

影像處理概論 HW3

題目：Color image enhancement

409410095 王子奕

繳交期限：June 5, 2023

繳交日期：June 5, 2023

● Technical description:

(四張 test images: “aloe.jpg” 、 “church.jpg” 、 “house.jpg” 、 “kitchen.jpg” 需與 hw3.py 存在同個資料夾下，此作業使用了 python 的 OpenCV 和 numpy 套件)。

1. 定義 RGB 色彩增強函數 (enhance_rgb): 將圖像轉換為 greyscale。將圖像拆分為個別的顏色通道 (藍色、綠色和紅色)。對每個通道應用增強操作 (可以根據需求自行修改)。將增強的通道合併為一個圖像。將增強的圖像轉換回 0-255 的範圍，並轉換為 uint8 數據類型。(如下圖)

```
import cv2
import numpy as np

def enhance_rgb(image):
    # Convert image to grayscale
    gray = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)

    b, g, r = cv2.split(image)

    r_enhanced = cv2.equalizeHist(r)
    g_enhanced = cv2.equalizeHist(g)
    b_enhanced = cv2.equalizeHist(b)

    enhanced_image = cv2.merge([b_enhanced, g_enhanced, r_enhanced])

    # Convert the enhanced image back to the range of 0-255
    enhanced_image = (enhanced_image * 255).astype(np.uint8)

    return enhanced_image
```

2. 定義 HSI 色彩增強函數 (enhance_hsi): 將圖像從 BGR 色彩空間轉換為 HSI 色彩空間。將 HSI 圖像拆分為個別的組件 (色

調 H、飽和度 S 和強度 I)。對強度 (I) 組件應用增強操作 (可以根據需求自行修改)。將增強的強度組件與原始的色調和飽和度組件合併。將增強的 HSI 圖像轉換回 BGR 色彩空間。(如下圖)

```
def enhance_hsi(image):  
    # Convert image from BGR to HSI color space  
    hsi_image = cv2.cvtColor(image, cv2.COLOR_BGR2HSV)  
  
    h, s, i = cv2.split(hsi_image)  
  
    i_enhanced = cv2.equalizeHist(i)  
  
    enhanced_hsi_image = cv2.merge([h, s, i_enhanced])  
  
    # Convert the enhanced HSI image back to the BGR color space  
    enhanced_bgr_image = cv2.cvtColor(enhanced_hsi_image, cv2.COLOR_HSV2BGR)  
  
    return enhanced_bgr_image
```

3. 定義 Lab*色彩增強函數 (enhance_lab)：將圖像從 BGR 色彩空間轉換為 Lab*色彩空間。將 Lab 圖像拆分為個別的組件 (亮度 L、a 和 b*)。對亮度 (L*) 組件應用增強操作 (可以根據需求自行修改)。將增強的亮度組件與原始的 a 和 b 組件合併。將增強的 Lab*圖像轉換回 BGR 色彩空間。(如下圖)

```
def enhance_lab(image):  
    # Convert image from BGR to L*a*b* color space  
    lab_image = cv2.cvtColor(image, cv2.COLOR_BGR2LAB)  
  
    l, a, b = cv2.split(lab_image)  
    l_enhanced = cv2.equalizeHist(l)  
    enhanced_lab_image = cv2.merge([l_enhanced, a, b])  
  
    # Convert the enhanced L*a*b* image back to the BGR color space  
    enhanced_bgr_image = cv2.cvtColor(enhanced_lab_image, cv2.COLOR_LAB2BGR)  
  
    return enhanced_bgr_image
```

4. 載入四個彩色圖像：將“aloe.jpg”、“church.jpg”、“house.jpg”、“kitchen.jpg”載入為圖像。(如下圖)

```
# 載入四個彩色圖像
image1 = cv2.imread('aloe.jpg')
image2 = cv2.imread('church.jpg')
image3 = cv2.imread('house.jpg')
image4 = cv2.imread('kitchen.jpg')
```

5. 在 RGB 色彩空間中進行圖像增強：對每個圖像進行 RGB 色彩增強，獲得增強後的 RGB 圖像。

在 HSI 色彩空間中進行圖像增強：對每個圖像進行 HSI 色彩增強，獲得增強後的 BGR 圖像。

在 Lab*色彩空間中進行圖像增強：對每個圖像進行 Lab*色彩增強，獲得增強後的 BGR 圖像。(如下圖)

```
# 在RGB色彩空間中進行圖像增強
enhanced_rgb_image1 = enhance_rgb(image1)
enhanced_rgb_image2 = enhance_rgb(image2)
enhanced_rgb_image3 = enhance_rgb(image3)
enhanced_rgb_image4 = enhance_rgb(image4)

# 在HSI色彩空間中進行圖像增強
enhanced_hsi_image1 = enhance_hsi(image1)
enhanced_hsi_image2 = enhance_hsi(image2)
enhanced_hsi_image3 = enhance_hsi(image3)
enhanced_hsi_image4 = enhance_hsi(image4)

# 在Lab*色彩空間中進行圖像增強
enhanced_lab_image1 = enhance_lab(image1)
enhanced_lab_image2 = enhance_lab(image2)
enhanced_lab_image3 = enhance_lab(image3)
enhanced_lab_image4 = enhance_lab(image4)
```

