**SVKM’s NMIMS**

**Mukesh Patel School of Technology Management & Engineering**

**Computer Engineering Department**

Program: B.Tech. Sem VI

**Course: Image Processing**

LAB Manual

PART A

**Experiment No.09**

**A.1 Aim:**

Write a program to apply following morphological operations on finger print image.

1. Opening
2. Opening followed by closing

**A.2 Prerequisite:**

1 Matlab programming syntax (Refer the Matlab manual).

2. Knowledge of fundamentals of morphological operations.

2. Availability of Soft copy of finger print image.

**A.3 Outcome:**

**After successful completion of this experiment students will be able to**

1. Apply Opening and closing operations on given image.
2. Differentiate the outputs of different methods of opening and closing.
3. Identify applications of morphological operations studied.

**A.4 Theory:**

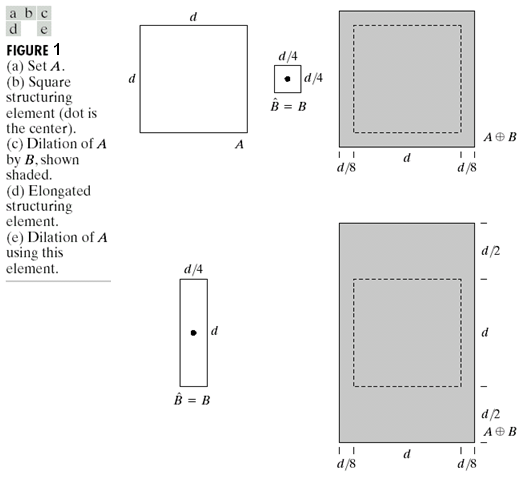
**A.4.1. Morphological operations**

**Morphological operations**

**1. Dilation**







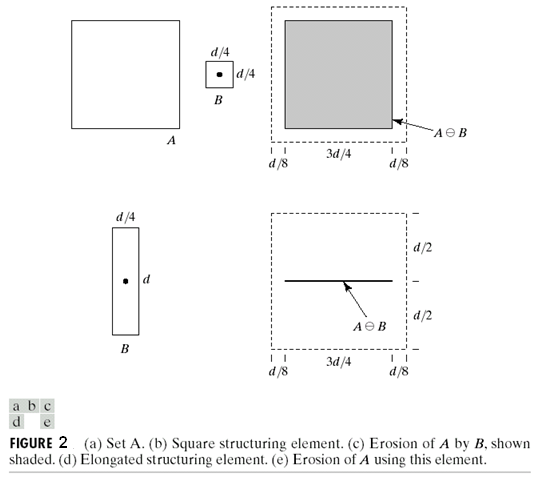
**2. Erosion**

Erosion can be defined as:

…… Eq (2)

E= {z| (B)z subset of A}

For example:



The figure is taken from Text book: “The fundamentals of Image processing” by Gonzalez Woods

**3. Opening:**

Opening is defined as

A ○ B = (A erode B) dilate B ….. Eq (3)

**4. Closing:**

Closing is defined as

A ● B = (A dilate B) erode B …… Eq (4)

**A.5 Procedure/Algorithm:**

**A.5.1:**

**TASK 1:**

1. Read the input finger print image.

2. Apply morphological operations following order and obtain 4 outputs

Separately.

1. Opening
2. Closing
3. Opening followed by closing.
4. Closing followed by opening

3. Display the original and the output images.

4. Observe/compare all outputs and complete PART B of lab manual.

5. Save and close the file and name it as **EX9\_Task1\_your Roll no.m**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

PART B

|  |  |
| --- | --- |
| Roll No. B032 | Name: NAMAN GARG |
| Class : CS B | Batch : B2 |
| Date of Experiment: 16/9/2020 | Date of Submission:19/9/2020 |
| Grade : | Time of Submission: |
| Date of Grading: |  |

**B.1 Software Code written by student:**

img=imread('cc.jpg');

img=rgb2gray(img);

img=imresize(img,[256,256],'nearest');

se1 = strel('Line',11,18);

se2 = strel('disk',15);

se3 = strel('diamond',25);

BW1= imdilate(img,se1);

BW2= imdilate(img,se2);

