



夜間影像強化

學生: 陳信樺

蘇建穎

指導老師: 黃敬群

專題成果報告

- 主要問題
- 使用方法
- 結果比較

專題成果報告

- 主要問題
- 使用方法
- 結果比較

主要問題

亮度不足的影像無法清楚辨識



夜間影像強化

- 主要問題
- 使用方法
- 結果比較

伽瑪校正

傳統的伽瑪校正公式：

$$Output = 255 \left(\frac{Input}{255} \right)^\gamma$$

其中 Input 是輸入的 RGB 值

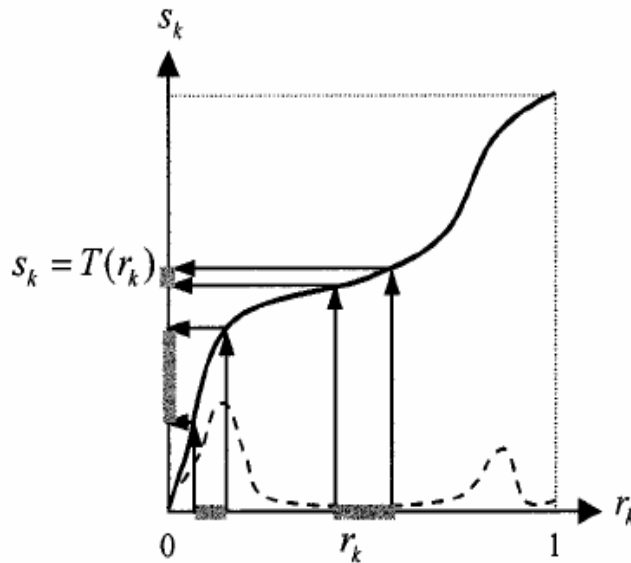
伽瑪校正



原圖

結果

直方圖均衡化



$$S_k = T(r_k) = \sum_{j=0}^k P_r(r_j) = \sum_{j=0}^k \frac{n_j}{n}$$

$$0 \leq r_k \leq 1 \text{ and } k = 0, 1, \dots, L-1$$

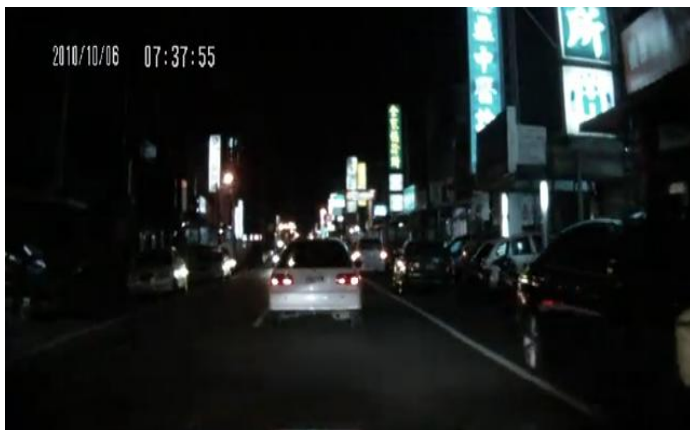
Transformation function for histogram equalization.

$P_R(r_j) = \frac{n_j}{n}$: 輸入圖像的 j 階層的概率密度函數 (PDF)

n : 在輸入圖像的總像素數

n_j : j 階層的輸入像素數

直方圖均衡化



原圖

結果

Retinex

- 演算法基本概念:

$$I'(x_i) = I(x_i) + \log(\rho^{x_i}) - \log(\rho^{x_j})$$

$I'(x_i)$ 為影像中的像素

$\log(\rho^{x_i})$ 為計算中像素的照明參數

$\log(\rho^{x_j})$ 為計算中像素周圍的像素的照明參數

Retinex

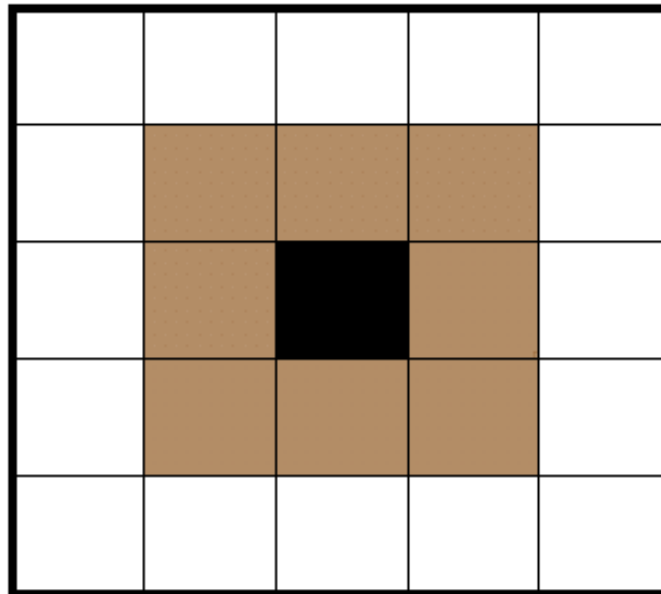
- 演算法:
 1. SSR (single scale Retinex)
 2. MSR(multi-scale Retinex)
 3. MSRCR (multi-scale Retinex with color restoration)
 4. IMSR(multi-scale Retinex)

Retinex

- SSR , MSR -- 計算灰階影像物體間相對的明度
- MSRCR -- 計算彩色影像物體間R,G,B相對的彩度
- IMSR -- 計算單獨提出彩色影像物體間的明度

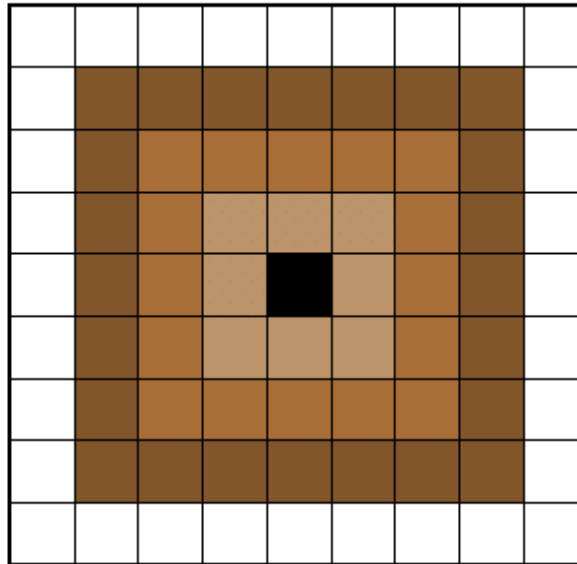
Retinex

- SSR (single scale Retinex)-單一濾波



Retinex

- MSR(multi-scale Retinex)-多個濾波



Retinex-SSR

$$R(x, y) = \log I(x, y) - \log[F(x, y) * I(x, y)]$$

$$F(x, y) = Ke^{\frac{-(x^2 + y^2)}{c^2}}$$

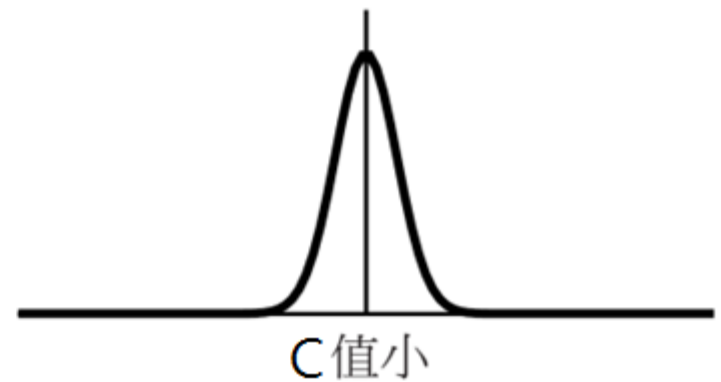
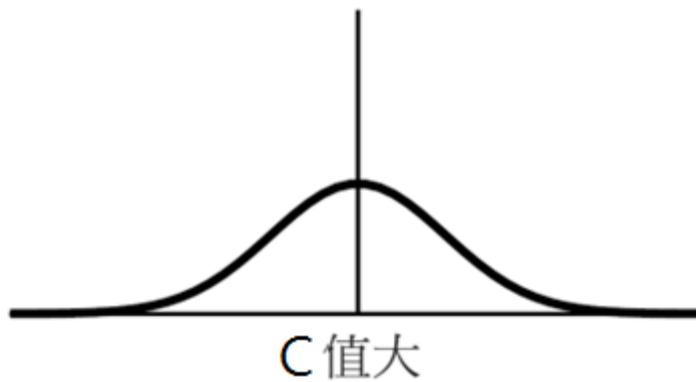
C 值影響影像輸出結果

$$\iint F(x, y) dx dy = 1$$

Retinex-SSR

缺點

- C值↑ : 包含較多Pixel , 保留較多資訊
- C值↓ : 包含較少Pixel , 保留較少資訊





原圖



$C = 2$



$C = 150$

Retinex-SSR

Problem

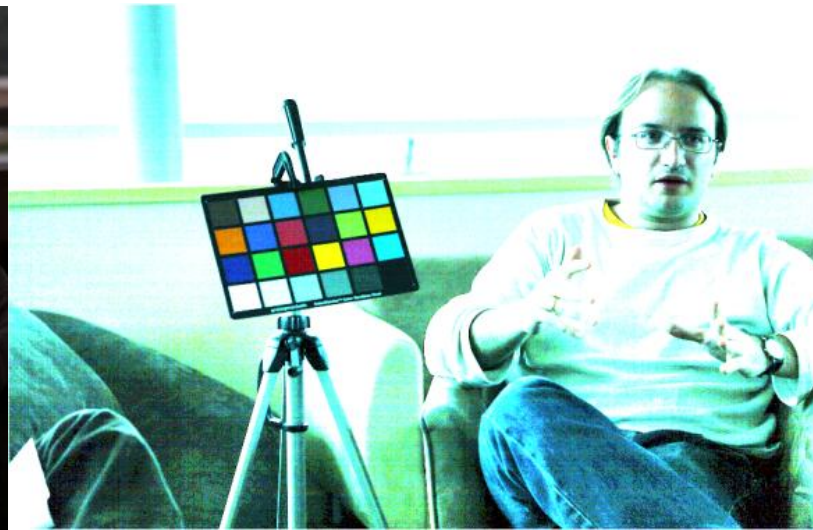
無法同時強化邊緣資訊跟保留較多的影像資訊。

Retinex-MSR

$$R(x, y) = \sum_{k=1}^K W_k (\log I(x, y) - \log [F(x, y) * I(x, y)])$$

W_k : 權重值 (Sum=1)

Retinex-MSR



原圖

結果

Retinex-MSR

缺點：

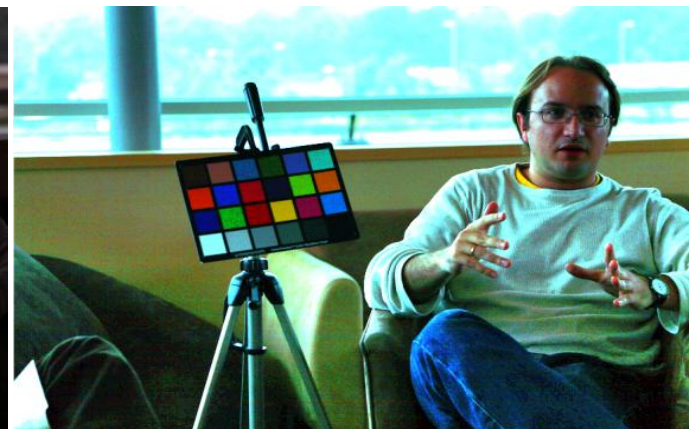
- 輸出時，顏色重現可能無法呈現出正常的顏色，而且看起來會不自然。
- 若是原始影像有偏向某個主色調，在經過**MSR** 處本理後，反而會有色偏的情況發生。
- 會連雜訊一起強化

Retinex-MSRCR

$$R'_{MSR_i}(x, y) = R_{MSR_i}(x, y) \times I'_i(x, y)$$

$$I'_i(x, y) = \beta \log(\alpha \frac{I_i(x, y)}{\sum_{i=1}^3 I_i(x, y)})$$

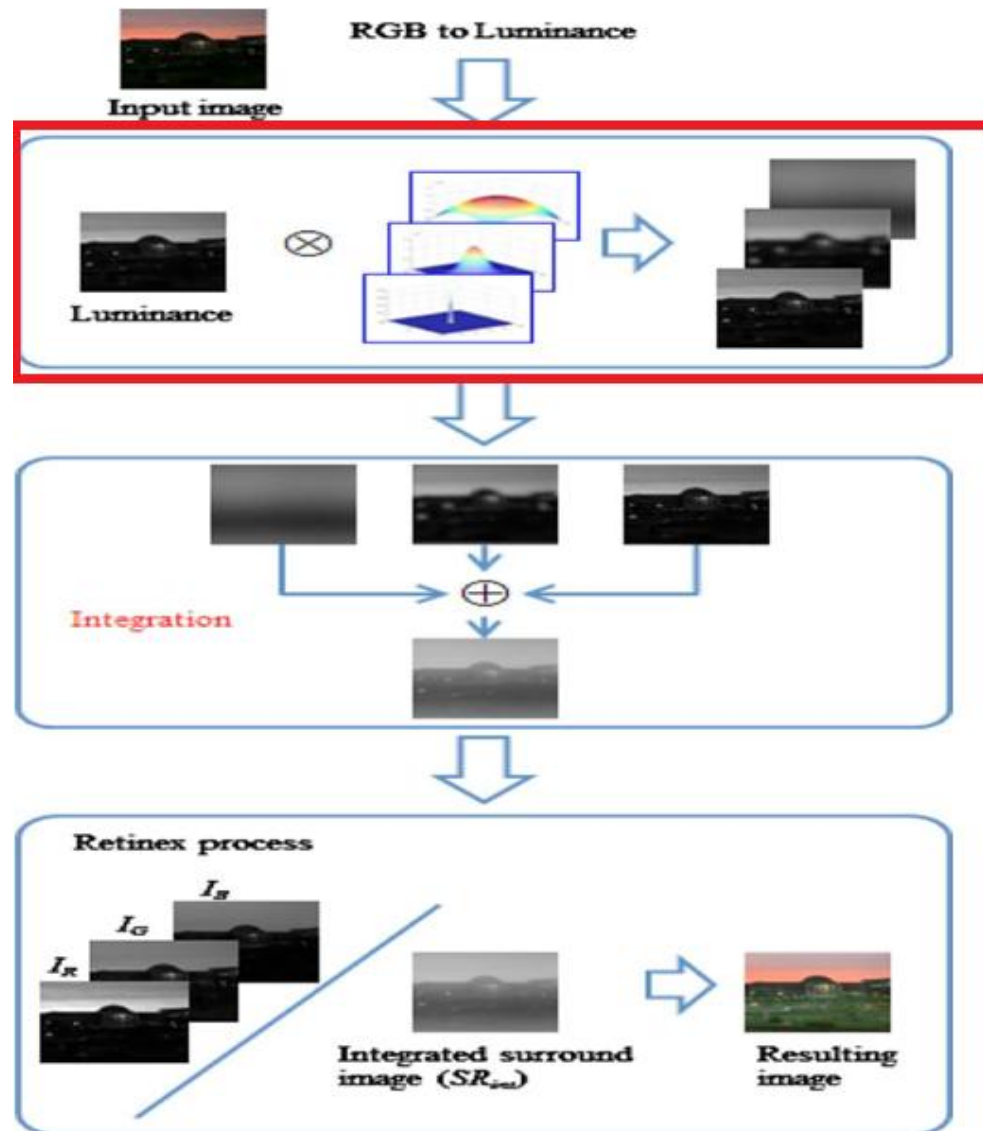
Retinex-MSRCR



原圖

結果

Retinex-IMSR



IMSR - 增強亮度

$$SR_m(x, y, \sigma_m) = G_m(x, y) \otimes Y(x, y),$$

$$G_m(x, y) = K_m \exp(-(x^2 + y^2)/\sigma_m^2),$$

$$\sum_{m=1}^M w(\sigma_m) = 1,$$

$$\iint G_m(x, y) dx dy = 1$$

$Y(x, y)$: 亮度圖像

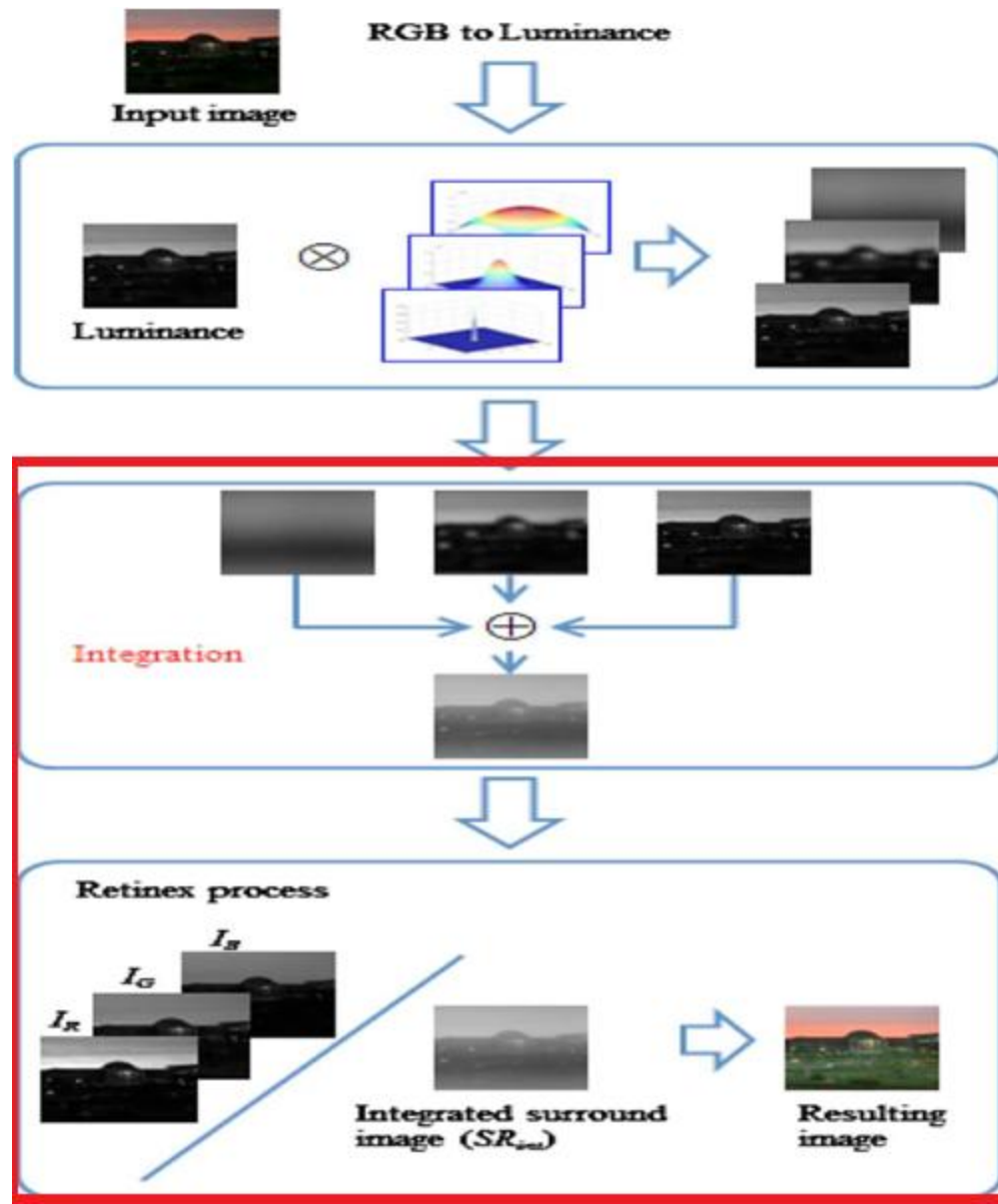
$G_m(x, y)$: 高斯濾波器

σ_m : 標準偏差

$w(\sigma_m)$: 自適應權值

SR_m : 環繞圖像

Retinex-IMSR



IMSR - 增強亮度

$$SR_{merge}(x,y) = \sum_{m=1}^M w(\sigma_m) SR_m(x,y,\sigma_m).$$

$$IR_{merge}(x,y) = \alpha \frac{I_i(x,y)}{SR_{merge}(x,y)}$$

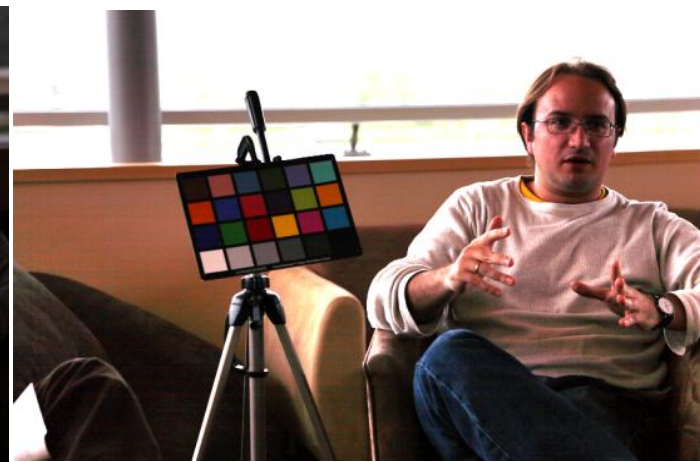
$SR_{merge}(x,y)$: 綜合環繞圖像

$I_i(x,y)$: 從輸入圖像的每個RGB通道

$IR_{merge}(x,y)$: 計算結果

α : 增益係數

Retinex-IMSR



原圖



結果

Retinex-IMSR 簡化版

$$S_{blur}(x, y) = B(x, y) \otimes Y_D(x, y)$$

$$GM = \frac{\sum_{n=0}^N \sum_{m=0}^M S_{blur}(n, m)}{NM}$$

S_{blur} = 模糊影像(亮度)

GM = 全域平均值

Y_D = 亮度通道

NM = 圖像總像素

Retinex-IMSR 簡化版

$$SR_{BM}(x, y) = A \frac{I_i(x, y)}{SGM_{blur}(x, y)}$$

$$SGM_{blur}(x, y) = w_1 S_{blur}(x, y) + w_2 GM$$

SGM_{blur} = 綜合圖像

$w(x)$ = 自適應權值

$I_i(x, y)$: 從輸入圖像的每個RGB通道

A : 增益係數

Retinex-IMSR 簡化版

伽瑪校正查找表

Y values	Gamma values	iGamma values
0	0	0
1	12	0
2	21	0
3	28	0
4	33	1
⋮	⋮	⋮
253	254	1005
254	254	1014
255	254	1024

期中專題成果報告

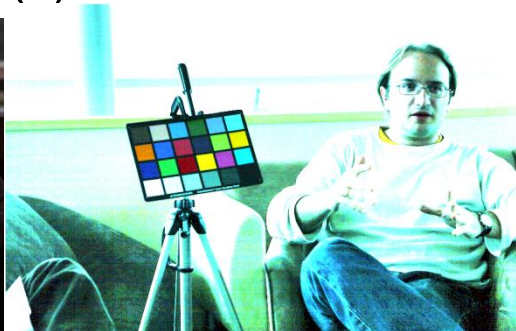
- 主要問題
- 使用方法
- 結果比較

結果比較

(a)原始影像



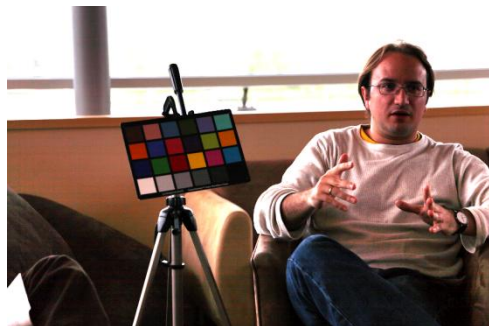
(b)MSR影像



(c)MSRCR影像



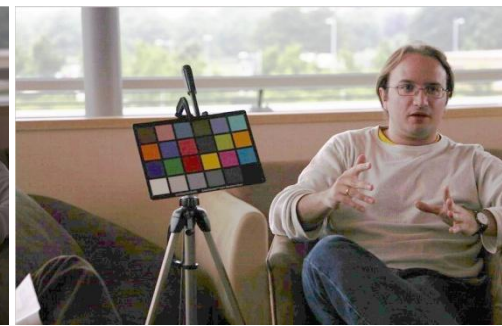
(d)IMSR影像



(e)伽瑪校正影像



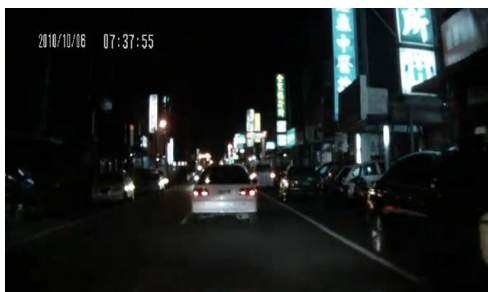
(f)直方圖均衡化影像



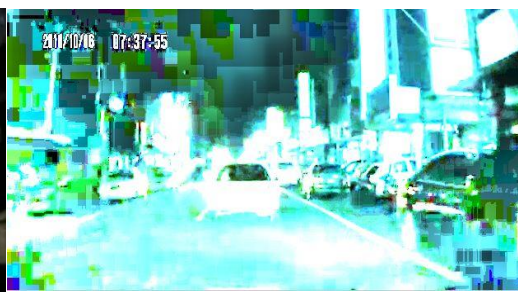
(a)為原本輸入影像 (b)為執行MSR結果 (c)為執行MSRCR結果 (d)為執行IMSR結果
(e)為執行伽瑪校正結果 (f)為執行直方圖均衡化結果

結果比較

(a)原始影像



(b)MSR影像



(c)MSRCR影像



(d)IMSR影像



(e)伽瑪校正影像



(f)直方圖均衡化影像



(a)為原本輸入影像 (b)為執行MSR結果 (c)為執行MSRCR結果 (d)為執行IMSR結果
(e)為執行伽瑪校正結果 (f)為執行直方圖均衡化結果

結果比較

(a)原始影像



(b)MSR影像



(c)MSRCR影像



(d)IMSR影像



(e)伽瑪校正影像



(f)直方圖均衡化影像



(a)為原本輸入影像 (b)為執行MSR結果 (c)為執行MSRCR結果 (d)為執行IMSR結果
(e)為執行伽瑪校正結果 (f)為執行直方圖均衡化結果

結果比較

(a)原始影像



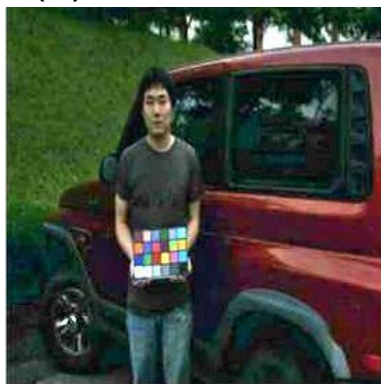
(b)MSR影像



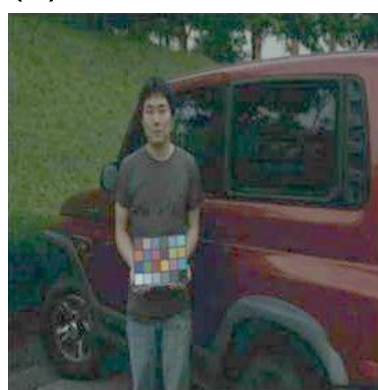
(c)MSRCR影像



(d)IMSR影像



(e)伽瑪校正影像



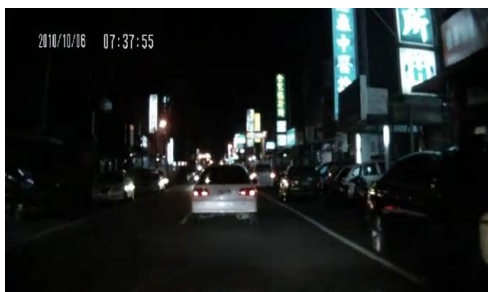
(f)直方圖均衡化影像



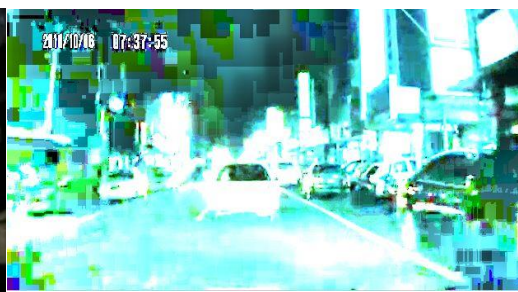
(a)為原本輸入影像 (b)為執行MSR結果 (c)為執行MSRCR結果 (d)為執行IMSR結果
(e)為執行伽瑪校正結果 (f)為執行直方圖均衡化結果

結果比較

(a)原始影像



(b)MSR影像



(c)MSRCR影像



(d)IMSR影像



(e)伽瑪校正影像



(f)直方圖均衡化影像

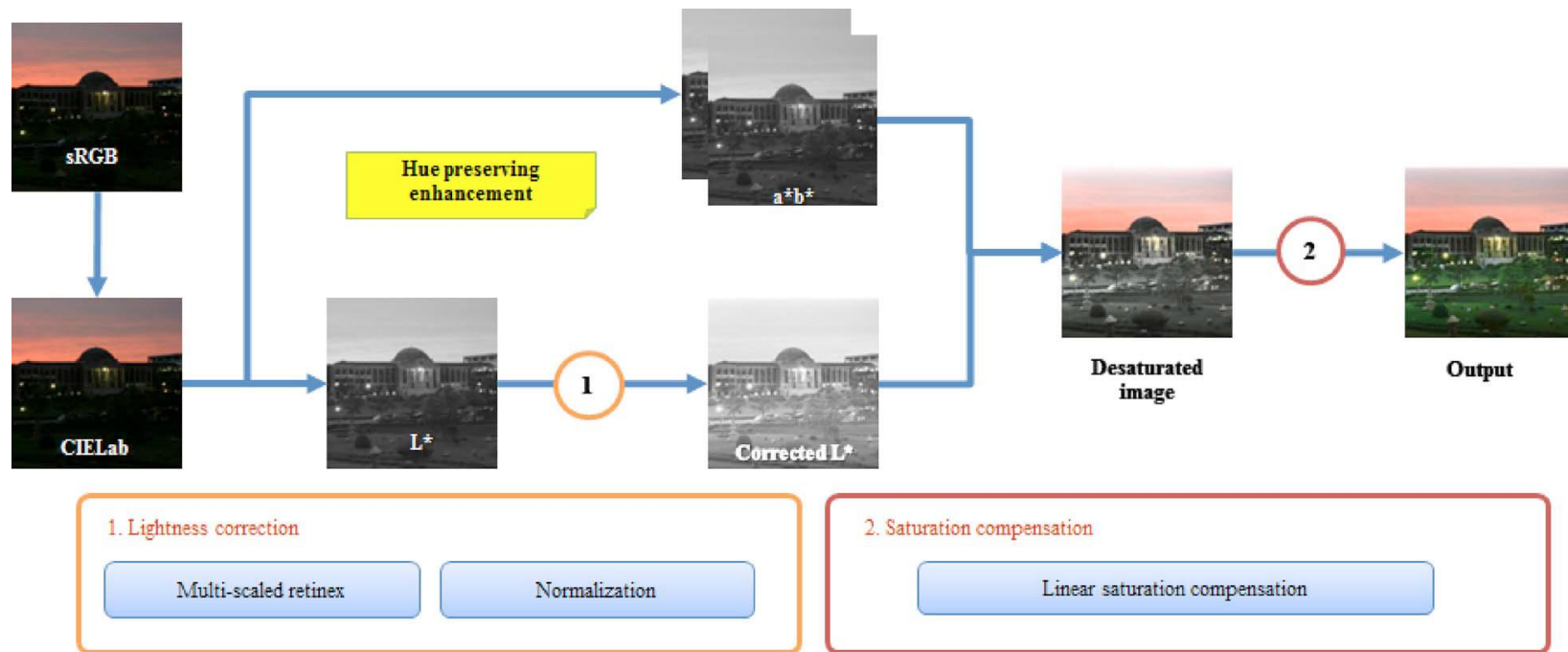


(a)為原本輸入影像 (b)為執行MSR結果 (c)為執行MSRCR結果 (d)為執行IMSR結果
(e)為執行伽瑪校正結果 (f)為執行直方圖均衡化結果

建議色調還原的方法

- 亮度增強
- 亮度校正
- 簡單圖像擴展色域飽和補償

建議色調還原的方法



建議色調還原的方法

- 亮度增強
- 亮度校正
- 簡單圖像擴展色域飽和補償

亮度增強

輸入圖像轉化成**CIEXYZ**排序優先排列

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0.4124 & 0.3576 & 0.1805 \\ 0.2126 & 0.7152 & 0.0722 \\ 0.0193 & 0.1192 & 0.9505 \end{bmatrix} \begin{bmatrix} R_{sRGB} \\ G_{sRGB} \\ B_{sRGB} \end{bmatrix}.$$

CIEXYZ值轉化成**CIELAB**值

$$L^* = \begin{cases} 116 \left(\frac{Y}{Y_n} \right)^{1/3} - 16 & \text{if } \frac{Y}{Y_n} > 0.008856 \\ 903.3 \left(\frac{Y}{Y_n} \right) & \text{if } \left(\frac{Y}{Y_n} \right) < 0.008856 \end{cases}$$

亮度增強

$$a^* = 500 \left(f\left(\frac{X}{X_n}\right) - f\left(\frac{Y}{Y_n}\right) \right),$$

$$b^* = 200 \left(f\left(\frac{Y}{Y_n}\right) - f\left(\frac{Z}{Z_n}\right) \right),$$

$$\text{where } f(x) = \begin{cases} x^{1/3} & \text{if } x > 0.00856 \\ 7.787x + \frac{16}{116} & \text{if } x < 0.00856 \end{cases}$$

X_n, Y_n , and Z_n are used as D65 illumination (95.05, 100.0, and 108.9 respectively)

亮度增強

這實際上是相同的過程在IMSR使用RGB值，但利用 $L^*(X, Y)$ 而不是 $li(X, Y)$

$$LR_{merge}^*(x, y) = \alpha \frac{L^*(x, y)}{SR_{merge}^*(x, y)},$$

$$SR_{merge}^*(x, y) = \sum_{m=1}^M w(\sigma_m) SR_m^*(x, y, \sigma_m),$$

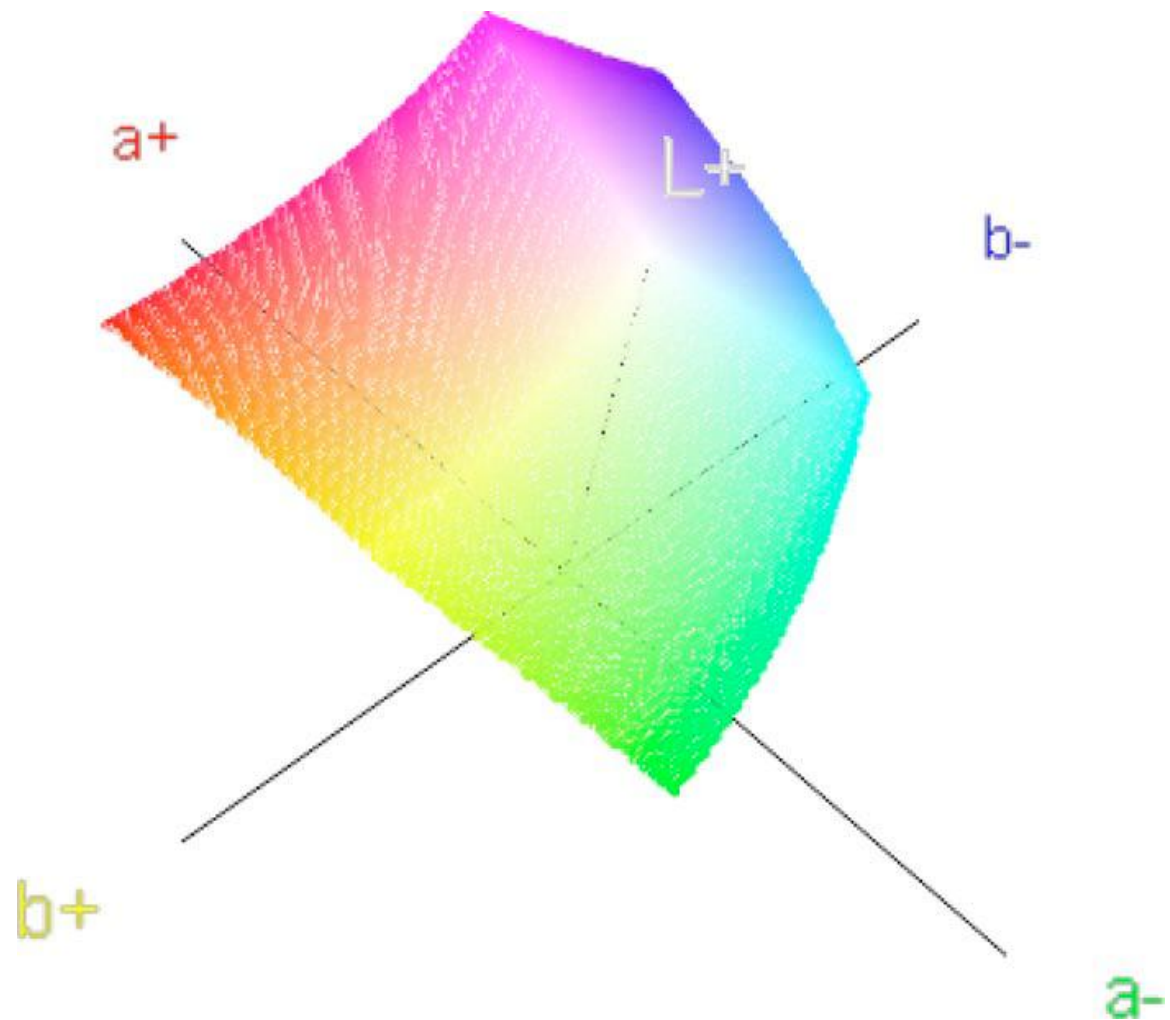
$$SR_m^*(x, y, \sigma_m) = G_m(x, y) \otimes L^*(x, y).$$

$LR_{merge}^*(x, y)$: 增強亮度值

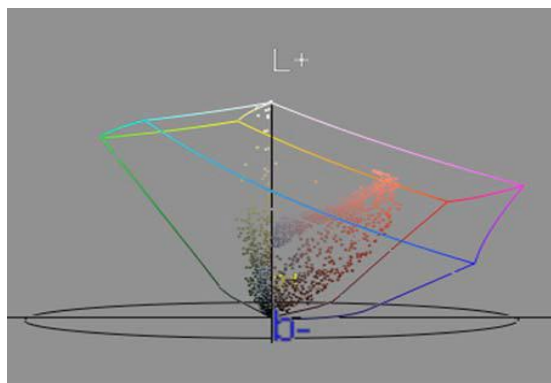
建議色調還原的方法

- 亮度增強
- 亮度校正
- 簡單圖像擴展色域飽和補償

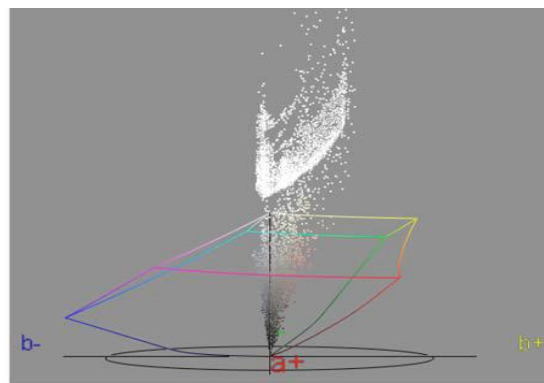
亮度校正



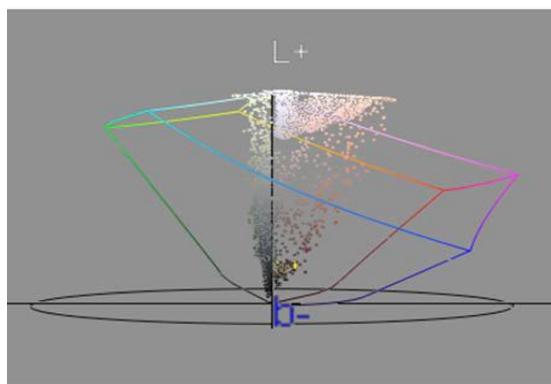
亮度校正



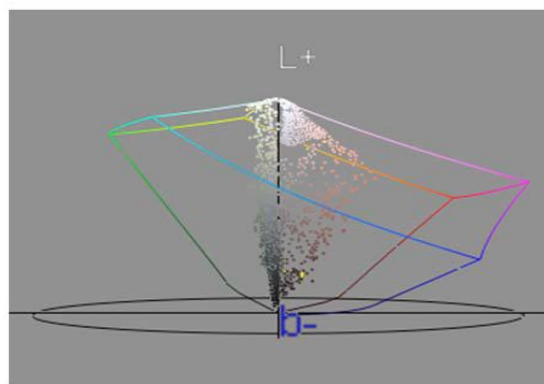
(a)



(b)



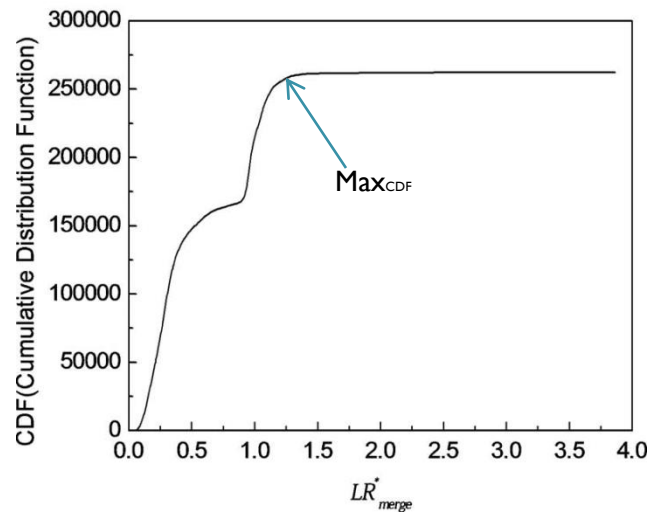
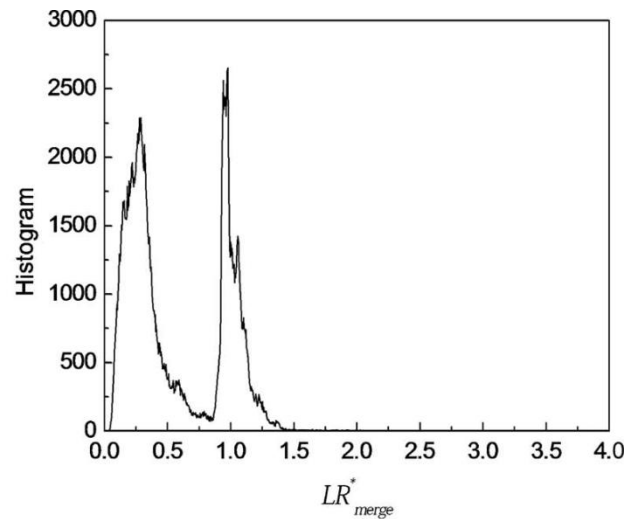
(c)



(d)

(a) 輸入圖像, (b) 是IMSR的CIELAB色彩空間的色域數據, (c) 正常化後的色域數據, (d) 色域數據後提出的方法。

亮度校正



the value with an almost zero gradient in the cdf is used as the normalization value.

亮度校正

$$LR_{correct}^*(x, y, \sigma_m) = Gb_{Hat}(a^*, b^*, x, y) \frac{LR_{merge}^*(x, y, \sigma_m)}{Max_{CDF}}$$

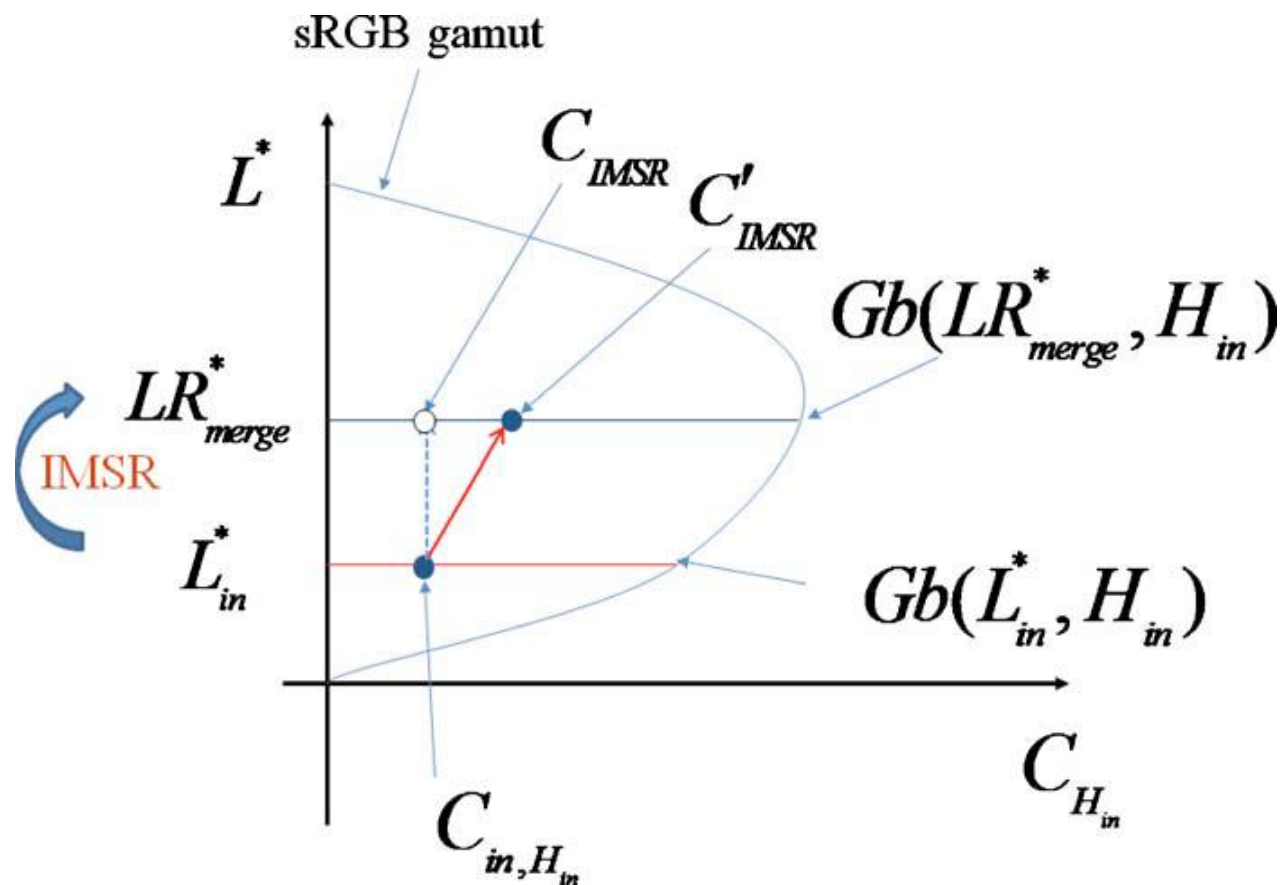
Max_{CDF} : 決定分佈的累積分佈函數.

Gb_{Hat}(a, b, x, y) : 表示在sRGB色域最大亮度值和頂端邊界在每個位置上sRGB色域亮度值之間的比值 (A* B*)。

建議色調還原的方法

- 亮度增強
- 亮度校正
- 簡單圖像擴展色域飽和補償

建議色調還原的方法



H_{in} : 輸入色調

L_{in}^* : 輸入亮度

Gb : 色域邊界

C : chroma值

建議色調還原的方法

$$C'_{IMSR}(x, y, H_{in}) = \frac{Gb(LR_{merge}^*)(x, y, H_{in})}{Gb(L_{in}^*)(x, y, H_{in})} C_{in}(x, y, H_{in})$$

C'_{IMSR} : 結果飽和補償