# CV Homework 5

R02922124 葉信良

Part 1 Gray-scale Dilation (using lena.bmp)

$ gcc –o hw5\_1\_GrayDil hw5\_1\_GrayDil.cpp

$ ./hw5\_1\_GrayDil

我直接使用c++讀入bmp file lena.bmp，並將header 1078個字元直接寫入到output。接著將image的數值0~255存到int array BBMap中，接著去check每個pixel周遭點(using 3-5-5-5-3 kernel)的值，並選出maximum來當作此pixel的值。接著輸出圖片。圖片檔名是hw5\_1\_GrayDil.bmp。  


Part 2 Gray-scale Erosion (using lena.bmp)

$ gcc –o hw5\_2\_GrayEro hw5\_2\_GrayEro.cpp

$ ./hw5\_2\_GrayEro

我直接使用c++讀入bmp file lena.bmp，並將header 1078個字元直接寫入到output。接著將image的數值0~255存到int array BBMap中，接著去check每個pixel周遭點(using 3-5-5-5-3 kernel)的值，並選出minimum來當作此pixel的值。接著輸出圖片。圖片檔名是hw5\_2\_GrayEro.bmp。  


Part 3 Gray-scale Opening (using lena.bmp)

$ gcc –o hw5\_3\_GrayOpen hw5\_3\_GrayOpen.cpp

$ ./hw5\_3\_GrayOpen

我直接使用c++讀入bmp file lena.bmp，並將header 1078個字元直接寫入到output。接著將image的數值0~255存到int array BBMap中，接著去check每個pixel周遭點(using 3-5-5-5-3 kernel)的值，並選出minimum來當作此pixel的temp值，存進TMap陣列；然後再把TMap當input，一樣check每個pixel周遭點的值並選出maximum來當作此pixel的值。接著輸出圖片。圖片檔名是hw5\_3\_GrayOpen.bmp。  


Part 4 Gray-scale Closing (using lena.bmp)

$ gcc –o hw5\_4\_GrayClos hw5\_4\_GrayClos.cpp

$ ./hw5\_4\_GrayClos

我直接使用c++讀入bmp file lena.bmp，並將header 1078個字元直接寫入到output。接著將image的數值0~255存到int array BBMap中，接著去check每個pixel周遭點(using 3-5-5-5-3 kernel)的值，並選出maximum來當作此pixel的temp值，存進TMap陣列；然後再把TMap當input，一樣check每個pixel周遭點的值並選出minimum來當作此pixel的值。接著輸出圖片。圖片檔名是hw5\_4\_GrayClos.bmp。  


🡪以上的source code、執行檔、result image皆附在壓縮檔中。