E94091063 資訊系三甲 鄭鈞智

使用環境：matlab2023a

資料夾檔案：

1. Histogram+Edge.bmp：測試用的參考圖片，來自實驗室網站
2. project3.m：主程式
3. sobel.m：內含自己實作的sobel edge detect function
4. Histogram\_Comparison：有原圖的灰階色彩histogram，與經過Histogram Equalizaion後的圖片之Histogram
5. Picture\_Histogram\_Equalization.png：經過Histogram Equlization的圖片
6. Picture\_Comparison.png：灰階的原圖與經過Histogram Equalization後的圖片對比
7. Edge\_Detection.png：將灰階後的原圖進行sobel edge detection後的結果圖
8. Edge\_Detection\_eq.png：將經過Histogram Equlization的圖片拿去進行sobel edge detection的結果
9. Edge\_Detection\_Comparison.png：7與8的對照圖

程式碼解釋：

project3.m

一張含有 文字, 字型, 行, 數字 的圖片

自動產生的描述

讀取圖片並轉換成灰階，還有取得圖片的長與寬

一張含有 文字, 螢幕擷取畫面, 字型, 行 的圖片

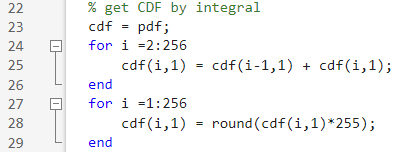
自動產生的描述

遍歷整張圖片，紀錄0~255出現的次數

一張含有 文字, 字型, 螢幕擷取畫面 的圖片

自動產生的描述

計算PDF



將PDF積分變成CDF，之後再對每個數值進行四捨五入

一張含有 文字, 字型, 數字, 螢幕擷取畫面 的圖片

自動產生的描述

藉由CDF的數值，將原圖的數值做轉換，生成Histogram Equalized的結果圖

一張含有 文字, 字型, 螢幕擷取畫面, 白色 的圖片

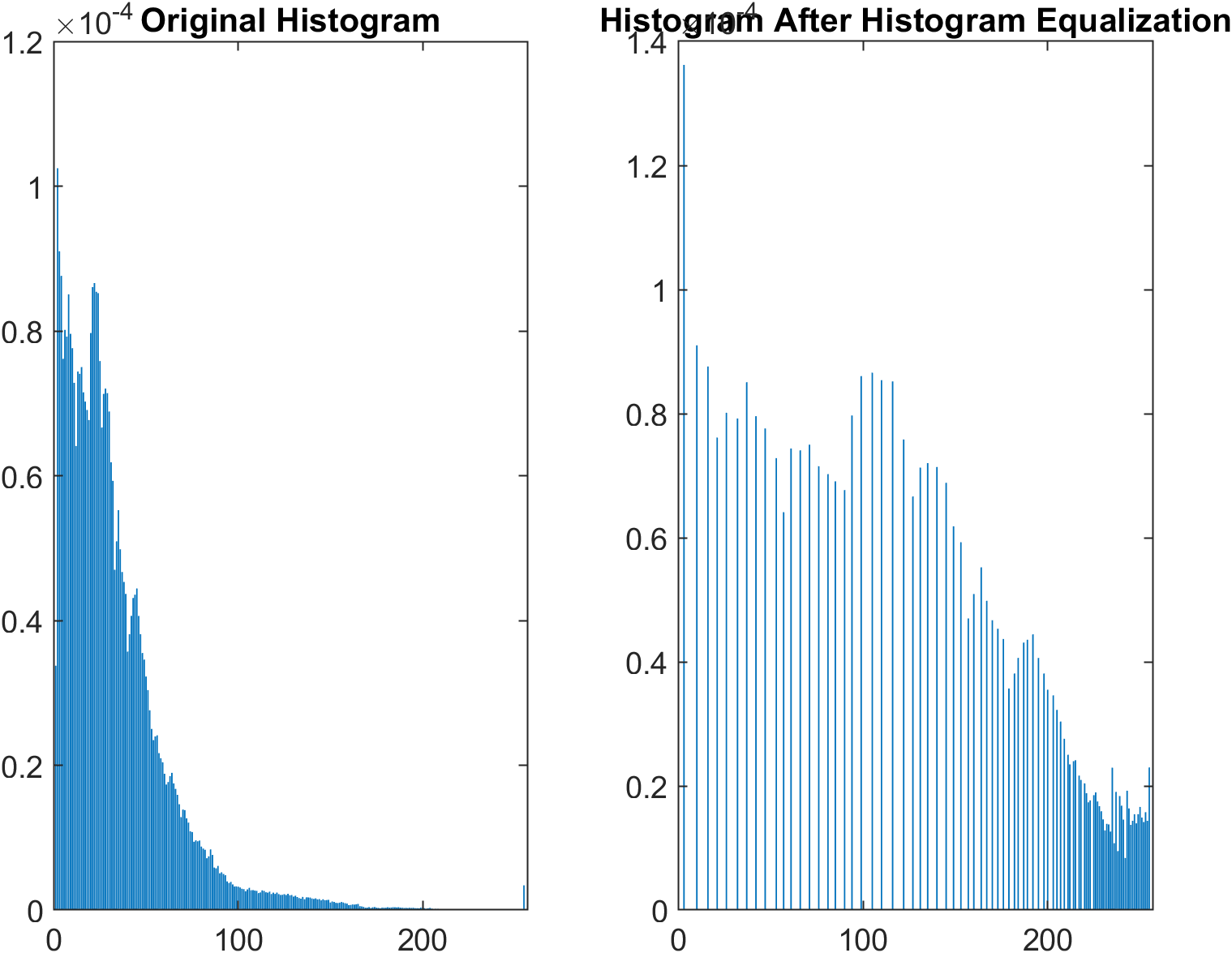
自動產生的描述

計算Histogram Equalized後的histogram

一張含有 文字, 螢幕擷取畫面, 字型 的圖片

自動產生的描述

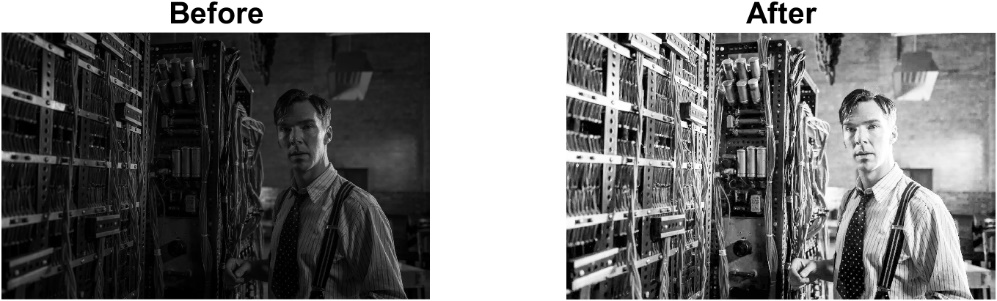
將結果圖輸出



一張含有 文字, 螢幕擷取畫面, 字型 的圖片

自動產生的描述

將灰階的原圖與Histogram Equlized的圖進行sobel edge detection，並且輸出結果。使用自訂function sobel（在sobel.m裡）



