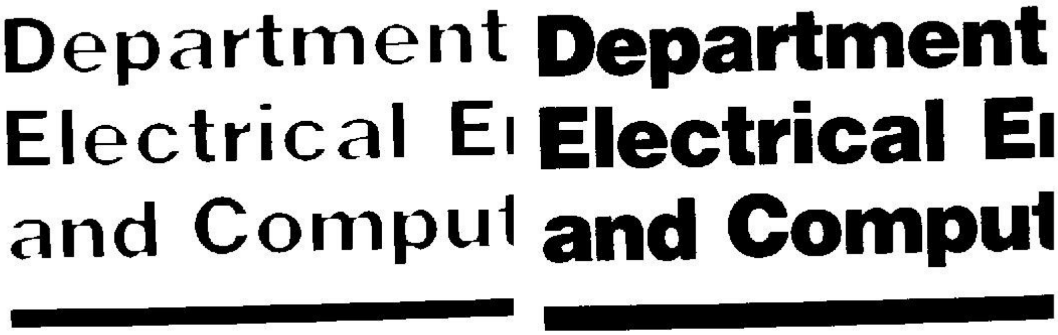
(4) Erosion:

Erosion is a type of morphological operator and it is used to erode away the boundary regions between the foreground and the background. By performing erosion operation, object becomes thinner.

By convolving image with the structuring element and checking condition for erosion which sum of convolution should be greater than or equal to size of kernel, we will get eroded image if value obtained is greater than or equal to size of kernel.

Kernel Size = 3x3

Here we have Eroded image on the left side and threshold image on the right side.



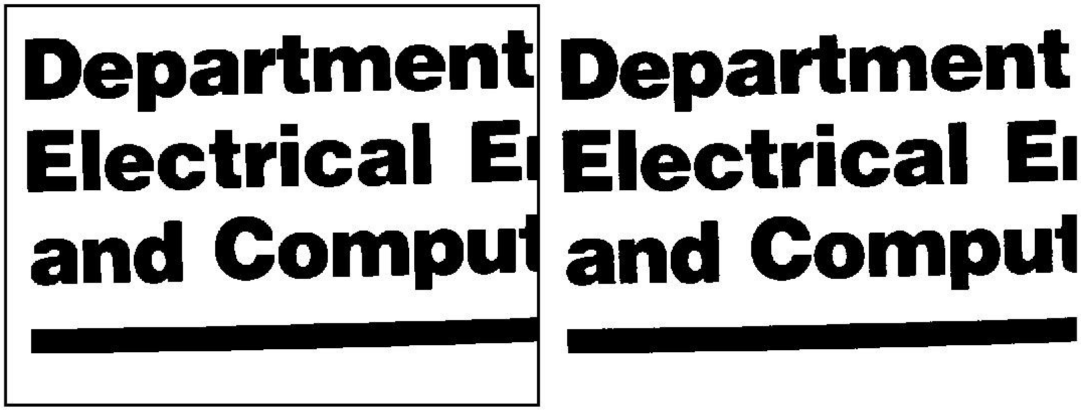
(5) Dilation

Dilation operator is also morphological operator used to enlarge the boundaries of the image.

Same procedure as erosion but if value obtained after convolution is more than 1, than it is considered as 1.In our experiment, we had to dilate an eroded image.

