

3DTV 作業一

I. 作業題目: 利用 Sobel Mask 完成一邊緣偵測器，用該偵測器處理影像。

- i. 依照投影片的步驟實現一 Sobel 邊緣濾波器
- ii. 使用 OpenCV 函示庫完成一 Sobel 邊緣濾波器

II. 編譯/執行環境

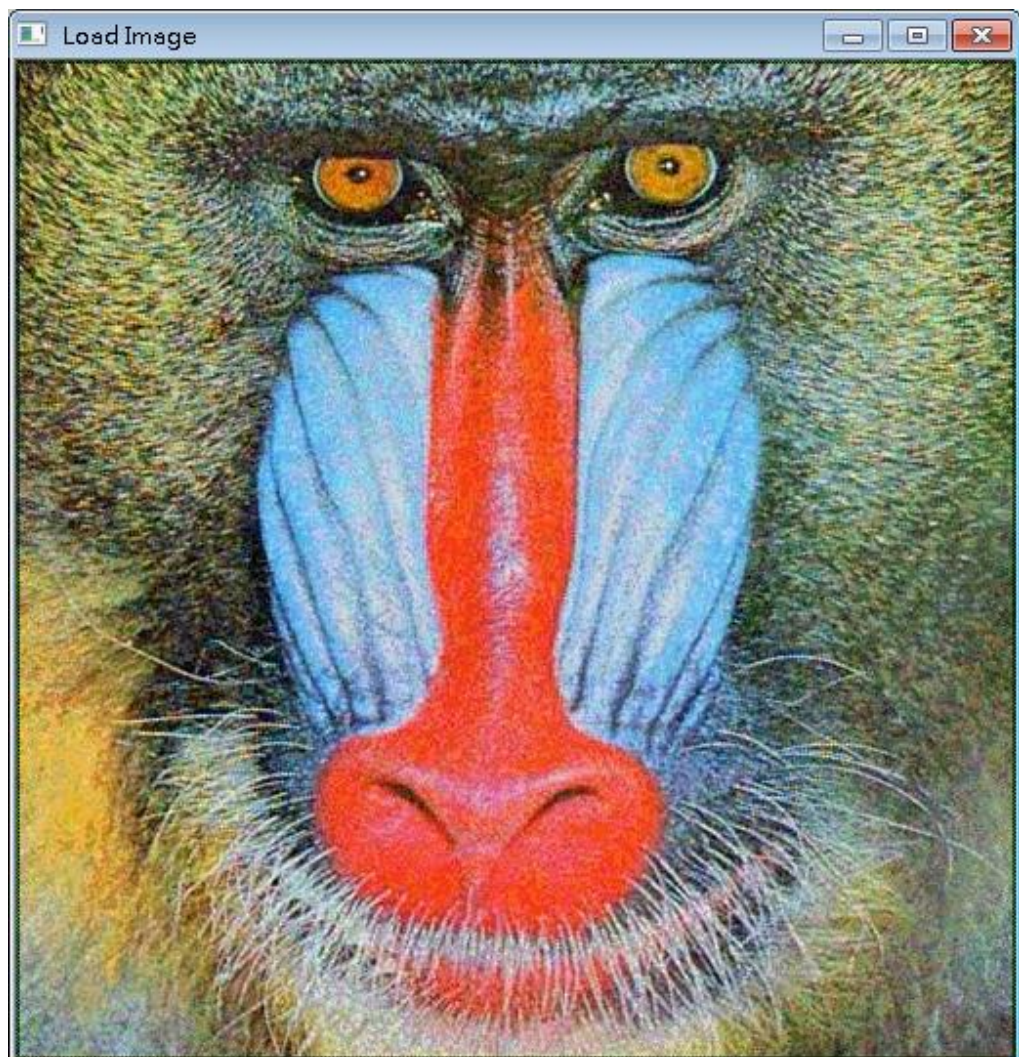
- i. Visual Studio 2008，OpenCV2.1。
- ii. Windows 7，64 位元作業系統。