sobel.m

一張含有 文字, 螢幕擷取畫面, 陳列, 軟體 的圖片

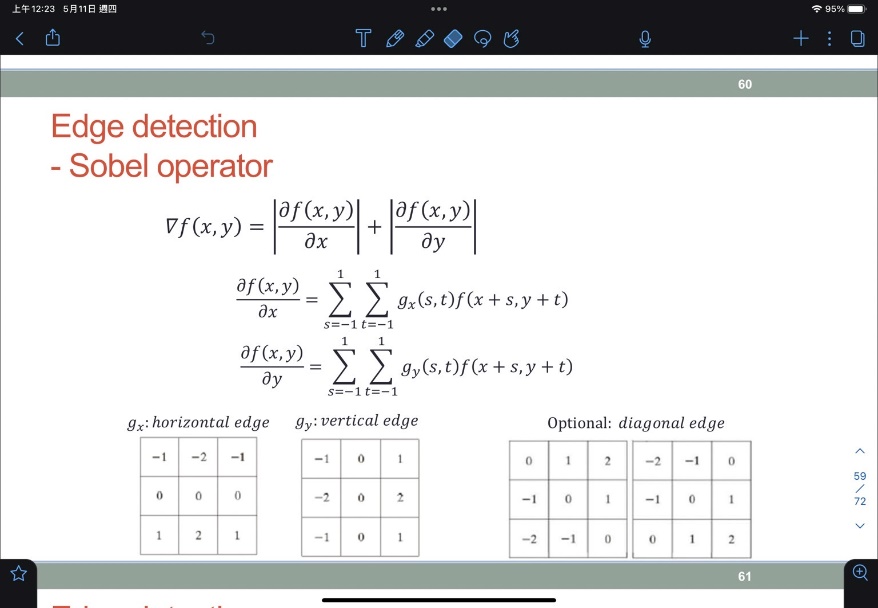
自動產生的描述

建立好兩個mask（horizontal edge與vertical edge），取得圖片的長與寬，初始化結果圖

一張含有 文字, 螢幕擷取畫面, 數字, 字型 的圖片

自動產生的描述

進行運算（以下圖片擷取自教授的講義）

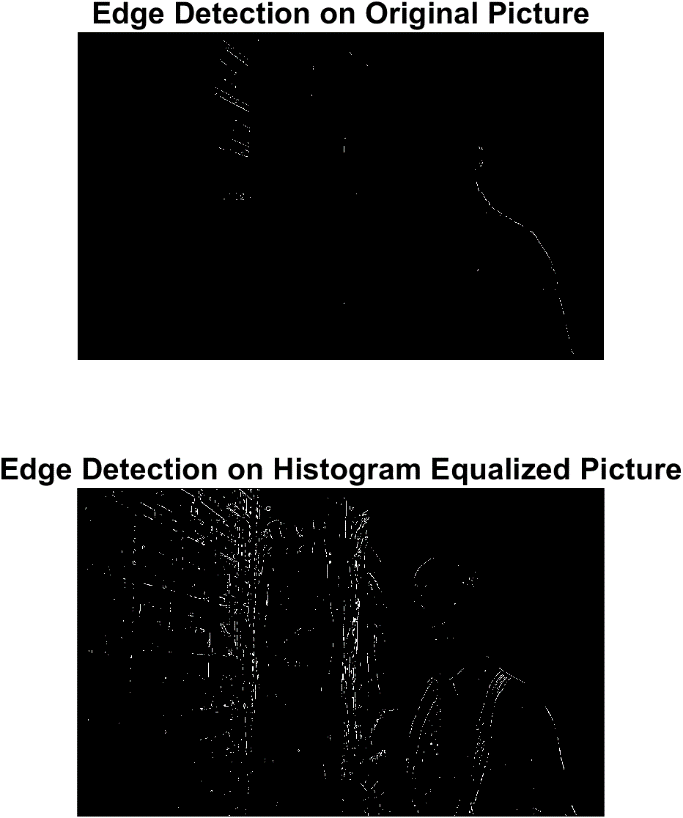


簡單來說就是遍歷整張Image，以每個pixel為九宮格中心，每一格與gx, gy對應格子相乘並加總，得到該pixel新的值。若遇到的pixel在邊界（即無法成為九宮格中心），則直接設為0。

