

VFX HW1—HDR report

作者：

- b02901030 電機三 陳尚甫
- b02901011 電機三 趙祐毅

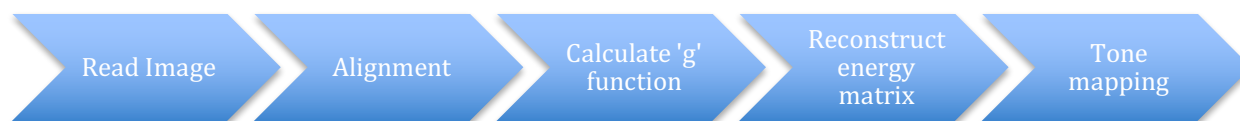
日期：

- 2016/3/29

一、 About HDR

一般而言，相機 **sensor** 的動態範圍有限，常會有照片的亮部過曝，或者是暗部曝光不足的問題，這樣的照片稱為 **LDR(low dynamic range)** 照片。我們可以經由一些處理，將多張曝光度不同的 **LDR** 照片合成一張 **HDR(high dynamic range)** 照片，在同一張照片中同時細緻地呈現出亮部和暗部的細節，希望這樣的照片更接近人眼所看到的結果。

二、 Steps



三、 HDR assembling

1. Read Image

為了大批讀進檔案，我們將一整批照片放在同一個資料夾，並且檔名設為：
img01.jpg ~ img10.img (假設有十張照片)，並且在 **matlab** 裡面設定兩個可以調整的參數：照片的張數、資料夾名稱。這樣每次要跑不一樣照片時就只要修改簡單的兩個參數。

在讀檔案時，也會同時用讀進照片的快門速度資訊，建立 **B array**。

由於原圖大小過大，因此在讀進檔案之後會先進行 **resize**，若 **resize** 為 **1/3**，花的時間約為原本的 **1/9** 倍。

2. Alignment (細節參考第四大項：Alignment)

建立兩個 **for loop**，將第一個照片輪流跟其他照片丟進 **imgalign** (我們的 **function**) 做比較，每次做 **alignment** 都會留下共同的部分，做兩次之後就能確保所有照片都做好 **alignment**。

3. Calculate 'g' function

此步驟運用老師上課所提供的 **code** 來解出 **non-linear curve**。分別對 **RGB** 三色解出一條 **curve**。

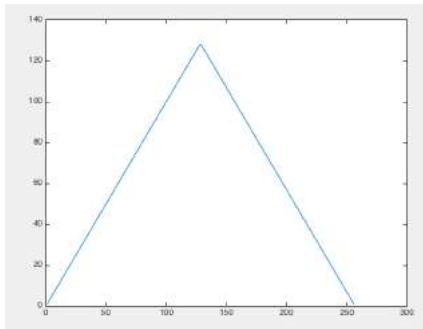
4. Reconstruct energy matrix

運用這個 non-linear curve，以及各張照片，我們使用 4 層 for loop 來計算出 hdr 照片：

1st for loop: RGB 3 色，所以要跑三圈

2nd & 3rd for loop: 圖片的長跟寬，因為每個 pixel 都要分別計算。

4th for loop: 從第一張到最後一張照片，通過前一步驟得到的 g function 得到那點的 energy，並經過三角波(如下圖所示)加權，最後寫進一張圖片，那張圖片即是我們的 hdr image.



5. Tone mapping (細節參考第五大項：Tone mapping)

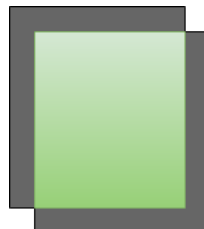
- 1) 在丟進 tone mapping 之前，我們必須先做 normalization，我讓原本的 hdr image 除上 R G B 三色裡面，最小的 energy 值中的。原本曾經 R G B 各自做 normalize，後來發現這樣會讓顏色失真，調回來之後就解決這樣的問題。
- 2) 拿回 tone mapping 過後的 matrix 之後，我們在分別乘上 255 後作四捨五入，以符合目前電腦讀取影像格式為 0~255 的需求。
- 3) 我們分別給 tone mapping 三種不一樣的 key 值，然後寫成 bmp 檔。再人工挑選最美麗的一張。

