Principles and Applications of Digital Image Processing

Homework 1 report

### Part 1: (50%) Histogram of an Image

Design a software program to read the special .64 image file described in our lecture. Translate the .64 text file into a 64x64 image with 32 gray levels and store the data in a 2-dimensional array. Process the image array to obtain the histogram of the image.

Test your program with the following .64 files and plot the histogram of each image.

LISA.64,   LINCOLN.64,   JET.64,   LIBERTY.64.

You may plot the image histogram directly in your software program or plot the histogram with any plotting software such as EXCEL or MATLAB. Designing a function in your program to display the image on the screen is encouraged.

這一題是要將範例中的.64 的文字檔案讀進程式中,可以使用 C/C++ 中的 fopen 或是 fstream 來完成,但這門課所使用的 framework 是 Qt,所以要秉持" When in Rome, do as the Romans do"的精神使用 Qt 中的 QFile 來開檔,再用 QTextStream 唯讀的方式來讀文字進程式,因為原始的影像有一點黑,所以我將所有的像素乘上 8
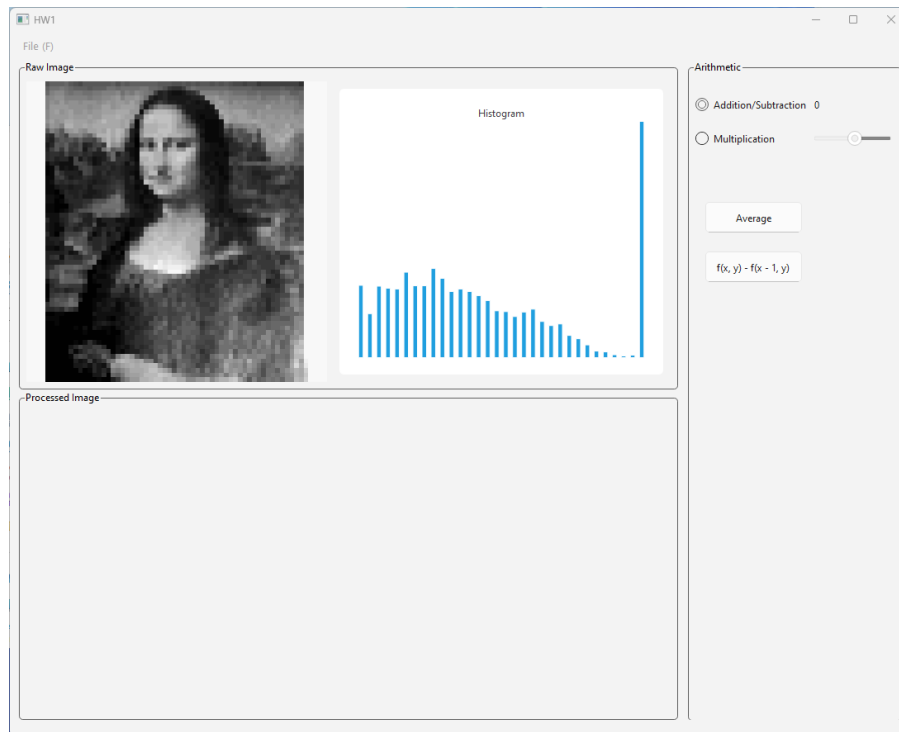
**Part 2: (50%) Arithmetic Operations of an Image Array**

Design a software program that will perform the basic tasks of arithmetic operations on an image or two images. Use the .64 image for this program. The assigned image processing operations are as follows:

1. Add or subtract a constant value to each pixel in the image.
2. Multiply a constant to each pixel in the image.
3. Create a new image which is the average image of two input images.
4. Create a new image $g(x,y)$ in which the value of each pixel is determined by calculating the pixel values of the input image $f(x,y)$ using the following equation:
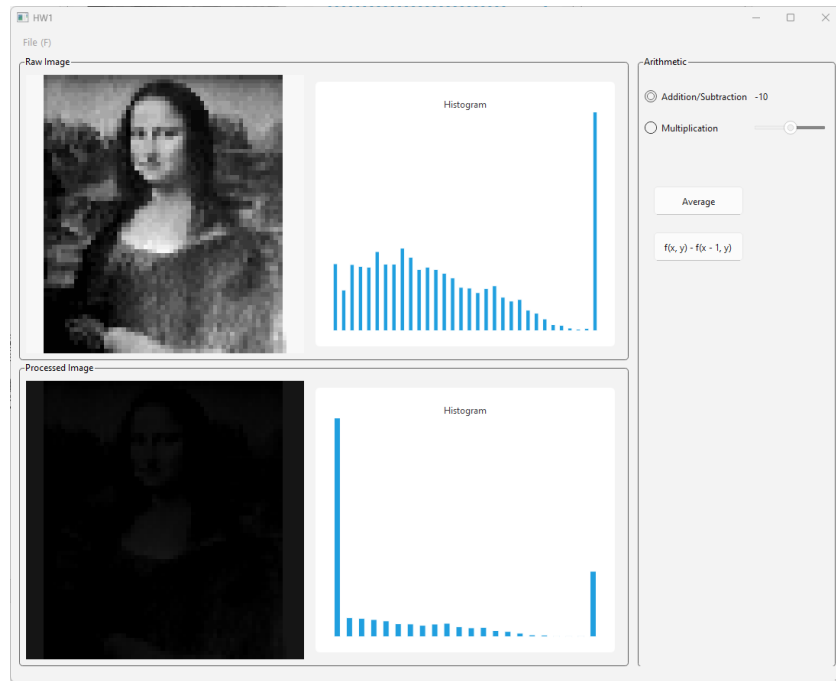
$$g(x,y) = f(x,y) - f(x-1,y)$$

