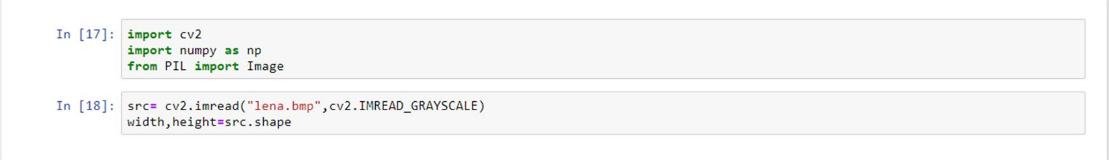
Computer Vision HW#1

R08942125 廖克允

Part 1.

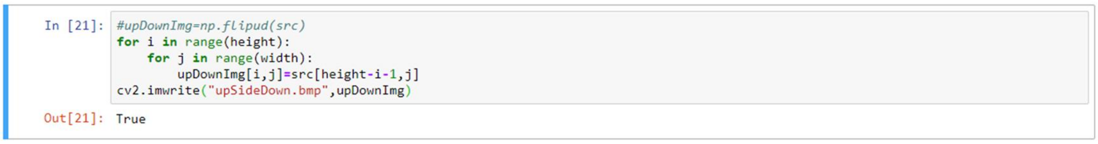
Import cv2 and numpy for basic image I/O



```
In [17]: import cv2
         import numpy as np
         from PIL import Image

In [18]: src= cv2.imread("lena.bmp",cv2.IMREAD_GRAYSCALE)
         width,height=src.shape
```

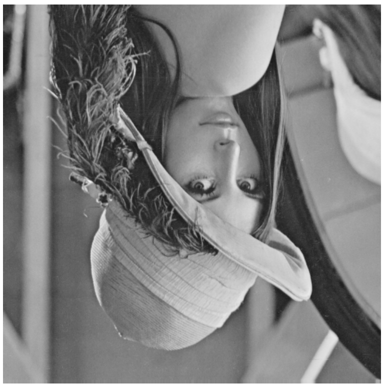
(a) upside-down lena.bmp

upside-down lena



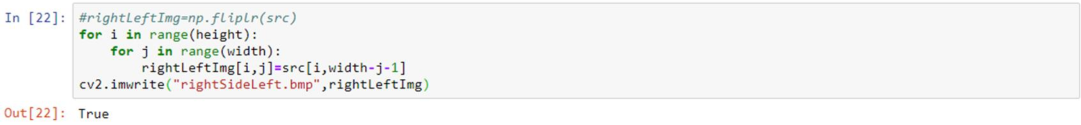
```
In [21]: #upDownImg=np.flipud(src)
         for i in range(height):
             for j in range(width):
                 upDownImg[i,j]=src[height-i-1,j]
         cv2.imwrite("upSideDown.bmp",upDownImg)

Out[21]: True
```

Result:



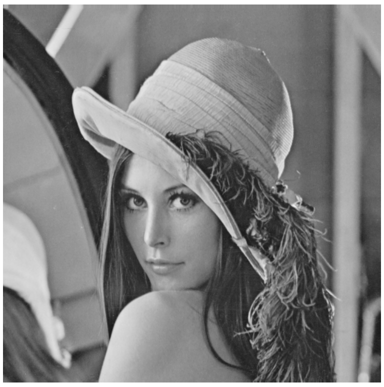
(b) right-side-left lena.bmp

right-side-left lena

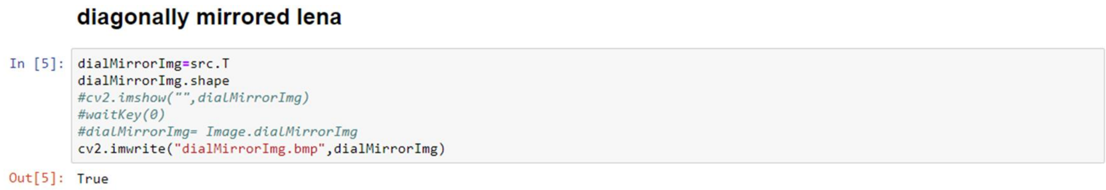


```
In [22]: #rightLeftImg=np.fliplr(src)
         for i in range(height):
             for j in range(width):
                 rightLeftImg[i,j]=src[i,width-j-1]
         cv2.imwrite("rightSideLeft.bmp",rightLeftImg)

Out[22]: True
```

Result:



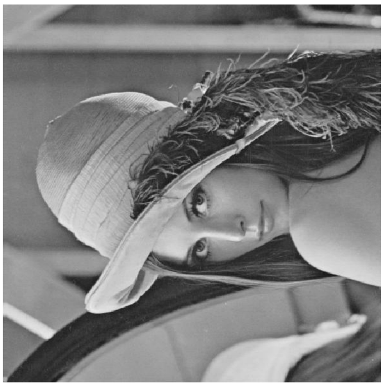
(c) diagonally mirrored lena.bmp



diagonally mirrored lena

```
In [5]: dialMirrorImg=src.T
        dialMirrorImg.shape
        #cv2.imshow("",dialMirrorImg)
        #waitKey(0)
        #dialMirrorImg= Image.dialMirrorImg
        cv2.imwrite("dialMirrorImg.bmp",dialMirrorImg)

Out[5]: True
```

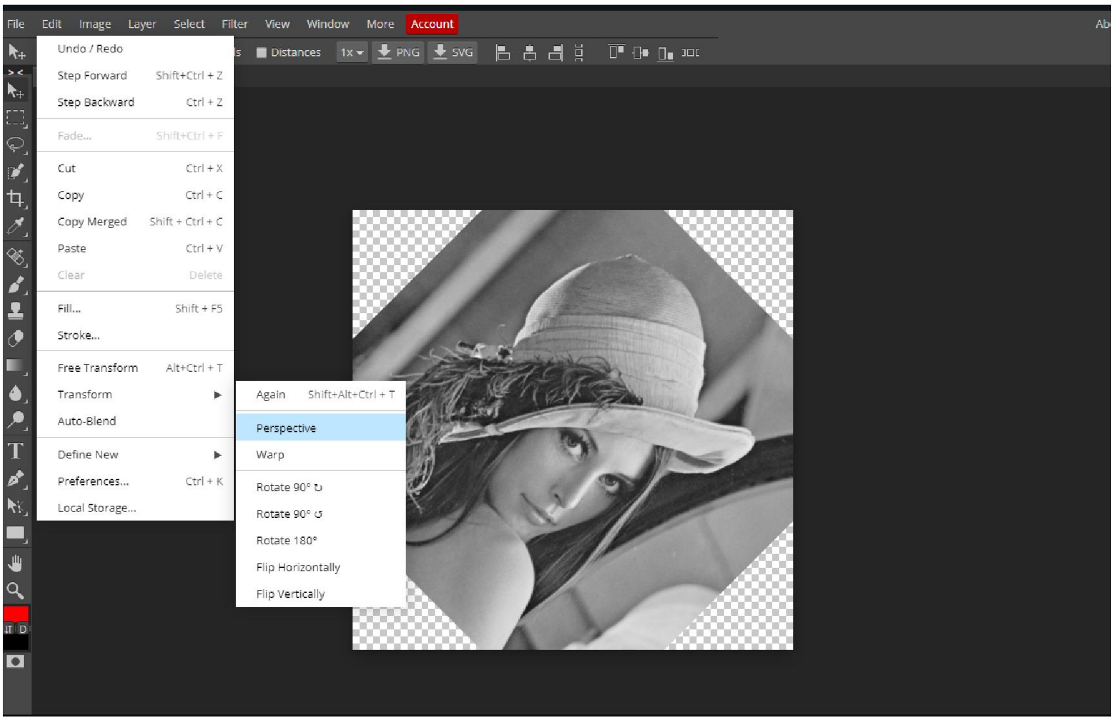
Result:



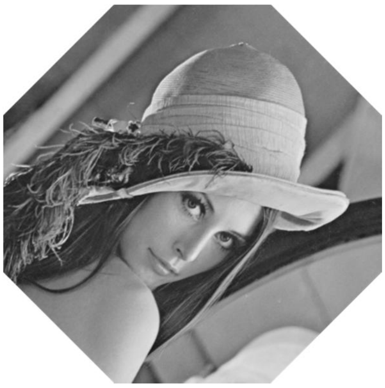
Part 2.

Use Photopea to get the requirement

(d) rotate lena.bmp 45 degrees clockwise



Result:



(e) shrink lena.bmp in half

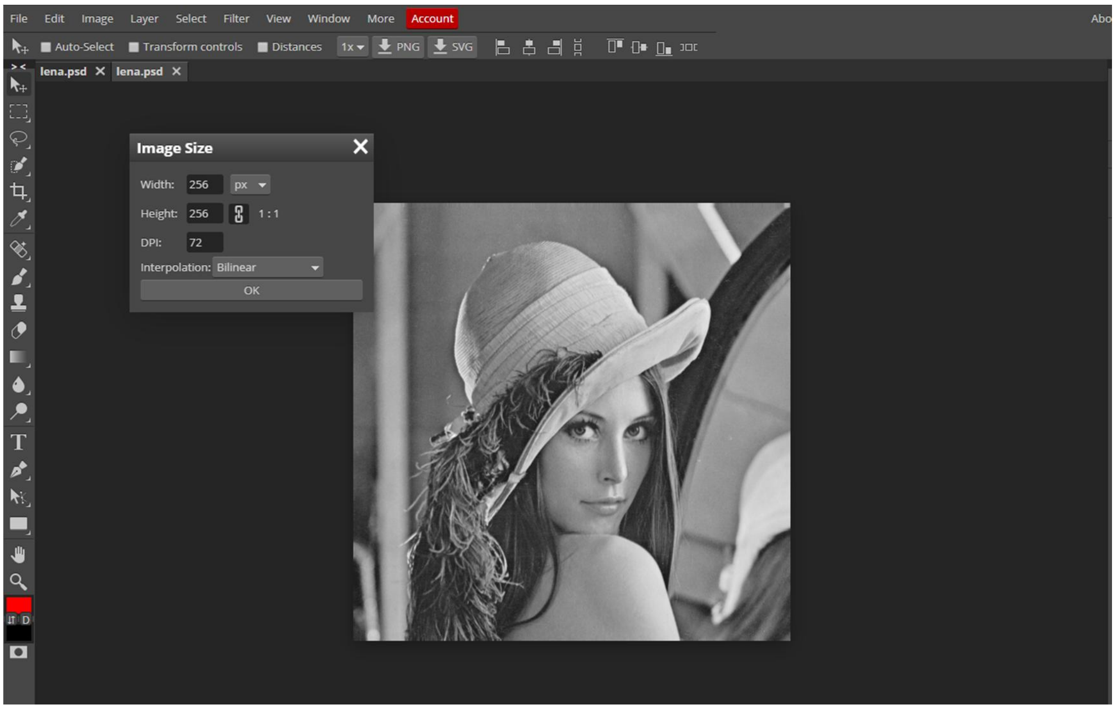


Image Size

Width: 256 px

Height: 256 1 : 1

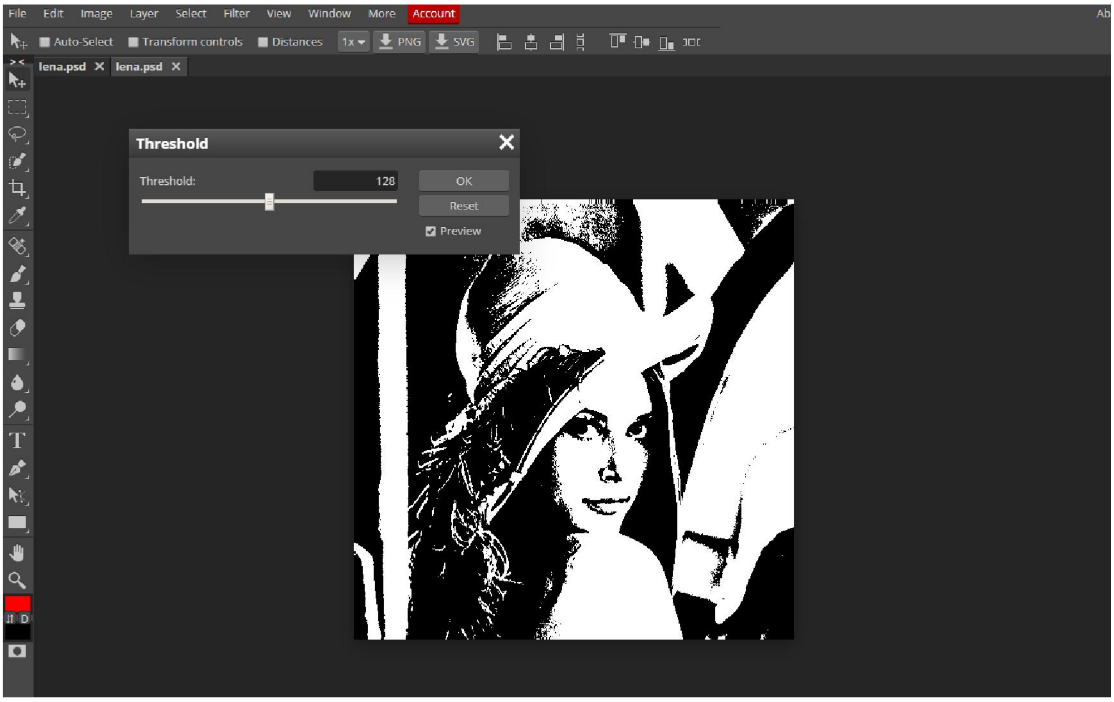
DPI: 72

Interpolation: Bilinear

OK

Result:

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



Threshold

Threshold: 128

☑ Preview

Result: