

**Advanced Computer Vision**

HOMEWORK 3

Lab : VPILab

Advisor : Cheng-Ming Huang(黃正民)

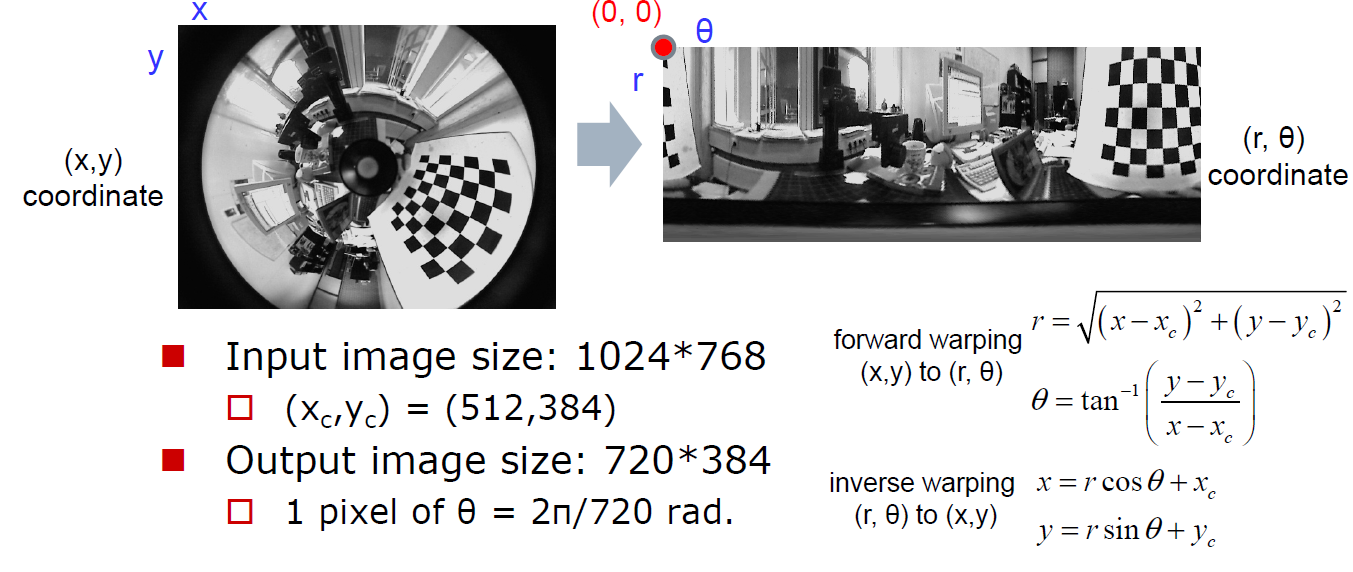
Student : Yu Cho(卓諭)

Student ID : 106318025

Data : 2017.11.30

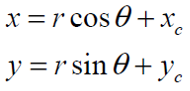
1. Rectify the image of omnidirectional camera to a panoramic image
2. Problem

Rectify the image of omnidirectional camera to a panoramic image



1. Method

使用invers warping，依據要輸出影像的各個座標位置，經過下面公式對應回原影像座標。



doublet=((double)theta/720)\*2\*M\_PI;

