ECE415 -- Homework 4

Name: Wang Yunjuan

UIN:661180568

**Problem 1**

Find the connected components in the image Connected.bmp.

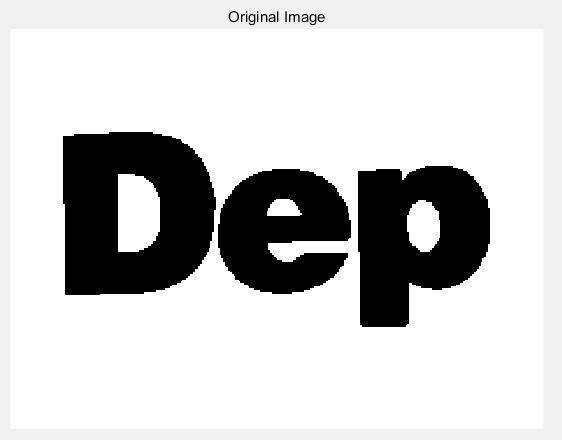
1. **Display the image**

Load the grayscale image Image.bmp into MATLAB and display the image. The picture is shown as  
follows.

im=imread('Connected.bmp');

imshow(im);

title('Original Image');



1. **Perform thresholding if necessary, to create binary image. Describe the method and threshold used. Display the image.**

I first write a function *histogram* to calculate the histogram for the intensity image, which can implement the built-in function of *imhist*. The MATLAB code is as follows.

function y=histogram(I)

[row col]=size(I);

h=zeros(1,256);

for i=1:row

for j=1:col

h(I(i,j)+1)=h(I(i,j)+1)+1;

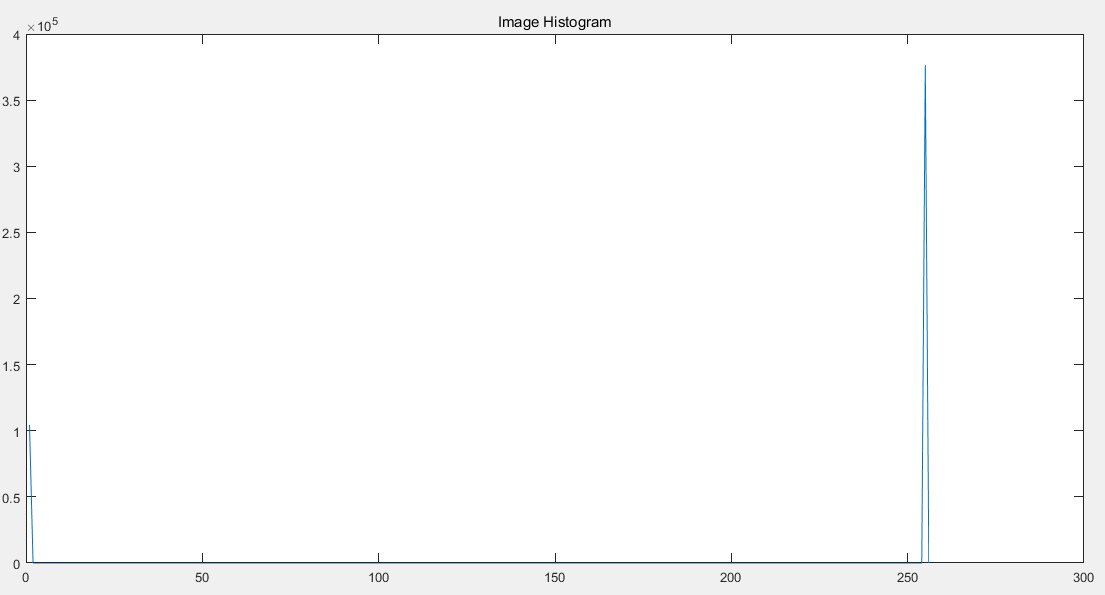
end

end

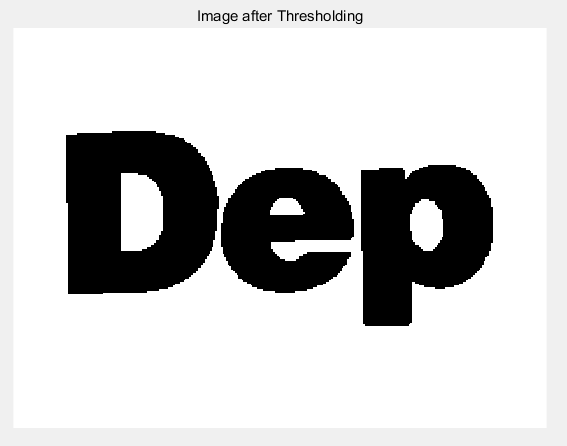
y=h;

end

Then I use function *histogram* I have write above to display the histogram of picture.



