

## DIP – 4

Cs20b1088

G.sachin sai reddy

```
# cs20b1088 G.sachin sai reddy
import cv2
import numpy as np
import math

def rotate_image(image, angle, center=None, scale=1.0):

    h, w = image.shape[:2]

    # If center is not specified, set it to the center of the image
    # this is for rotation w.r.t center.
    if center is None:
        center = (w // 2, h // 2)
    M = cv2.getRotationMatrix2D(center, angle, scale)
    rotated_predefined = cv2.warpAffine(image, M, (w, h),
    flags=cv2.INTER_LINEAR, borderMode=cv2.BORDER_REPLICATE)

    # Get the rotation matrix
    # used to convert degrees to radians
    theta = angle * np.pi / 180
    c, s = np.cos(theta), np.sin(theta)

    rotated = np.zeros_like(image)
    for i in range(h):
        for j in range(w):
            x, y = i - center[0], j - center[1]
            new_x, new_y = int(round(x * c + y * s)), int(round(-x * s
+ y * c))
            if 0 <= new_x + center[0] < h and 0 <= new_y + center[1] <
w:
                x1, y1 = new_x + center[0], new_y + center[1]
                x2, y2 = x1 + 1, y1 + 1
                a, b = x1 - new_x - center[0], y1 - new_y - center[1]
                # below is bilinear interpolation
                if 0 <= x1 < h and 0 <= y1 < w and 0 <= x2 < h and 0 <=
y2 < w:
```

```
        rotated[i, j] = (1 - a) * (1 - b) * image[x1, y1] +  
a * (1 - b) * image[x1, y2] + (1 - a) * b * \  
        image[x2, y1] + a * b * image[x2,  
y2]  
    return rotated, rotated_predefined  
  
# Load the image  
img = cv2.imread(r"C:\Users\sachi\OneDrive\Desktop\S6\DIP\PISA.jpg")  
  
# Rotate the image by 6 degrees  
rotated, rotated_predefined = rotate_image(img, 6)  
  
# Show the result  
cv2.imshow("Rotated", rotated)  
cv2.waitKey(0)  
cv2.destroyAllWindows()  
  
cv2.imshow("rotated_predefined", rotated_predefined)  
cv2.waitKey(0)  
cv2.destroyAllWindows()
```