一張含有 文字, 字型, 螢幕擷取畫面, 行 的圖片

自動產生的描述

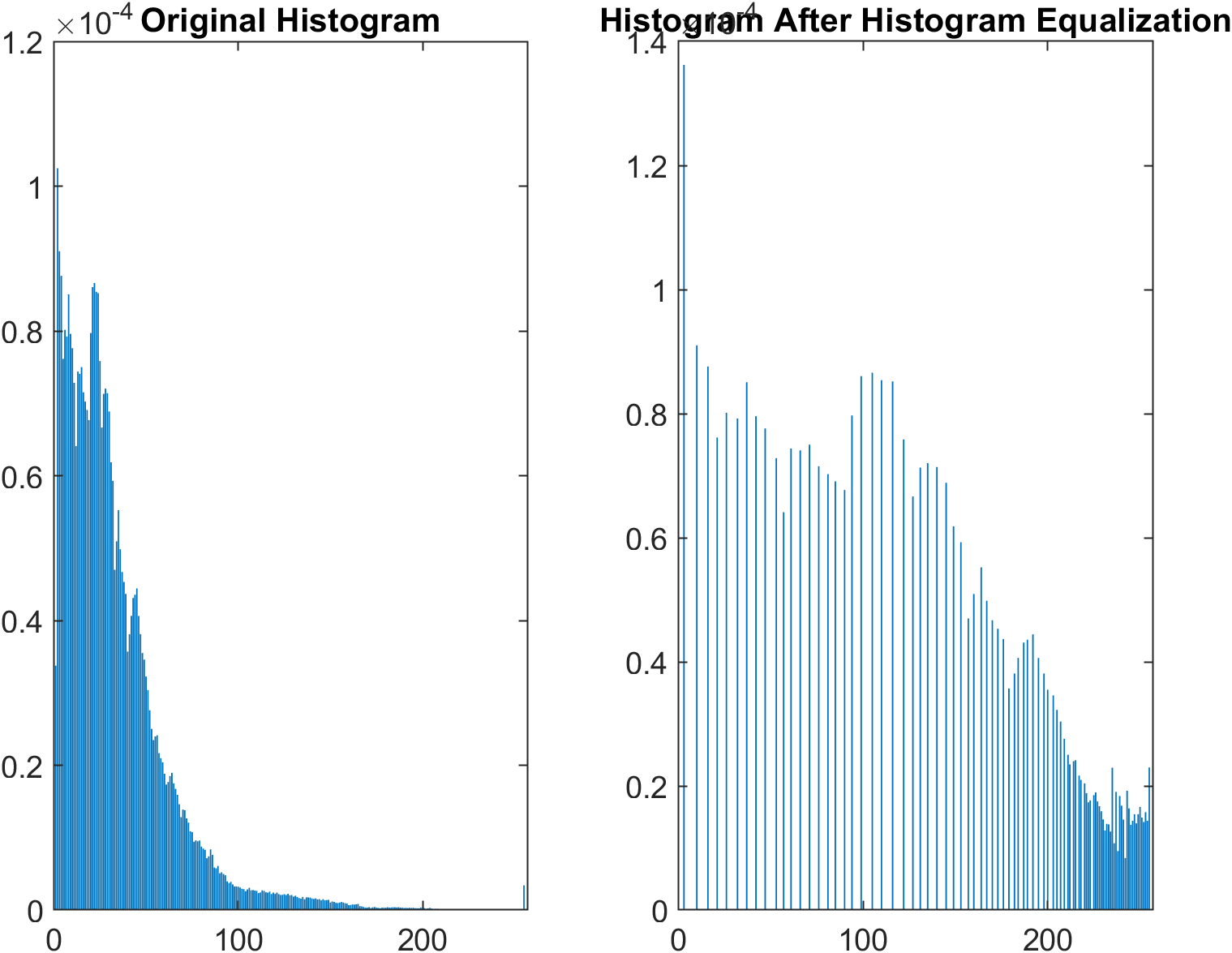
將結果map回0~255



藉由比對圖可以得知，將圖片Histogram