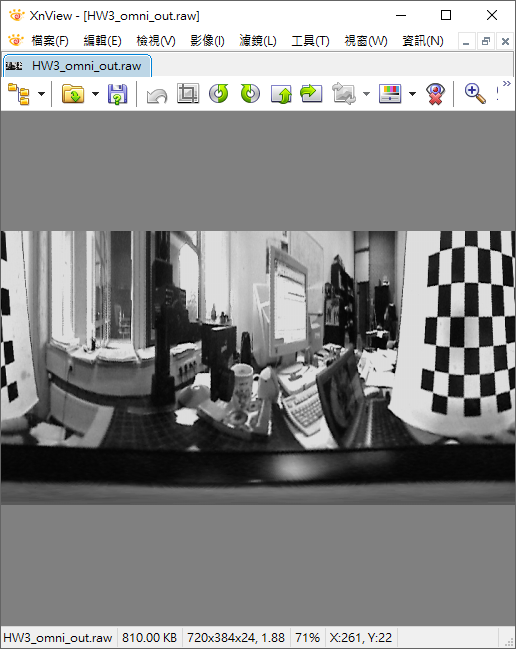
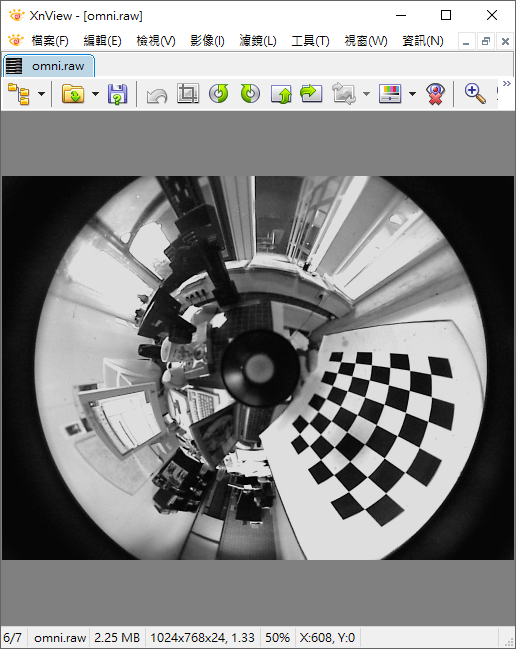
intx=(r\*cos(-t)+512);

inty=(r\*sin(-t)+384);

使用三個for迴圈分別對列、行以及通道逐行逐列掃描，並將(x, y)坐標系對應到的原影像像素點，存入(r, )坐標系的輸出影像。

1. Result

(Original) (Output)

