Computer Vision

Home Work 1

Name – Rohit Das Student ID - ntnu_61047086s

Project – An image transformation program.

Language and library used: Python, OpenCV, Numpy

Description: This program will perform the following functions while executing lena.bmp image file:

Part 1(Without using any prebuilt libraries):

1. Flip the image vertically
2. Flip the image horizontally
3. Flip the image diagonally

Part 2(Using OpenCV Library):

1. Rotate the image 45°.
2. Shrink the image in half.
3. Binarize the image at 128 threshold

Algorithm Used –

Part 1: Basic mathematics to change row/columns/both elements of the matrix.

1. Flip Image Vertically – Flips the matrix(x,y) to (width-x-1,y)
2. Second Horizontally – Flips the matrix(x,y) to (width,height-y-1)
3. First diagonally – Flips the matrix(x,y) to (width-x-1, height-y-1)

Part 2: Image Transformation using libraries (OPENCV)

1. Flip the image 45°- After finding the centre point of the image (centre_x,centre_y), getRotationMatrix2D and WrapAffine are being used to transform the image to 45°.
2. Shrink the image in half- After equally reducing the dimensions to half, the new dimension values are being added to cv2.rotate function with no interpolation.
3. Binarize Lena at 128 – Applied cv2.threshold with 128 threshold value and the maxvalue as 255.

Parameters: None. Please Copy-paste the image path inside the program.

Principal Code:

Part 1:

```
image_file = r"C:\Users\Rohit\Desktop\Fall 2021 NTNU\Computer Vision
NTU\Chapter-1\Homework\lena.bmp"
lena = cv2.imread(image_file,cv2.COLOR_BGR2RGB)
flipped_lena_horizontal = np.full((512, 512, 3),0, dtype = np.uint8)
flipped_lena_vertical = np.full((512, 512, 3),0, dtype = np.uint8)
```

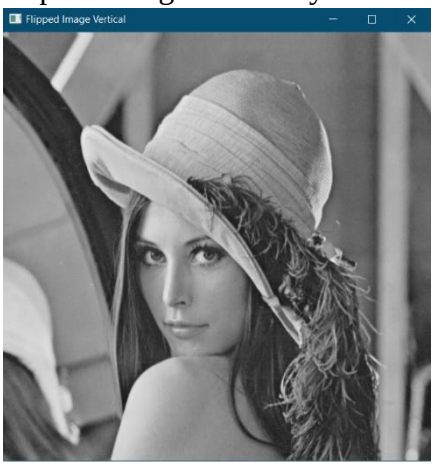
```
flipped_lena_diagonal = np.full((512, 512, 3),0, dtype = np.uint8)
height = lena.shape[0]
width = lena.shape[1]

for h_pixel in range(height):
        for w_pixel in range(width):
            flipped_lena_horizontal[h_pixel,w_pixel] = lena[height-h_pixel-
1,w_pixel]
            flipped_lena_vertical[h_pixel,w_pixel] = lena[h_pixel,width-
w_pixel-1]
            flipped_lena_diagonal[h_pixel,w_pixel] = lena[height-h_pixel-
1,width-w_pixel-1]

cv2.imshow('Original Image',lena)
cv2.imshow('Flipped Image Horizontal',flipped_lena_horizontal)
cv2.imshow('Flipped Image Vertical',flipped_lena_vertical)
cv2.imshow('Flipped Image Diagonal',flipped_lena_diagonal)
cv2.waitKey(0)
```

Part 2:

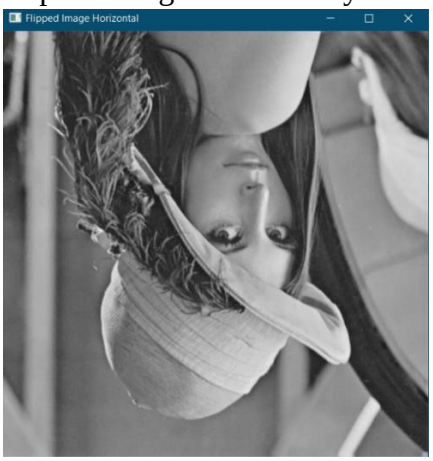
```
(centre_X, centre_Y) = (height // 2, width // 2)
r_matrix = cv2.getRotationMatrix2D((centre_X, centre_Y), 45, 1.0)
rotated_lena = cv2.warpAffine(lena, r_matrix, (width, height))
scale_percent = 50
reduced_width = int(lena.shape[1] * scale_percent / 100)
reduced_height = int(lena.shape[0] * scale_percent / 100)
reduced_dimension = (reduced_height, reduced_width)
shrink_lena = cv2.resize(lena,reduced_dimension,interpolation=None)
th,binarize_lena = cv2.threshold(lena,128,255,cv2.THRESH_BINARY)
cv2.imshow('Original Image',lena)
cv2.imshow('Rotated Image',rotated_lena)
cv2.imshow('Shrinked lena',shrink_lena)
cv2.imshow('Binarized Lena',binarize_lena)
cv2.waitKey(0)
```

Example:

1. Flip the image Vertically.

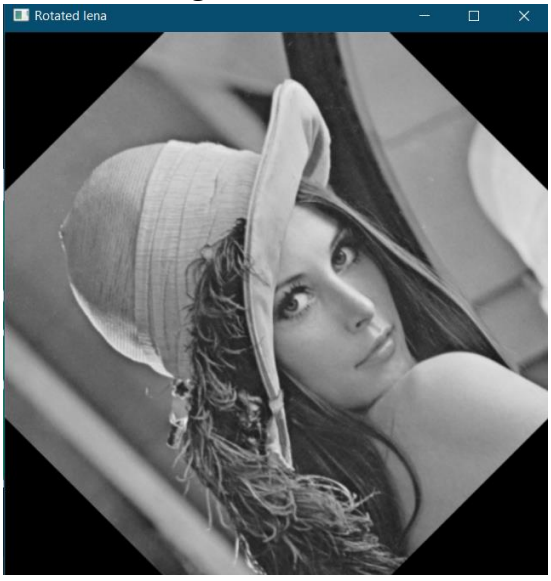


2. Flip the image Horizontally.

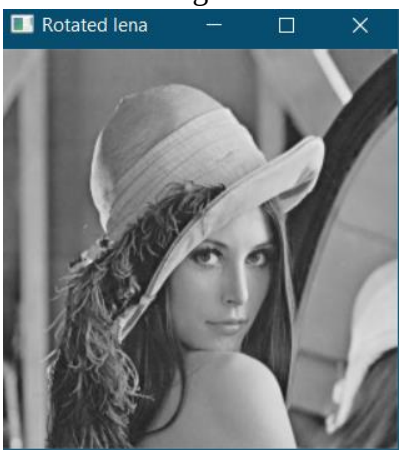


3. Flip the image Diagonally

4. Rotate the image 45°



5. Shrink the image in half



6. Binarize the image at 128 threshold.