高等數位影像處理

作業#(2)

姓名：闕楷宸
學號：114318047
指導老師：黃正民

作業說明:

作業架構:

```
.
├── CMakeLists.txt
├── HW2.cpp
├── lena256.raw
├── lena512.raw
├── lena_eye50.raw
└── output_images
    ├── lena_eye50_out.raw
    ├── task1_1_blur.raw
    ├── task1_1_no_blur.raw
    ├── task1_2_bilinear.raw
    └── task1_2_nearest_neighbor.raw

1 directory, 10 files
```

主程式HW2.cpp

```
================ Results Menu ================
 1) 1-1
 2) 1-2
 3) 2-1
 4) 2-2
 0) Exit
Enter the question number: 4
Image Quality Metrics
MSE= 36.6012
PSNR= 32.4959 dB
```

輸入0~4即可輸出結果

建置執行(window,linux)

Linux:

1.使用opencv 4.5.4(sudo apt install libopencv-dev)

2.建置與執行:

```
cd HW1_114318047/
cmake -S . -B build
cmake --build build -j
./build/HW1
```

Window:

```
cd window_user
./HW1.exe
```

Window 建置:

1:Cmake:

winget install Kitware.CMake

2:OpenCV for window:

C:\opencv\build\x64\vc16\bin(make sure DLLs are in this dir)

3:Confirm CMake config file exists:

C:\opencv\build\x64\vc16\lib\OpenCVConfig.cmake

4:Build and run

cd "C:\HW1_114318047"

5:Configure:

cmake -S . -B build -G "Visual Studio 17 2022" -A x64 -DOpenCV_DIR="C:\opencv\build\x64\vc16\lib"

6:Produce exe:

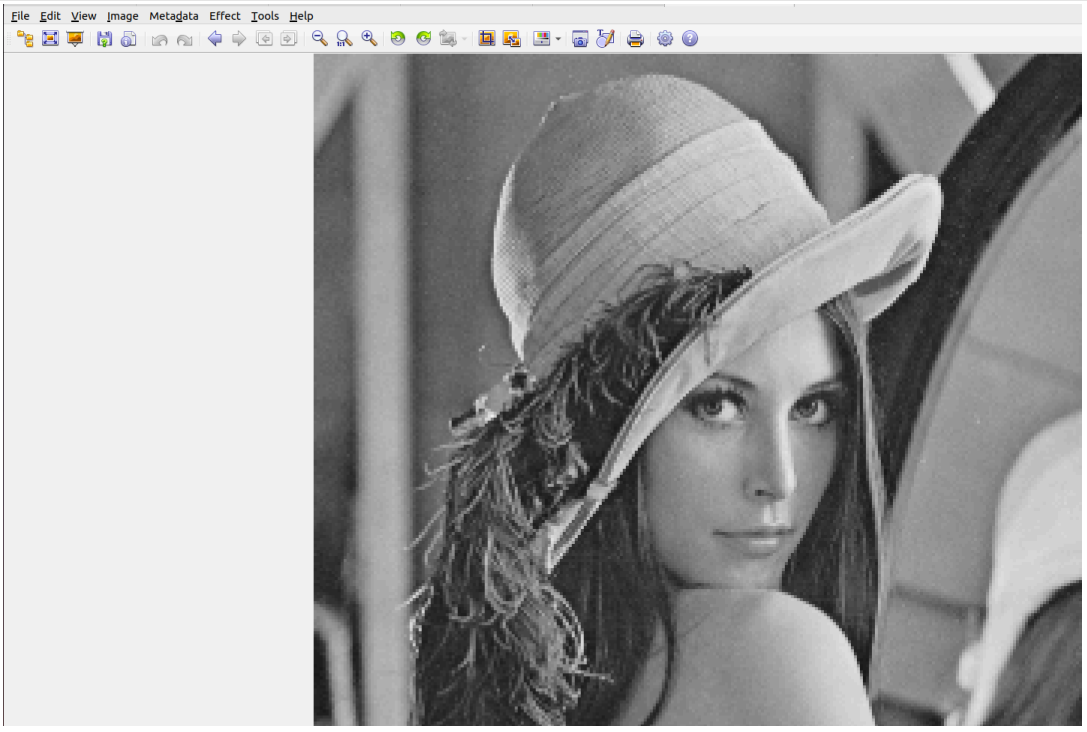
cmake --build .\build --config Release

7:Run exe

.\build\Release\HW1.exe

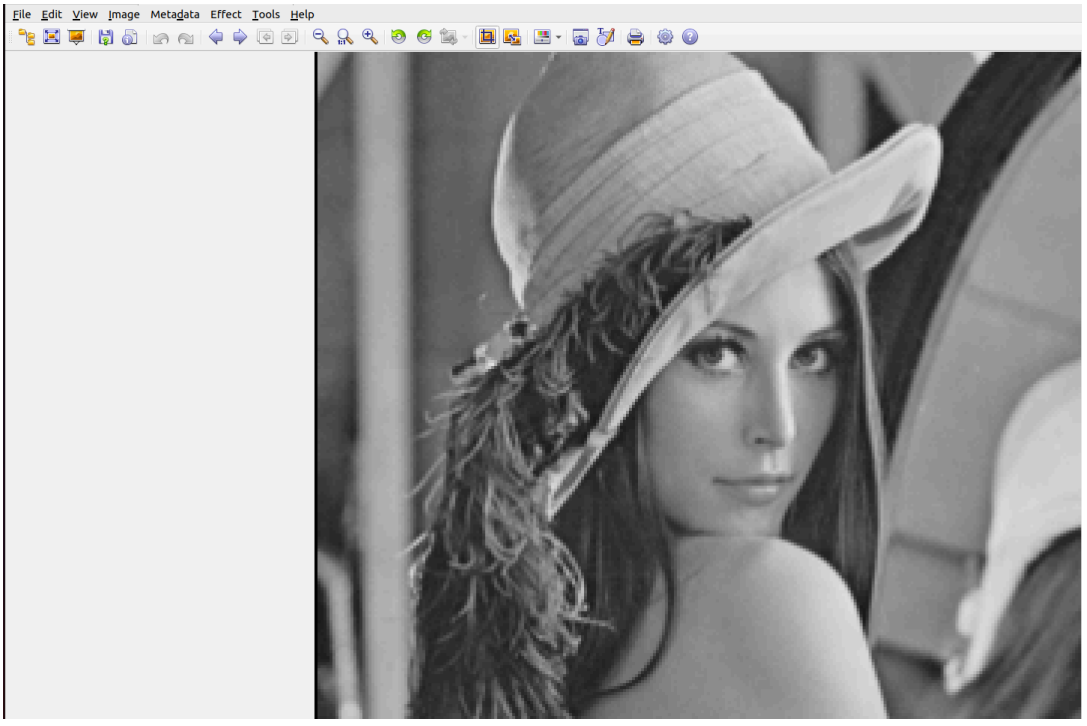
1-1:Zooming and Shrinking

Shrinking tha image lena512.raw with ratio 1:2 using row-column deletion. Compare the output with lena256.raw.Check and discuss your result with ot without blurring before shrinking(use XnView to blur the image yourself



Size 65.5 kB (65,536 bytes)

未模糊(task1.raw)



模糊處理(task1_1_blur.raw)

Discussion(1)

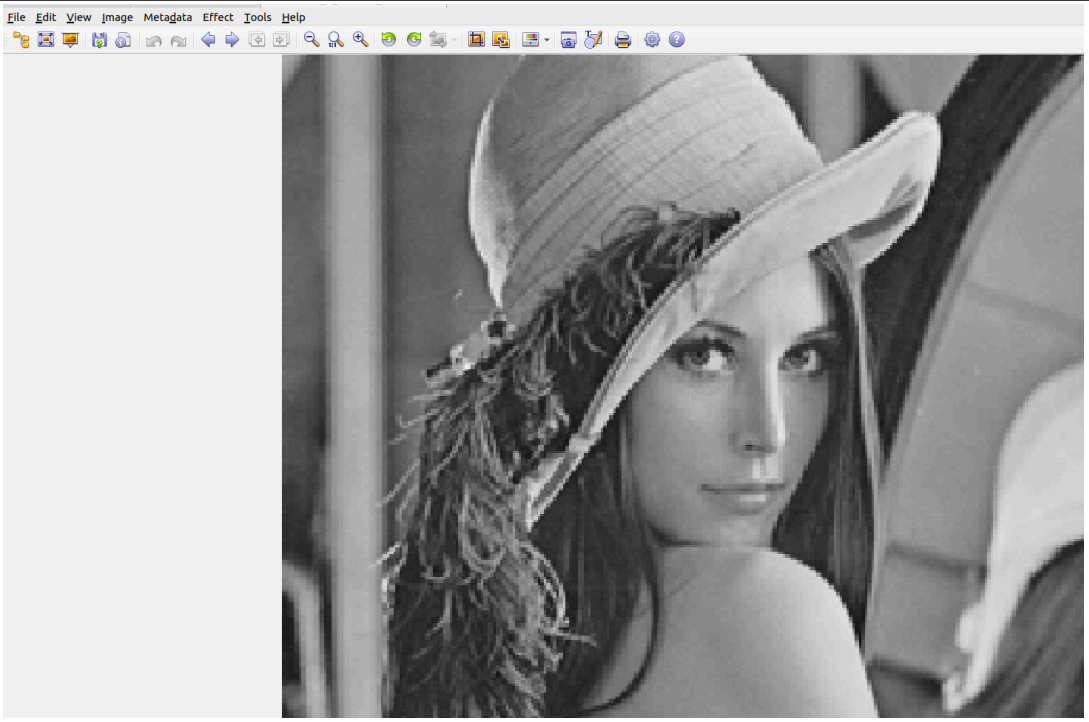
上圖為有無經過模糊處理的結果
(左為未經過模糊的輸出圖像，右為經過模糊處理的輸出圖像)