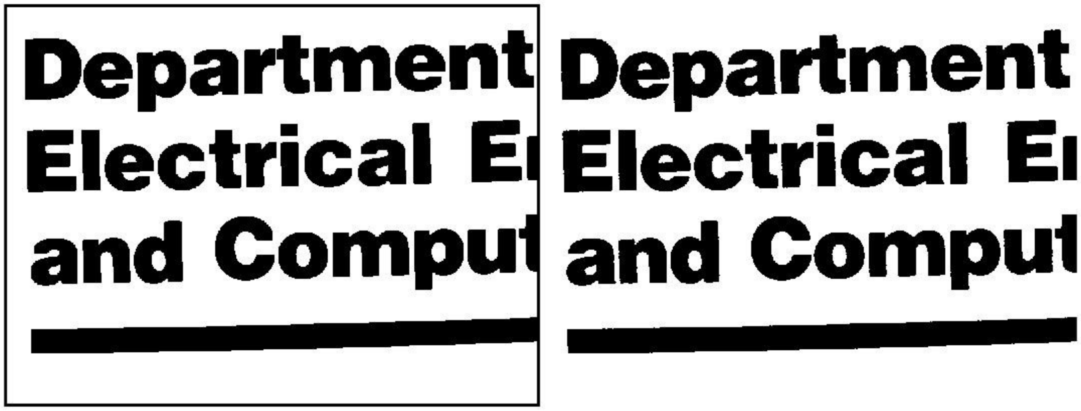
Kernel Size = 3x3

Here we have dilated image on the left side and threshold image on the right side.



(6) No, There is difference between image obtained in 5 and image obtained in 3.

The output of the eroded image (left) have much thicker objects as compared to the threshold image. Also noise is added to image.

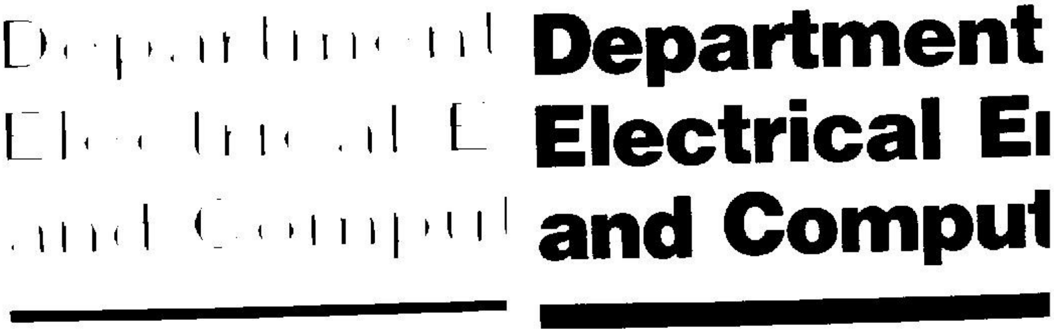


(7)

First transformation of image into double format is done for both images. Then by taking absolute difference between pixels of images. After normalizing image, convert it back to uint8, we get difference. We get thicker image from erosion result, we get white pixels on edge only where we have effect of dilation as over there pixel intensities are 1 but original pixels have corresponding value 0.

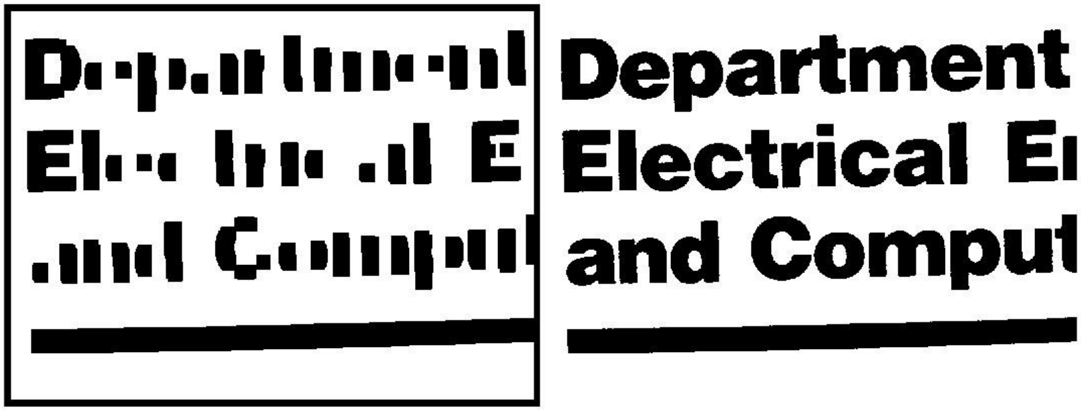
(8) Erosion 10 Times:

By applying 10 times erosion operation on to image, objects become thinner as compared to previous result where we applied for 3 times. And object lost almost all information.