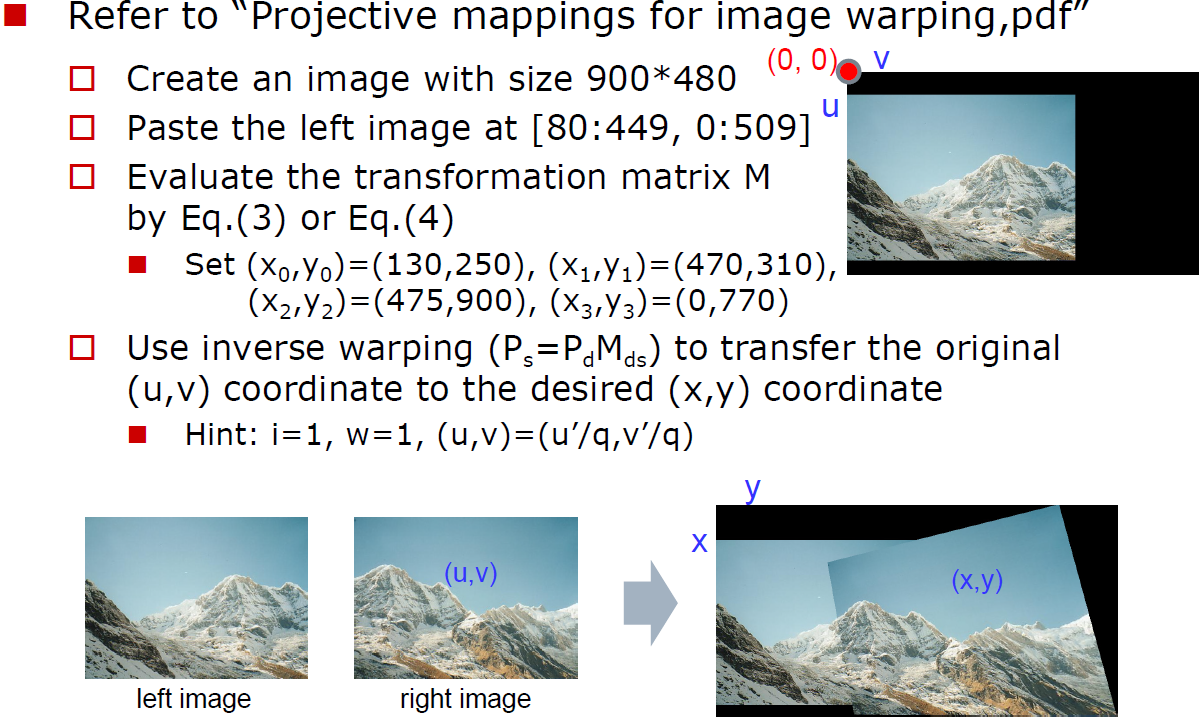


1. Discussion

在轉換的過程中，要注意的是，因為全向影像是360o的環景，而要拉伸至720個像素點，必須做正規化，1 pixel of =。將原影像存入輸出影像的步驟中，必須讓像素點由輸出圖左下開始存入，否則影像會上下顛倒(.\*raw 是從左上存入像素值)。因為r是從0開始遞加，相當於從原影像的中心點往外推，中心點應在空間中的底部位置，越從中心點外推至圓邊界則越靠近空間中的天花板位置。

1. Image stitching with the projective transform
2. Problem

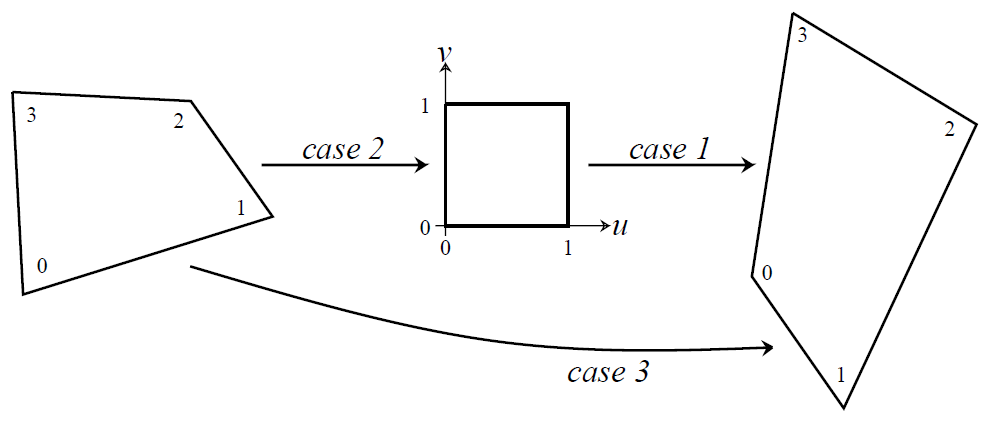
Image stitching with the projective transform



1. Method

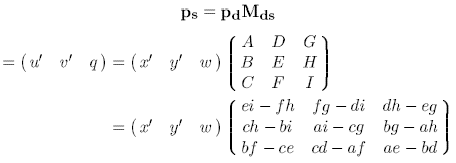
Input image: left.raw (510\*370)、right.raw (510\*370)

Output image: HW3\_image\_warping.raw (700\*480)



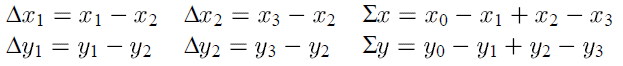
***Step1.*** 將right. raw經過上圖的Case2 轉至單位圖像，單位圖像位置為(u, v)=