此題有無模糊處理的圖像沒有太大差異的原因有幾個，一個是原圖lena512.raw本身並沒有太多邊緣特徵明顯的部份，第二個也許是3x3的平均濾波器的強度不夠，也或許是因為是經過row-column deletion無法將模糊的效果體現出來，因為row-column deletion是取一個pixel，但模糊是取周圍pixel。

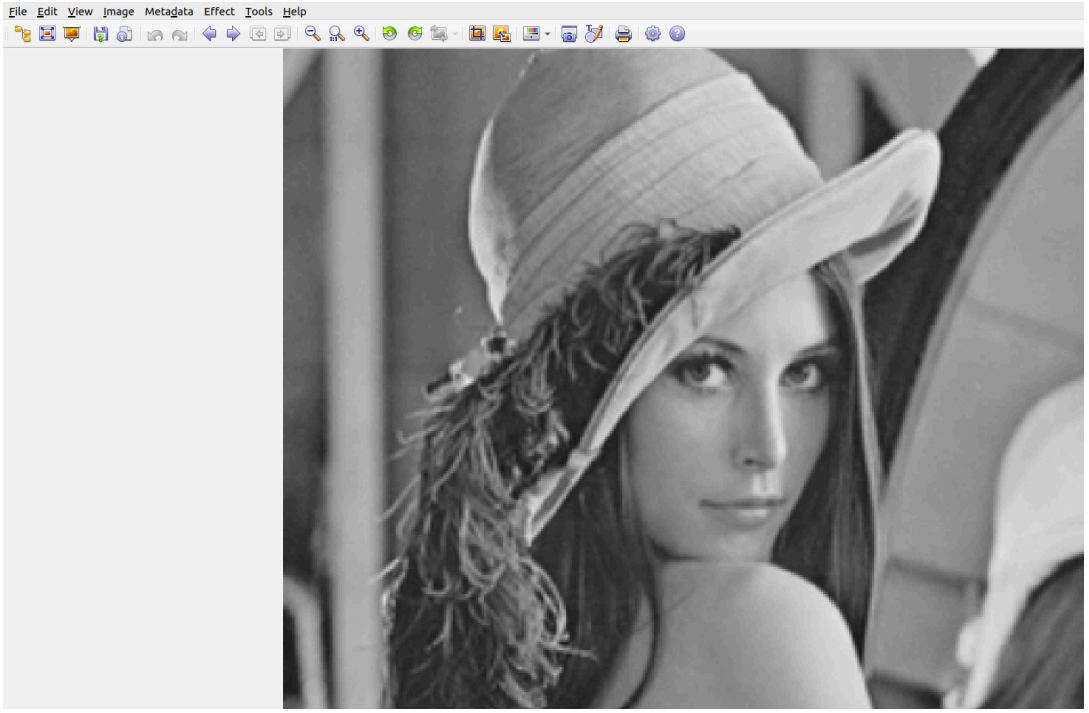
但即便如此，我們還是能從上圖稍微看出經過模糊處理過得圖像在紅色方框下有些微差異，比如說帽子的顆粒感沒這麼重，或是頭髮也比較柔順。

1-2:Zooning and Shrinking

Use nearest neighbor and bilinear algorithms to resize the image lena256.raw into size 384x384.Compare the results from different algorithm

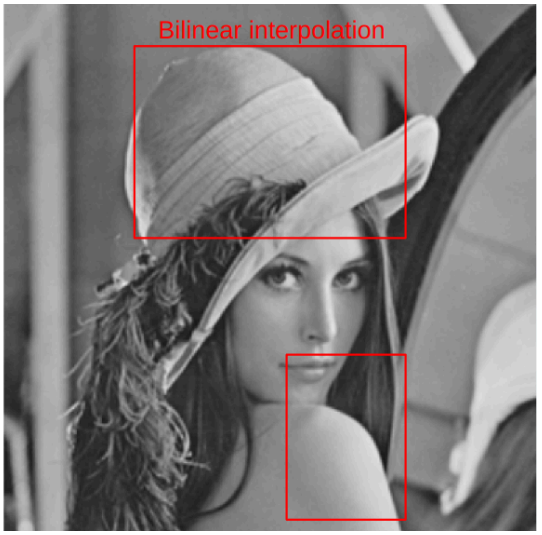


Nearest Neighbor Interpolation(**task1_2_nearest_neighbor.raw**)



