**--Lab-9--To perform various pattern clustering and analysis techniques on images.—**

import cv2

import numpy as np

# Load the image

img = cv2.imread('CV\_Test.jpg')

# Reshape the image to a 2D array of pixels and their color values

# (height \* width, 3 for BGR channels)

Z = img.reshape((-1, 3))

Z = np.float32(Z)

# Define criteria for k-means (type, max\_iter, epsilon)

criteria = (cv2.TERM\_CRITERIA\_EPS + cv2.TERM\_CRITERIA\_MAX\_ITER, 10, 1.0)

# Number of clusters (K)

K = 8

# Apply k-means

compactness, labels, centers =

cv2.kmeans(Z, K, None, criteria, 10, cv2.KMEANS\_RANDOM\_CENTERS)

# Convert centers to uint8 and reshape back to the original image dimensions

centers = np.uint8(centers)

res = centers[labels.flatten()]

res2 = res.reshape((img.shape))

# Display the segmented/quantized image

cv2.imshow('Quantized Image', res2)

cv2.waitKey(0)

cv2.destroyAllWindows()