

## 高等電腦視覺 HW#3



學號：107318050

姓名：黃佳婷

本次作業為實現 Connected components，使用 C++ 完成這次的作業，程式執行流程就是輸入檔案名稱後觀看相關資訊及結果，直接執行 exe 檔就會看到存下來的圖了，只是程式沒做甚麼防護措施，報告裡的圖片都有附原圖檔可以查看，也可以直接執行 exe 檔產生。

### #Program\_3-1(純 C++)

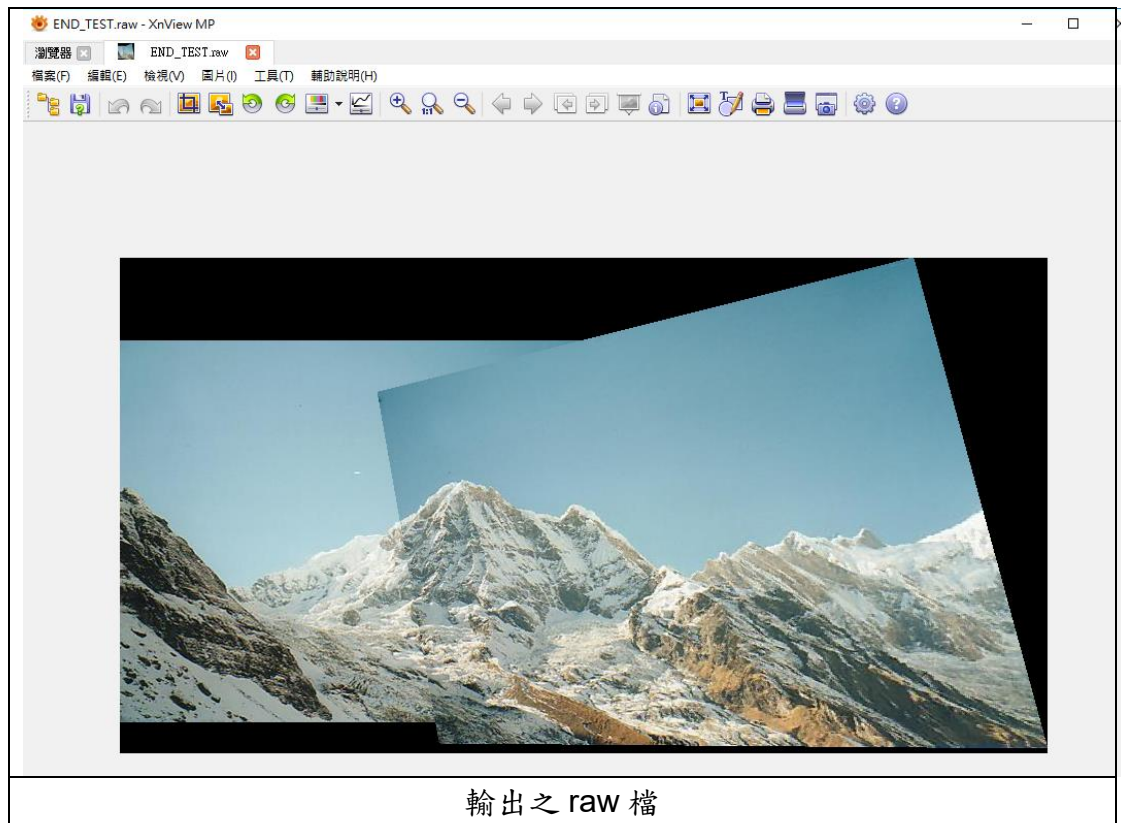
首先將用 forward warping 先定義好新圖的大小，再利用 inverse warping 映射回原圖填像素值，因為 forward warping 是從原圖計算對應到新圖的座標值，但是有可能對應到的座標有小數點並不存在，導致輸出圖會有一些點是黑的，用 inverse warping 則是相反，是從新圖的座標計算對應到原圖的座標值，就可以確保新圖的每個像素點都會填到值，以下為純 C++ 之結果圖，皆有附輸出圖檔可查看。

	
distortion.bmp(原圖)	輸出結果

### #Program\_3-2(C++)

這題做的是將兩張圖拼接在一起，參考作業說明附 x、y 型變參數以及的 paper 的公式( $P_s = P_d M_d$ )——代入矩陣運算，就會得到輸出的拼接圖，因為上一題寫 BMP 檔寫格式寫得有點痛苦，所以這題輸入輸出都改用 RAW 檔處理及顯示，一樣有附輸出截圖，也可以用 XnView 查看 raw 檔。

	
left	right



心得：

這次做 inverse warping 其實卡了很久，主要是卡在寫入 BMP 檔格式的時候一直出問題，只是卡了很久的輸出結果最後也是有問題的，讓我覺得相當絕望，應該是中間有公式代錯但是我沒有發現到。因為第一題的教訓，第二題程式整個重新寫，改成讀寫都用 raw 檔處理，整體輕鬆愜意了許多，只是變成察看結果需要用 XnView 不太方便，下次有機會再改成輸出可以直接查看的格式，第二題主要的障礙是卡在看 paper 看了很久，把所有的變數搞清楚之後再實現成為程式，這之間也花了相當久的時間，想出這個拼接影像方法的人真的相當了不起。