影像處理作業報告

HW3

Spatial Image Enhancement

授課教授：柳金章

學　　生：楊憲閔

學 　號：613410047

Due date：2024/12/27

Date hand in：2024/12/19

目錄

[Technical description 3](#_Toc181092894)

[Experimental results 9](#_Toc181092895)

[Discussions 16](#_Toc181092896)

[References and Appendix 17](#_Toc181092897)

# Technical description

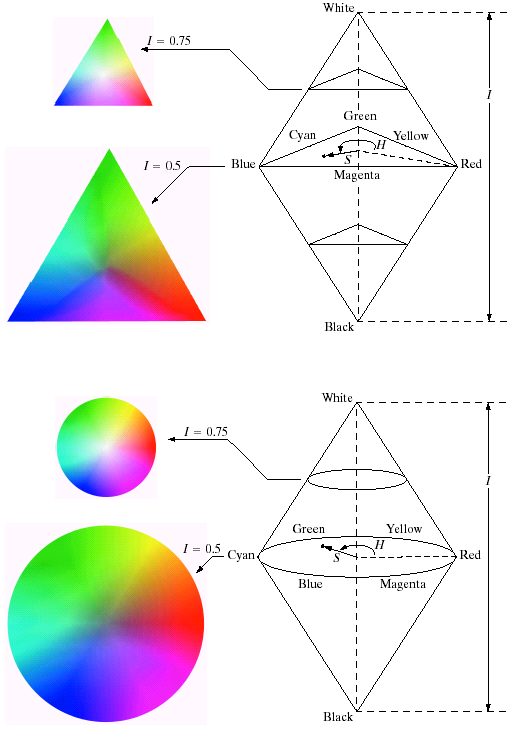
　　彩色影像是日常生活中很常見的，在表示上也有很多種方法，例如說RGB、CMY、HSI、YCbCr、YUV、YIQ與L\*a\*b\*等等，本次作業即是要在RGB、HSI與L\*a\*b\*進行影像處理，讓影像變得更適合需求。

1. RGB

又稱三色原光模式，其中三色為紅、綠、藍三種，是最常見的色彩空間，也是一種依賴於裝置的顏色空間：不同裝置對特定RGB值的檢測和重現都不一樣，因為顏色物質（螢光劑或者染料）和它們對紅、綠和藍的單獨回應水平隨著製造商的不同而不同，甚至是同樣的裝置不同的時間也不同。

1. HSI

HSI是將RGB色彩模型表示在圓形或是三角形坐標系的表示法，其中H代表Hue(色調)，S代表Saturation(飽和度)，I代表Intensity(強度)。其中圖示HIS color model與RGB轉換成HSI的公式如下：



圖(1) HIS color model based on triangular and circular color planes.

其中轉換前須先對RGB值normalize到[0,1]，轉換後我們是對I進行處理，因為H與S是與色彩有關係，我們在這就是對亮度做處理。處理完後再將HSI轉回到RGB呈現，轉換的公式會分為三個部分，詳細如下：

*RG sector* (0°≦H<120°)：

*GB sector* (120°≦H<120°)：

*BR sector* (240°≦H≦360°)：

1. L\*a\*b\*

是常用來描述人眼可見的所有顏色的色彩模型，L\*為亮度，a\*為紅色與綠色間的位置，b\*則為黃色與藍色間的位置，這個與RGB最大的差異在於L\*a\*b\*是與設備無關的模型，也因此他是絕對色彩空間，故RGB與L\*a\*b\*間是無法直接轉換的，必須透過XYZ色彩空間當中介做轉換。其中步驟為：

(1)sRGB變換為RGB：

我們所說的RGB通常為sRGB，因此我們先需進行歸一化與轉換為真正的RGB：

歸一化：

gamma變換(t代表R,G,B)：

(2)RGB透過線性映射轉換為XYZ：

(3)XYZ轉換到L\*a\*b\*：

透過白色參考點(Xref\_white、Yref\_white、Zref\_white)歸一化：

非線性變換(t代表x,y,z)：

最後線性變換：

得到後就對L\*進行處理，a\*與b\*皆不動。處理完後則是做相反動作將L\*a\*b\*轉回RGB，步驟如下：

(1) L\*a\*b\*轉換到XYZ：

線性變換:

非線性變換(t代表x,y,z)：

透過白色參考點(Xref\_white、Yref\_white、Zref\_white)反歸一化：

(2)XYZ透過線性變換成RGB：

(3)RGB轉成sRGB：

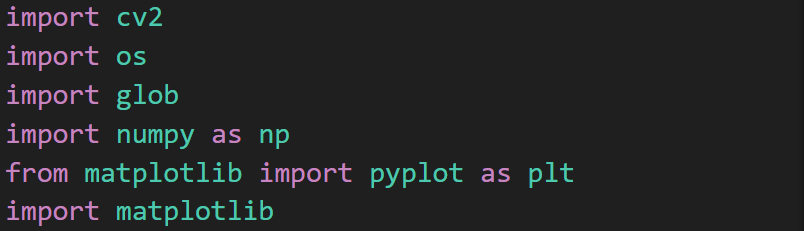
gamma變換(t代表R,G,B)：

裁減：

反歸一化：

# Experimental results

1. 程式執行流程:
2. 確保已安裝相關module，本次作業使用module如下所示:

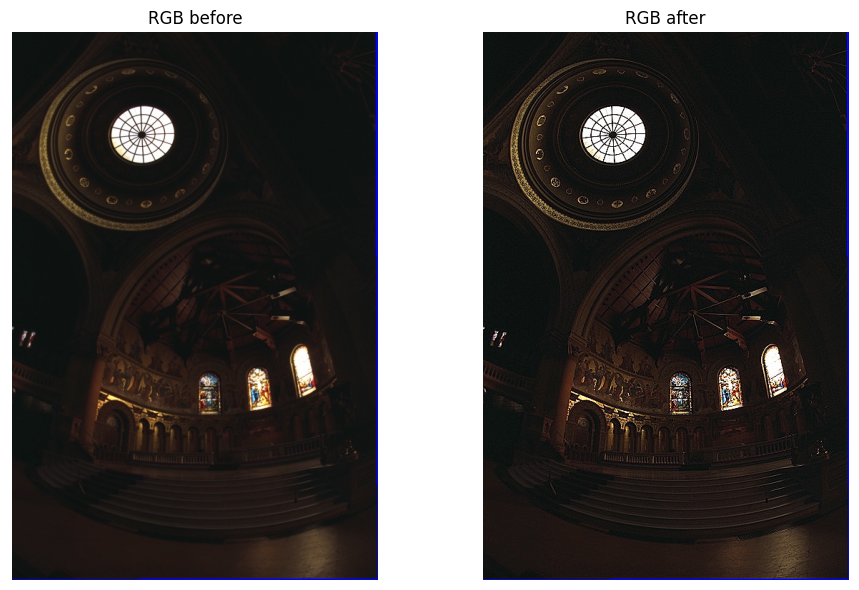


圖(2) 會使用到的module

1. 進到作業的目錄底下，會看到一個名為HW3\_test\_image的資料夾，一個main.py，還有這份pdf，點右鍵按在終端中開啟，輸入python main.py，程式即開始執行。
2. 程式會讀取HW3\_test\_image資料夾底下的圖片，並輸出對每個圖片進行transform的結果，本次作業是使用合成型Laplacian operator(中間係數為5)進行處理，但在處理前會先進行padding。順序會是讀一張圖片，輸出對該圖片在RGB進行處理後的結果，關掉視窗後會輸出對該圖片在HSI進行處理後的結果，再關掉視窗後會輸出該圖片在L\*a\*b\*進行處理後的結果。到此一張圖片輸出結束，會繼續讀取下一張圖片，並做一樣的順序，直到所有圖片都被讀取完，即結束程式。
3. 程式執行結果:
4. RGB:



圖(3) aloe.jpg原圖與轉換後之影像



圖(4) church.jpg原圖與轉換後之影像

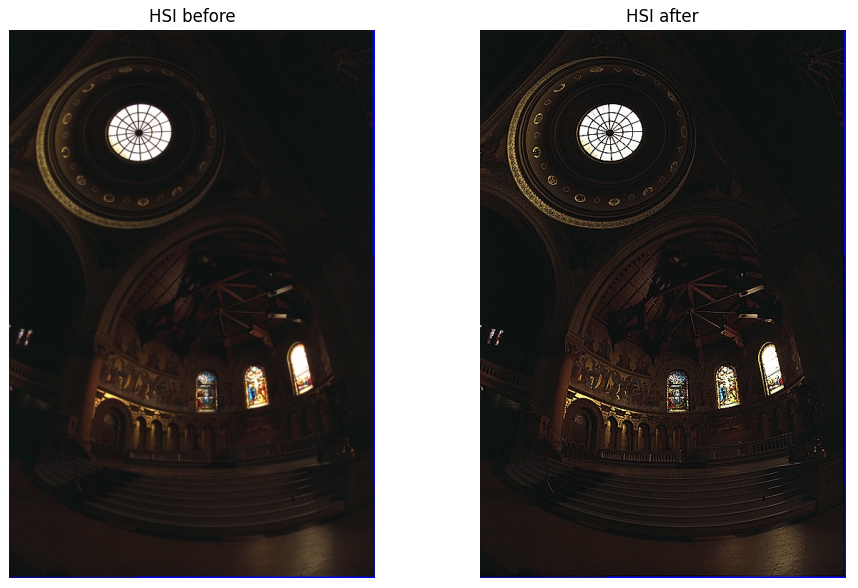
圖(5) kitchen.jpg原圖與轉換後之影像



圖(6) house.jpg原圖與轉換後之影像

1. HSI:



圖(7) aloe.jpg原圖與轉換後之影像

圖(8) church.jpg原圖與轉換後之影像

圖(9) kitchen.jpg原圖與轉換後之影像



圖(10) house.jpg原圖與轉換後之影像

1. L\*a\*b\*:



圖(11) aloe.jpg原圖與轉換後之影像



圖(12) church.jpg原圖與轉換後之影像



圖(13) kitchen.jpg原圖與轉換後之影像



圖(14) house.jpg原圖與轉換後之影像

# Discussions

這次的作業相較於前面兩次而言費力許多，因為要考慮到許多，也遇到了許多困境，這邊講一下一些遇到的問題：

* 1. 顏色輸出怪怪的，原本我寫的讀取圖片為單純的cv2.imread()，因此會是BGR通道順序，也因此在ax.imshow()時會變成輸出顏色錯誤，思考一下後就想到好像有這麼一回事，便利用cv2.cvtColor()轉換回去以便展示圖片
  2. 在查詢L\*a\*b\*的轉換方法時結果都不太一樣，但大抵上是相同的，也因此我決定了其中一個當作本次作業實作，在實作時我的輸出變得很奇怪，會是全黑的，這時我沒意識到是我在np.clip時出現了問題，因為我都是將laplacian後的結果進行np.clip到[0,1]，也因此L\*a\*b\*中的L\*會被clip掉，但實際上他的範圍應該是[0,100](黑色到白色)，導致說輸出會是全黑的。

# References and Appendix

RGB

<https://zh.wikipedia.org/zh-tw/%E4%B8%89%E5%8E%9F%E8%89%B2%E5%85%89%E6%A8%A1%E5%BC%8F>

HSI

<https://ithelp.ithome.com.tw/m/articles/10297458>

padding

<https://medium.com/hccuse/numpy%E5%AD%B8%E7%BF%92%E7%AD%86%E8%A8%98-%E5%A1%AB%E5%85%85%E6%96%B9%E6%B3%95-np-pad-5331f5a2dfb7>

XYZ

<https://www.getop.com/post/xyz%E8%89%B2%E5%BD%A9%E7%A9%BA%E9%96%93?srsltid=AfmBOorThg695PwMw9RGJnGLplEJoLlgn9zqVmfU2gI8l1WHB8c6rids>

RGB to L\*a\*b\*

<https://blog.csdn.net/bby1987/article/details/109522126>

<https://blog.csdn.net/lz0499/article/details/77345166>