III. 程式碼(附件在壓縮檔內)

- i. 實作 Sobel：main_Sobel.cpp。
- ii. 使用 OpenCV：main_cvSobel.cpp。

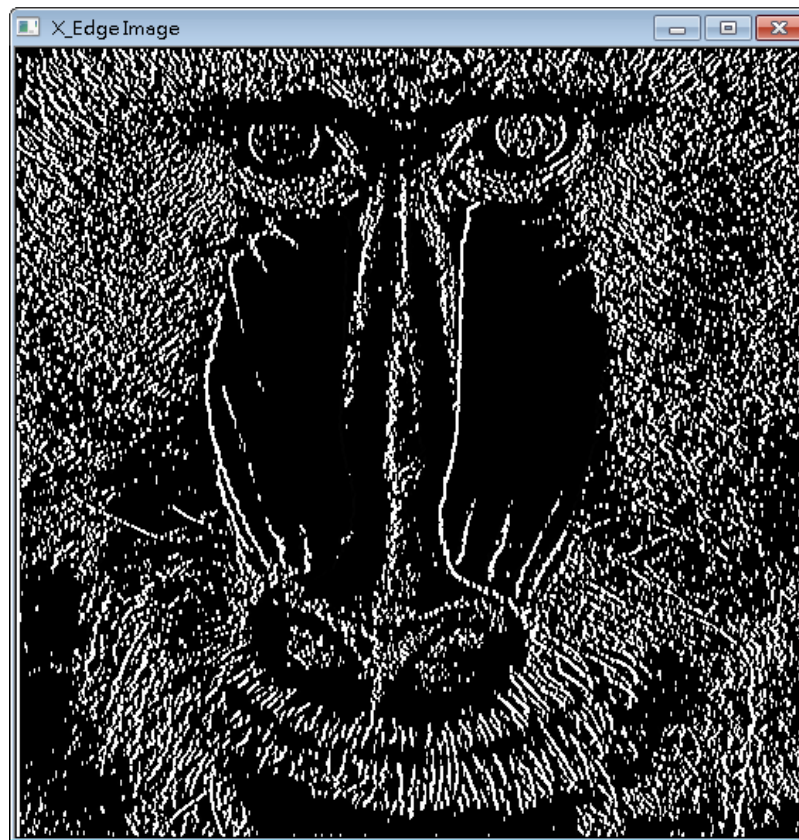
IV. 實驗結果

- i. 原圖

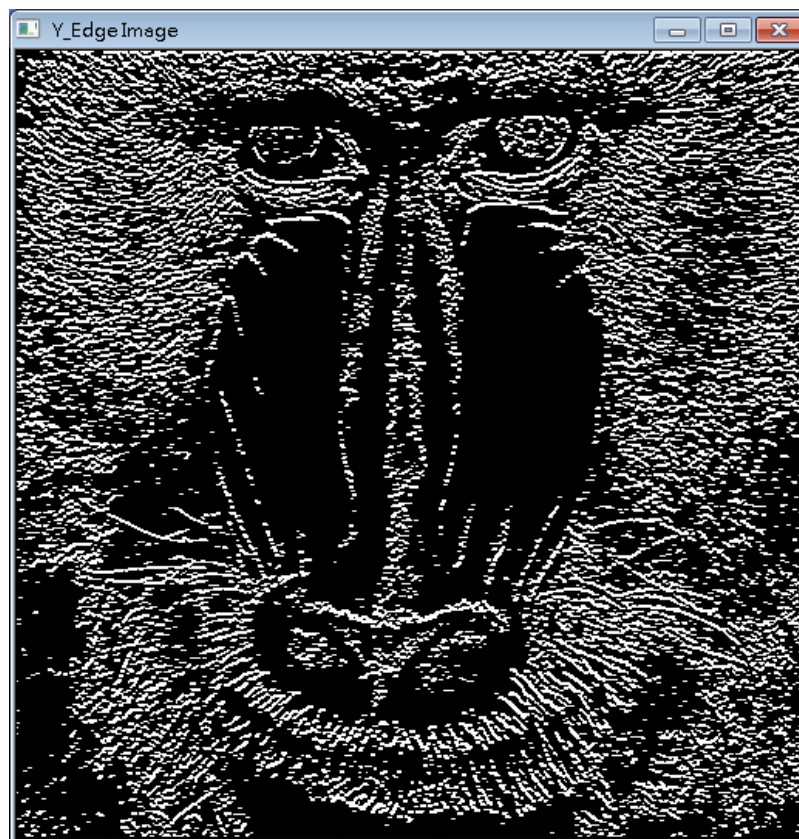


ii. 實作 Sobel

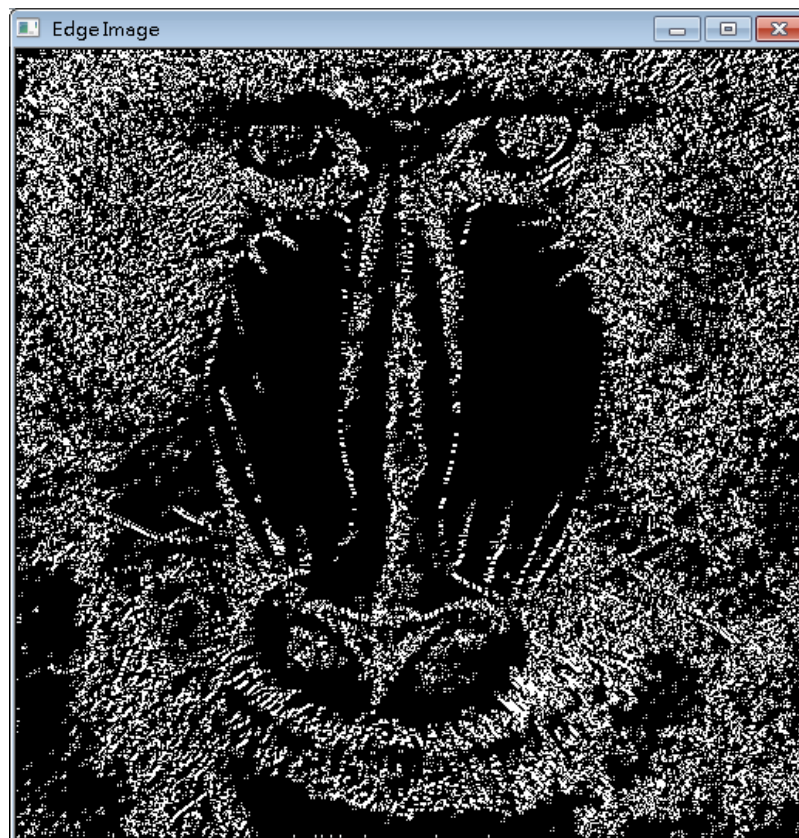
1. X 方向邊緣資訊



2. Y 方向邊緣資訊

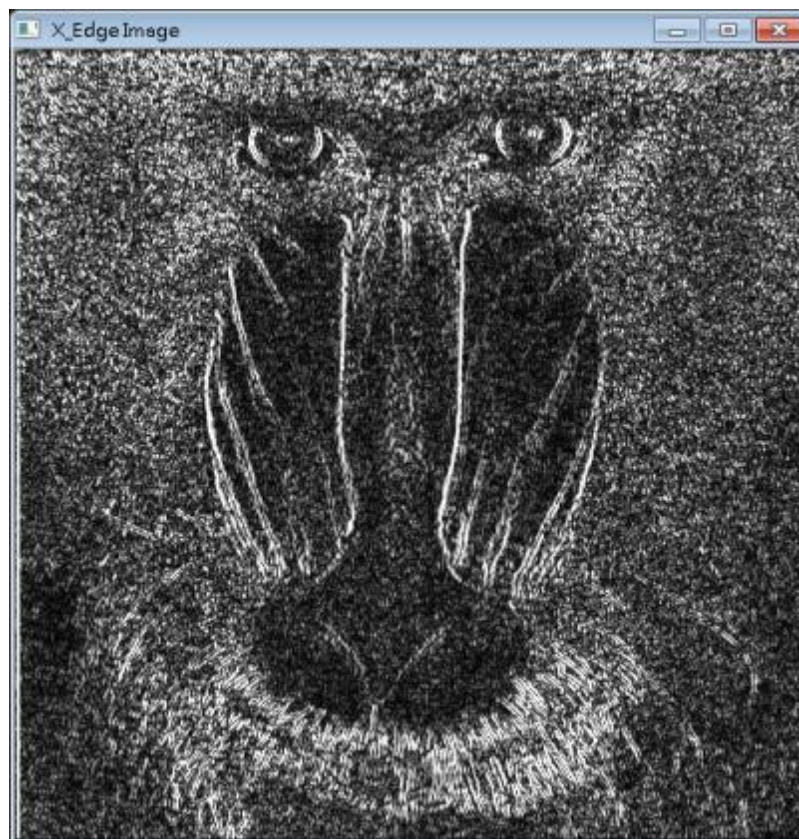