Bilinear interpolation(**task1_2_bilinear.raw**)

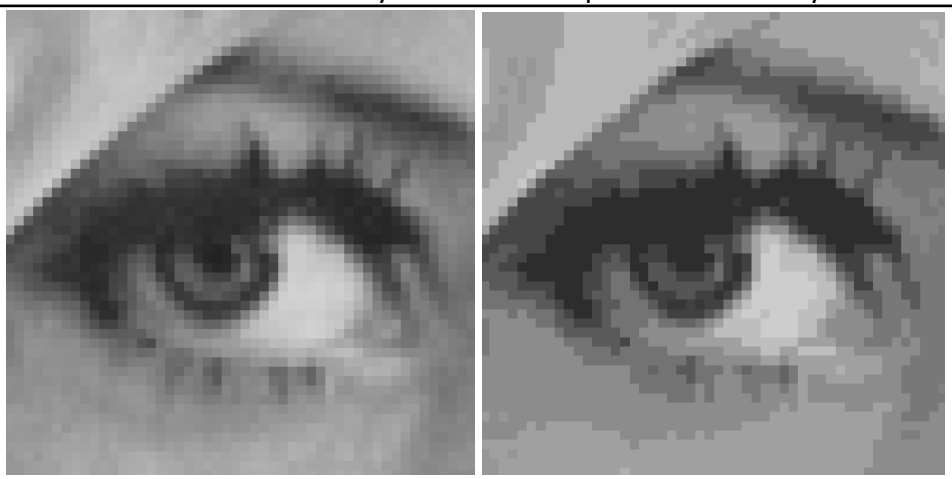
Discussion (2)



此題比較nearest neighbor跟bilinear在做resize之間的差異，我們可以發現bilinear效果勝過nearest neighbor非常多

2-1.Distance and Region

Given the image lena_eye50.raw,the task is to process the image by grouping pixels based on their intensity values and spatial connectivity.First,divide the 8-bit grayscale range [0,255] into eight intervals to from sets Vi for i = 1 …8 as follows:[0,57],[58,81],[82,105],[106,129],[130,153],[154,177],[178,201],[202,255]. Starting from the upper-left corner at coordinate (0,0),iterate through all pixels in lexicographical order(row by row, from left to right).Using 8-neighbor adjacency, group pixels into connected component such that any pixel p and its neighbor q belong to the same component if they are adjacent and fall within the same intensity interval Vi.For each connected component,calculate the mean of all pixel intensities within that component and assign this rounded mean value to all pixels in the group.Finally, save the resulting image as lena_eye50_out.raw and display it.The output image should reflect the smoothed and segmented result based on the defined intensity intervals and spatial connectivity.



Original image(**lena_eye50.raw**)　Output image(**lena_eye50_out.raw**)

Discussion (3)

此題稍加複雜，我使用BFS遍歷所有pixel，並使用兩個判斷式(是否在8-neighbor)與(檢查是否在同個interval中)，同時成立再enqueue，並且使用offset(const int dr[8] = {-1, -1, -1, 0, 0, 1, 1, 1};)與 const int dc[8] = {-1, 0, 1, -1, 1, -1, 0, 1}; 算出neighbor是否超出範圍，並且計算出同個component的mean，並assign給所有pixel。

2-2:

Implement two image quality assessment methods PSNR,MSE using C/C++

$$MSE = \frac{1}{m\,n}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}[I(i,j) - K(i,j)]^2$$

$$PSNR = 10 \cdot \log_{10}\left(\frac{MAX_I^2}{MSE}\right)$$

Calculate the MSE and PSNR between lena_eye50.raw and lena_eye50_out.raw

```
Enter the question number: 4
Image Quality Metrics
MSE= 36.6012
PSNR= 32.4959 dB
```

Discussion (2-2)

此題比較原圖與經過處理的圖的MSE 與PSNR，MSE都是負責檢查兩者的箱四處，MSE越低，代表兩者越相似，PSNR則是越高，代表越好

$$\mathrm{PSNR} = 10\log_{10}\left(\frac{MAX_I^2}{\mathrm{MSE}}\right)$$