HW1

- **A discription of your homework**

  Programming language used: Python 2.7

  Library used: Numpy, PIL, Scipy.misc

- **Your parameters**

  tem: 用於儲存每個像素的灰階數值 0~255

  i: row

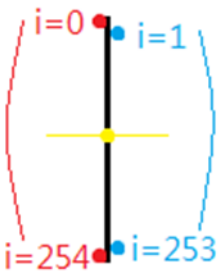
  j: column

- **The algorithm you used**

1.

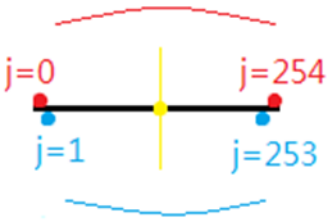
(a) upside-down lena.im

i 以照片中心水平線為對稱軸，對稱軸上下兩側像素互換，j 不變



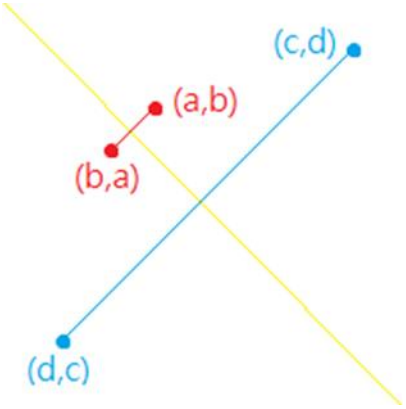
(b) right-side-left lena.im

j 以照片中心鉛垂線為對稱軸，對稱軸左右兩側像素互換，i 不變



(c) diagonally mirrored lena.im

以斜對角線為對稱軸，對稱軸兩側像素互換 → i, j 互換



2.

(a) rotate lena.im 45 degrees clockwise

使用 rotate 函式

原理：旋轉矩陣

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos(45°) & -\sin(45°) \\ \sin(45°) & \cos(45°) \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

$$\rightarrow \begin{bmatrix} j' \\ i' \end{bmatrix} = \begin{bmatrix} \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} j \\ i \end{bmatrix}$$

(b) shrink lena.im in half

使用 resize 函式

長寬各為原本的 0.5

(c) binarize lena.im at 128 to get a binary image

Threshold: 128

灰階像素 > 128 → 白(1)

灰階像素 < 128 → 黑(0)

- **Principal code fragment**

```
def up_down(x):
    tem = np.zeros(x.shape) # array full of zeros
    for i in range(x.shape[0]): # shape[0]: row
        for j in range(x.shape[1]): # shape[1]: column
            tem[i][j] = x[x.shape[0]-i-1][j]
    return tem

def right_left(x):
    tem = np.zeros(x.shape)
    for i in range(x.shape[0]):
        for j in range(x.shape[1]):
            tem[i][j] = x[i][x.shape[0]-j-1]
    return tem

def diag(x):
    tem = np.zeros(x.shape)
    for i in range(x.shape[0]):
        for j in range(x.shape[1]):
            tem[i][j] = x[j][i]
    return tem

def resize(x):
    width = int(x.size[1]*0.5)
    height = int(x.size[0]*0.5)
    nim = x.resize((width,height),Image.BILINEAR)
    return nim

def rotate(x):
    nim = x.rotate(-45) # counterclockwise
    return nim

def bi(x):
    tem = np.zeros(x.shape)
    for i in range(x.shape[0]):
        for j in range(x.shape[1]):
            if x[i][j]<128: # threshold 128
                tem[i][j] = 0
            else:
                tem[i][j] = 1
    return tem
```

- **Resulting images**

1.

(a) upside-down lena.im

Result by using Python

(b) right-side-left lena.im

Result by using Python

(c) diagonally mirrored lena.im

Result by using Python

2.

(a) rotate lena.im 45 degrees clockwise

Result by using Python

Since the the corners of “lena” by using Python is cut off, result by using the rotation function in Word2013 is presented here as well. The image is shrinked slightly here so it does not exceed the width of the page.

(b) shrink lena.im in half

Result by using Python

(c) binarize lena.im at 128 to get a binary image

Result by using Python