



chiahan / vfx-project1-hdr

Watch 2

Star 0

Fork 0

62 commits

1 branch

0 releases

2 contributors



branch: master ▾

vfx-project1-hdr / +



test

chiahan authored Apr 8, 2015

latest commit e403cdf71c

artifact

original files

Mar 30, 2015

image

original files

Mar 30, 2015

instruction

original files

Mar 30, 2015

program

original files

Mar 30, 2015

report

original files

Mar 30, 2015

README.md

test

Apr 8, 2015

README.md

Code

Issues

Pull requests

Pulse

Graphs

HTTPS clone URL

https://github.com

You can clone with [HTTPS](#) or [Subversion](#).

Download ZIP

vfx 2015 Spring @ CSIE.NTU.EDU.TW

project #1: High Dynamic Range Imaging (original link)

程式執行方式

使用matlab2014撰寫，執行program資料夾中的main.m，參數都在main.m中最上方修改，output會有hdr和tone mapping後的結果在program資料夾中

```
% folder: the (relative) path containing the image set.  
% type_: 'global' or 'local' tone mapping  
% phi: used by local tone mapping  
% epsilon: used by local tone mapping (find the max gaussian scale)  
% lambda: smoothness factor for gsolve.  
% prefix: output LDR file's prefix name  
% [srow scol]: the dimension of the resized image for sampling in gsolve.  
% shift_bits: the maximum number of bits in the final offsets in  
% alignment.  
  
function main(folder, type_, alpha_, delta_, white_, phi, epsilon, lambda, prefix, srow,
```

實作內容

1. alignment
2. hdr
3. tonemapping(global operator)
4. tonemapping(local operator)

Algorithm

#1: Alignment (Median Threshold Bitmap)

將圖片RGB資訊轉成灰階，方便後續採取Median Threshold Bitmap方法對位，為了增加對位的精準以及處理速度，把原圖尺寸一直縮小二分之一到最後有七張不同尺寸等級的圖片，以論文裡的演算法對每兩張不同曝光時間的照片互相比較，並將每次位移暫存並移動，其餘部分填上黑色，從最小尺寸等級一直到原尺寸，完成alignment對位。

1. 直接利用matlab指令將圖片RGB資訊轉成灰階
2. 原圖尺寸一直縮小二分之一，到最後每張不同曝光時間的照片一共分別有七張不同尺寸
3. 計算所有圖片的中位數，圖片裡的像素若大於中位數計為1，否則為0。第二種方法為找出所有像素在中位數 ± 4 範圍內，並記為0，其餘記為1。
4. 使用上述兩種方法對每兩張不同曝光時間的圖片進行八個不同方位的比較，算出一個最佳位置
5. 從尺寸最小的圖片往上做比較，進行對位

#2: HDR

1. 我們參考Paul E. Debevec論文中所提到的演算法，最小化目標方程式

$$\mathcal{O} = \sum_{i=1}^N \sum_{j=1}^P \{w(Z_{ij}) [g(Z_{ij}) - \ln E_i - \ln \Delta t_j]\}^2 + \lambda \sum_{z=Z_{min}+1}^{Z_{max}-1} [w(z)g''(z)]^2$$

2. 滿足矩陣 $Ax=b$ ，以SVD求解矩陣 x ，得到 g 函數，帶入權重值及曝光時間 $\ln \Delta t$

$$\ln E_i = \frac{\sum_{j=1}^P w(Z_{ij})(g(Z_{ij}) - \ln \Delta t_j)}{\sum_{j=1}^P w(Z_{ij})}$$

3. 計算多張不同曝光時間同像素的點，找出其算術平均值得到HDR圖像

#3: Tone Mapping (global operator)

1. Tone Mapping 目的為將 HDR 轉為 LDR，假設

$$L_w = 0.27R + 0.67G + 0.06B$$

2. 先在圖像中取得每一點的亮度值並取 \log ，取得平均值再做 exponential

$$\bar{L}_w = \frac{1}{N} \exp \left(\sum_{x,y} \log (\delta + L_w(x,y)) \right)$$

3. 定義 normal-key a 值為 0.18，並代入平均亮度 L_w 至下列方程式求得 L_m

$$L(x,y) = \frac{a}{\bar{L}_w} L_w(x,y)$$

4. 設定場景中最大亮度為 1.5，求得 L_d

$$L_d(x, y) = \frac{L(x, y) \left(1 + \frac{L(x, y)}{L_{\text{white}}^2}\right)}{1 + L(x, y)}$$

5. 最後重新計算LDR每個channel的亮度值

$$\begin{bmatrix} R_d \\ G_d \\ B_d \end{bmatrix} = \begin{bmatrix} L_d \frac{R_w}{L_w} \\ L_d \frac{G_w}{L_w} \\ L_d \frac{B_w}{L_w} \end{bmatrix}$$

#4: Tone Mapping (local operator)

1. 找出一定範圍內，使得所有像素亮度都差不多，沒有任何的sharp contrast，對亮度分佈Lm與Gaussian profile做摺積 (convolution) 轉換，並拿兩張Gaussian blurred images相減，由於要找出V1與V2在相近區域內有相同的亮度梯度 (luminance gradient)，給予一個threshold ε ，流程如下

$$L_s^{blur}(x, y) = L_m(x, y) \otimes G_s(x, y)$$

$$V_s(x, y) = \frac{L_s^{blur}(x, y) - L_{s+1}^{blur}(x, y)}{2^\phi a/s^2 + L_s^{blur}}$$

$$s_{\max} : |V_{s_{\max}}(x, y)| < \varepsilon$$

2. 求得 L_d ，最後重新計算 LDR 每個 channel 的亮度值

$$L_d(x, y) = \frac{L_m(x, y)}{1 + L_{s_{\max}}^{blur}(x, y)}$$

Result

#1: Alignment (Median Threshold Bitmap)

由於原圖之間差異太小，看不出來效果，我們故意拿一張圖位移很多來測試，設定在7層（最大位移為 2^7 ）時會移回來 input: 圖一



圖一位移後



output:



另外，照片之間差異小時，層數設太大反而會因為照片曝光差太多而失敗，通常設定3層可以得到較好的比較結果

#2: HDR

input: f4 2"



f4 1"



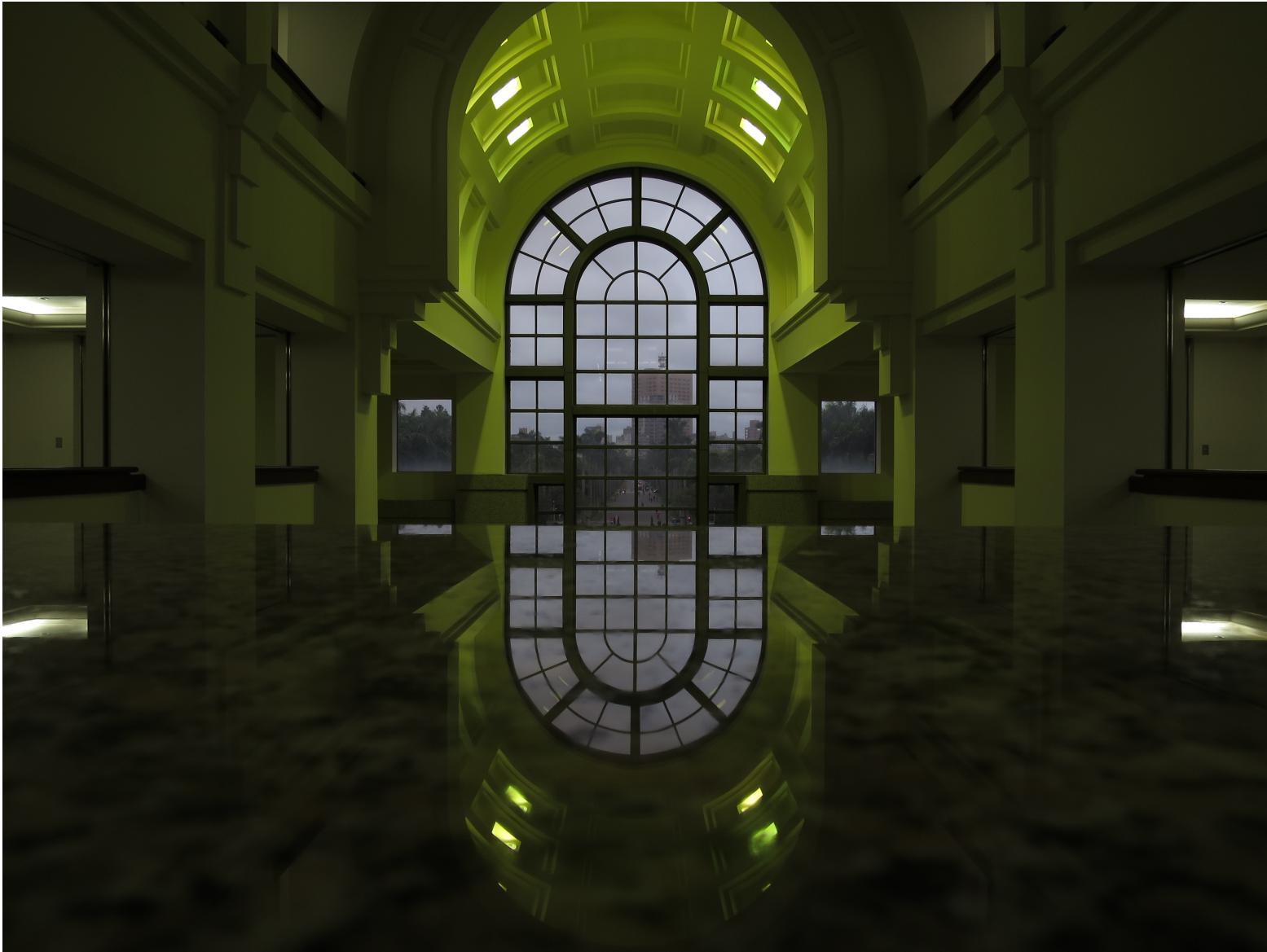
f4 0.5"



