CLASS : B.E. E &TC **SUBJECT: DIVP**

EXPT. NO. : 9 **DATE:09-10-2020**

TITLE : TO APPLY MORPHOLOGICAL OPERATORS ON AN IMAGE.

CO 2: Given a gray image, select an appropriate technique (similarity based or discontinuity based) to segment it. Derive the mask coefficients of First order Derivative (FoD) and Second order Derivative (SoD) to detect an edge in an image. Considering an appropriate test case, analyze and compare the performance of FoD and SoD using parameters like response to constant intensity and isolated intensities in an image.

CO4: Carry out experiments as an individual and in a team, comprehend and write a laboratory record and draw conclusions at a technical level.

AIM : To implement morphological operations in Matlab.

SOFTWARE REQUIREMENT: Matlab R2007b or above, python

THEORY:

9.1 Morphological Image Processing

Mathematical morphology is a tool for extracting image components that are useful in the representation and description of region shape, such as boundaries, skeletons, and the convex hull. We are interested also in morphological techniques for pre- or post processing such as morphological filtering, thinning, and pruning.

While commonly used on binary images, this approach can also be extended to gray-scale images.

In the general case, morphological image processing operates by passing a structuring element over the image in an activity similar to convolution. The structuring element can be of any size and it can contain any complement of 1's and 0's. At each pixel position, a specified logical operation is performed between the structuring element and the underlying binary image. The binary result of that logical operation is stored in the output image at that pixel position. The effect created depends upon the size and content of the structuring element and upon the nature of the logical operation.

9.2 Translation & reflection

We need two additional definitions that are used extensively in morphology but generally are not found in basic texts on set theory .The reflection of set B is defined as

$$\hat{B} = \{w \mid w=-b, \text{ for } b \in B\}$$

The translation of set A by point z = (z1, z2), denoted $(A)_z$, is defined as

$$(A)_z = \{c \mid c=a+z, \text{ for } a \in A\}$$

9.3 Binary image processing

Binary images have only two gray levels and they constitute an important subset of digital images. A binary image normally results from an image segmentation operation. If the initial segmentation is not completely satisfactory, some form of processing done on the binary image can often improve the situation.

Recall the two rules of the connectivity. In the binary image, any pixel, together with its eight neighbors, represents nine bits of information.

9.4 Erosion

The process is also known as “shrinking”. The manner and extent of shrinking is controlled by a structuring element. Simple erosion is a process of eliminating all the boundary points from an object, leaving the object smaller in area by one pixel all around its perimeter. If the object is circular, its diameter decreases by two pixels with each erosion. If it narrows to less than three pixels thick at any point, it will become disconnected (into two objects) at that point. Objects no more than two pixel thick in any direction are eliminated.

The erosion of A by B, denoted as A Θ B, is defined as

$$A \ominus B = \{z \mid (B)_z \cap A^c \neq \phi\}$$

In other words, erosion of A by B is the set of all structuring element origin locations where the translated B has no overlap with the background of A.

Original Image

Erosion by 3*3 structuring element

Erosion by 5*5 structuring element

9.5 Dilation:

Simple dilation is the process of incorporating into the object all the background points that touches it, leaving it larger in area by that amount. If the object is circular, its diameter increases by two pixels with each dilation. If two objects are separated by less than three pixels at any point, they will become connected (merged into one object) at that point.

Dilation is an operation that “grows” or “thickens” objects in a binary image. The specific manner and extend of this thickening is controlled by a shape referred to as structuring element.

Mathematically, dilation is defined in terms of set operations. The dilation of A by B, denoted

$A \oplus B$, is defined as

$$A \oplus B = \{z \mid (\hat{B})_z \cap A \neq \phi\}$$

Original Image | **Dilation by 3*3 structuring element** | **Dilation by 3*3 structuring element**

9.6 Opening and Closing

9.6.1 Opening

Opening generally smoothens the contour of an image, breaks narrow isthmuses, and eliminates thin protrusions. The opening of set A by structuring element B, denoted A◦B, is defined as

$$A \circ B = (A \ominus B) \oplus B$$

In other words, opening of A by B is simply the erosion of A by B, followed by dilation of the result by B.

The opening operation satisfies the following properties.

1) A◦B is a subset of A.
2) If C is a subset of D, then C◦B is a subset of D◦B.
3) (A ◦ B) ◦ B = A◦B. This is also called as idempotence.