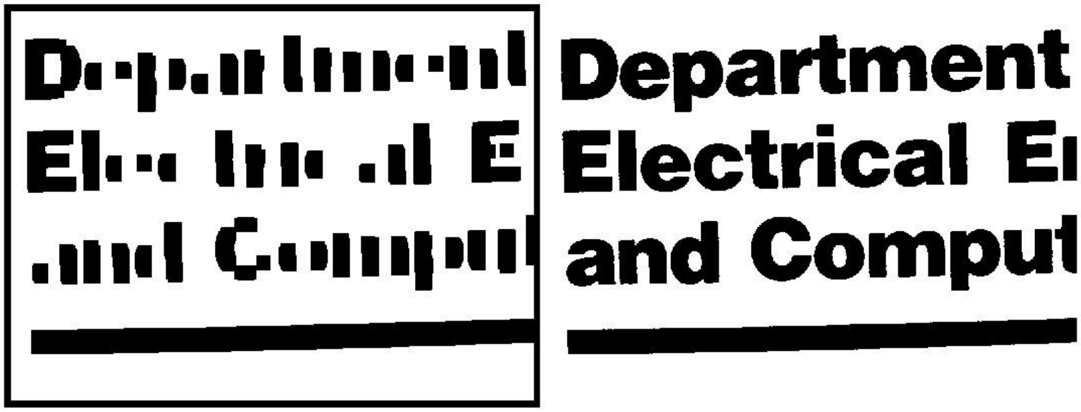
(9) Dilation 10 Times:

By applying 10 times Dilation operation on to image, objects become more thinner as compared to previous result where we applied for 3 times.



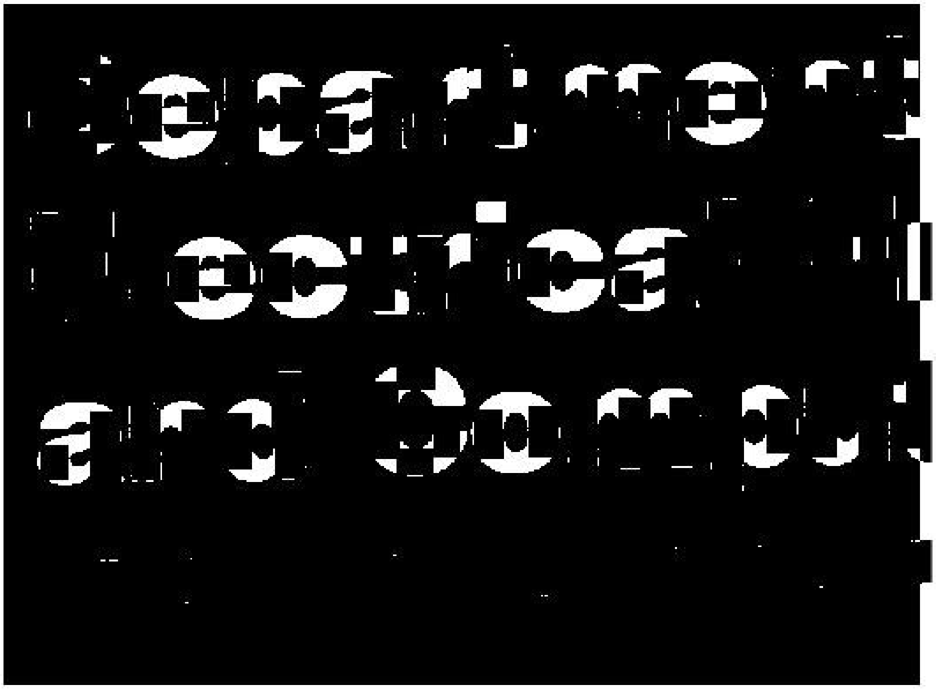
(10) We see that we could not retrieve the original image after erosion, objects get thicker and they start merging and therefore we lost image information. Also the added noise is too large and it hampers the process of getting back towards the original image.

