Basic Image Manipulation

資工所碩一 R07922164 林群崴

Part1

(a) upside-down	
(b) right-side-left	
(c) diagonally mirrored	

The following program generates the upside-down, right-side-left and diagonally mirrored lena when execute once.

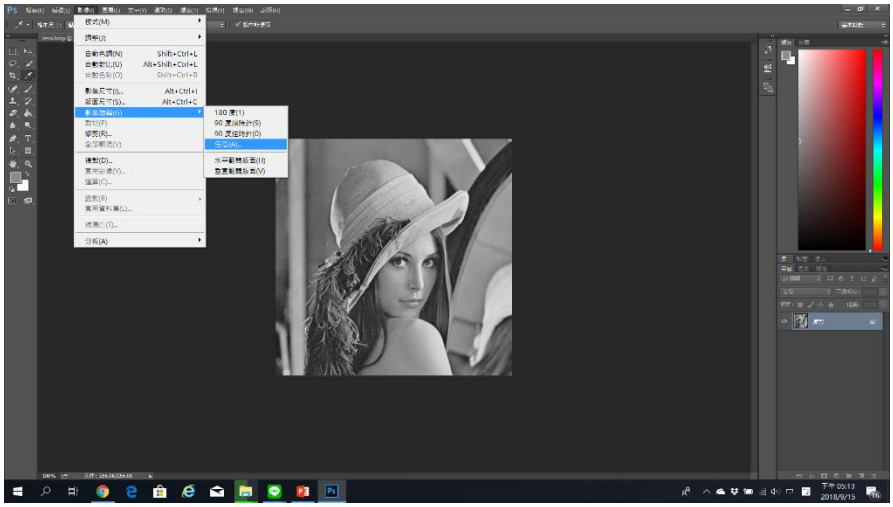
```
/*********************************************************
Name : Qun-Wei, Lin
Date : 2018/9/17
Project : Computer Vision HW1
*********************************************************/
#include "pch.h"
#include <iostream>
#include <opencv2/opencv.hpp>

using namespace cv;

int main(int argc, char const *argv[]) {
	//讀圖，以灰階讀入
	Mat lena = imread("lena.BMP", CV_LOAD_IMAGE_GRAYSCALE);

	//複製圖片準備處理
	Mat upside_down, right_side_left, diagonally_mirrored;
	upside_down = lena.clone();
	right_side_left = lena.clone();
	diagonally_mirrored = lena.clone();

	//定義圖片寬度
	int widthLimit = lena.channels() * lena.cols;

	//上下顛倒
	uchar temp;
	for (int height = 0; height <= upside_down.rows / 2; height++) {
		//至多處理到高度的一半
		for (int width = 0; width < widthLimit; width++) {
			temp = upside_down.at<uchar>(height, width);
			upside_down.at<uchar>(height, width) = upside_down.at<uchar>(upside_down.rows - height - 1, width);
			upside_down.at<uchar>(upside_down.rows - height - 1, width) = temp;
			//利用暫時的變數temp進行像素的swap
		}
	}
```

```
    //左右顛倒
    temp = 0;
    for (int height = 0; height < right_side_left.rows; height++) {
        for (int width = 0; width <= widthLimit / 2; width++) {
            //至多處理到寬度的一半
            temp = right_side_left.at<uchar>(height, width);
            right_side_left.at<uchar>(height, width) = right_side_left.at<uchar>(height, right_side_left.cols -
width - 1);
            right_side_left.at<uchar>(height, right_side_left.cols - width - 1) = temp;
            //利用暫時的變數temp進行像素的swap
        }
    }

    //對角線鏡射
    temp = 0;
    for (int height = 0; height < diagonally_mirrored.rows; height++) {
        for (int width = height; width < widthLimit; width++) {
            temp = diagonally_mirrored.at<uchar>(height, width);
            diagonally_mirrored.at<uchar>(height, width) = diagonally_mirrored.at<uchar>(width, height);
            diagonally_mirrored.at<uchar>(width, height) = temp;
            //利用暫時的變數temp進行像素的swap
        }
    }

    //存圖
    imwrite("upside_down.bmp", upside_down);
    imwrite("right_side_left.bmp", right_side_left);
    imwrite("diagonally_mirrored.bmp", diagonally_mirrored);

    exit(0);
}
```