9.6.2 Closing:

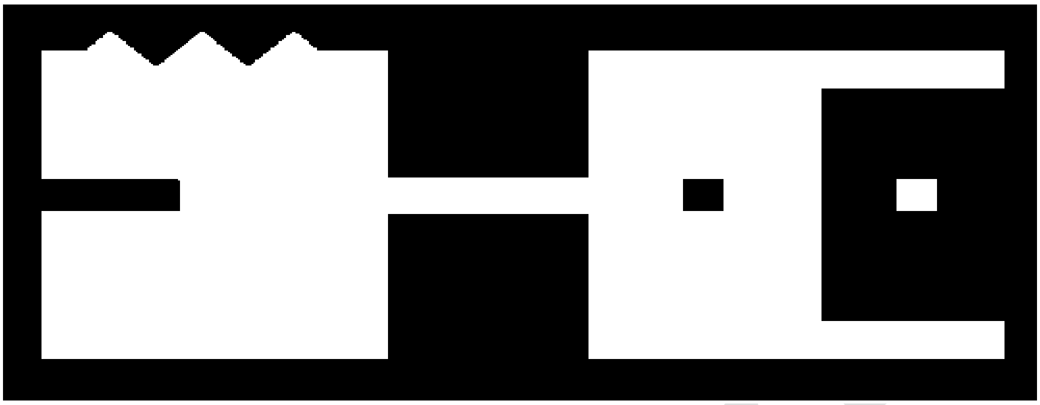
Closing smoothens sections of contours but, as opposed to opening, it generally fuses narrow breaks and long thin guffs, eliminates small holes, and fills gaps in the contour. The closing of set A by structuring element B, denoted A•B is defined as

$$A \bullet B = (A \oplus B) \ominus B$$

In other words, closing of A by B is simply the dilation of A by B, followed by erosion of the result by B.

The closing operation satisfies the following properties

1) A•B is a subset of A.
2) If C is a subset of D, then C•B is a subset of D•B.
3) (A • B) • B = A • B. This is also called as idempotence.



Original Image

Image after opening

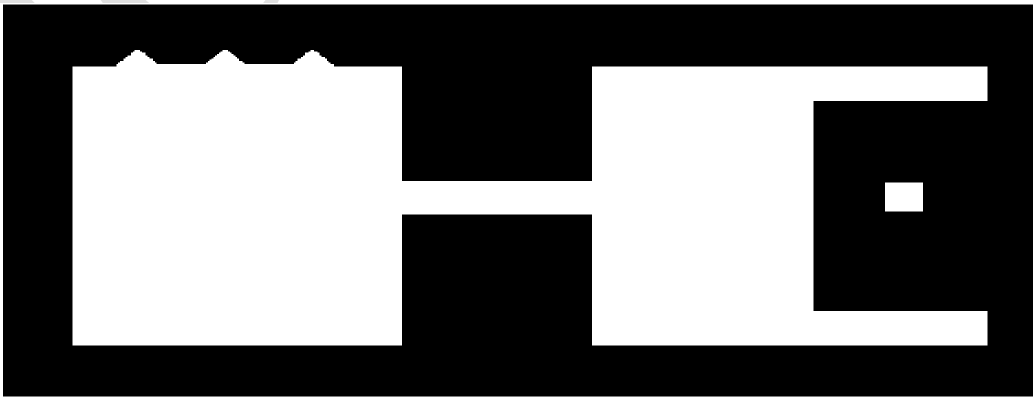


Image after closing

9.7 Algorithms:

9.7.1 Erosion:

1. Start.
2. Read the image.
3. Initialize the Structuring Element (Here, 3x3 matrix, with origin at center element).
4. Move the structuring element mask over entire image and each time multiply the each element with respective pixel in image.
5. For erosion image, select the minimum element of new matrix and replace it with origin place.
6. Stop.

9.7.2 Dilation:

1. Start.
2. Read the image.
3. Initialize the Structuring Element (Here, 3x3 matrix, with origin at center element).
4. Move the structuring element mask over entire image and each time multiply the each element with respective pixel in image.
5. For erosion image, select the maximum element of new matrix and replace it with origin place.
6. Stop.

9.7.3 Opening

1. Start.
2. Read the image.
3. Perform the erosion on image.
4. Perform the dilation on new image.
5. Stop.

9.7.4 Closing

1. Start.
2. Read the image.
3. Perform the dilation on image.
4. Perform the erosion on new image.
5. Stop.

9.8 Applications of Morphological Image Processing:

1. The simplest application of dilation is for bridging gaps.
2. Morphological operations are used to construct filters similar in concept to the spatial filters.
3. A morphological filter consisting of opening followed by closing can be used to eliminate the noise and its effects on the image.
4. Erosion is useful for removing from a segmented image objects that are too small to be of interest.
5. Dilation is useful for filtering holes in segmented objects.