四、 Image alignment

我們使用老師上課所介紹的 Median bitmap threshold 的方法來實作這個問題，寫成一個 matlab function: `[r1, r2] = imgalign(img1,img2,iter)`

img1、img2 為想要對齊的照片，且會以 img1 為基準，最後一個 argument iter 是在使用 multiple scale technique 所希望進行的 iteration 次數，舉例來說，若將 iter 設定為 4，則最多可以調整回來 $8+4+2+1=15$ pixels 的誤差。調整完之後，沒有重疊的部分(下圖中的灰色部分)會被裁掉，留下兩張圖片都有的部分。

實作時，會用迴圈將所有曝光度的照片對齊，並且裁掉不重疊的部分，如此設計儘管對齊後所有照片都會變得較小，但每個 pixel 都能拿到所有曝光程度的資訊，能較完整還原出 HDR image。



五、 Tone mapping

我們使用老師上課介紹 Photographic method 中的 Global operator，函數定義如下：

```
function [mapImg]=toneMapping(img, lightness)
```

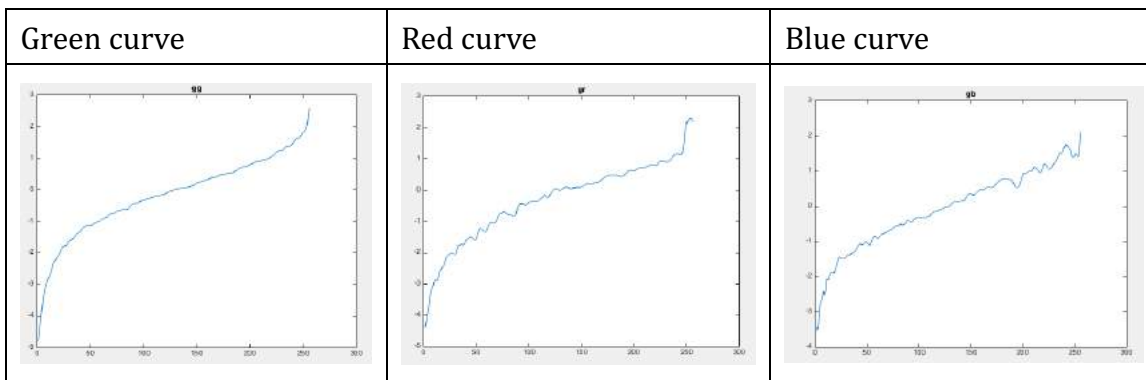
吃進含有 RGB 資訊的 `img`，以及給予亮度參數 `lightness`，也就是此方法中的 key 值：'a'。'δ'值我們給 natural number 'e'，讓取 log 不會出現負號。如此就能將所有 energy 投影回到 0 跟 1 之間。