We can easily find out the pixel value basically either 0 or 255. We just choose 100 to create the binary image.



1. **Find the connected components using the component labeling two-pass algorithm assuming 4 connectivity. How many components are there?**

I find **three** components.

First pass:

Iterate the image from top left to bottom right. Each time if the pixel (x, y) is black, do the following thing:

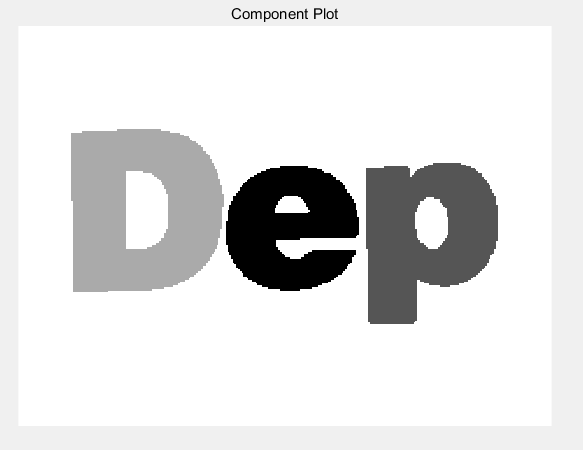
1. If both pixel (x, y-1) and (x-1, y) are white, give this pixel a new label.
2. If either pixel (x, y-1) or (x-1, y) is black, give this pixel the same label as the black pixel next to it.
3. If both pixel (x, y-1) and (x-1, y) are black, give this pixel the same label as either of the neighbor pixel and record that the label of pixel (x, y-1) and (x-1, y) are equivalent.

Second pass:

Assign same label to all the pixel with equivalent labels.

1. **Substituting the pixel values in binary image with the component number they belong to, and mapping the component numbers to the full dynamic range of 8 bits display the image to demonstrate the found connected components. Comment!**

According to the picture, there are three different colors, which means we find out three components.



MATLAB code:

%% find connected component

flag=1;

count=[];

label=zeros(row,col);

x=0;

for i=1:row

for j=1:col

if im(i,j)==0

if im(i,j-1)==255&&im(i-1,j)==255

count=[count,flag];

label(i,j)=count(x+1);

flag=flag+1;

x=x+1;

continue;

end

if im(i,j-1)==255&&im(i-1,j)==0

label(i,j)=count(label(i-1,j));

continue;

end

if im(i,j-1)==0&&im(i-1,j)==255

label(i,j)=count(label(i,j-1));

continue;

end

if im(i,j-1)==0&&im(i-1,j)==0

label(i,j)=label(i-1,j);

for k=1:x

if count(k)==count(label(i,j-1))

count(k)=count(label(i-1,j));

end

end

continue;

end

end

end

end

component=unique(count)

len=size(component)

output=zeros(row,col);

for i=1:row

for j=1:col

if label(i,j)==0

output(i,j)=0;

else

output(i,j)=count(label(i,j));

end

end

end

for i=1:row

for j=1:col

if label(i,j)==0

output(i,j)=0;

else

output(i,j)=count(label(i,j));

end

end

end

for i=1:row

for j=1:col

for k=1:len(2)

if output(i,j)==component(k)

output(i,j)=k;

end

end

end

end

for k=0:len(2)

for i=1:row

for j=1:col

if output(i,j)==k

output(i,j)=-255/len(2)\*(k-3);

end

end

end

end

output=uint8(output);

subplot(2,2,3);

imshow(output);

title('Component Plot');

**Problem 2**

Load the image Image.bmp.