So image obtained from 9 and 3 are not same.



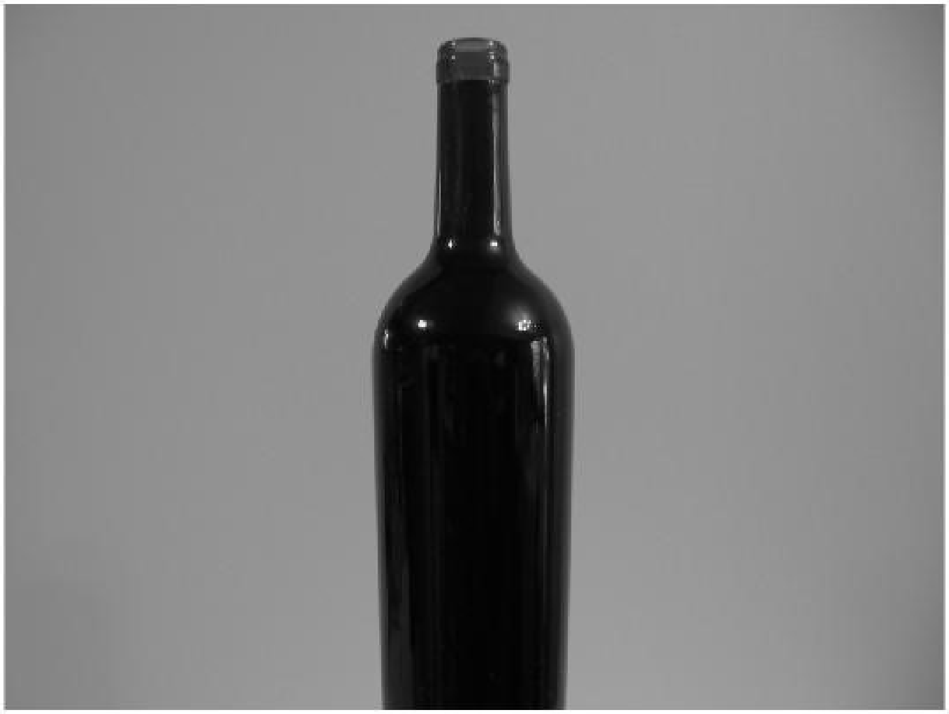
(11)

If we take the difference between dilated image which is done for 10 times and threshold image, and as we get thicker image from erosion result, we get white pixels on edge only where we have effect of dilation as over there, pixel intensities are 1 but original pixels have corresponding value 0. If we compare with result 6, it is clearly seen that difference is more and we get more white pixels as objects become thicker.