***Step2.*** 要經由轉換矩陣由(x, y)計算(u, v)映射像素點

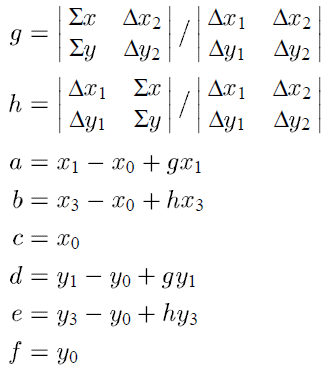




***Step3.*** 然而生成A,B,C,D,E,F,G,H,I的參數由以下公式求得。

先定義，

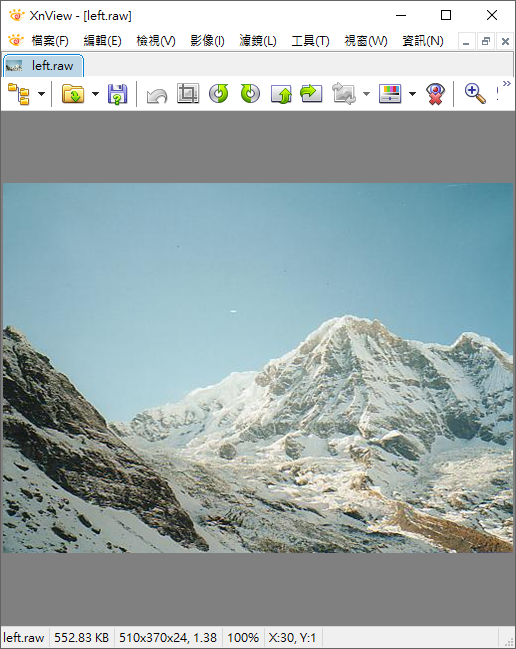
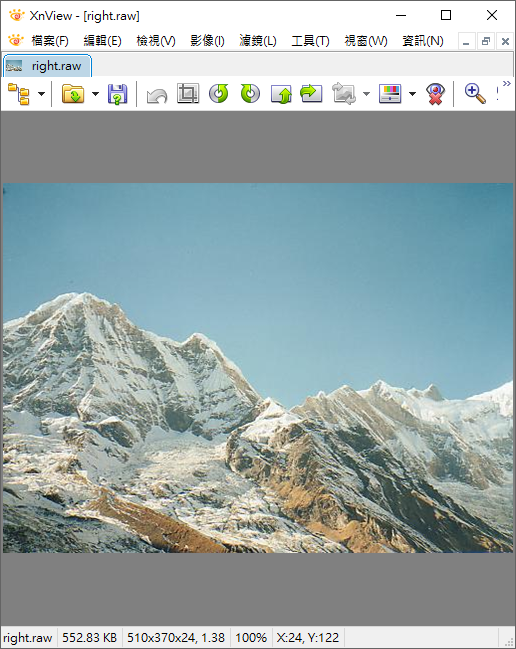
求參數a, b, c, d, e, f, g, h，



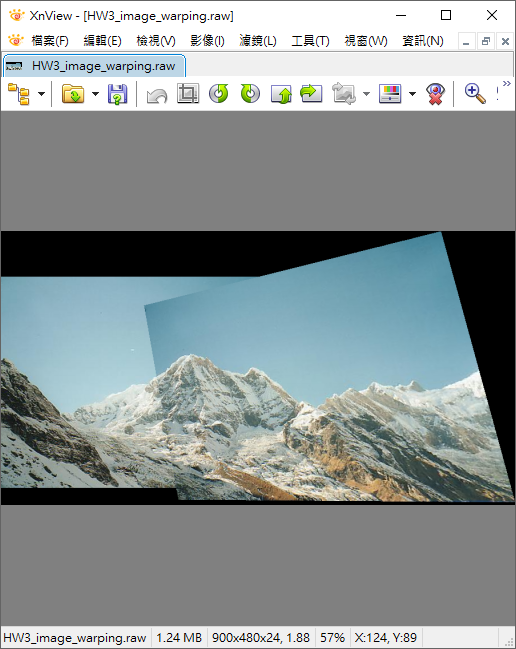
***Step4.*** 將題目提供的warping後4個頂點帶入上述公式求得參數。

1. Result

(left.raw) (right.raw)

