

# 3DTV 作業五

I. 作業題目：使用附檔之左影像和右影像以 SSD 演算法實作 stereo matching，並與 ground truth 作比較，計算錯誤率。演算法如投影片所示，步驟可參考以 NCC 演算法完成的程式內容。

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)

main stereo matching.cpp。

IV. 實驗流程與結果

i. 實驗流程

1. 將附檔之左影像及右影像讀入程式。
2. 將讀入的影像轉換成灰階圖。
3. 以左影像為參考影像，依序對每個像素在右影像中找出匹配點。首先以目標像素為中心，展開一個 window。在右影像以 SSD 演算法搜尋最相似的 window，由於本次作業影像僅向左偏移且無垂直方向位移，因此只須水平搜尋右影像的左半邊。依作業規定，搜尋範圍為以目標像素向左最多偏移 64 pixel。
4. SSD 演算法核心公式如下所式，計算出的值越小越相似。

$$SSD = \sum_{j=-w}^w \sum_{i=-w}^w (Il(x+i, y+j) - Ir(x+i, y+j))^2$$

其中，(x,y)為目標像素座標，w 為像素偏移量(window size 為 2w+1)，Il、Ir 分別為左影像及右影像。

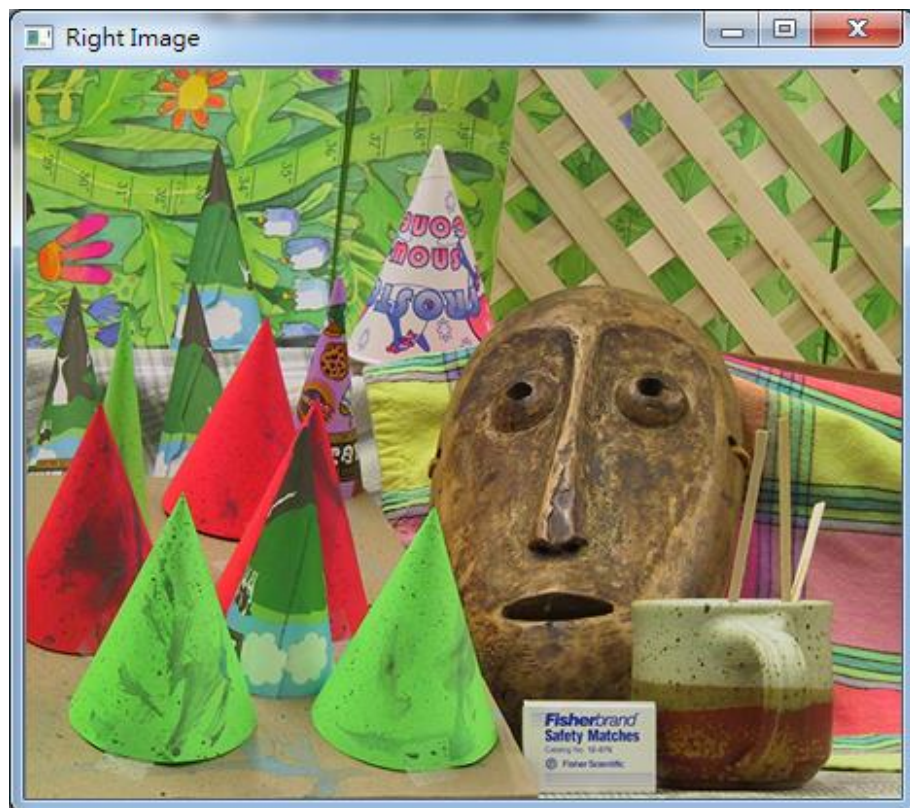
5. 對於每個像素，當找到最相似的 window 將偏移量紀錄下來，建構 disparity map。
6. 將 disparity map 與 ground truth 做比較，計算錯誤率。
7. 輸出最後結果圖(disparity map)及錯誤率。