9.9 Conclusion:

1. In grayscale images, erosion reduces the brightness (and therefore the size) of bright objects on a dark background by taking the neighborhood minimum when passing the structuring element over the image
2. In grayscale images, dilation increases the brightness of objects by taking the neighborhood maximum when passing the structuring element over the image.
3. Opening is a process in which first erosion operation is performed and then dilation operation is performed. It eliminates the thin protrusions of the obtained image.
4. Opening is used for removing internal noise of the obtained image.
5. Closing is a process in which first dilation operation is performed and then erosion operation is performed. It eliminates the small holes from the obtained image.
6. Closing is used for smoothening of contour and fusing of narrow breaks.

9.10 References:

1." Digital Image Processing " , by Gonzalez and Woods.

2. "Digital Image Processing " , S. Jayaraman, S. Esakkirajan, T. Veerakumar

3. Pictures taken from:

http://www.imageprocessingplace.com/root_files_V3/image_databases.htm

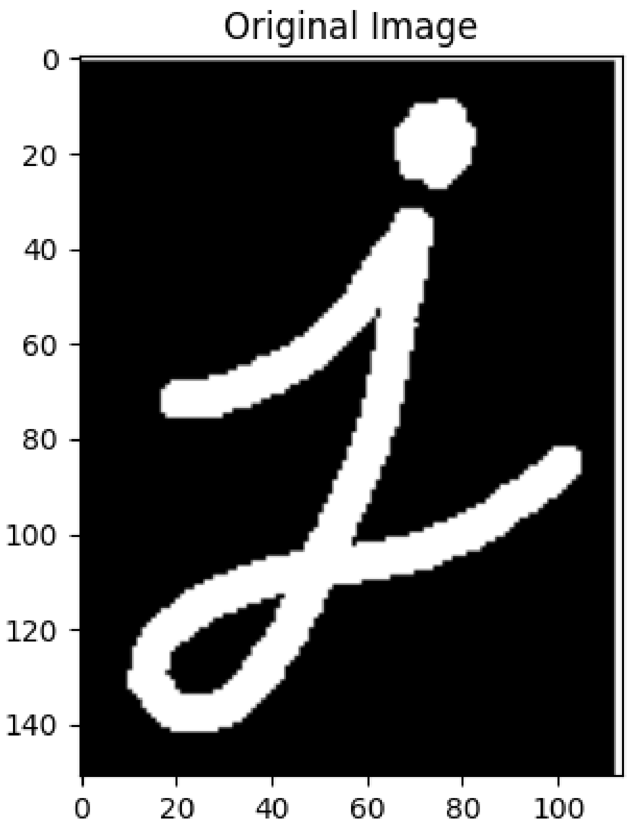
(Course Teacher)

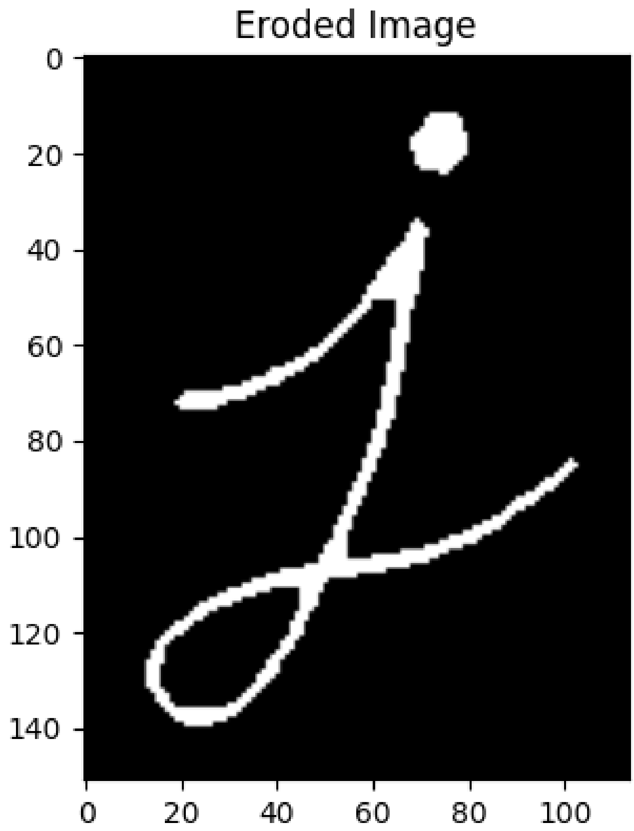
CLASS	**:**	**B.E (E &TC)**	**COURSE**	**:**	**DIVP**
AY	**:**	**2020-21 (SEM- I)**	**DATE**	**:**	**09-10-2020**
EXPT. NO.	**:**	**9**	**CLASS & ROLL NO**	**:**	**BE VIII 42428**
TITLE	**:**	**TO APPLY MORPHOLOGICAL OPERATORS ON AN IMAGE**			