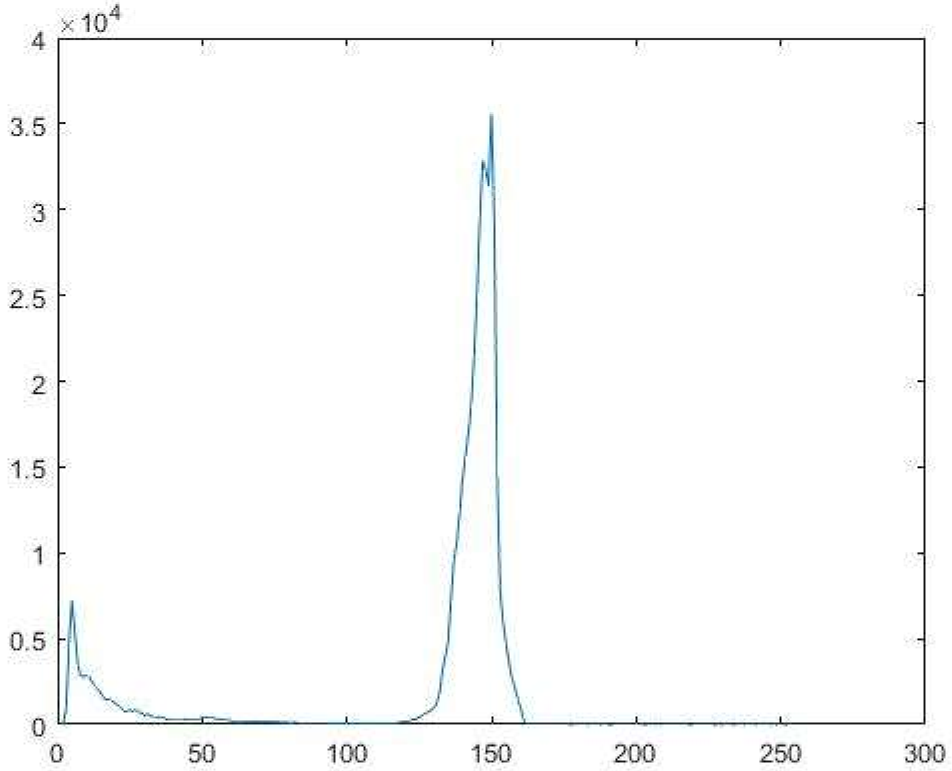
Answer – 2

Image Display



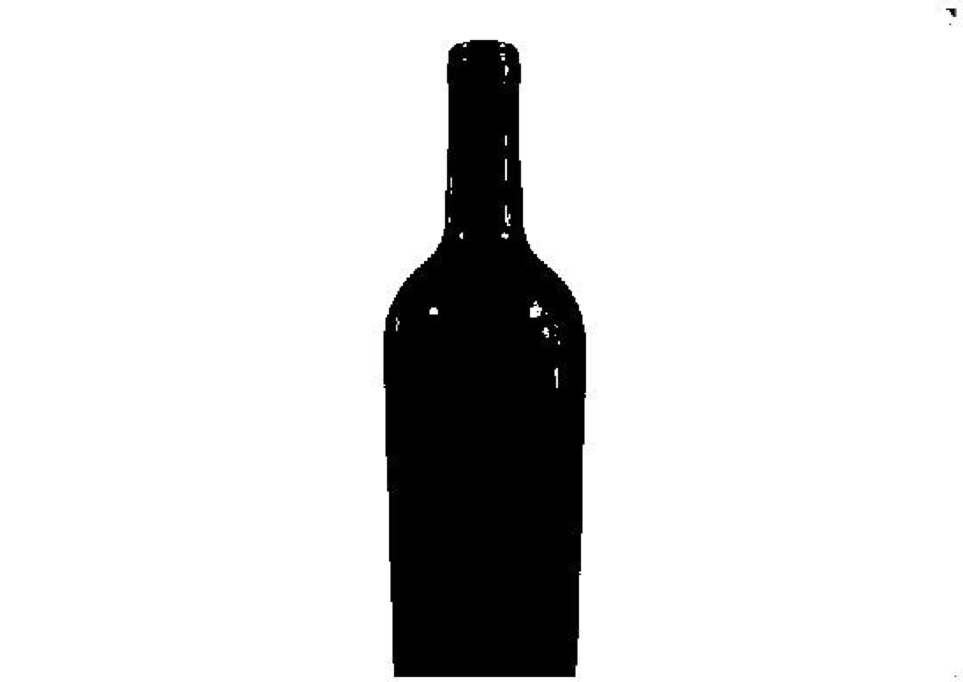
Histogram:

Histogram of an image is basically the graph of all the pixel intensities of the image. It tells us the number of times a particular intensity occurs in the image.



Threshoding:

Thresholding The mean of all the values of the pixels of the image is taken as the threshold of the image. Any pixel that is less than this value is taken as a 0 and any value greater than the threshold is taken as 1. This gives Threshold value is selected as the mean of all the intensities present in the input image. At the end we get a black and white image.



4. Removal of the holes from the black and white image

To remove the holes from the black and white image, we use a technique called morphological closing. The morphological closing is used to remove the holes from objects without changing shape and size of the object. To implement this technique, we need to dilate the image first and then erode it. The dilation first, thickens and enlightened the boundaries and the erosion, shrinks the background color holes. So, when we apply dilation and erosion once, we see that after the dilation that the holes inside have shrunk a little and when we erode it the holes are removed. By applying dilation thrice, the holes get closed. When we erode it, we get the boundaries of the image as they were in the original but couldn't revive the holes as the there were no 1s in the image to help the process.