BW3= imdilate(img,se3);

subplot(2,3,2),imshow(img),title('Original');

subplot(2,3,4),imshow(BW1),title('Dilation - Line');

subplot(2,3,5),imshow(BW2),title('Dilation - Disk');

subplot(2,3,6),imshow(BW3),title('Dilation - Diamond');

img=imread('H.jpeg');

img=rgb2gray(img);

img=imresize(img,[256,256],'nearest');

se1 = strel('Line',11,18);

se2 = strel('disk',15);

se3 = strel('diamond',20);

BW1= imerode(img,se1);

BW2= imerode(img,se2);

BW3= imerode(img,se3);

subplot(2,3,2),imshow(img),title('Original');

subplot(2,3,4),imshow(BW1),title('Erode - Line');

subplot(2,3,5),imshow(BW2),title('Erode - Disk');

subplot(2,3,6),imshow(BW3),title('Erode - Diamond');

img=imread('cc2.jpg');

img=rgb2gray(img);

img=imresize(img,[256,256],'nearest');

eroded\_image=erosion(img);

opened\_image=dilation(eroded\_image);

dilated\_image=dilation(img);

closed\_image=erosion(dilated\_image);

subplot(2,2,1),imshow(uint8(dilation(img))),title('1 Level Dilation');

subplot(2,2,2),imshow(uint8(erosion(img))),title('1 Level Erosion');

subplot(2,2,3),imshow(uint8(opened\_image)),title('1 Level Opening');

subplot(2,2,4),imshow(uint8(closed\_image)),title('1 Level Closing');

function res=dilation(img)

res=zeros(256);

for i=2:255

for j=2:255

res(i,j)=max(reshape(img(i-1:i+1,j-1:j+1),[1,9]));

end

end

res=uint8(res);

end

function res2=erosion(img)

res2=zeros(256);

for i=2:255

for j=2:255

res2(i,j)=min(reshape(img(i-1:i+1,j-1:j+1),[1,9]));

end

end

res2=uint8(res2);

end

img=imread('H.jpeg');

img=rgb2gray(img);

img=imresize(img,[256,256],'nearest');

eroded\_image=erosion(img);

opened\_image=dilation(eroded\_image);

dilated\_image=dilation(img);

closed\_image=erosion(dilated\_image);

open\_then\_close=erosion(opened\_image);

close\_then\_open=dilation(closed\_image);

subplot(5,2,1),imshow(uint8(dilation(img))),title('1 Level Dilation');

subplot(5,2,2),imshow(uint8(dilation(dilation(img)))),title('2 Level Dilation');

subplot(5,2,3),imshow(uint8(dilation(dilation(dilation(img))))),title('3 Level Dilation');

subplot(5,2,4),imshow(uint8(erosion(img))),title('1 Level erosion');

subplot(5,2,5),imshow(uint8(erosion(erosion(img)))),title('2 Level erosion');

subplot(5,2,6),imshow(uint8(erosion(erosion(erosion(img))))),title('3 Level erosion');

subplot(5,2,7),imshow(uint8(opened\_image)),title('Opening');

subplot(5,2,8),imshow(uint8(closed\_image)),title('Closing');

subplot(5,2,9),imshow(uint8(open\_then\_close)),title('Open then Close');

subplot(5,2,10),imshow(uint8(close\_then\_open)),title('Close then Open');

function res=dilation(img)

res=zeros(256);

for i=2:255

for j=2:255

res(i,j)=max(reshape(img(i-1:i+1,j-1:j+1),[1,9]));

end

end

res=uint8(res);

end

function res2=erosion(img)

res2=zeros(256);

for i=2:255

for j=2:255

res2(i,j)=min(reshape(img(i-1:i+1,j-1:j+1),[1,9]));

