2020 Computer Vision [Homework 1] 資工所 r08922123
Date: 17th Sep., 2020 Author: 王韻豪

I. Problem Statement

- Part1. Write a program to do the following requirement.

  (a) up-side-down lena.bmp

  (b) right-side-left lena.bmp

  (c) diagonally flip lena.bmp

- Part2. Write a program or use software to do the following requirement.

  (d) rotate lena.bmp 45 degrees clockwise

  (e) shrink lena.bmp in half

  (f) binarize lena.bmp at 128 to get a binary image

II. Programming Tools

- Programming language: Python 3.8.5
- Library: Numpy 1.19.1, OpenCV 4.0.1

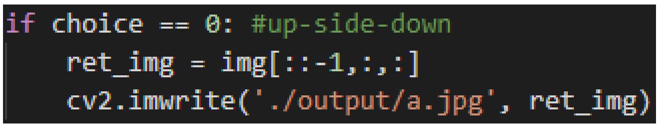
III. Problem-Solving Process

First, I use the 'cv2.imread()' method to load 'lena.bmp' from the file.

Second, I process image with Numpy.

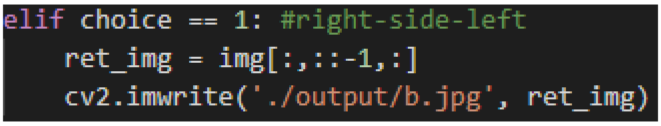
Finally, I save image by the 'cv2.imwirte()' method to the output file.

a. up-side-down lena.bmp



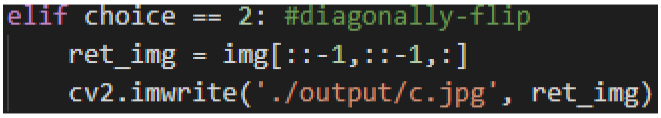
```
if choice == 0: #up-side-down
    ret_img = img[::-1,:,:]
    cv2.imwrite('./output/a.jpg', ret_img)
```

b. right-side-left lena.bmp



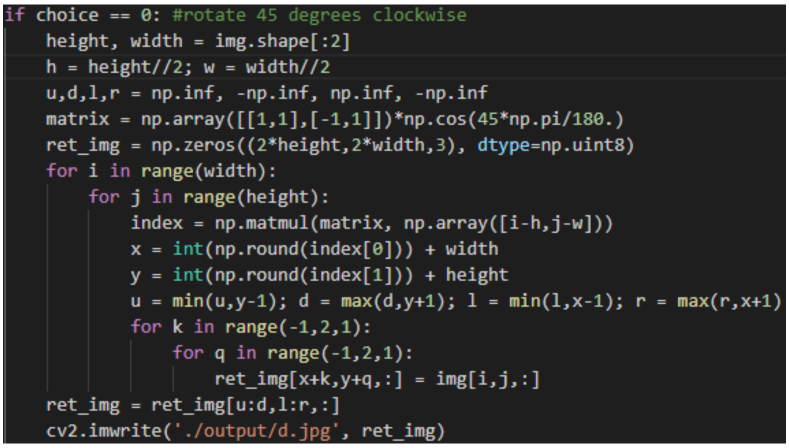
```
elif choice == 1: #right-side-left
    ret_img = img[:,::-1,:]
    cv2.imwrite('./output/b.jpg', ret_img)
```

c. diagonally flip lena.bmp



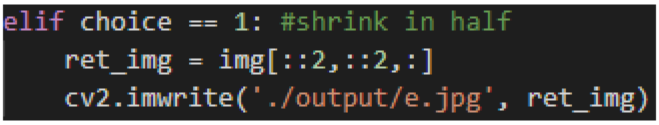
```
elif choice == 2: #diagonally-flip
    ret_img = img[::-1,::-1,:]
    cv2.imwrite('./output/c.jpg', ret_img)
```