(5) Distance transform:

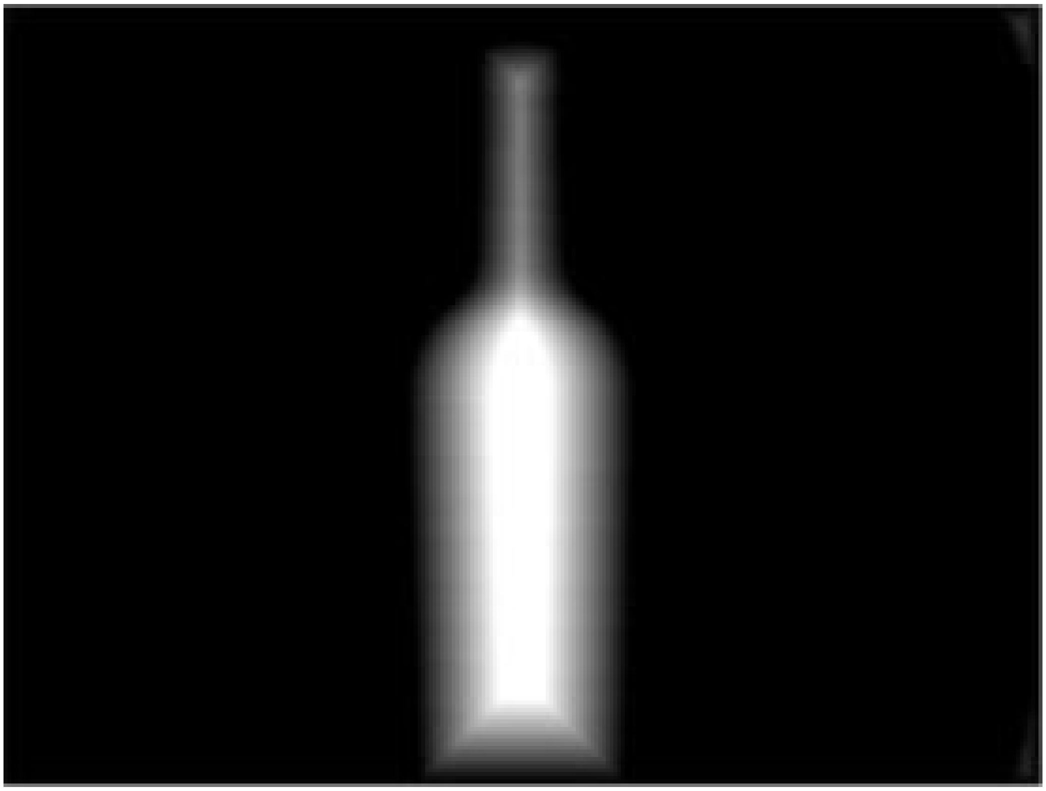
To perform distance transform operation, we have to perform two pass algorithm, forward pass followed by backward pass.

Max value of this distance transform is 88

58 pixels have this maximum value

(6)

Mapping of the distance transform to the full dynamic range of 8 bits we get the following image. The distance transform gives us a gray level output image. In this image the pixel intensities are such that they give the corresponding distance from the background of the image.



(7)

Area = 79692 pixels

Perimeter = 4191 pixels

Centroid = (352.22, 393.05)

To find out the area, counted the total number of pixels whose value was greater than or equal to 1 in the distance transformed image. I found the perimeter by counting the total number of 1s in the image I get after doing the forward and backward transform. I did this because, when we calculate the distance transforms only the outer layer of the image has al 1s in it. The rest either have zeros or a higher value. I found out the centroid by adding all the indexes of the pixels having a pixel intensity of more than one and then dividing it by 600 for x co-ordinate and 800 for y co-ordinate.