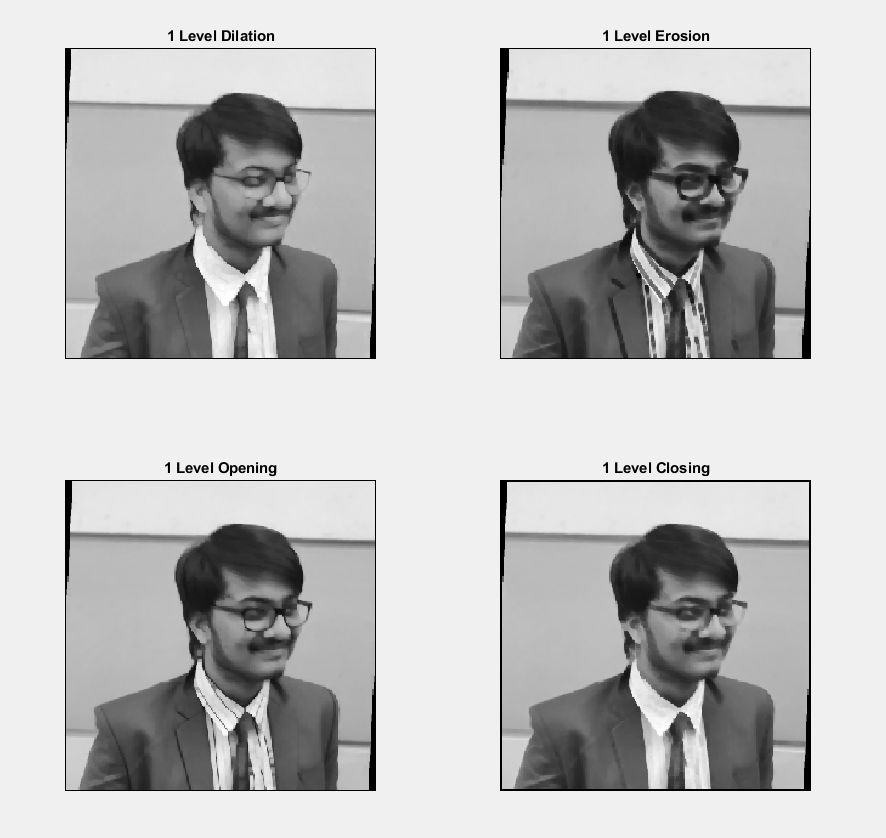
end

end

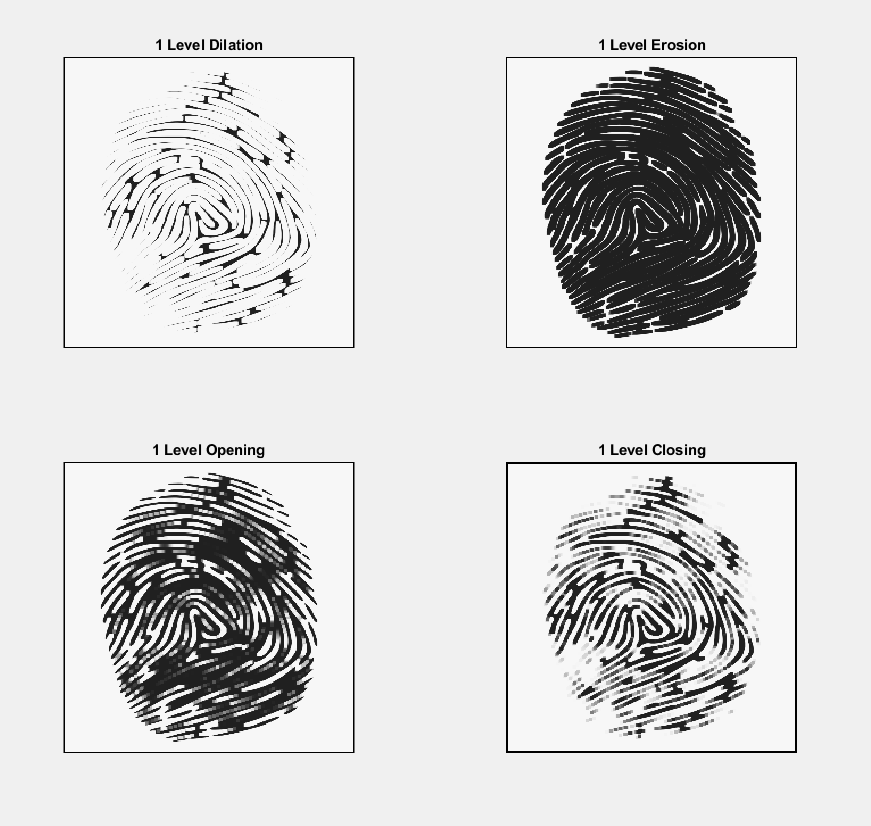
res2=uint8(res2);