ii. 實驗結果

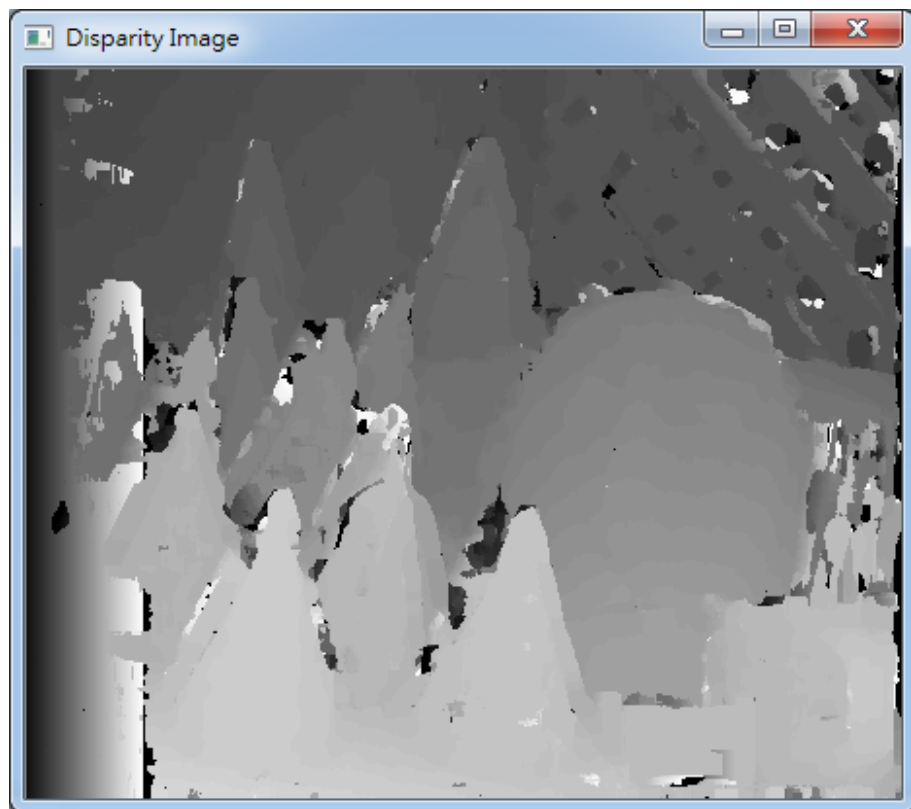
1. 左影像(參考影像)



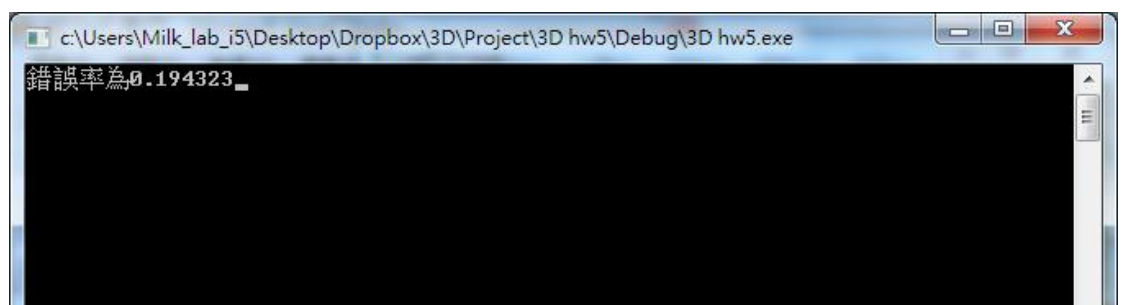
2. 右影像(偏移影像)



3. 結果圖(disparity map)



4. 錯誤率



## V. 結論與心得

這次作業比想像中容易一些，主要原因除了演算法本身不困難之外，作業提供的附檔並沒有垂直的偏移量，因此程式其實就單純許多。其實將垂直偏移考慮進來是比較嚴謹的，有視差的影像很難保證只有水平偏移。

其實類似的程式曾經寫過，也知道其運作原理，但每次想的時候還是會被一堆 for 迴圈搞得暈頭轉向，尤其是這種要開 window 類型的程式，常常會忘記哪個變數是負責指定目標像素的還是用來搜尋的，但還好有之前的經驗，這次在實作上有稍微順利一些，希望以後可以越來越順手。