MATLAB CODE:

ANSWER (1)

```
clc;
clear all;
original_image = imread('text.bmp');
figure(1)
imshow(original_image);
%%
max_intensity = max(max(original_image));
histo = zeros(1,max_intensity+1);

%Histogram creation
for i = 1:600
    for j = 1:800
        temp = original_image(i,j);
        histo(1,temp+1) = histo(1,temp+1) + 1;

    end
end
figure(2)
plot(histo);
%%
%Thresholding
meanIntensity = mean(mean(original_image));
for i = 1:600
    for j = 1:800
        if original_image(i,j) >= meanIntensity
            tempImage(i,j) = 1;
        else
            tempImage(i,j) = 0;
        end
    end
end
figure(3)
imshow(tempImage)

%% Erosion

original_image = tempImage;
structuring_element = [0 0 0;0 0 0;0 0 0];
temp = original_image;
surrounding_size = 3;
for z =1:5
    temp = padarray(temp,[1,1],1);
    for i = 2:601
        for j = 2:801
            m = 1;
            for k = i-1:i+1
                n= 1;
                for l = j-1:j+1
                    surrounding(m,n) = temp(k,l);
                    n = n+1;
                end
                m = m+1;
            end
            if(surrounding == structuring_element)
                x(i-1,j-1) = 0;
            else
```

```
                x(i-1,j-1) = 1;
            end
        end
    end
    temp = uint8(x);
end
figure(4)
imshowpair(temp,tempImage,'montage')

%% Dilation

structuring_element = [1 1 1;1 1 1;1 1 1];

for z =1:5
    temp = padarray(temp,[1,1],0);
    for i = 2:601
        for j = 2:801
            m = 1;
            for k = i-1:i+1
                n= 1;
                for l = j-1:j+1
                    surrounding(m,n) = temp(k,l);
                    n = n+1;
                end
                m = m+1;
            end
            if(surrounding == structuring_element)
                x(i-1,j-1) = 1;
            else
                x(i-1,j-1) = 0;
            end
        end
    end
    temp = x;
end

temp2 = uint8(temp);

figure(5)
imshowpair(temp2,tempImage,'montage')

%%
figure(6)
imshowpair(temp2,tempImage,'montage')

%%

difference = abs(double(temp2) - double(tempImage));
max_value = max(max(difference));
min_value = min(min(difference));
final_image1 = round((difference)/(max_value-min_value))*255;
final_image = uint8(final_image1);
figure(7)
imshow(final_image);
```

```
%% Erosion

original_image = tempImage;
structuring_element = [0 0 0;0 0 0;0 0 0];
temp = original_image;
surrounding_size = 3;
for z =1:10
    temp = padarray(temp,[1,1],1);
    for i = 2:601
        for j = 2:801
            m = 1;
            for k = i-1:i+1
                n= 1;
                for l = j-1:j+1
                    surrounding(m,n) = temp(k,l);
                    n = n+1;
                end
                m = m+1;
            end
            if(surrounding == structuring_element)
                x(i-1,j-1) = 0;
            else
                x(i-1,j-1) = 1;
            end
        end
    end
    temp = uint8(x);
end
figure(8)
imshowpair(temp,tempImage,'montage')

%% Dilation

structuring_element = [1 1 1;1 1 1;1 1 1];

for z =1:10
    temp = padarray(temp,[1,1],0);
    for i = 2:601
        for j = 2:801
            m = 1;
            for k = i-1:i+1
                n= 1;
                for l = j-1:j+1
                    surrounding(m,n) = temp(k,l);
                    n = n+1;
                end
                m = m+1;
            end
            if(surrounding == structuring_element)
                x(i-1,j-1) = 1;
            else
                x(i-1,j-1) = 0;
            end
        end
    end
    temp = uint8(x);
end

temp2 = temp;

figure(9)
imshowpair(temp2,tempImage,'montage')

%%
figure(10)
imshowpair(temp2,tempImage,'montage')

%%

difference = abs(double(temp2) - double(tempImage));
max_value = max(max(difference));
min_value = min(min(difference));
final_image1 = round((difference)/(max_value-min_value))*255;
final_image = uint8(final_image1);
figure(11)
imshow(final_image);
```