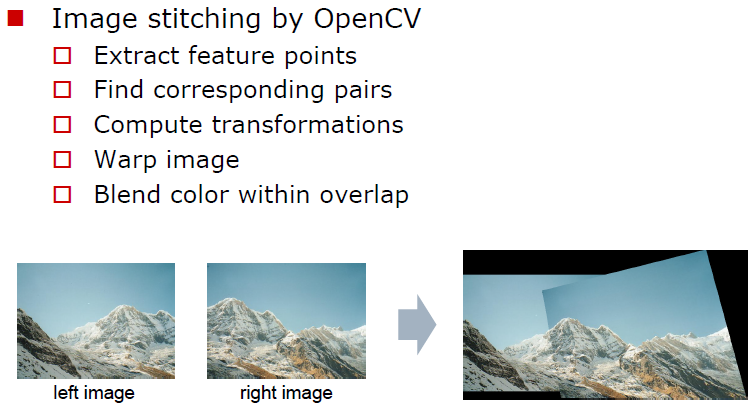
(HW3\_image\_warping.raw)



1. Discussion

參考Projective Mappings for Image Warping, Paul Heckbert 的論文，理解轉換矩陣的實現原理，算出轉換矩陣後可以知道要形變的影像4個頂點位置，並且將影像拉伸，最後將兩張影像疊合，完成stitching，這部分只實現形變，並沒有整套的影像拼接流程，影像拼接的大致上流程為，特徵點的偵測、提取、比對，再來式影像的形變，最後做色調的一致化。若要實現影像拼接，使用OpenCV函式庫會較方便也更有效率實現影像拼接。

1. Bonus
2. Problem



1. Method

Input : left.bmp、right.bmp

Output : keypoints\_left.bmp、keypoints\_right.bmp、

Corresponding\_Pairs&Warping.bmp、Image Stitching.bmp

Step1. 利用SURF提取特徵點

Step2. 特徵點描述，為之後的特徵點匹配做準備

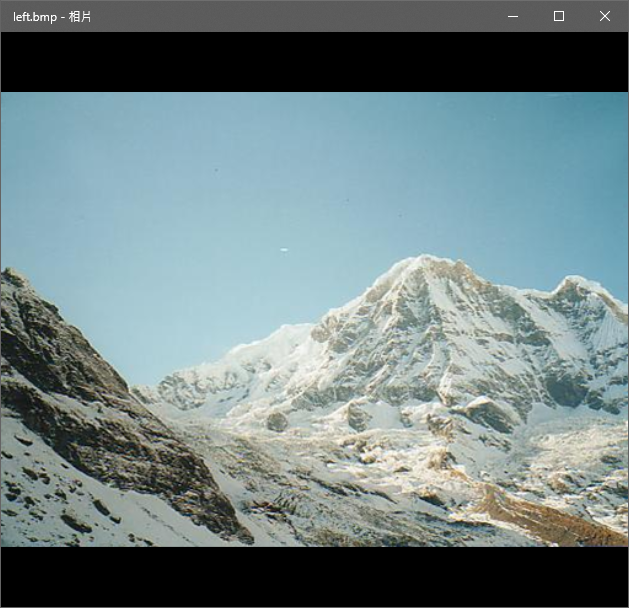
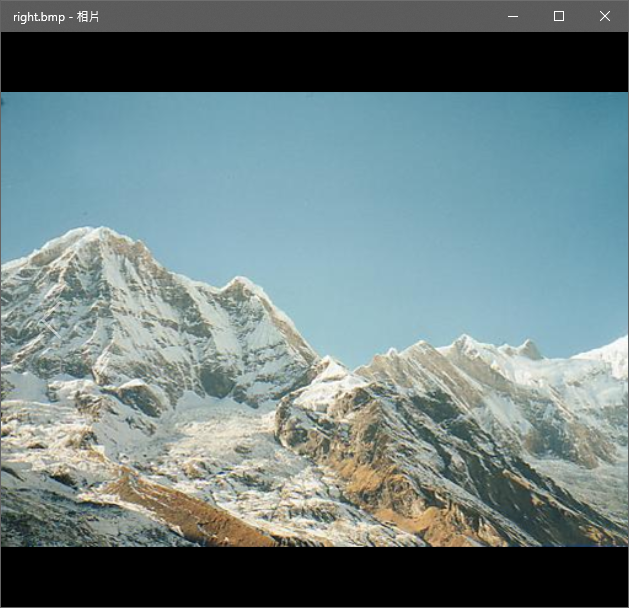
Step3. 獲得匹配特徵點，並提取最優配對(定義匹配距離小於3倍的最小匹配距離)