d. rotate lena.bmp 45 degrees clockwise



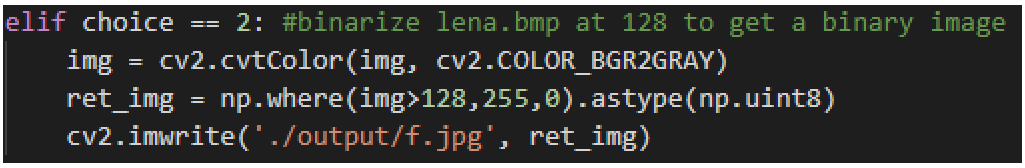
```
if choice == 0: #rotate 45 degrees clockwise
    height, width = img.shape[:2]
    h = height//2; w = width//2
    u,d,l,r = np.inf, -np.inf, np.inf, -np.inf
    matrix = np.array([[1,1],[-1,1]])*np.cos(45*np.pi/180.)
    ret_img = np.zeros((2*height,2*width,3), dtype=np.uint8)
    for i in range(width):
        for j in range(height):
            index = np.matmul(matrix, np.array([i-h,j-w]))
            x = int(np.round(index[0])) + width
            y = int(np.round(index[1])) + height
            u = min(u,y-1); d = max(d,y+1); l = min(l,x-1); r = max(r,x+1)
            for k in range(-1,2,1):
                for q in range(-1,2,1):
                    ret_img[x+k,y+q,:] = img[i,j,:]
    ret_img = ret_img[u:d,l:r,:]
    cv2.imwrite('./output/d.jpg', ret_img)
```

e. shrink lena.bmp in half



```
elif choice == 1: #shrink in half
    ret_img = img[::2,::2,:]
    cv2.imwrite('./output/e.jpg', ret_img)
```

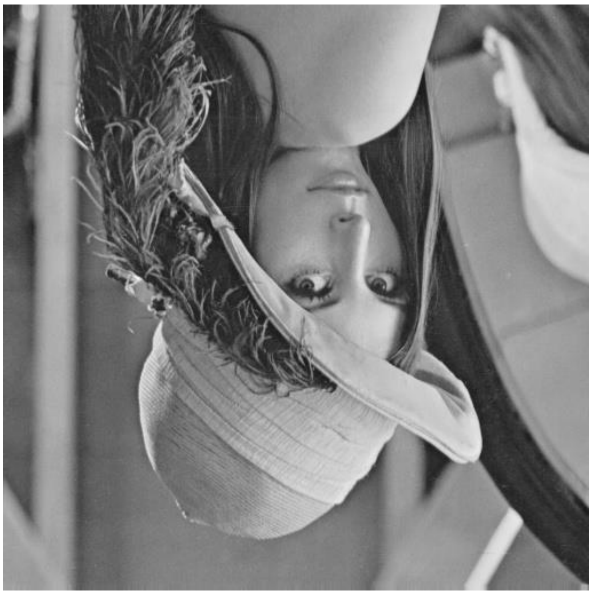
f. binarize lena.bmp at 128 to get a binary image



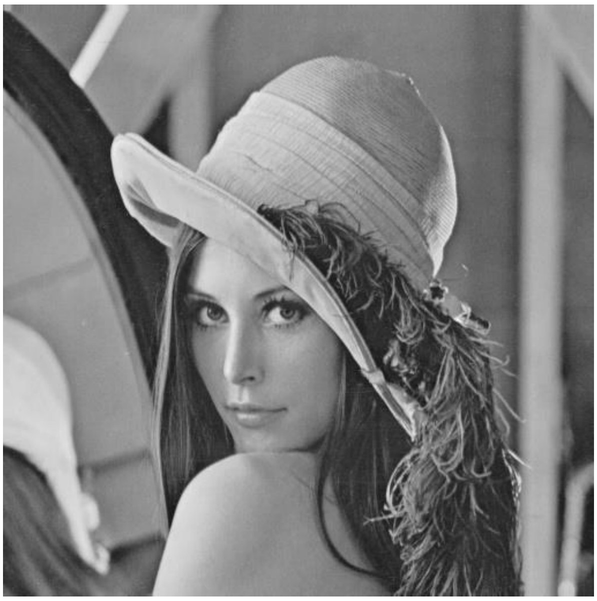
```
elif choice == 2: #binarize lena.bmp at 128 to get a binary image
    img = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
    ret_img = np.where(img>128,255,0).astype(np.uint8)
    cv2.imwrite('./output/f.jpg', ret_img)
```