MATLAB CODE

ANSWER (2)

```
clc;
clear all;
close all;

original_image = imread('bottle.bmp');
figure(1)
imshow(original_image);

max_intensity = max(max(original_image));
histo = zeros(1,max_intensity+1);

%Histogram creation
for i = 1:600
    for j = 1:800
        temp = original_image(i,j);
        histo(1,temp+1) = histo(1,temp+1) + 1;
    end
end
figure(2)
plot(histo);

%Thresholding

meanIntensity = mean(mean(original_image));
for i = 1:600
```

```
    for j = 1:800
        if original_image(i,j) >= meanIntensity-10
            tempImage(i,j) = 1;
        else
            tempImage(i,j) = 0;
        end
     end
end
figure(3)
imshow(tempImage);

o = original_image;
x = tempImage;
original_image = tempImage;
temp = original_image;
surrounding_size = 3;


structuring_element = [1 1 1;1 1 1;1 1 1];
for z =1:3
temp = padarray(temp,[1,1],1);
    for i = 2:601
    for j = 2:801
        m = 1;
        for k = i-1:i+1
            n= 1;
            for l = j-1:j+1
                surrounding(m,n) = temp(k,l);
                n = n+1;
            end
            m = m+1;
        end
        if(surrounding == structuring_element)
            x(i-1,j-1) = 1;
        else
            x(i-1,j-1) = 0;
        end
    end
    end
    temp = x;
end

figure(4)
imshow(temp)
structuring_element = [0 0 0;0 0 0;0 0 0];

for z =1:3
temp = padarray(temp,[1,1],1);

    for i = 2:601
    for j = 2:801
        m = 1;
        for k = i-1:i+1
            n= 1;
            for l = j-1:j+1
                surrounding(m,n) = temp(k,l);
                n = n+1;
            end
            m = m+1;
        end
        if(surrounding == structuring_element)
            x(i-1,j-1) = 0;
        else
            x(i-1,j-1) = 1;
        end
    end
    end
    temp = x;
end
figure(5)
imshow(temp);


for i =1:600
    for j = 1:800
        if(temp(i,j) ==1)
            temp(i,j) = 0;
        else
            temp(i,j) = 1;
        end
    end
end
figure(6)
imshow(temp);


tempImage = temp;
tempImage = padarray(tempImage,[1,1],0);

%forward pass
forward_temp = tempImage;
backward_temp = tempImage;
for i = 2:601
    for j= 2:801
        if(forward_temp(i,j) >= 1)
            forward_temp(i,j) = min((forward_temp(i,j-1)+1),forward_temp(i-
1,j)+1);
        end
    end
end

for i = 601:-1:2
    for j = 801:-1:2
        if(backward_temp(i,j) >= 1)
            temp_min =min((backward_temp(i,j+1)+1),(backward_temp(i+1,j)+1));
            backward_temp(i,j) = min(temp_min,forward_temp(i,j));
        end
    end
end
```

```
final = uint8(backward_temp);
figure(7)
imshow(final)
final_image = final(2:601,2:801);

max_value=max(max(final_image));
final = round(final_image*(256/max_value+1));
final=uint8(final);
area = 0;
for i = 1:600
    for j = 1:800
        if(final(i,j)>0)
            area = area +1;
        end
    end
end


count_peri = 0;
for i = 2:601
    for j = 2:802
        if(backward_temp(i,j) == 1)
            count_peri  = count_peri+1;
        end
    end
end

x_sum = 0;
y_sum = 0;
tot_count = 0;
for i = 1:600
    for j = 1:800
        if(final(i,j)>=1)
            x_sum = x_sum + i;
            y_sum = y_sum + j;
            tot_count = tot_count +1;
        end
    end
end

cent_x = x_sum/tot_count;
cent_y = y_sum/tot_count;
```