六、 Result

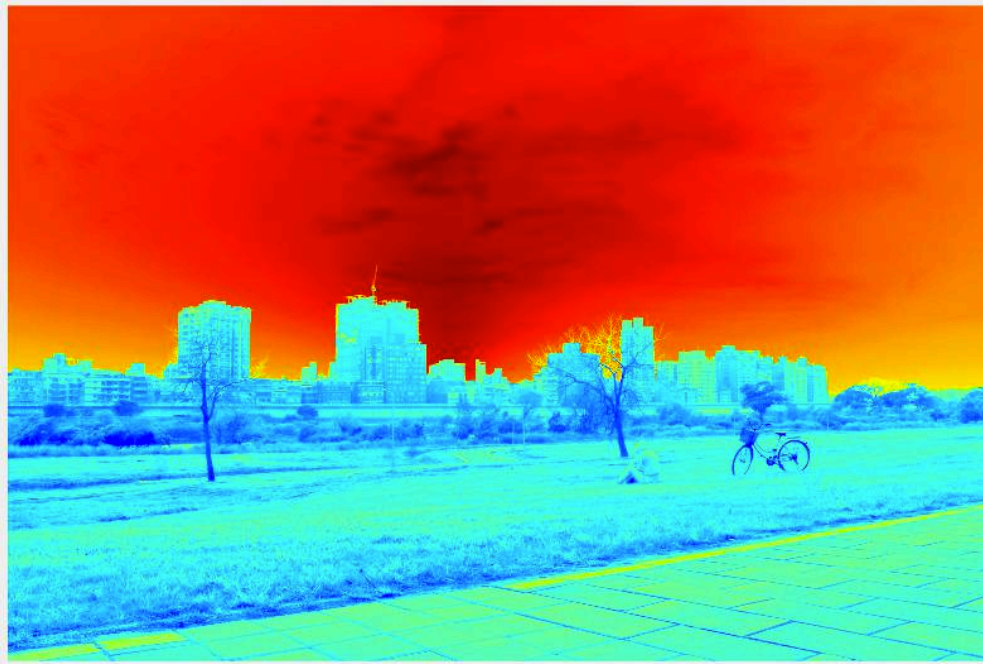
1. Original Pictures





2. 'g' function


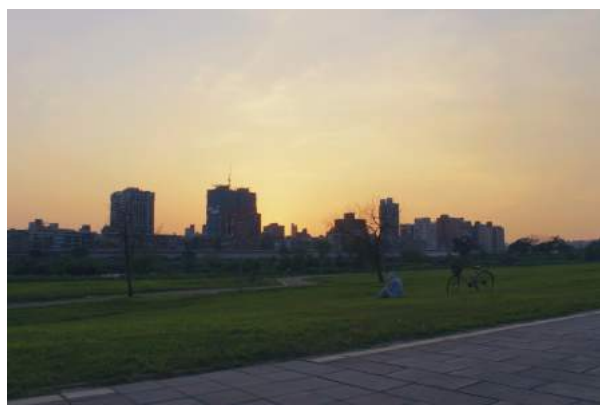


3. HDR histogram(reconstructed HDR)

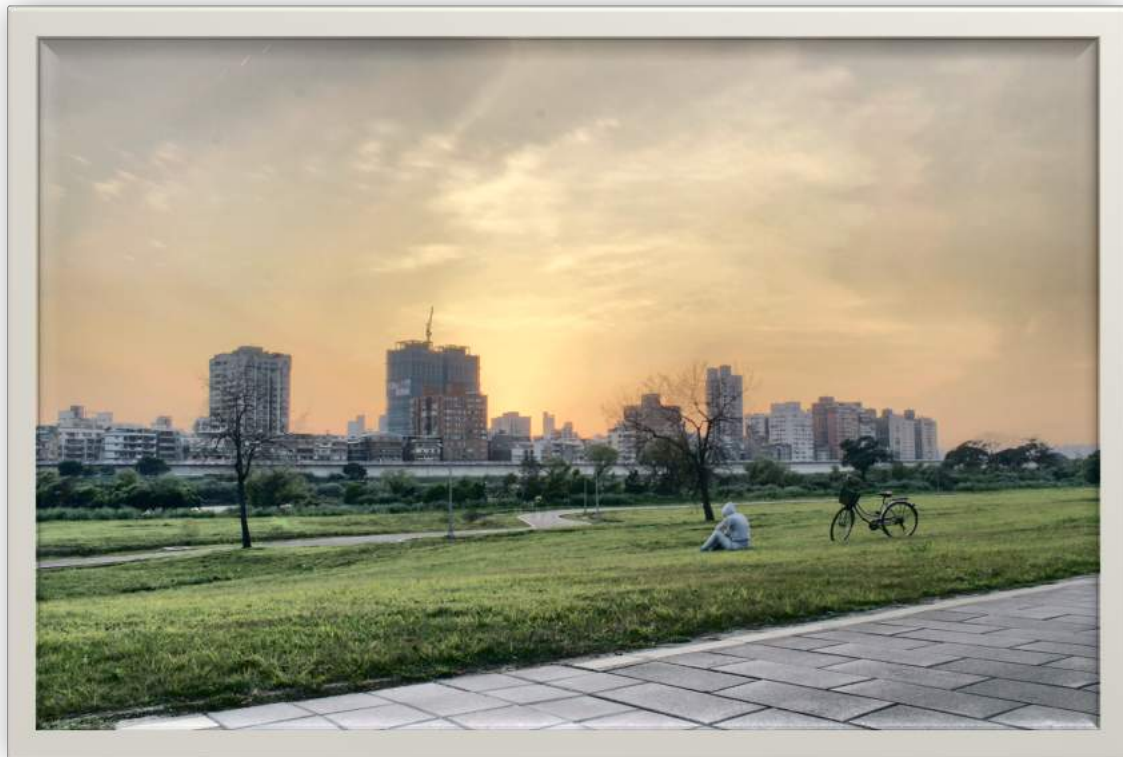


4. Tone mapping result

Our tone mapping	Matlab tone mapping
	

Our tone mapping(low key)	Our tone mapping(middle key)
	