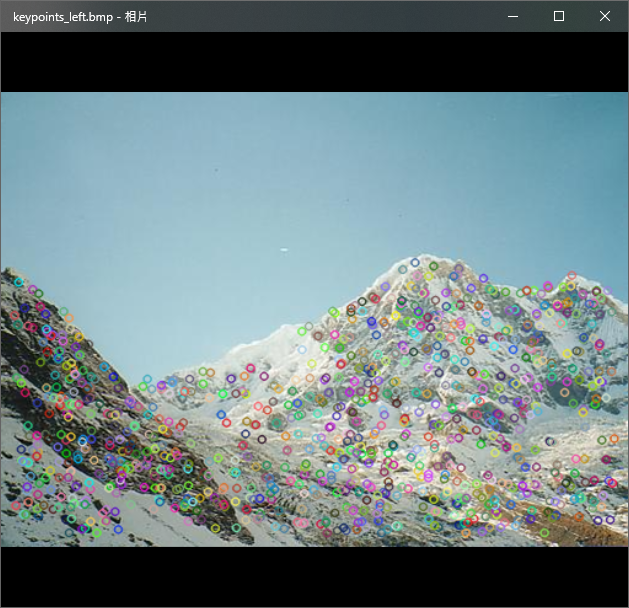
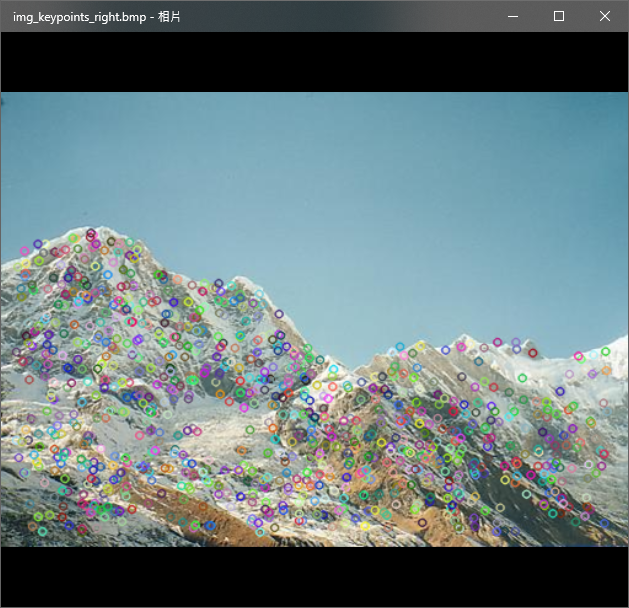
Step4. 獲取兩影像之間的投影映射矩陣，尺寸為3\*3