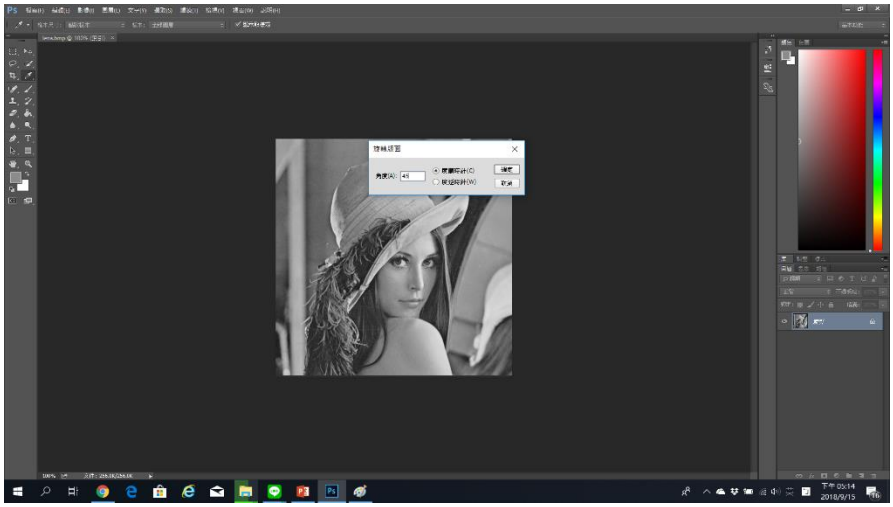
Part2

(a) rotate lena.im 45 degrees clockwise

Step1:影像→影像旋轉→任意

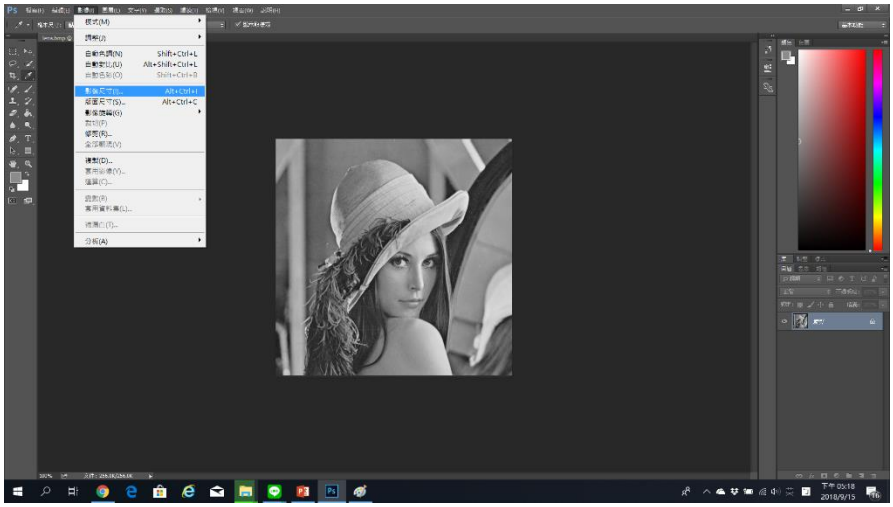


Step2:角度輸入 45°

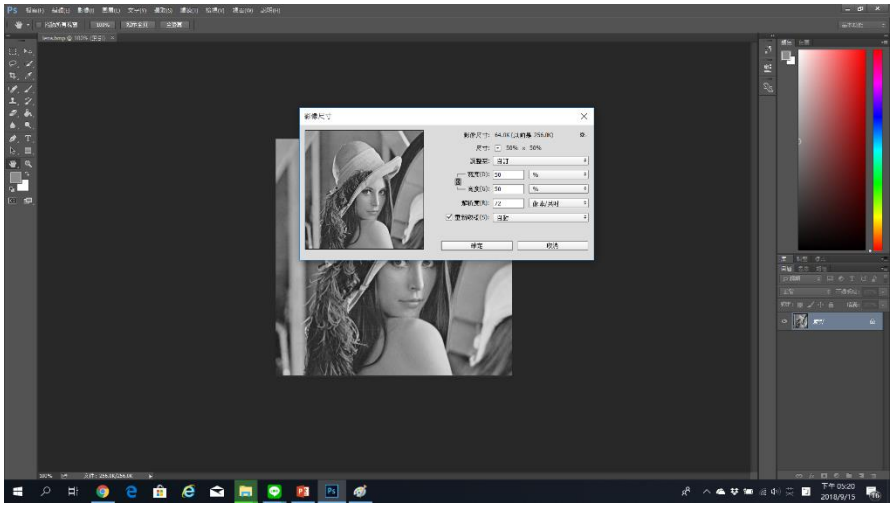


(b) shrink lena.im in half

Step1:影像→影像尺寸

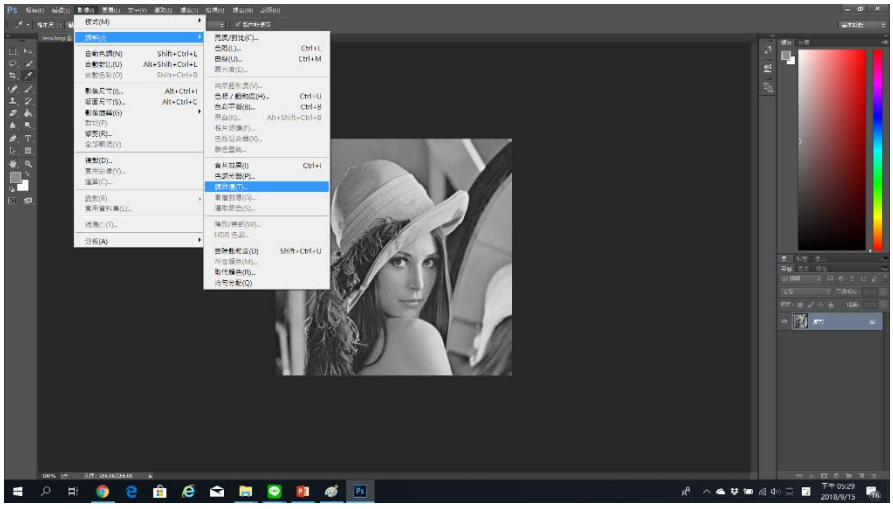


Step2:尺寸選擇50% × 50%



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

Step1:影像→調整→臨界值(Threshold)



Step2:域值輸入 128