IV. Results

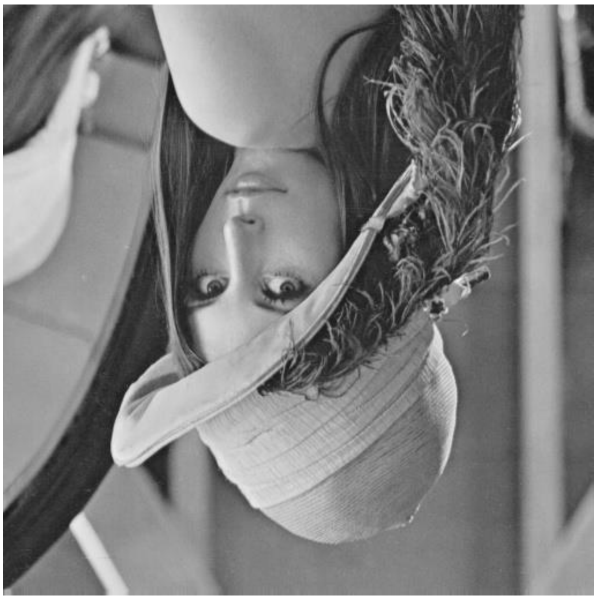
a. up-side-down lena.bmp



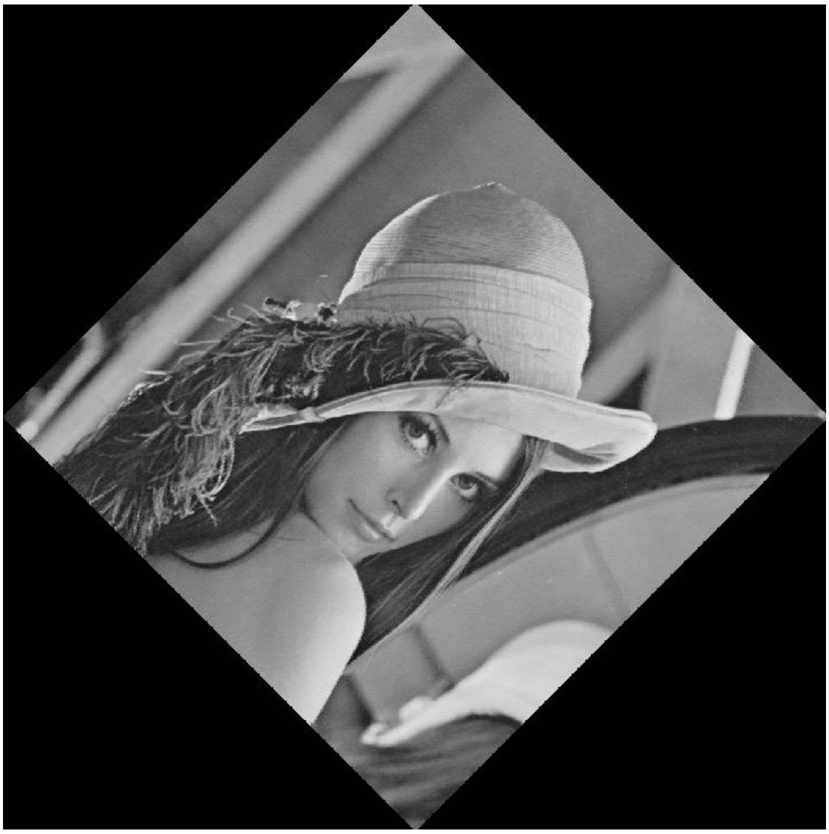
b. right-side-left lena.bmp



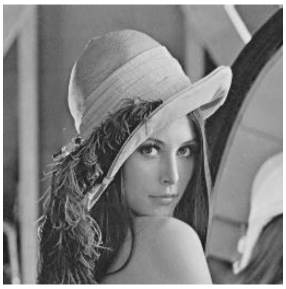
c. diagonally flip lena.bmp



d. rotate lena.bmp 45 degrees clockwise



e. shrink lena.bmp in half



f. binarize lena.bmp at 128 to get a binary image