1. **Display the image.**

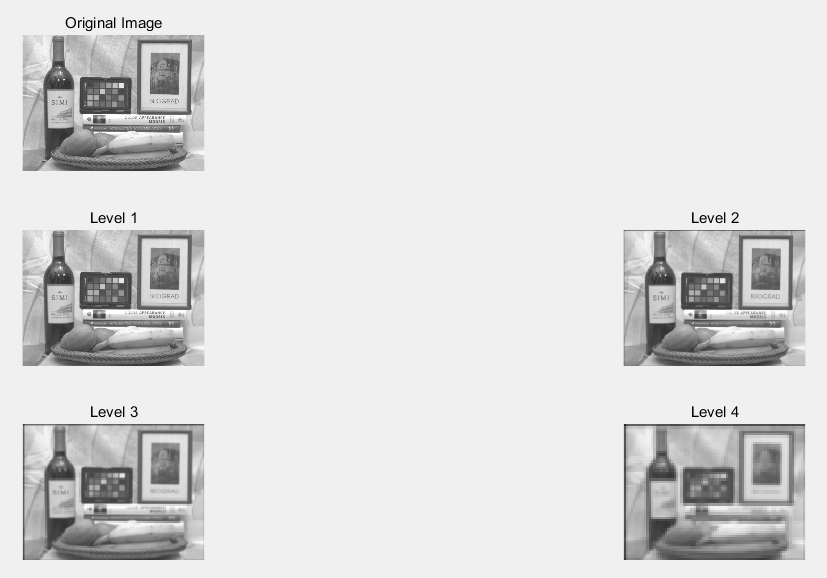
im=imread('Image.bmp');

imshow(im);

title('Original Image');



1. **Create a Gaussian pyramid for the Image.bmp. The pyramid should have the original image and four more levels. Display all five levels of the pyramid.**



MATLAB code:

clear;clc;

im=imread('Image.bmp');

subplot(3,2,1);

imshow(im);

title('Original Image');

[row, col]=size(im);

im1=double(im);

output1=zeros(row,col);

outputmid1=zeros(row+4,col);

im1=[zeros(2,col);im1;zeros(2,col)];

im1=[zeros(row+4,2) im1 zeros(row+4,2)];

for i=3:row+2

for j=1:col

outputmid1(i,j)=round(im1(i,j)/16+im1(i,j+1)/4+6\*im1(i,j+2)/16+im1(i,j+3)/4+im1(i,j+4)/16);

end

end

for i=1:col

for j=1:row

output1(j,i)=round(outputmid1(j,i)/16+outputmid1(j+1,i)/4+6\*outputmid1(j+2,i)/16+outputmid1(j+3,i)/4+outputmid1(j+4,i)/16);

end

end

for j=col:-2:2

output1(:,j)=[];

end

for i=row:-2:2

output1(i,:)=[];

end

output1=uint8(output1);

subplot(3,2,3);

imshow(output1);

title('Level 1');

[row1, col1]=size(output1);

im2=double(output1);

output2=zeros(row1,col1);

outputmid2=zeros(row1+4,col1);

im2=[zeros(2,col1);im2;zeros(2,col1)];

im2=[zeros(row1+4,2) im2 zeros(row1+4,2)];

for i=3:row1+2

for j=1:col1

outputmid2(i,j)=round(im2(i,j)/16+im2(i,j+1)/4+6\*im2(i,j+2)/16+im2(i,j+3)/4+im2(i,j+4)/16);

end

end

for i=1:col1

for j=1:row1

output2(j,i)=round(outputmid2(j,i)/16+outputmid2(j+1,i)/4+6\*outputmid2(j+2,i)/16+outputmid2(j+3,i)/4+outputmid2(j+4,i)/16);

end

end

for j=col1:-2:2

output2(:,j)=[];

end

for i=row1:-2:2

output2(i,:)=[];

end

output2=uint8(output2);

subplot(3,2,4);

imshow(output2);

title('Level 2');

[row2, col2]=size(output2);

im3=double(output2);

output3=zeros(row2,col2);

outputmid3=zeros(row2+4,col2);

im3=[zeros(2,col2);im3;zeros(2,col2)];

im3=[zeros(row2+4,2) im3 zeros(row2+4,2)];

for i=3:row2+2

for j=1:col2

outputmid3(i,j)=round(im3(i,j)/16+im3(i,j+1)/4+6\*im3(i,j+2)/16+im3(i,j+3)/4+im3(i,j+4)/16);

end

end

for i=1:col2

for j=1:row2 output3(j,i)=round(outputmid3(j,i)/16+outputmid3(j+1,i)/4+6\*outputmid3(j+2,i)/16+outputmid3(j+3,i)/4+outputmid3(j+4,i)/16);

end

end

for j=col2:-2:2

output3(:,j)=[];

end

for i=row2:-2:2

output3(i,:)=[];

end

output3=uint8(output3);

subplot(3,2,5);

imshow(output3);

title('Level 3');

[row3, col3]=size(output3);

im4=double(output3);

output4=zeros(row3,col3);

outputmid4=zeros(row3+4,col3);

im4=[zeros(2,col3);im4;zeros(2,col3)];

im4=[zeros(row3+4,2) im4 zeros(row3+4,2)];

for i=3:row3+2

for j=1:col3

outputmid4(i,j)=round(im4(i,j)/16+im4(i,j+1)/4+6\*im4(i,j+2)/16+im4(i,j+3)/4+im4(i,j+4)/16);

end

end

for i=1:col3

for j=1:row3

output4(j,i)=round(outputmid4(j,i)/16+outputmid4(j+1,i)/4+6\*outputmid4(j+2,i)/16+outputmid4(j+3,i)/4+outputmid4(j+4,i)/16);

end

end

for j=col3:-2:2

output4(:,j)=[];

end

for i=row3:-2:2

output4(i,:)=[];

end

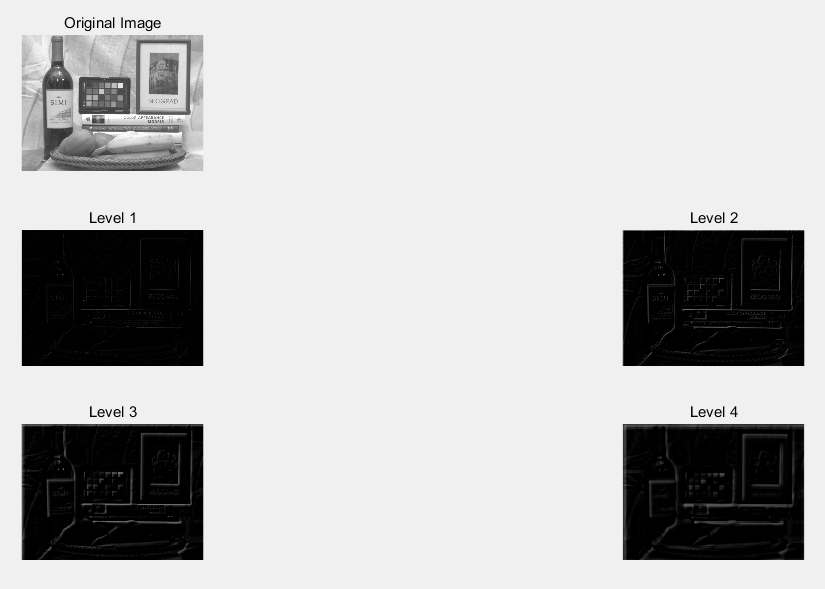
output4=uint8(output4);

subplot(3,2,6);

imshow(output4);

title('Level 4');

1. **Create a Laplacian pyramid for the Image.bmp. The pyramid should have the original image and four more levels. Display all five levels of the pyramid.**



MATLAB code:

upoutput1=double(output1);

for j=col1:-1:1

upoutput1=[upoutput1(:,1:j-1) zeros(row1,1) upoutput1(:,j:end)];

end

for i=row1:-1:1

upoutput1=[upoutput1(1:i-1,:); zeros(1,2\*col1); upoutput1(i:end,:)];

end

upoutput1=[zeros(2,2\*col1);upoutput1;zeros(2,2\*col1)];

upoutput1=[zeros(2\*row1+4,2) upoutput1 zeros(2\*row1+4,2)];

upoutputmid1=zeros(2\*row1+4,2\*col1);

lapout1=zeros(2\*row1,2\*col1);

for i=1:2\*row1+4

for j=1:2\*col1

upoutputmid1(i,j)=round(upoutput1(i,j)/8+upoutput1(i,j+1)/2+6\*upoutput1(i,j+2)/8+upoutput1(i,j+3)/2+upoutput1(i,j+4)/8);

end

end

for i=1:2\*col1

for j=1:2\*row1

lapout1(j,i)=round(upoutputmid1(j,i)/8+upoutputmid1(j+1,i)/2+6\*upoutputmid1(j+2,i)/8+upoutputmid1(j+3,i)/2+upoutputmid1(j+4,i)/8);

end

end

laplace1=uint8(double(im)-lapout1(1:row,1:col));

subplot(3,2,3);

imshow(laplace1);

title('Level 1');

upoutput2=double(output2);

for j=col2:-1:1

upoutput2=[upoutput2(:,1:j-1) zeros(row2,1) upoutput2(:,j:end)];

