

PRACTICAL 4

NAME-Shivam Tiwari

Roll no- B3-51

AIM:- 1. Perform morphological operations 2. Thresholding 2. Dilation
3.Erosion 4.Opening 5. Closing 6. gradient

THEORY : -

In image processing, morphological operations are used to process binary images (black and white). The process begins with thresholding, which converts grayscale images to binary by setting pixel values above a certain threshold to white and those below it to black.

Common morphological operations include:

- **Dilation:** Expands white areas, making objects larger.
- **Erosion:** Shrinks white areas, reducing object size.
- **Opening:** Removes small noise by first eroding then dilating the image.
- **Closing:** Fills small holes in objects by first dilating then eroding the image.
- **Gradient:** Highlights edges by calculating the difference between dilation and erosion.

These operations help enhance features, remove noise, and highlight structures in an image, making them useful in various computer vision tasks like object detection and image cleaning. The kernel, a small matrix, is used to define how the transformation applies to the image.

CODE:=

```
import cv2
import numpy as np
import matplotlib.pyplot as plt

image_path = "img.jpg"
img = cv2.imread(image_path, cv2.IMREAD_GRAYSCALE)

_, binary_image = cv2.threshold(img, 127, 255, cv2.THRESH_BINARY_INV)

kernel = np.ones((5,5),np.uint8)

# dialted image
dilated_image = cv2.dilate(binary_image, kernel , iterations=1)

# eroded image
eroded_image = cv2.erode(binary_image, kernel, iterations=1)

# opening
opened_image = cv2.morphologyEx(binary_image, cv2.MORPH_OPEN, kernel)

# closed
closed_image = cv2.morphologyEx(binary_image, cv2.MORPH_CLOSE, kernel)

# gradient
gradient_image = cv2.morphologyEx(binary_image, cv2.MORPH_GRADIENT, kernel)

# top hat
top_hat = cv2.morphologyEx(binary_image, cv2.MORPH_TOPHAT, kernel)

# black hat
black_hat = cv2.morphologyEx(binary_image, cv2.MORPH_BLACKHAT, kernel)

titles = ["original" , "dialted" , "eroded" , "opened" , "closed" , "gradient", "top hat" , "black hat"]
images = [binary_image , dilated_image , eroded_image , opened_image , closed_image , gradient_image , top_hat , black_hat]

for i in range(8):
    plt.subplot(2,4,i+1)
    plt.imshow(images[i] , cmap="gray")
    plt.title(titles[i])
    plt.axis("off")
```

original image



dilated



eroded



opened



closed



gradient



top hat



black hat



Conclusion

Morphological operations like dilation, erosion, opening, closing, gradient, blackhat, and tophat are powerful tools for enhancing and analyzing binary images. They help in noise removal, edge detection, and feature extraction, making them essential in various image processing tasks, including object detection and shape analysis.