Equlized後，可以得到較佳的edge detection結果

Results:

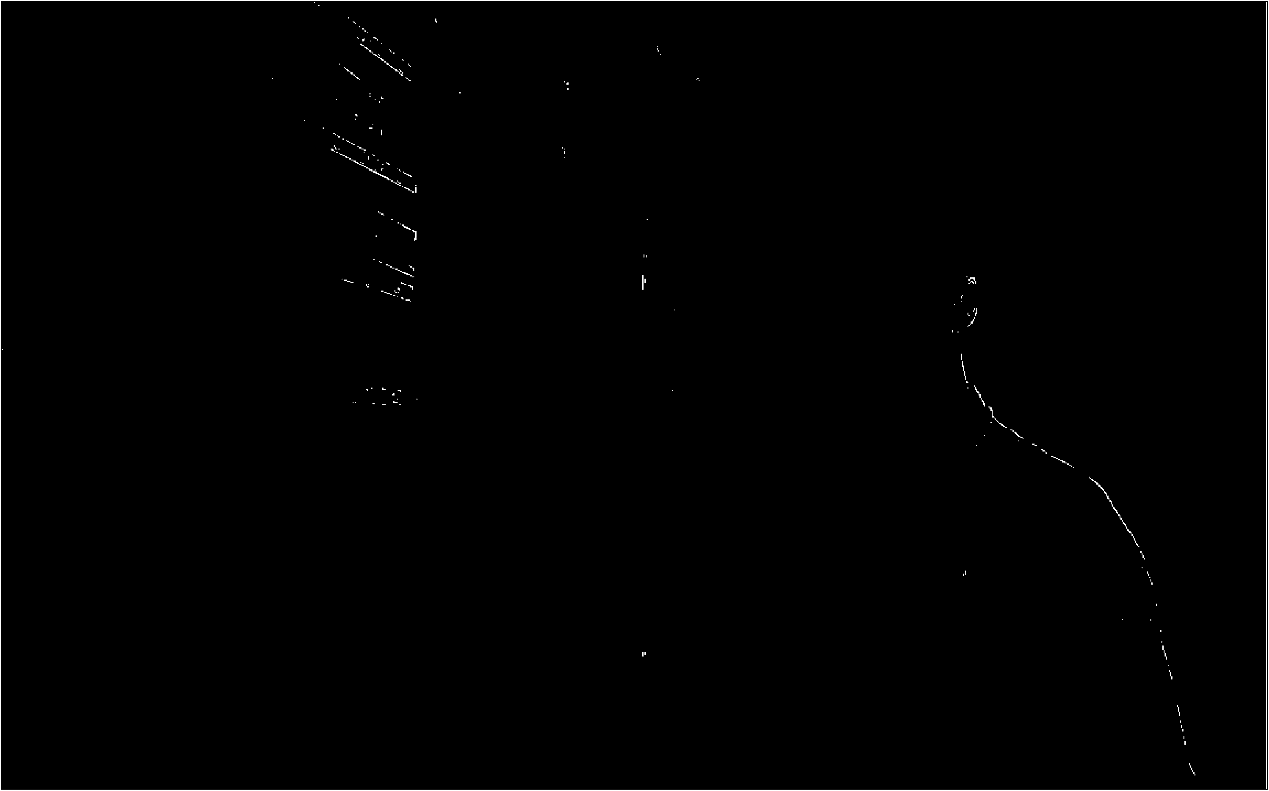
1. Original histogram, equalized histogram