Calculate the histograms of the processed images from the above arithmetic operations and compare them with the histograms of the original image. Briefly discuss your results.
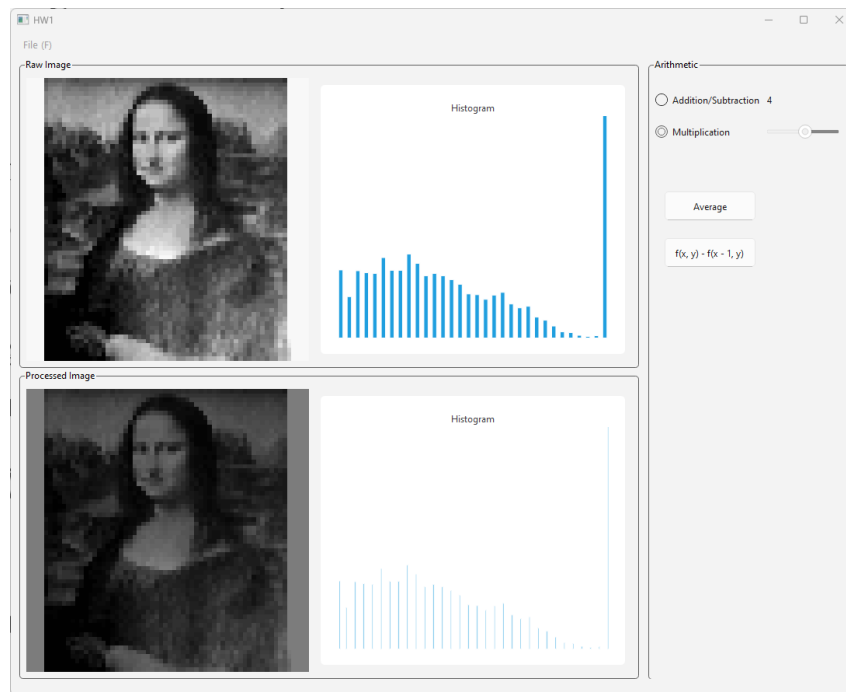
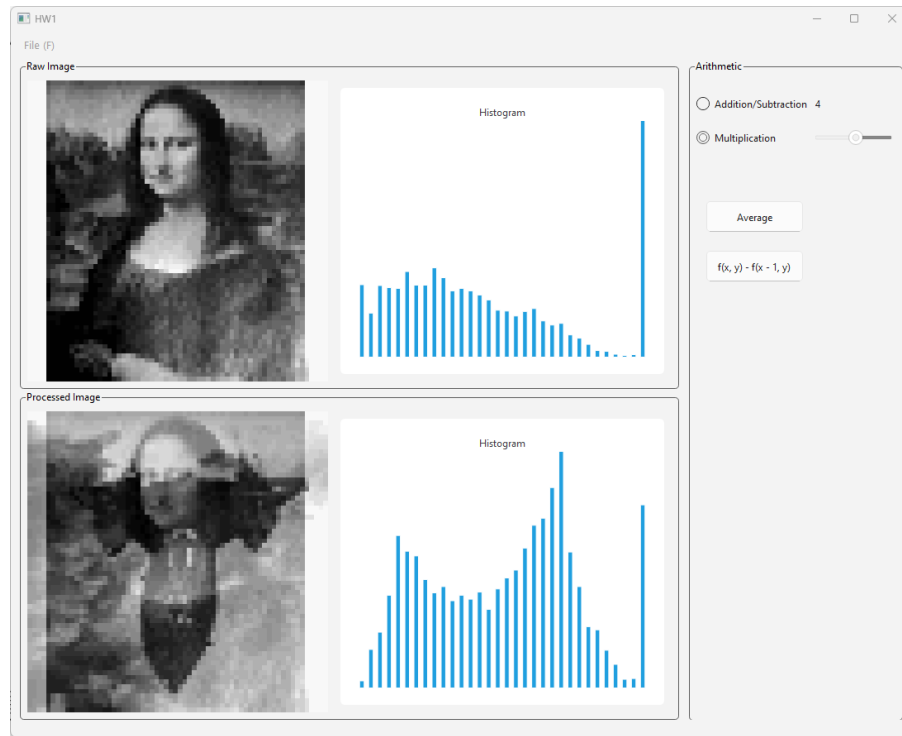這四小題是要實作出加法和減法的運，為了實作上的方便我都是用一維的 QVector 來存像素的，加法和減法的結果是將影像的亮度上增減
