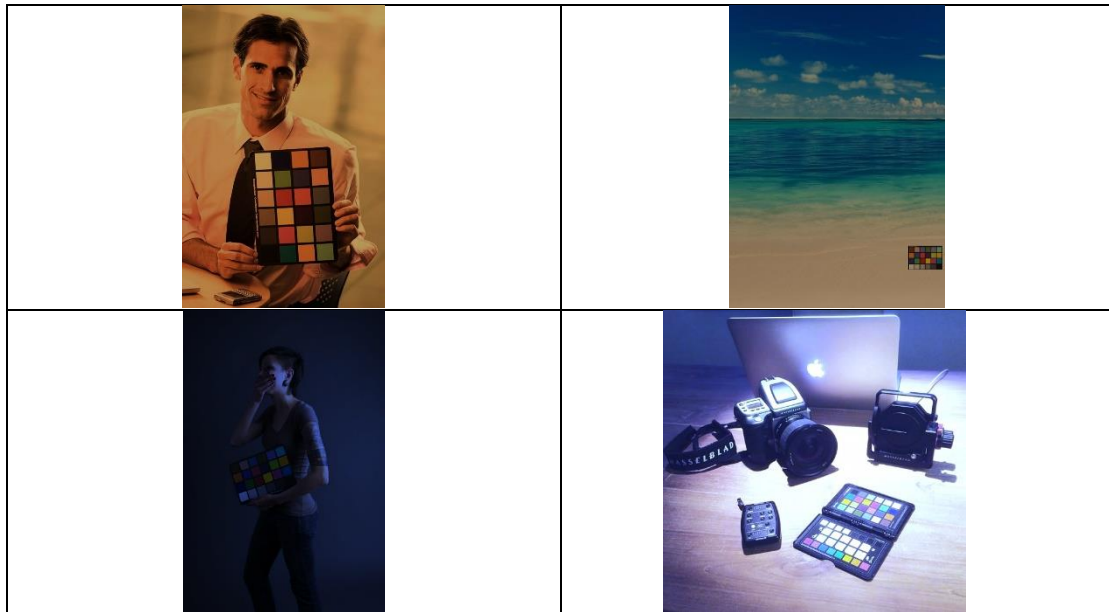


# Digital Image Processing (2023) HW3

## I. Chromatic Adaptation



▲上方為 4 張原圖



▲上方為 4 張經過 Chromatic Adaptation 的圖片

針對第一題，實作了兩種 Chromatic Adaptation，分別是 MaxRGB 與 Gray World。

### A. MaxRGB

是依據 White Patch 假設 RGB 的最大響應是由圖像中的白色表面引起的，此算法則將 RGB 通道中的最大值作為該通道的光照顏色。而由於最大值是分開通道計算的，因此在圖像缺乏白色、灰色等三通道數值相

近的點，就會使得表現較差(eg. input1.bmp，輸出仍然為橙色調)。

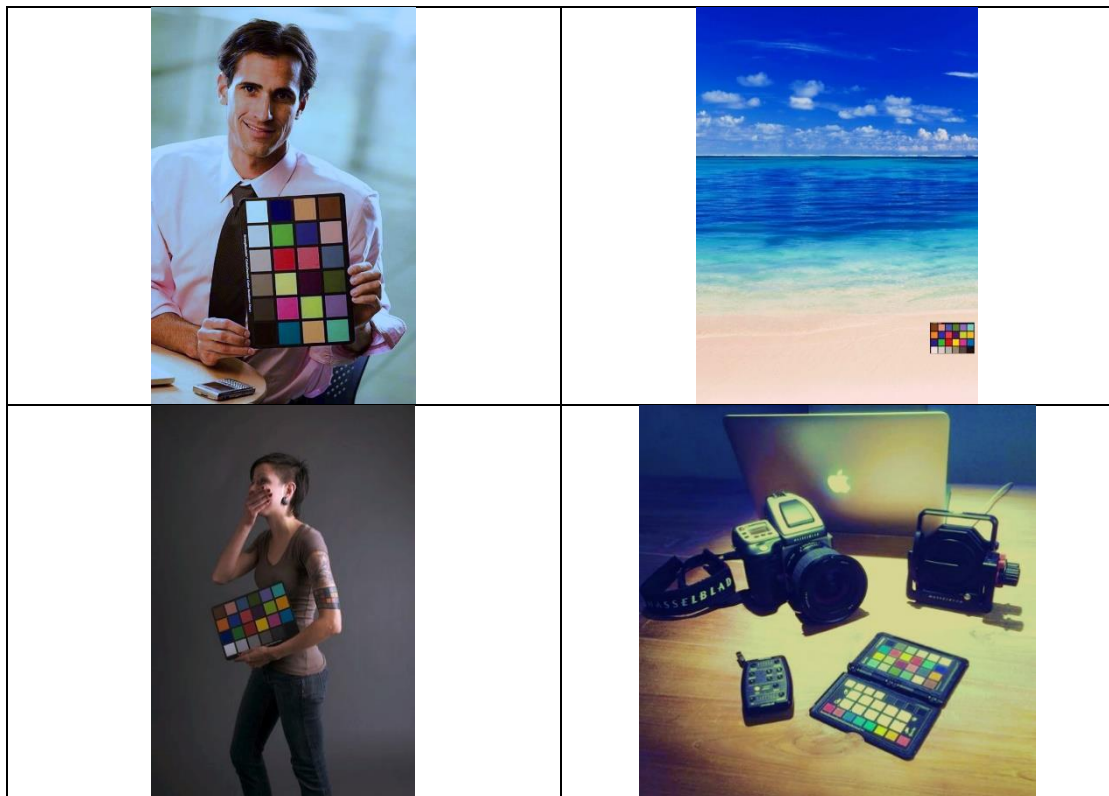
實作上，先計算三通道各自的最大值  $R_{max}$ 、 $G_{max}$ 、 $B_{max}$ ，接著計算增益  $k=[R_{max}, G_{max}, B_{max}]/Max$ ，其中我將  $Max$  用 255 代替。最後將原圖的三通道分別都乘上增益  $(1/k)$ 。