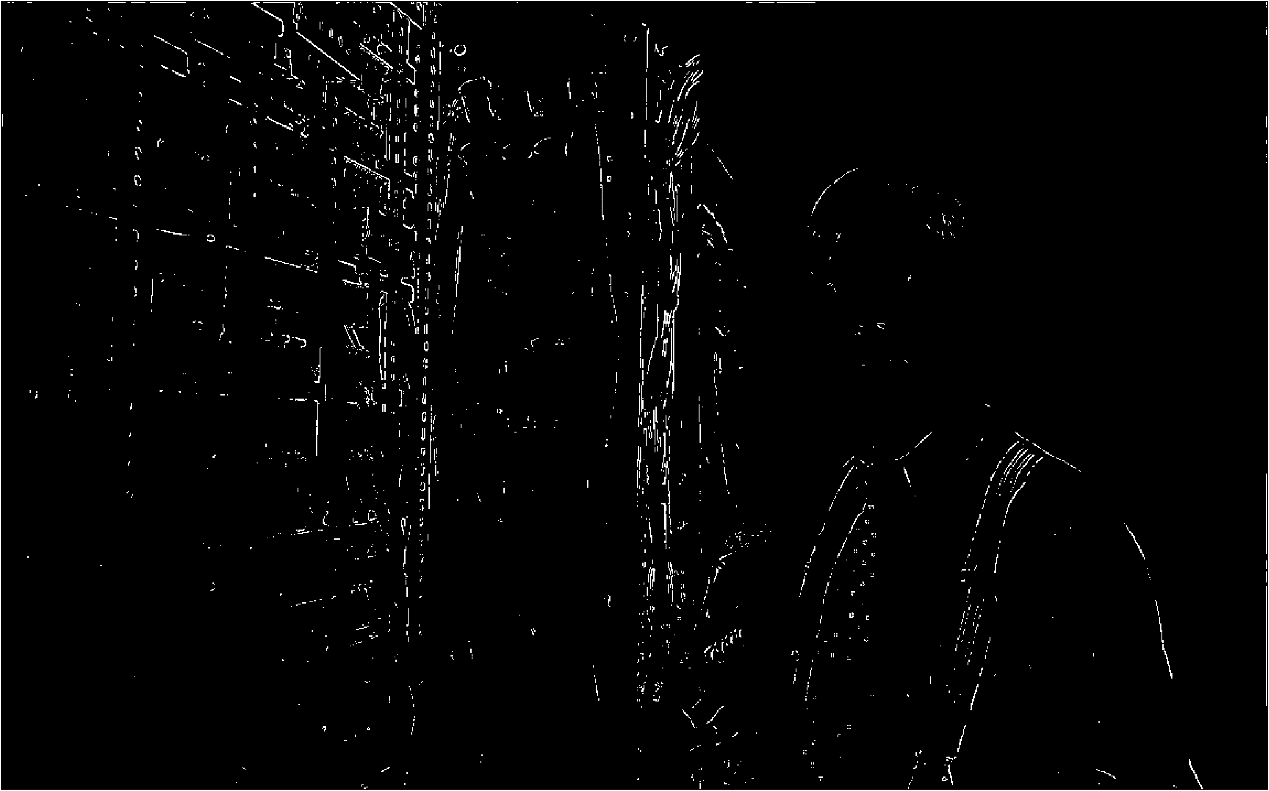
1. processed image after histogram equalization



1. Detected edges in the image after edge detection



另外使用image after histogram equalization進行edge detection的結果如下圖：



另外我還拿了其他圖片進行測試：