f4 1/5



f4 1/8



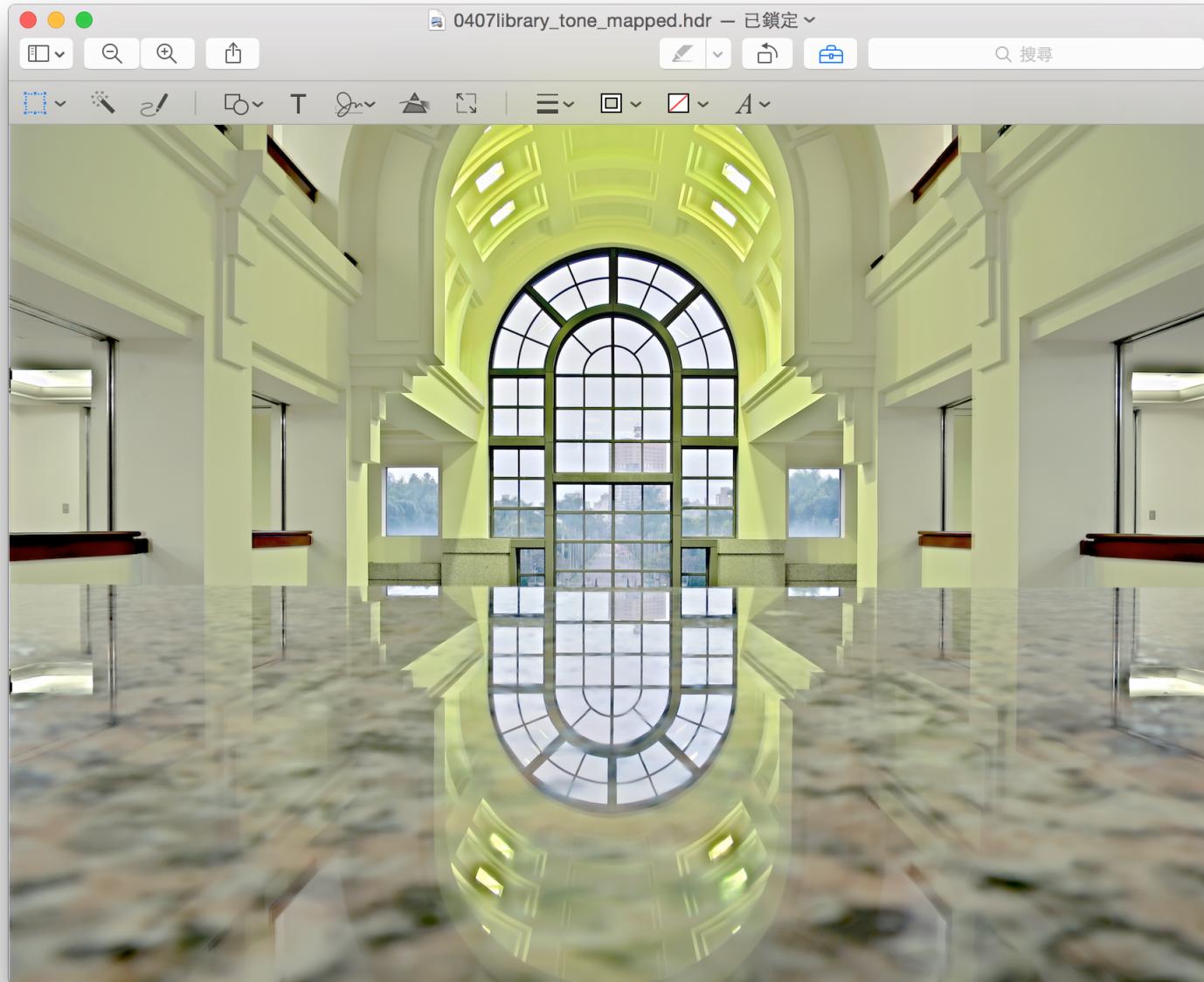
f4 1/15



f4 1/30



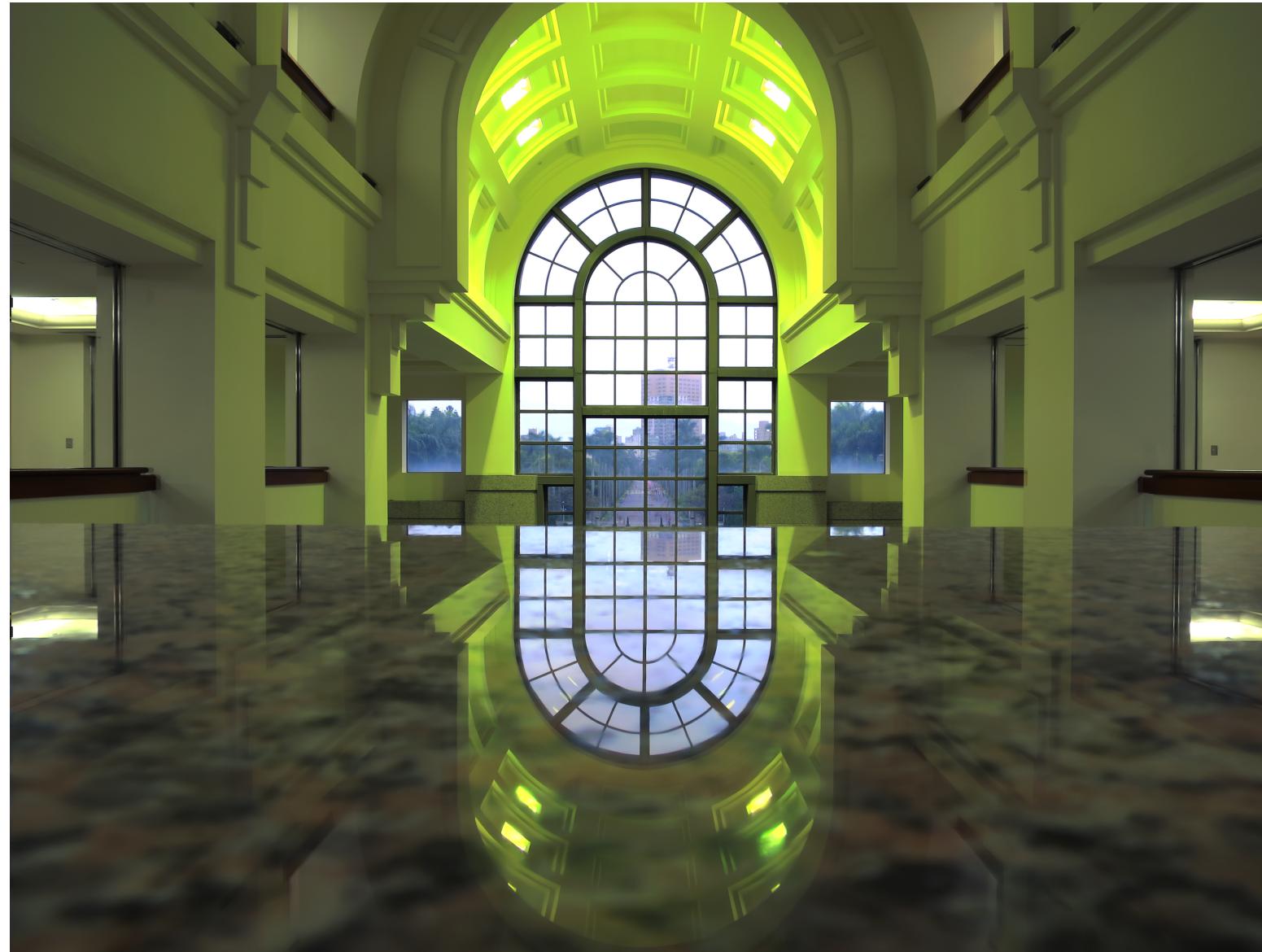
output:



#3: Tone Mapping (global operator)

input:用hdr演算法得到的output(同上圖)

output:a越大圖整體越亮，white越大亮部細節增加，但亮部細節增加的同時暗部細節也會消失，因此需要local的tone mapping來改善



#4: Tone Mapping (local operator)

input: 用hdr演算法得到的output(同上上圖)

output: $a = 0.5$, $\phi = 8$, $\epsilon = 0.05$

可看出和global的結果比起來，local的邊緣較銳利，顏色相近的區域中對比較高



比較：使用 **matlab tone mapping function** 得到的結果（下圖）

整體色調和相機拍到的照片以及hdr檔差很多，從paper實作的結果似乎比較好