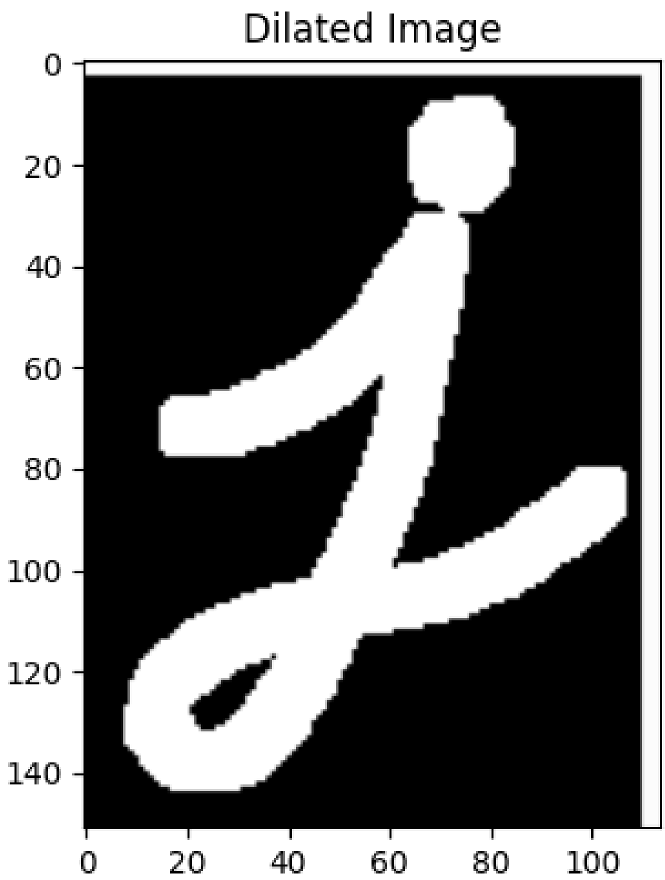
I. CODE:

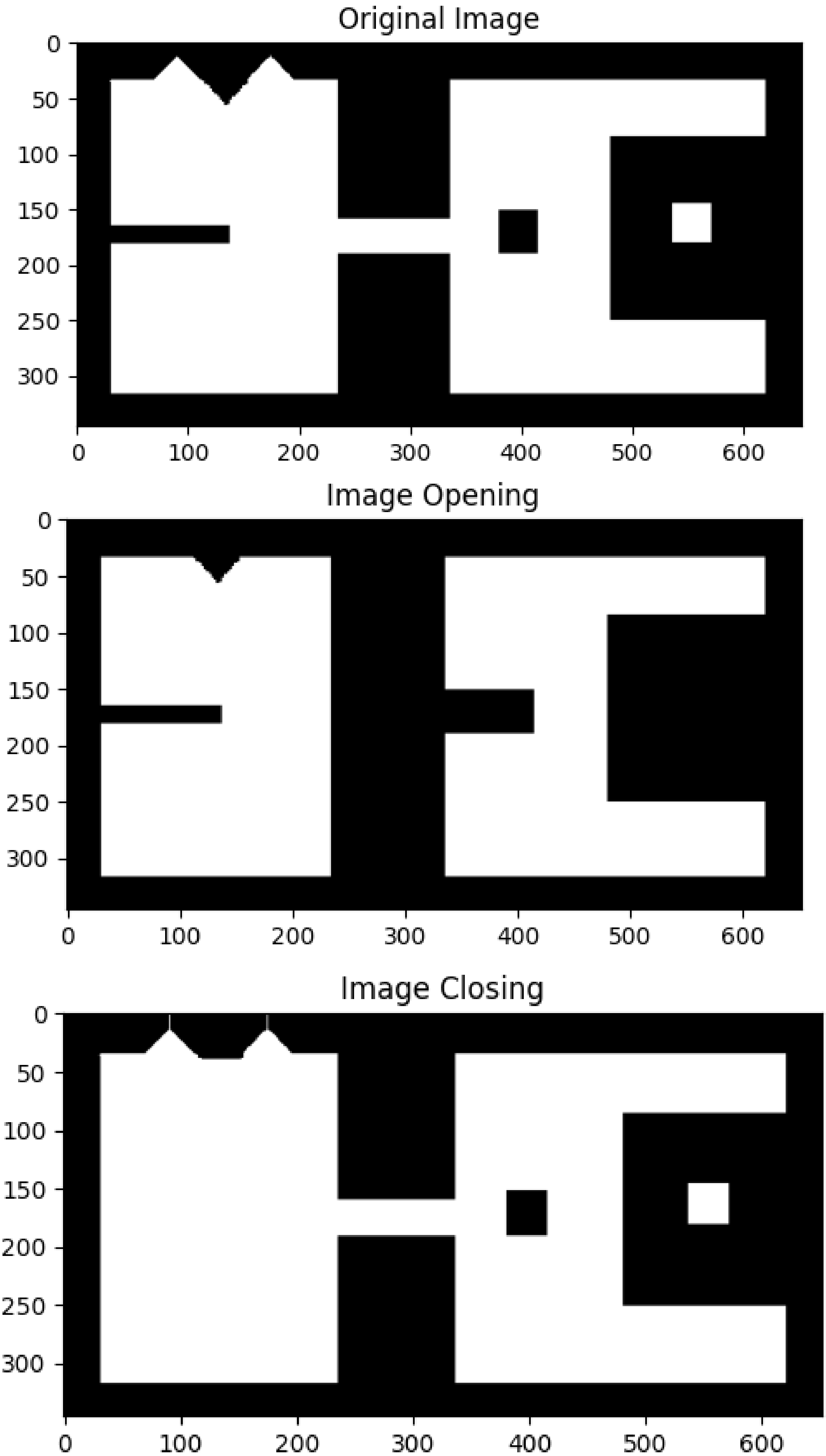
```
import cv2

# import Numpy
import numpy as np
from matplotlib import pyplot as plt

# read a image using imread
img = cv2.imread('Images/morphological/ero_dila.png',0)
img1 = cv2.imread('Images/morphological/open_close.png',0)

# Taking a matrix of size 5 as the kernel
kernel = np.ones((5,5), np.uint8)
open_kernel = np.ones((45,45), np.uint8)
close_kernel = np.ones((30,30), np.uint8)
def opening(img,kernel):
   img_ero = cv2.erode(img, kernel, iterations=1)
   img_open = cv2.dilate(img_ero, kernel, iterations=1)
   return img_open

def closing(img,kernel):
   img_dil = cv2.dilate(img, kernel, iterations=1)
   img_close = cv2.erode(img_dil, kernel, iterations=1)
   return img_close

# The first parameter is the original image,
# kernel is the matrix with which image is
# convolved and third parameter is the number
# of iterations, which will determine how much
# you want to erode/dilate a given image.
img_erosion = cv2.erode(img, kernel, iterations=1)
img_dilation = cv2.dilate(img, kernel, iterations=1)
img_opening = opening(img1, open_kernel)
img_closing = closing(img1, close_kernel)

plt.subplot(2, 3, 1),plt.imshow(img, 'gray'),plt.title('Original Image')
plt.subplot(2, 3, 2),plt.imshow(img_erosion, 'gray'),plt.title('Eroded Image')
plt.subplot(2, 3, 3),plt.imshow(img_dilation, 'gray'),plt.title('Dilated Image')
plt.subplot(2, 3, 4), plt.imshow(img1, 'gray'),plt.title('Original Image')
plt.subplot(2, 3, 5), plt.imshow(img_opening, 'gray'),plt.title('Image Opening')
plt.subplot(2, 3, 6), plt.imshow(img_closing, 'gray'),plt.title('Image Closing')
plt.show()
```

II. RESULTS:


Original Image
0
20
40
60
80
100
120
140
0 20 40 60 80 100


Eroded Image
0
20
40
60
80
100
120
140
0 20 40 60 80 100


Dilated Image
0
20
40
60
80
100
120
140
0 20 40 60 80 100


Original Image
0
50
100
150
200
250
300
0 100 200 300 400 500 600
Image Opening
0
50
100
150
200
250
300
0 100 200 300 400 500 600
Image Closing
0
50
100
150
200
250
300
0 100 200 300 400 500 600