5. *Best result*

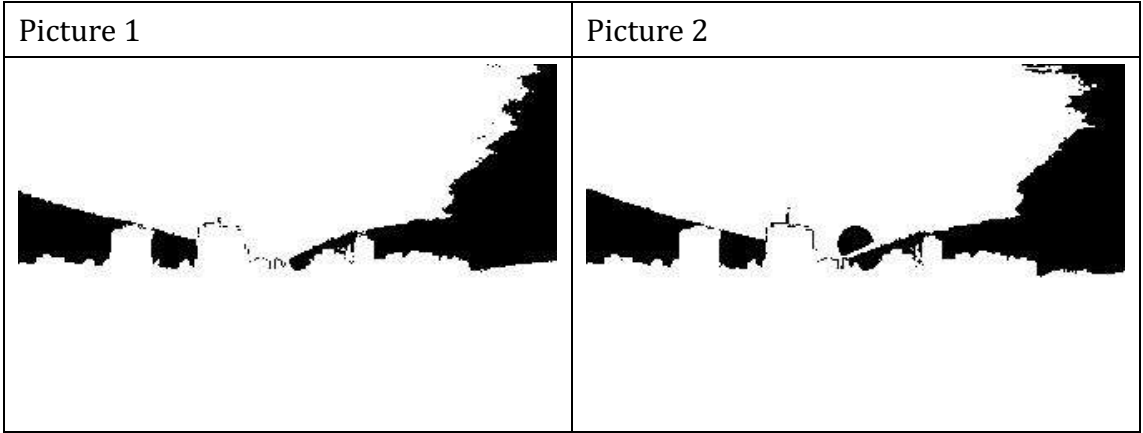


6. *Alignment result*

Original picture 1	Original picture 2
	



Exclusion map:



7. Other results

	Our result	Matlab result
1		
2		
3		



8. 分析與討論

- 1) 我們到學校各處及河濱公園去取景，包含白天的景和晚上的景，從這些照片裡面挑選出最滿意的一張。
- 2) Matlab 的 **tone mapping** 也是利用我們做出來的 **hdr image** 再去轉換，由此可以知道我們所計算出的 **hdr image** 是品質良好的。
- 3) 比較我們做完 **tone mapping** 後的照片和 **matlab** 內建的 **tone mapping**，發現我們所做的結果顏色較為接近原圖，但細節對比度較不強烈，暗部和亮部的細節略顯不足；而 **matlab** 的結果讓亮暗部的細節都能呈現出來，但是顏色略顯偏差，有時細節也過於突出而顯得不自然。比較和討論後，認為是因為我們的 **tone mapping** 沒有做 **local** 的 **dodge and burning**。
- 4) 我們去分析白天的照片和晚上的照片的對比度，上圖(1)為夕陽直射，亮部與暗部對比約為 800:1，上圖(4)為晚上的行政大樓，亮暗對比度約為 8000:1，推測原因為：白天的時候，暗部不會真的很暗，亮部的部分感光元件未能完全記錄下來（太陽過於強烈），夜晚的時候暗部真的夠暗，即使亮的地方能量沒有很強，整體對比度依然很高。
- 5) 我們在做 **alignment** 的時候，一開始遇到沒辦法完全對齊好，後來發現 **alignment** 第一張照片的重要性

七、 Time Analysis

- Number of picture: 6

Size	1556*1037	3112*2074	Analysis
Aignment	23.32s	56.47s	$O(n^2)$
'g' function	0.17s	0.15s	$O(1)$
remap HDR image	43.62s	182.61s	$O(n^2)$
Tone mapping	3.06s	18.38s	$O(n^2)$

n: the width of a picture.

八、 What we have learned from this project?

- 1) 在這次的作業的一開始，我們兩個人都還是 **matlab** 新手，這樣的作業讓我們學習使用 **matlab** 的功能，也體會到它的強大。
- 2) 在上課時聽到相當多神奇的演算法，感覺不容易做到，透過這樣的練習，讓我們知道妥善運用 **matlab** 裡面的工具，可以真的把 **paper** 上的理論運用出來。
- 3) 上課時原本以為 **tone mapping** 的 **local operator** 不會很難實踐，在真正運用上還是遇到相當的困難，也翻出 **paper** 來研讀，但最後沒能成功寫出好的 **tone mapping**，下次應該要多討論並實驗不同的參數。

2016/3/29