

網頁上呈現高動態範圍影像

作者 Peter Webb

現代的數位相機已可與人眼的色彩感知與空間解析度相互匹敵，但是仍無法像眼睛一樣控制對比度，儀器設備進行錄製或顯影時，影像感測元件或顯示媒介的**對比率**定義為最深黑色與最亮白色之對比，眼睛對比率為 1:100,000 大約是典型數位相機對比率 1:4096 的 24 倍，因此，即使在正確曝光的相片中，陰影通常太暗而明亮區太亮，導致損失大量細節。

透過將對比率與基於觀察者的色彩感知模型結合在一起並結合單一場景的多次曝光，高動態範圍 (HDR) 數位影像系統可提高視覺真實度。它們使用稱為色調映射運算子的數學轉換，在諸如電腦顯示器等低動態範圍 (LDR) 設備上顯示 HDR 影像。

在本文中，我們使用 MATLAB®、Image Processing Toolbox 以及 MATLAB Builder JA 建立 HDR Explorer，一個基於 Web 的試驗測試可用於快速評估不同的 HDR 演算法如何呈現 LDR 影像的平台。本文將說明如何將演算法的數學運算式轉換為 MATLAB 程式，然後顯示如何將該程式轉換為 Web 應用程式。

MATLAB 中的 HDR 影像處理

HDR 影像處理的目的是使用場景的一系列相片，創造出最接近人眼看到的影像。要將相片轉換為 HDR 影像，我們首先從相片中獲得輻射亮度資訊，然後將它映射到單個低動態範圍影像中，利用矩陣運算重建演算法及色調映射，使得它們能夠輕鬆地在 MATLAB 中執行。

重建輻射亮度資訊

單張數位相片的低動態範圍僅包含正確曝光像素的精確輻射亮度資訊，改變曝光可捕捉不同像素集的輻射亮度資訊，為了精確地分析一系列相片以獲得整個場景的輻射亮度資訊，我們必須考慮相機感測元件對不同級別的光線作出反應的方法。數位相機透過**相繼響應函數**將場景的輻射亮度映射到影像像素值中，該函數是結合感測元件位置 i 上的輻射亮度及曝光時間 Δt ，得出像素為 i 的色彩值 C 。

$$C_i = f(E_i \Delta t) \quad (1)$$

在 Debevec 與 Malik 開發的方法中，輻射亮度重建過程對曝光組進行採樣，以決定逆向相繼響應函數， g 為可逆相機函數的自然對數， Z_{min} 與 Z_{max} 為像素值界限， N 為採樣數， P 為相片數， w 為促使 g 趨於平滑的 'hat' 權重函數。為了確定 g 的範圍，我們將目標函數最小化：

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

系數矩陣初始化之後，僅需使用兩行 MATLAB 程式碼即可使用高斯消去法求解此系統方程式。

$$X = A \setminus b; \quad g = X(1:256);$$

我們透過查詢表決定 g 值，在每次曝光中使用像素相機響應函數的加權平均值，結合多次曝光重建影像真實亮度：

$$\ln E_i = \frac{\sum_{j=1}^P \sum_{i=1}^N w(C_{ij}) [g(C_{ij}) - \ln \Delta t_j]}{\sum_{j=1}^P w(C_{ij})} \quad (3)$$

註： N 是每張影像中像素的總數，不是採樣像素的數量。

以 MATLAB 程式碼表達方程式 (3) 中的分子是很簡單的

```
for j = 1:P
    relRad = relRad + weightFcn(C(j)) .*
        (filmResponse(C(j)) - logExposure(j));
end
```

在此處，for 循環(for loop)取代了外部加法運算子。在 MATLAB 程式碼中，數學運算子自動對其矩陣輸入進行疊代，使得內含內部加法運算子於整個程式碼中。將這些數學方程式轉換為有用的算法需要大約 200 行 MATLAB 程式碼。

MATLAB 內建提供一豐富的函式庫，以簡化該操作。例如，載入 JPEG 影像僅僅需要呼叫單一函式，如同從影像中繼資料中擷取曝光資訊一樣。

色調映射輻射亮度資料

色調映射將大範圍 HDR 資料的動態範圍壓縮成顯示設備的較小動態範圍，一個好的演算法能產生與原場景類似的知覺影像，由於亮度很大程度上控制我們的對比度感知，大多數色調映射演算法適用於 HDR 影像的亮度通道，最簡單的壓縮形式是線性縮放，涉及縮放 HDR 資料庫的亮度值，以符合輸出設備的亮度範圍。但是，線性縮放去掉了大量細節（圖 1C）。

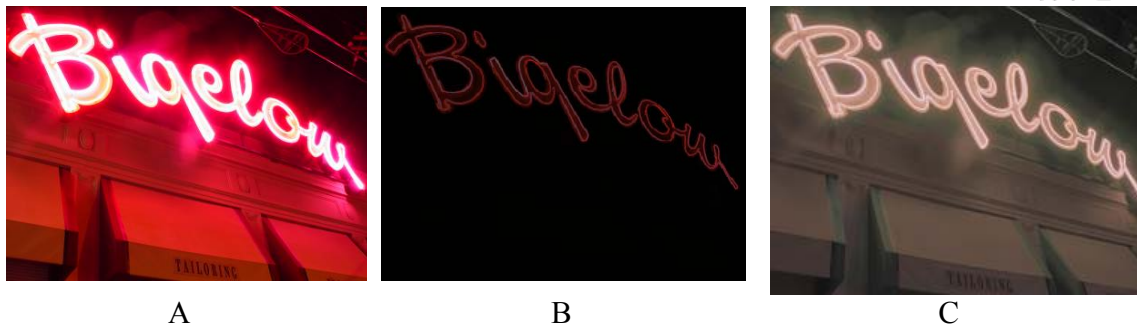


圖 1：單次 LDR 曝光 (A)、線性縮放 (B)、對數壓縮 (C)。

由於人眼對亮度的反應具有對數性質，所以對數變換會得到更加可辨認的影像，在標準化之前得出色彩值的自然對數可將許多 HDR 值轉移到 LDR 設備可存取的範圍內。此方法恢復了很多細節，但是降低了顏色的飽和度及對比度。許多色調映射演算法在對數運算中操作，每個運算不同地解決對數壓縮不足的問題。

Ward 提出的對比限制調適的直方圖均化幫助維持 HDR 影像與 LDR 影像之間的感知等效性，恢復 HDR 資料表現場景中的亮度而非影像中感知的色彩，Ward 的色調映射演算法利用如同人間對光線處理進行對數，為能完全利用可用的動態範圍，我們透過直方圖均衡化增加影像的對比度，並透過對數亮度壓縮恢復飽和度及對比度。

色調映射演算法執行五個步驟：

1. 計算 HDR 影像的對數並將它標準化為 [0 1] 的區間，從而將 HDR 資料的動態範圍壓縮為 LDR 設備可顯示的區間。
2. 將標準化的影像轉換為 CIE L*a*b* 色彩空間，可在該空間中直接調整亮度。
3. 對亮度通道採用對比限制調適的直方圖均化。
4. 根據使用者特定參數調整影像對比度。
5. 根據使用者要求增加或減少色彩飽和度。

使用 MATLAB 與 Image Processing Toolbox，我們可在七行程碼中執行這些步驟：

```
% Compute normalized log of HDR data
hdr = lognormal(hdr);

% Convert from the sRGB color space to the CIE L*a*b* colorspace
Lab = sRGB2Lab(hdr);
Luminance = Lab(:, :, 1);

% Increase contrast via histogram equalization
Luminance = adapthisteq(Luminance, 'NumTiles', numtiles);

% Adjust contrast (ignore shadows, clip highlights)
Luminance = imadjust(Luminance, LRemap, [0 1]);

% Restore color saturation by increasing a* and b* channels
```

```
Lab(:,:,2) = Lab(:,:,2) * saturation;
Lab(:,:,3) = Lab(:,:,3) * saturation;
```

比較圖 1C 的對數縮放與圖 5 中的色調映射影像，顯然，色調映射對數可產生更加逼真的影像，每個影像的直方圖範例說明了原因，以對數縮放的影像的直方圖為鋸齒形（圖 2），指出損失了平滑的色調變化，同時沿著亮度軸壓縮，浪費了可用的動態範圍。

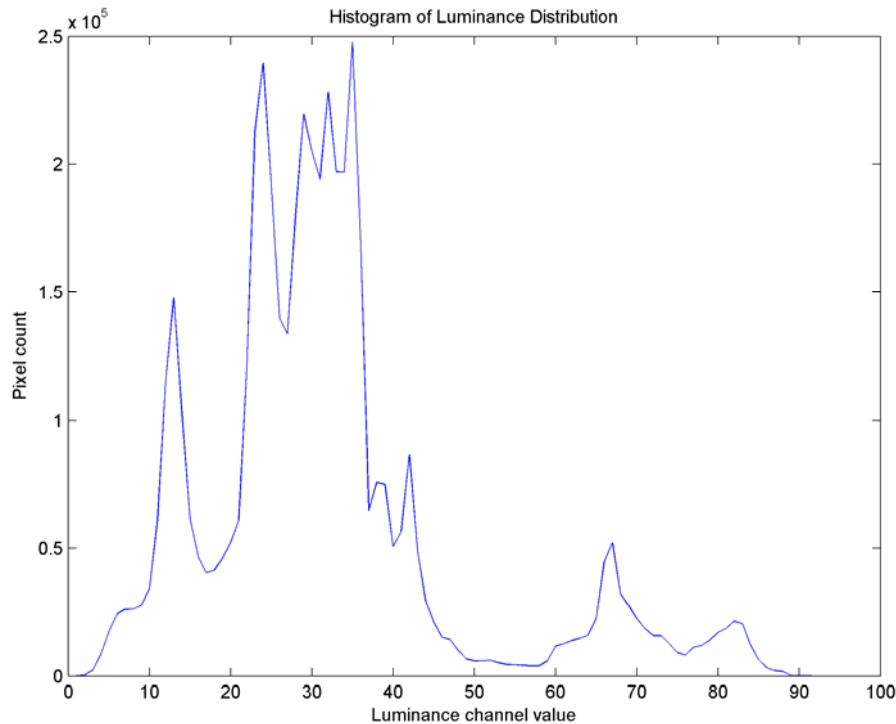


圖 2：從圖 1：單次 LDR 曝光 (A)、線性縮放 (B)、對數壓縮 (C)。的影像 C 中得到的對數壓縮 HDR 資料的亮度。

色調映射影像的直方圖（圖 3）不同於對數縮放影像的直方圖（圖 2）。相同區域中都有兩個高峰，但是 LDR 直方圖延伸到了整個 x 軸，而 HDR 直方圖有幾乎

10% 的 x 軸未使用。

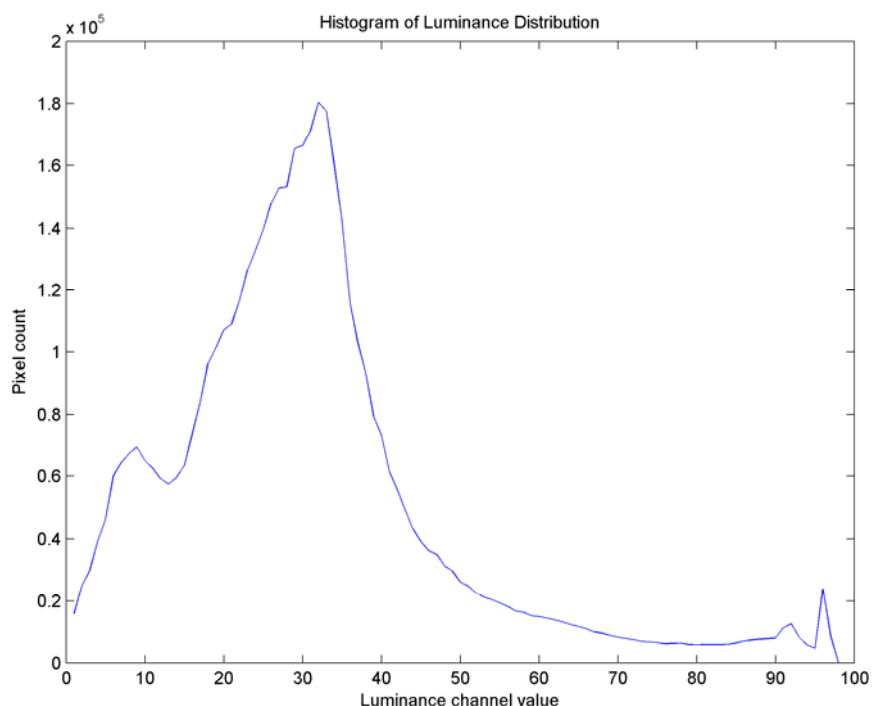


圖 3：色調映射影像的直方圖。

由於影像在中低色調範圍包含相對較多的黑色像素於 LDR 直方圖，而不是該範圍兩端的平滑形狀，顯示出加亮區或陰影都沒有被削減，透過比較圖 1C 與圖 5 HDR 資料與色調映射 LDR 影像，可清楚地發現這一點。圖 3 中直方圖的平滑度顯示出色調映射避免了色調分離（色彩空間壓縮造成的影像斑點）並在影像色調中恢復了細微變化

圖 4 顯示了高對比度夜間場景中的一系列 10 LDR 影像的邊界影像之最小與最大曝光範圍，合併全部 10 張影像以構成 HDR 資料組，然後對此資料組進行色調映射，以便電腦顯示器能夠產生如圖 5 中的影像，色調映射影像比邊界影像擁有更多細節，並且色彩逼真度更加精確，進一步處理會減少光圈效應，鏡頭炫光及直方圖均化的邊緣增強屬性在某種程度上導致了該效應。



圖 4：最小與最大 LDR 曝光範圍（最多 10 個）。



圖 5：HDR 資料，色調映射道 LDR 影像。

部署 HDR Explorer 作為網頁應用程式基於 MATLAB 之網頁應用程式要求可用於 MATLAB 算法的伺服器端介面的 GUI 瀏覽器，以及整合兩者的 Java Script。HDR

Explorer 應用程式的 GUI 瀏覽器包含一個 JavaServer Pages™ (JSP) 表單、Java servlet 的伺服器端介面以及 MATLAB Builder JA 產生的 Java API，Java Script 功能能夠手工編寫整合程式碼，此應用程式提供兩個功能：LDR 影像組預覽與 HDR 資料產生以及色調映射。

設計 GUI 瀏覽器

我們的 HDR Explorer 應用程式需要一個指定輸入影像的方法、調節變換參數的 GUI 以及儲存色調映射 LDR 輸出結果的位置。為了簡便起見，該應用程式只接受 JPEG 影像並要求每個 LDR 曝光組擁有一個獨立的目錄，GUI 包含數個資料輸入欄位、兩個按鈕以及一個影像顯示區域（圖 6）。使用者定義包含 LDR 影像組的目錄之後，HDR Explorer 應用程式顯示該組的中等曝光影像，使用者可調整 HDR 的建立及色調映射參數，然後建立並顯示自動儲存為 TIFF 檔案的色調映射 LDR 影像。

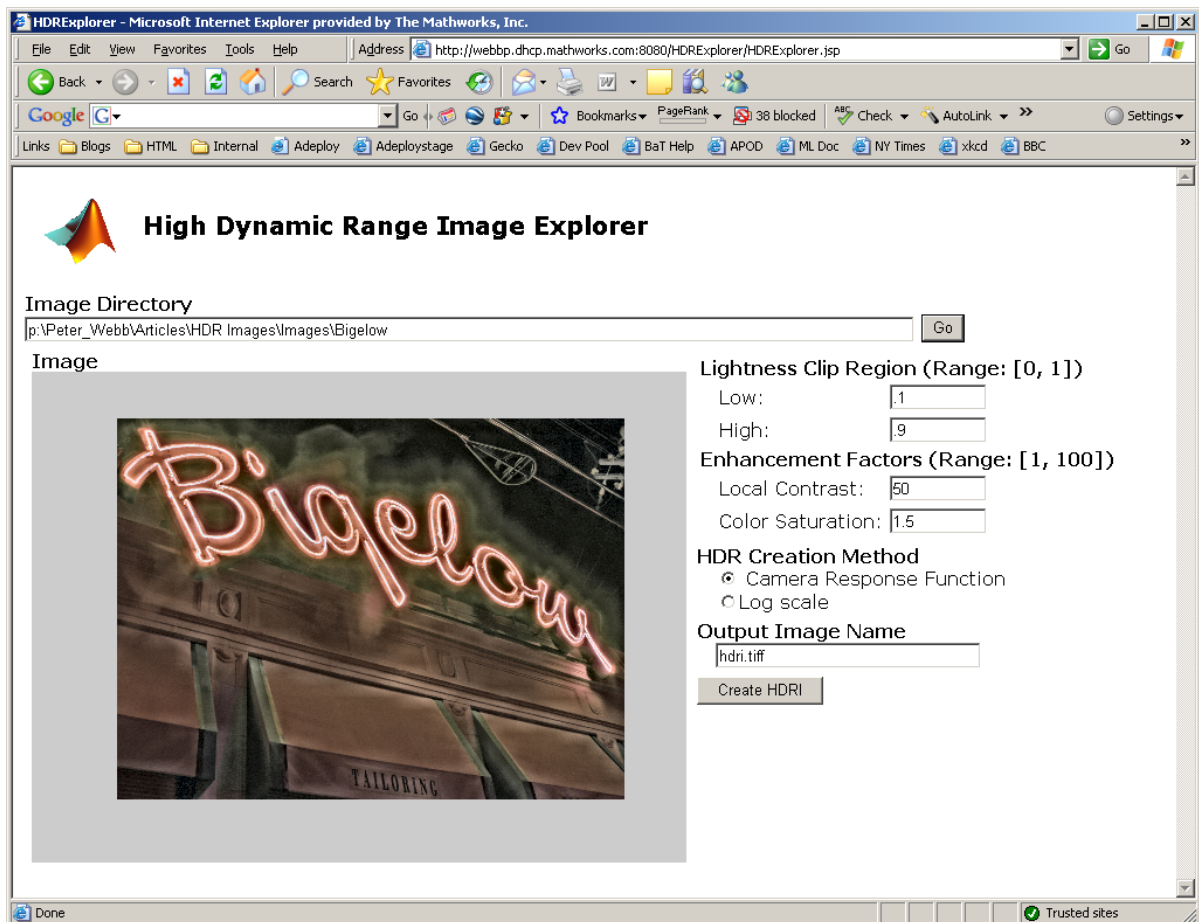


圖 6：Web 介面：包含 MATLAB Web 圖的 JSP 表單。

GUI 作為一個實驗 HDR 資料建立以及色調映射演算法的前端，它非常簡單，因此可執行手工編碼，使用原始 HTML GUI 元件來建立選擇按鈕或其他選取器，以便在 HDR 影像建立演算法與提交 JSP 表單的 POST 方法之間進行選擇。

建立 Java API

我們利用 MATLAB Builder JA 自動從 MATLAB HDR 演算法建立 Java API。下列命令透過兩個高階包裝函數- previewImageStack 與 jpg2hdr，建立匯出 HDR 算法的 Java 元件：

```
>> mcc -v -W "java:com.mathworks.HDRExplorer,HDRExplorer" jpg2hdr.m  
getWebFigure.m previewImageStack.m
```

