電機所-陳治言-R06921084

Description：

一開始，先處理圖片檔的標頭資訊，包括 size，offset，width，height，bits 等資訊，之後再將圖片的色彩值讀入，因為這次範例圖片只有 8 位元 色彩，所以每個畫素只需要 1byte 即可儲存。而作業主要是做圖片的 equalization。並將圖片的 histogram 給繪製出來，再將標頭與畫素資料一起寫出並存檔，並且分別產生 2 個檔案，equalization_lena 和 histogram2_lena 二張輸出圖片，即可完成這次的作業。

Algorithm：

一開始先統計圖片的 histogram，得出 0 到 255 的所有像素的出現次數。再來，利用累積分布函數，$s_k = 255\sum_{j=0}^{k}\frac{n_j}{n}$ 來計算每個像素的新值。最後在統計 histogram 即可。

Principal Code：

```
//先統計原圖的 histogram
for(i=0; i<bmpInfo.biHeight; ++i)
        for(j=0; j<bmpInfo.biWidth; ++j)
                ++histogram[BMPdata[i][j].color];
//開始計算累積分布函數
pdf[0] = histogram[0];
for(i=1; i<256; ++i)
        pdf[i] = histogram[i] + pdf[i-1];
//利用比例算出新的像素值並存入
for(i=0; i<bmpInfo.biHeight; ++i)
        for(j=0; j<bmpInfo.biWidth; ++j)
        BMPdata[i][j].color = pdf[BMPdata[i][j].color] * 255 / pdf[255];
```

Parameters：

編譯程式碼 g++ -o lena lena.cpp

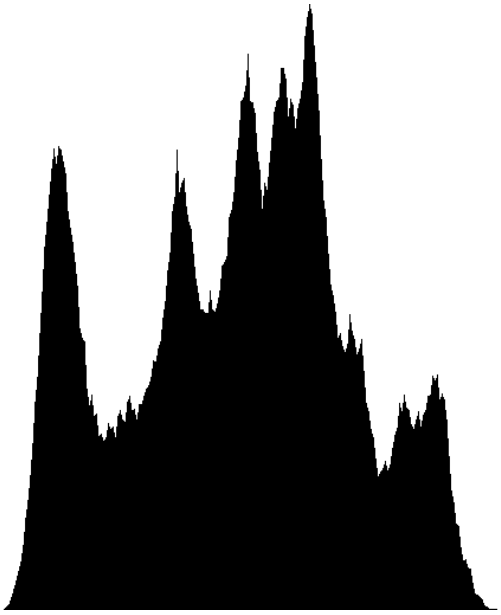
執行程式 ./lena lena.bmp

lena.bmp 是我們的 InputImage

Resulting Images：

equalization_lena

histogram_lena



histogram2_lena