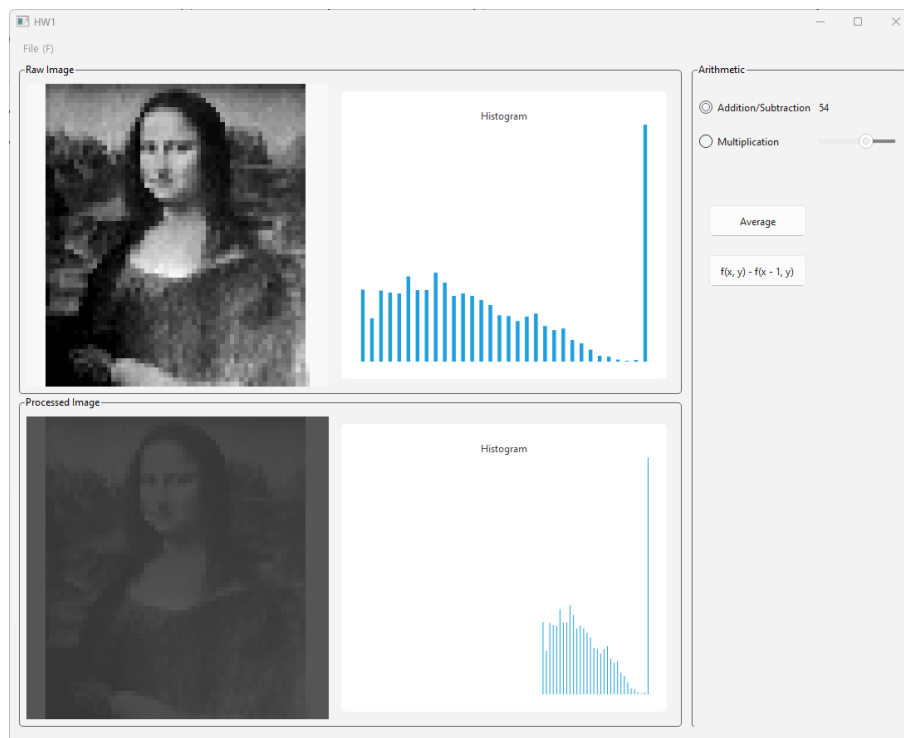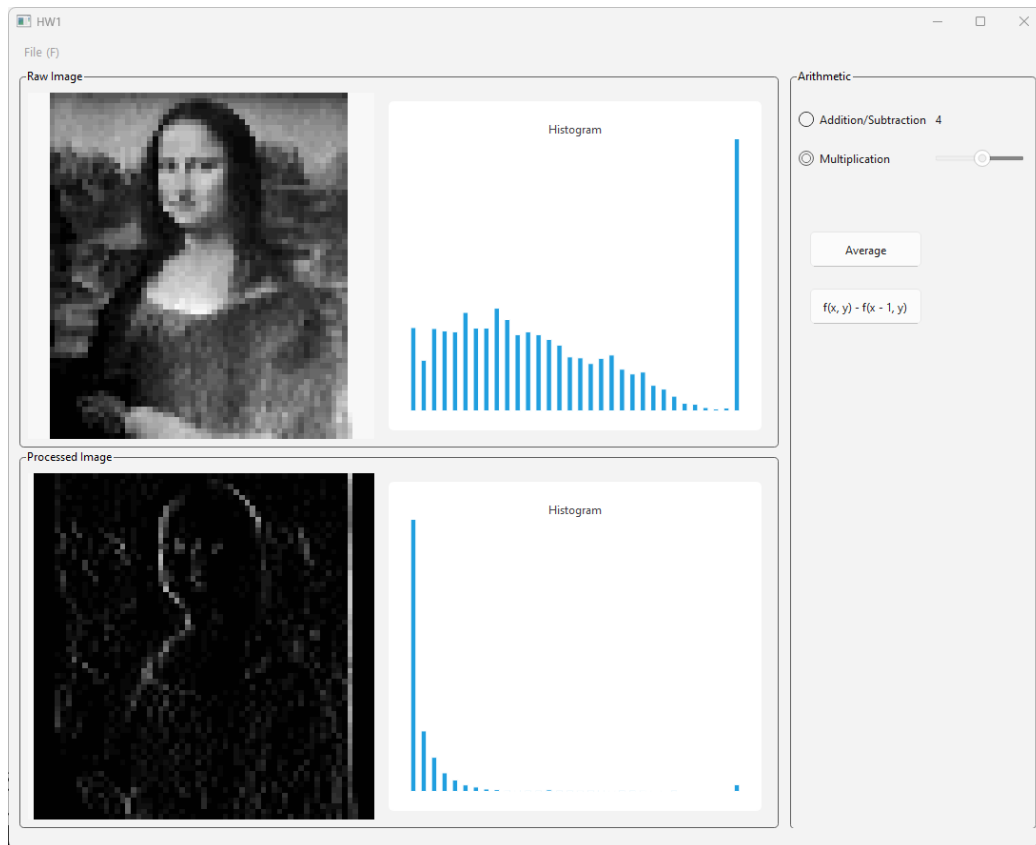

如上圖，不難看出加法的 histogram 為原始的向右偏了一些，整體的像素的分佈是不變的

減法的較果出來看出它的 0 像素會愈來愈來高

乘法運算上以我的觀點來看也是將影像變亮，histogram 可以看出有一點分散，但形狀上與原始的非常相似

這一個是將兩張影像平均，我發現中間有一點多，可能是常態分佈的原因所導致，但影像可以看出就是兩張影像疊加在一起

使用$g(x, y) = f(x, y) - f(x - 1, y)$這一個方程式可以看出影像的邊緣的部分會比較明顯，整體可以看出這張影像的輪廓