MATLAB Builder JA 可確實處理所有棘手的工作：確定哪些函數包括在元件中、產生可直接從 Java 中呼叫 MATLAB HDR 函數的函數以及將元件封裝到 JAR 檔案中，這些產生的函數還控制可重新分配的 MATLAB 運行時間，並自動在原始 MATLAB 與 Java 類型之間轉換資料，安裝在伺服器後，JSP servlet 可存取它們。

整合瀏覽器 GUI 與 MATLAB 演算法

HDR Explorer GUI 將參數傳送至 previewImageStack 與 jpg2hdr，並能控制顯示全部 MATLAB 圖形的 MATLAB Builder JA Web 圖，透過 MATLAB Builder JA，JSP 表單產生 HDRExplorer 與 Java Bean 的包裝類別連結 MATLAB 運行時間。

例如，在 jspInit 中建立 componentObject (HDRExplorer 類型) 包裝類別實例之後，該頁面內嵌一個 scriptlet 以便在使用者按 **Create HDRI (建立 HDRI)** 按鈕 (該按鈕在 JSP 表單的 HTML 中名為 SubmitHDR) 時能夠啓用 jpg2hdr：

```
if (request.getParameter("SubmitHDR") != null)  
{  
    MWCellArray flags = new MWCellArray(2,1);  
    flags.set(1,"Show");  
    flags.set(2,"ReturnWebFigure");  
  
    Object[] result =  
        componentObject.jpg2hdr(  
            1, // One output -- a structure array  
            guiState.getDirectory(),  
            guiState.getOutputName(),  
            guiState.getCreateHDRGroup(),  
            flags, 5, 250,  
            guiState.getLowLight(),  
            guiState.getHighLight(),  
            guiState.getSaturation(),
```



```
        guiState.getContrast(),  
        figureHandle);  
  
    UpdateWebFigure(result, "webfig");  
}
```

此處的 `guiState` 是 Java Bean，用於在用戶端與伺服器之間傳遞資料，`WCellArray` 類型是用於 MATLAB 單元陣列資料類型的 Java 鏈結 (Java proxy) (單元陣列是 N 維名稱與值關聯清單的概括，類似 C++ 標準範本庫類型 `map`)。當 `jpg2hdr` 修改在 Web 圖中顯示的影像，`UpdateWebFigure` 通知用戶端 Web 圖從伺服器中獲取序列影像資料。

HDR 影像處理演算法是基於矩陣且高度數學特性，使得它們很容易在 MATLAB 中執行，其數學公式通常逐一條件地映射到 MATLAB 運算式或陳述式中，MATLAB Builder JA 透過自動產生瀏覽器可調用的 Java API，簡化了將 MATLAB 程式轉換為 Web 基礎的應用程式的過程，利用 MATLAB Builder JA 與您喜歡的 HTML 或 servlet 開發工具，您可獲得編程人員協同作業平台，向網際網路中的任何桌上型電腦傳送數學功能豐富的應用程式。

致謝

感謝 Jeff Mather 向我介紹用於建立及處理 HDR 影像的技巧以及他對 Ward 的色調映射運算子的執行，還要感謝 Newton MA 的 Bigelow Cleaners 允許我使用他們著名的相片。

更多資訊

Debevec, P. and Malik, J., "Recovering High Dynamic Range Radiance Maps from Photographs," in *Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, pg. 369-378, ACM SIGGRAPH, 1997

MATLAB Builder Java JA Users Guide

Reinhard, E., Ward, G., Pattanaik, S., and Debevec, P., *High Dynamic Range Imaging*, Morgan Kaufmann, Boston, 2006

Ward, G., Rushmeier, H., and Piatko, C., "A Visibility Matching Tone Reproduction Operator for High Dynamic Range Scenes," Lawrence Berkeley National Laboratories, <http://radsite.lbl.gov/radiance/papers/>.