

電腦視覺 HW1

B05902011 資工四 梁振寧

Part 1.

source code : src/img_process.py

using language : python

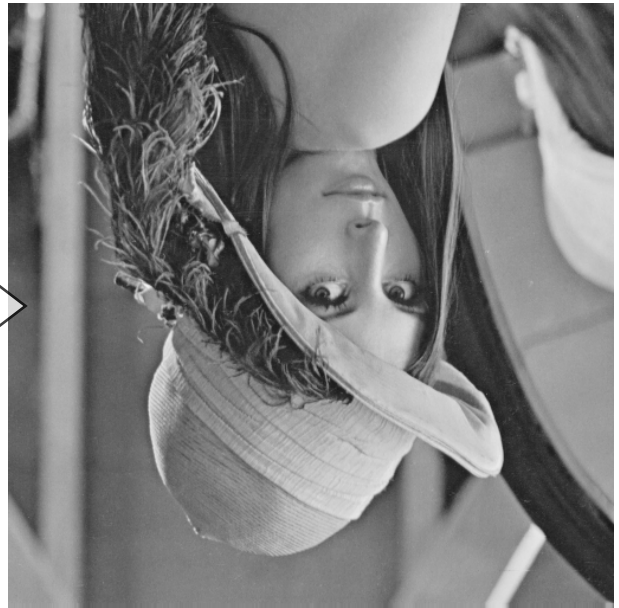
using material : numpy, PIL

* following images are resized to fit in the page

(a). upside-down



src/lena.bmp



src/ud_im.bmp

(b). right-side-left



src/lena.bmp



src/lr_im.bmp

(c). diagonally mirrored



src/lena.bmp

src/d_im.bmp

For (a) and (b), the process was simply done by reversing the index of the array. (line 19 to 20)
For (c), I used a 2-layered-for-loop to traverse through the 2d-array, and swap the values diagonally. (line 22 to 27)

Reference:

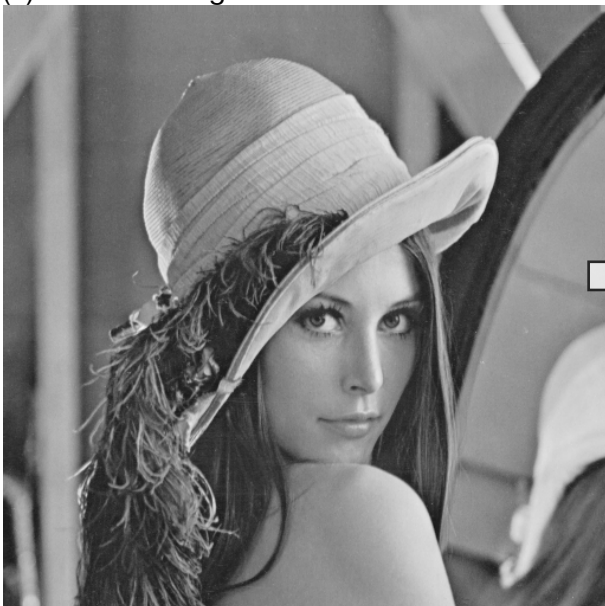
<https://stackoverflow.com/questions/9154120/how-can-i-flip-an-image-along-the-vertical-axis-with-python>

<https://medium.com/jameslearningnote/資料分析-機器學習-第2-5講-資料視覺化-matplotlib-seaborn-plotly-75cd353d6d3f>

Part 2.

source code (for (f).) : src/img_binarize.py
using language (for (f).) : python
using material (for (f).) : numpy, PIL
using software (for (d). and (e).) : MediBangPaintPro
* following images are resized to fit in the page

(d). rotate 45 degrees



src/lena.bmp

src/rotated_lena.bmp

(d). shrink in half



src/lena.bmp



src/halfed_lena.bmp

(f). binarize at 128



src/lena.bmp



src/binarized_lena.bmp

The algorithm I used for (f) was found online, which was originally written in Matlab language. I rewrote it into python language. Also, some minor arguments were configured to fit in the required task.

Reference:

<https://www.geeksforgeeks.org/matlab-converting-a-grayscale-image-to-binary-image-using-thresholding/>