end

for i=row2:-1:1

upoutput2=[upoutput2(1:i-1,:); zeros(1,2\*col2); upoutput2(i:end,:)];

end

upoutput2=[zeros(2,2\*col2);upoutput2;zeros(2,2\*col2)];

upoutput2=[zeros(2\*row2+4,2) upoutput2 zeros(2\*row2+4,2)];

upoutputmid2=zeros(2\*row2+4,2\*col2);

lapout2=zeros(2\*row2,2\*col2);

for i=1:2\*row2+4

for j=1:2\*col2

upoutputmid2(i,j)=round(upoutput2(i,j)/8+upoutput2(i,j+1)/2+6\*upoutput2(i,j+2)/8+upoutput2(i,j+3)/2+upoutput2(i,j+4)/8);

end

end

for i=1:2\*col2

for j=1:2\*row2

lapout2(j,i)=round(upoutputmid2(j,i)/8+upoutputmid2(j+1,i)/2+6\*upoutputmid2(j+2,i)/8+upoutputmid2(j+3,i)/2+upoutputmid2(j+4,i)/8);

end

end

laplace2=uint8(double(output1)-lapout2(1:row1,1:col1));

subplot(3,2,4);

imshow(laplace2);

title('Level 2');

upoutput3=double(output3);

for j=col3:-1:1

upoutput3=[upoutput3(:,1:j-1) zeros(row3,1) upoutput3(:,j:end)];

end

for i=row3:-1:1

upoutput3=[upoutput3(1:i-1,:); zeros(1,2\*col3); upoutput3(i:end,:)];

end

upoutput3=[zeros(2,2\*col3);upoutput3;zeros(2,2\*col3)];

upoutput3=[zeros(2\*row3+4,2) upoutput3 zeros(2\*row3+4,2)];

upoutputmid3=zeros(2\*row3+4,2\*col3);

lapout3=zeros(2\*row3,2\*col3);

for i=1:2\*row3+4

for j=1:2\*col3

upoutputmid3(i,j)=round(upoutput3(i,j)/8+upoutput3(i,j+1)/2+6\*upoutput3(i,j+2)/8+upoutput3(i,j+3)/2+upoutput3(i,j+4)/8);

end

end

for i=1:2\*col3

for j=1:2\*row3

lapout3(j,i)=round(upoutputmid3(j,i)/8+upoutputmid3(j+1,i)/2+6\*upoutputmid3(j+2,i)/8+upoutputmid3(j+3,i)/2+upoutputmid3(j+4,i)/8);

end

end

laplace3=uint8(double(output2)-lapout3(1:row2,1:col2));

subplot(3,2,5);

imshow(laplace3);

title('Level 3');

[row4, col4]=size(output4);

upoutput4=double(output4);

for j=col4:-1:1

upoutput4=[upoutput4(:,1:j-1) zeros(row4,1) upoutput4(:,j:end)];

end

for i=row4:-1:1

upoutput4=[upoutput4(1:i-1,:); zeros(1,2\*col4); upoutput4(i:end,:)];

end

upoutput4=[zeros(2,2\*col4);upoutput4;zeros(2,2\*col4)];

upoutput4=[zeros(2\*row4+4,2) upoutput4 zeros(2\*row4+4,2)];

upoutputmid4=zeros(2\*row4+4,2\*col4);

lapout4=zeros(2\*row4,2\*col4);

for i=1:2\*row4+4

for j=1:2\*col4

upoutputmid4(i,j)=round(upoutput4(i,j)/8+upoutput4(i,j+1)/2+6\*upoutput4(i,j+2)/8+upoutput4(i,j+3)/2+upoutput4(i,j+4)/8);

end

end

for i=1:2\*col4

for j=1:2\*row4

lapout4(j,i)=round(upoutputmid4(j,i)/8+upoutputmid4(j+1,i)/2+6\*upoutputmid4(j+2,i)/8+upoutputmid4(j+3,i)/2+upoutputmid4(j+4,i)/8);

end

end

laplace4=uint8(double(output3)-lapout4(1:row3,1:col3));

subplot(3,2,6);

imshow(laplace4);

title('Level 4');

1. **Comment on the difference between the Gaussian and Laplacian pyramid of the image.**

The Laplace pyramid can be calculated by subtracting interpolated filtering pyramid from Gaussian pyramid. With level increase, the picture is more blur in Gaussian pyramid because we down sample more information. As a result, with level increase, the edges in Laplacian pyramid is thicker.