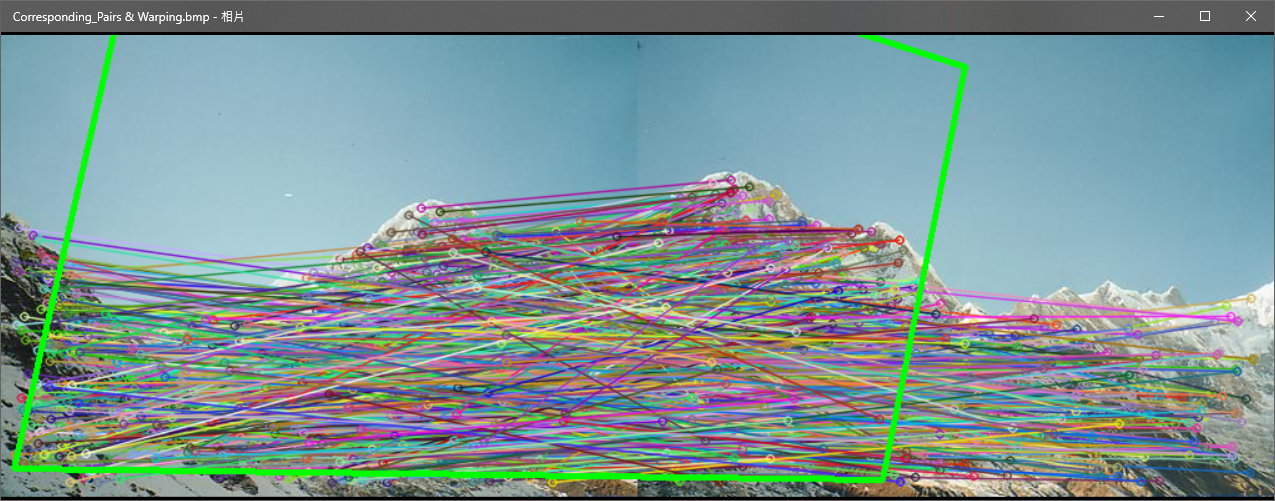
Step5. 獲取最佳配對點在原始圖像和矩陣變換後圖像上的對應位置，用於圖像拼接的定位

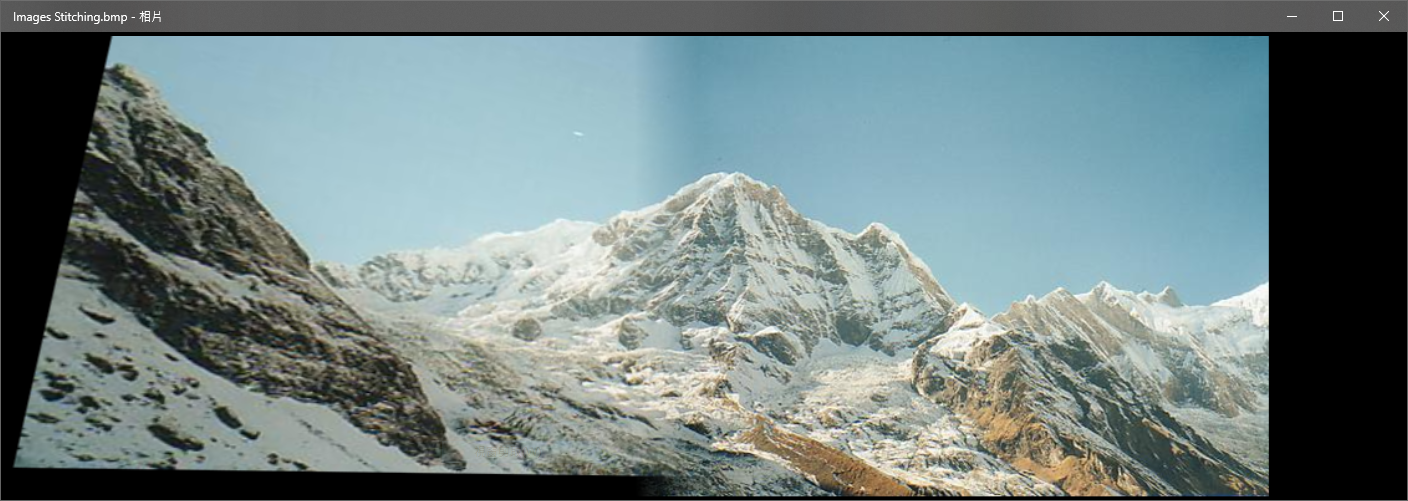
Step6.進行接縫處色調接合

1. Result





1. Discussion

使用OpenCV函示庫實現影像拼接可以更直覺地完成每個步驟的目的，最花時間的是去理解要如何使用OpenCV函示庫去實現各個步驟，相關的函式使用方式都可以從OpenCV官網上找到。