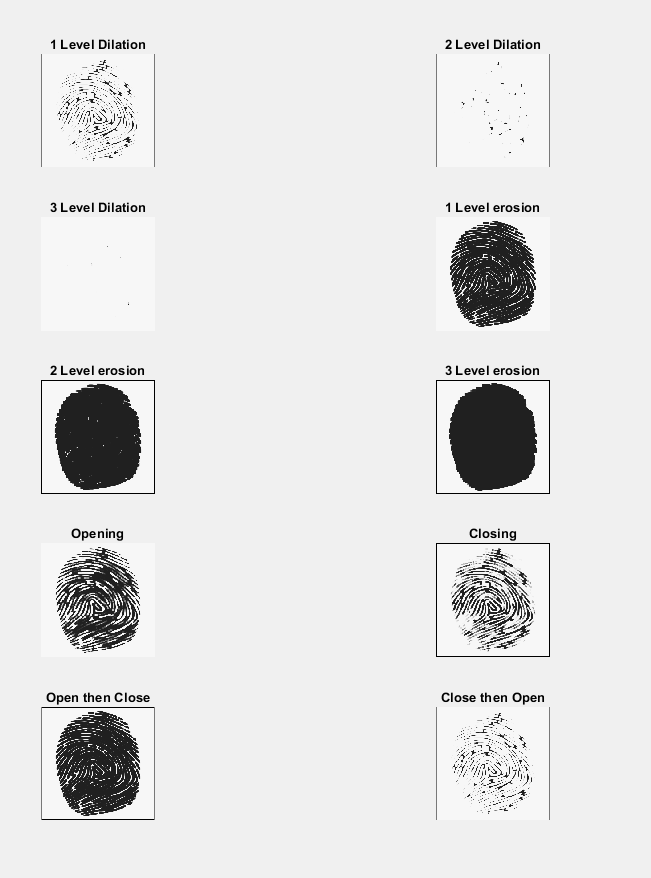
end

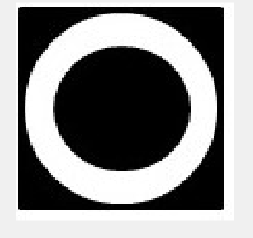
**B.2 Input and Output:**

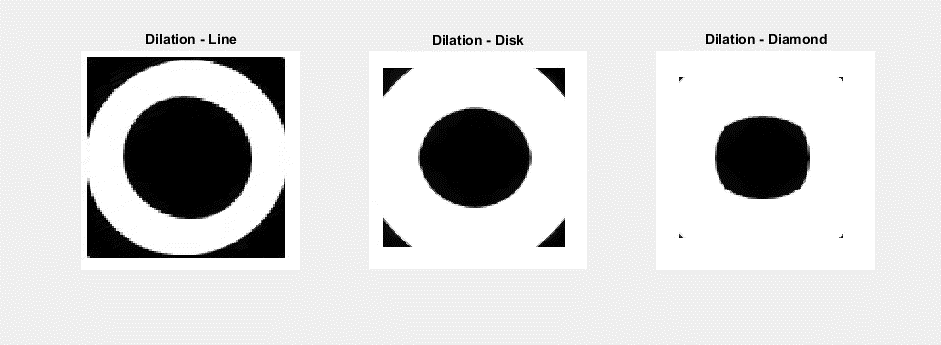
****

****

****

****

****



**B.3 Observations and learning:**

Erosion removes the pixels on the perimeter and reduces the brightness of the image in general and removes the pixels or objects smaller than the element.

As seen in outputs when we use dilation on an image and then erosion on the same the image retains most of its original details and doesn’t look much different from the original one. But still it is differentiable

B.4 Conclusion:

We were able to learn the real application of dilation and erosion morphology on images as shown above in which we performed a 3 level dilation and erosion on an image followed by combination of both

B.5 Question of Curiosity

List of out real life applications of morphological operations:

Applications of morphological dilation and erosion operations

1. Real mechanical surface reconstruction
2. Freeform surface deviation evaluation
3. Surface scanning

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*