## B. Gray World

Gray World 假設所有物理表面的平均反射都是灰色的，也可以理解為一張影像的所有像素對其 3 通道做平均後，會呈現某種灰色。此算法將整個圖像平均顏色作為圖像光照顏色。此設條件相較 MaxRGB 更為寬鬆，但當圖像顏色較單一時(eg. input2.bmp、input3.bmp)，會造成平均色偏離灰色，因此新圖片會整個色調偏離。(input2 會變成橙色調、input3 會偏藍)

實作上，先計算三通道各自的均值  $R_{avg}$ 、 $G_{avg}$ 、 $B_{avg}$ ，接著計算增益  $k=[R_{avg}, G_{avg}, B_{avg}]/AVG$ ，其中我將  $AVG$  用  $G_{avg}$  代替。因此實際上只需要計算 R、B 的增益，最後將原圖的 R、B 分別都乘上增益  $(1/k)$ 。

## II. Image Enhancement



由於用 C++ 實作一個自適應的影像增強偏難，因此此題分別依照 4 張圖的需求個別處理。

### A. output1\_1.bmp

轉換至 HSL 色彩空間，將每個像素 S、L 值分別乘上 2、1.1，增加每個像數點的飽和度與明度，隨後轉回 RGB 空間並調整 Gamma ratio  $(1/1.2)$ ，調整照片整體亮度，隨後在 R 通道乘上 0.75、G 通道乘上

0.87，弱化因為飽和調高導致紅色調過於明顯。

#### B. output2\_1.bmp

轉換至 HSL 色彩空間，並針對  $250 > h > 130$  的區塊用 gamma 的方式做加強，此區塊代表的是藍色區塊，而以下操作可以使淡藍加深。

```
s = pow(s / 1.0, 1.0 / 3);
if(h > 130 && (h < 250)){
    h = pow(h / 360, 1.0 / (1 + 0.18 * ((h - 130) / (225 - 130)))) * 360;
}
```

L 通道也呈上 1.05 倍，隨後轉回 RGB 空間，並增強 RGB 三通道的對比度。

```
r = (r - 127) * 1.1 + 127;
g = (g - 127) * 1.1 + 127;
b = (b - 127) * 1.1 + 127;
```

最後將得到的新圖片與舊圖片做線性疊合。

```
pixel.red = clamp((uint8_t)(pixel.red * 0.5 + clamp(r) * 0.5) * 1);
pixel.green = clamp((uint8_t)(pixel.green * 0.5 + clamp(g) * 0.5) * 0.9);
pixel.blue = clamp((uint8_t)(pixel.blue * 0.5 + clamp(b) * 0.5) * 0.9);
```

其中 clamp 函數是避免  $<0$ 、 $>255$  的情況發生，若發生則會填上飽和值 0 或 255。

#### C. output3\_1.bmp

這題經過 MaxRGB 算法之後已經很極為接近 target image，因此我觀察背景，似乎 target image 的背景亮度更亮一點，所以我將圖片 RGB 分別調整 gamma ratio ( $1/1.7$ )，增亮後直接輸出。

#### D. output4\_1.bmp

由於這題顏色偏冷色、藍色通道成分偏重、中央處則因為強白光，導致 RGB 三通道皆爆滿到 255 極值，單純對 RGB、HSV 做 Histogram Equalization 會被大量 255 極值干擾到壞掉。因此我觀察圖片特性後，僅想盡可能將桌面做出暖一點的色彩。

首先針對 RGB 通道 gamma ratio ( $1/0.75$ ) 將圖片調暗，並針對 B 通道做一個非對稱的對比加強，針對暗處增強對比，針對亮處則減少對比。

```
double cons_thre_r=127, cons_thre_g=127, cons_thre_b = 45;
if((gammaTable[pixel.blue]*0.8-(gammaTable[pixel.green]+gammaTable[pixel.red])/3)>10)
    b_c = 0.85;
else
    b_c = 1;

//pixel.red = clamp((gammaTable[pixel.red]-cons_thre_r)*r_c+(cons_thre_r));
//pixel.green = clamp((gammaTable[pixel.green]-cons_thre_g)*g_c+(cons_thre_g));
//pixel.blue = (gammaTable[int(pixel.blue*0.65)]-127)*0.9+127;
pixel.red = clamp((gammaTable[int(pixel.red*0.8)]));
pixel.green = clamp((gammaTable[int(pixel.green*0.8)]));
pixel.blue = clamp((gammaTable[int(pixel.blue*0.6)]-cons_thre_b)*b_c+(cons_thre_b));
```

隨後用 gamma ratio 的方式調強 HSL 的 S 與調落 L，最後再用 MaxRGB 重新校正一次白平衡，即可拿到顏色更暖、更飽和的圖片。

### III. Reference

1. 課堂講義
2. 白平衡算法之 Gray World、White Patch、SoG

[https://blog.csdn.net/weixin\\_43194305/article/details/101758864](https://blog.csdn.net/weixin_43194305/article/details/101758864)