3. 加總邊緣資訊



iii. 使用 OpenCV

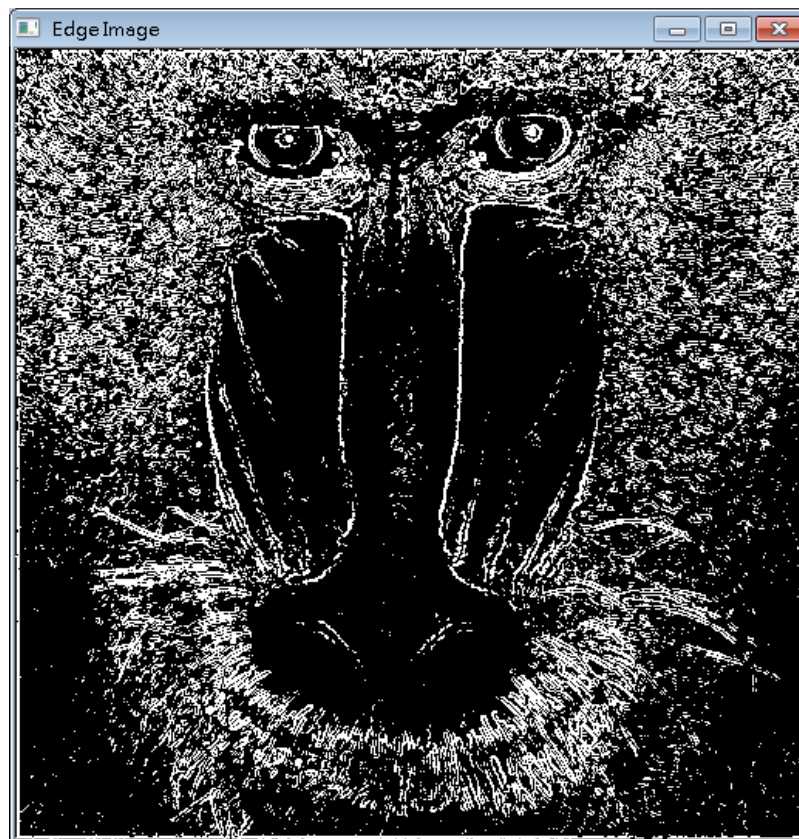
1. X 方向邊緣資訊



2. Y 方向邊緣資訊



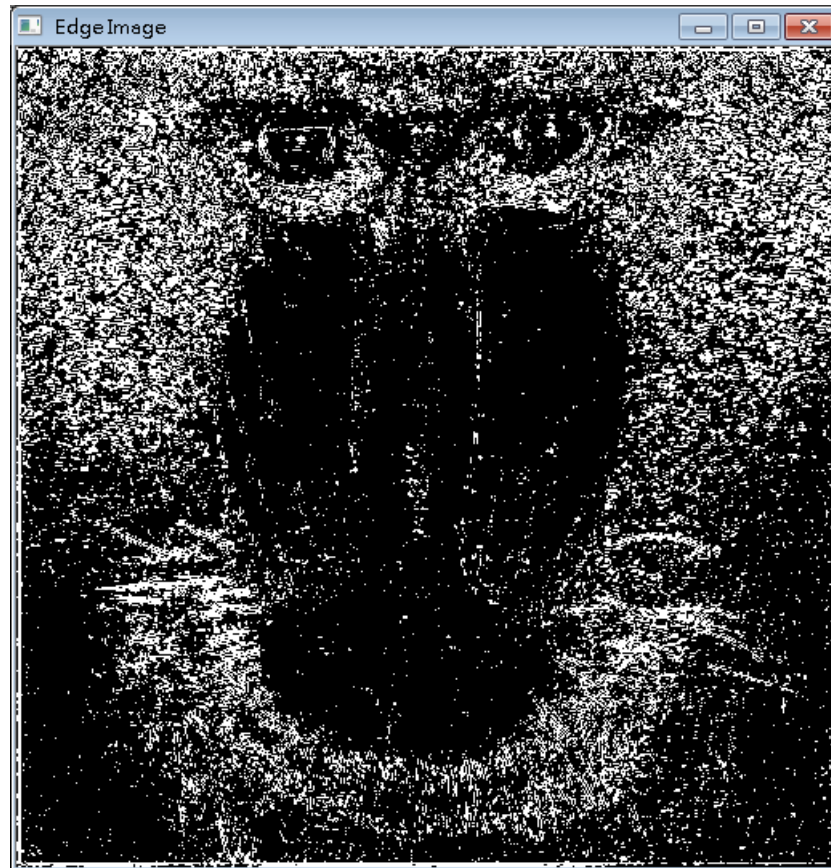
3. 加總邊緣資訊



V. 討論

Sobel 的演算法在於使用一個 3×3 的矩陣對於目標圖像做 convolution 的處理。主要分兩種：第二個 column vector 為 0 的是對 X 方向處理的 mask；而第二的 row vector 為 0 的則是對 Y 方向做運算的 mask。藉由助教提供教材中的 convolution 運算，便能得到邊緣偵測後的影像，結果如前面報告所述。

其中，Sobel 的 mask 還有一項所謂的”階數”。階數升高有種加權的意味存在，實際做出來的圖如下所示，相較之下資訊少了一些，但相對的也表示是較重要的邊緣資訊。



除此之外，使用 `cvSobel` 有個要注意的地方，由於 convolution 會算出負值，原本預設 `unsigned 8-bit` 的資料型態不夠儲存資訊，因此要改成 `signed 16-bit` 才可以正常運作。但想要顯示出圖片的話，還是必須使用 `unsigned 8-bit` 的資料型態，這時可以使用 `cvConvertScaleAbs()`，將型態再做轉換，就可以正常顯示出影像了。