6. 顯示增強的 RGB 圖像、增強的 HSI 圖像、增強的 Lab*圖像。

使用 `cv2.waitKey(0)` 等待按鍵輸入。使用

`cv2.destroyAllWindows` 關閉所有窗口。(如下圖)

```
# 顯示或保存增強的圖像
cv2.imshow('Enhanced RGB Image 1', enhanced_rgb_image1)
cv2.imshow('Enhanced RGB Image 2', enhanced_rgb_image2)
cv2.imshow('Enhanced RGB Image 3', enhanced_rgb_image3)
cv2.imshow('Enhanced RGB Image 4', enhanced_rgb_image4)

cv2.imshow('Enhanced HSI Image 1', enhanced_hsi_image1)
cv2.imshow('Enhanced HSI Image 2', enhanced_hsi_image2)
cv2.imshow('Enhanced HSI Image 3', enhanced_hsi_image3)
cv2.imshow('Enhanced HSI Image 4', enhanced_hsi_image4)

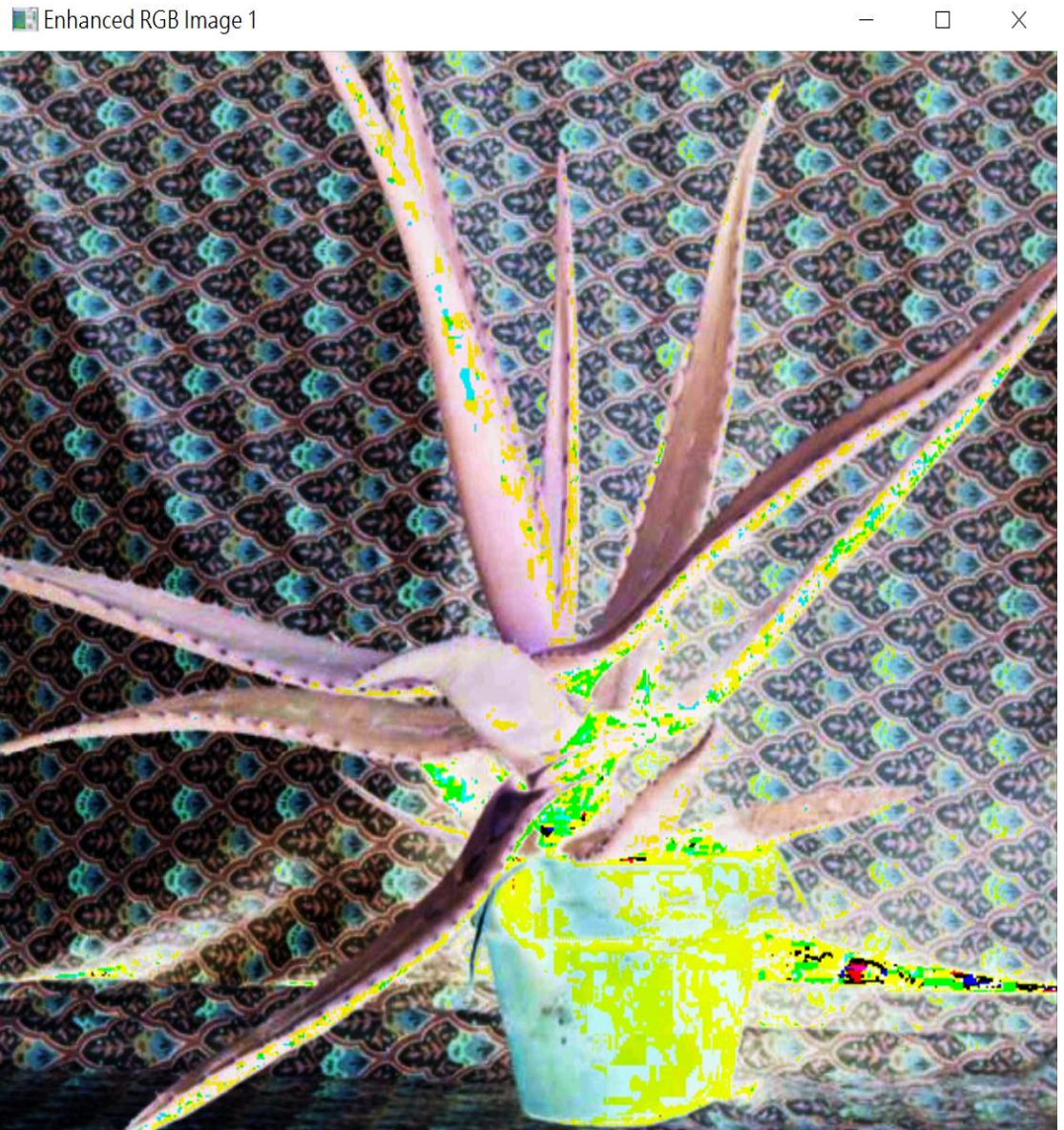
cv2.imshow('Enhanced L*a*b* Image 1', enhanced_lab_image1)
cv2.imshow('Enhanced L*a*b* Image 2', enhanced_lab_image2)
cv2.imshow('Enhanced L*a*b* Image 3', enhanced_lab_image3)
cv2.imshow('Enhanced L*a*b* Image 4', enhanced_lab_image4)

cv2.waitKey(0)
cv2.destroyAllWindows()
```

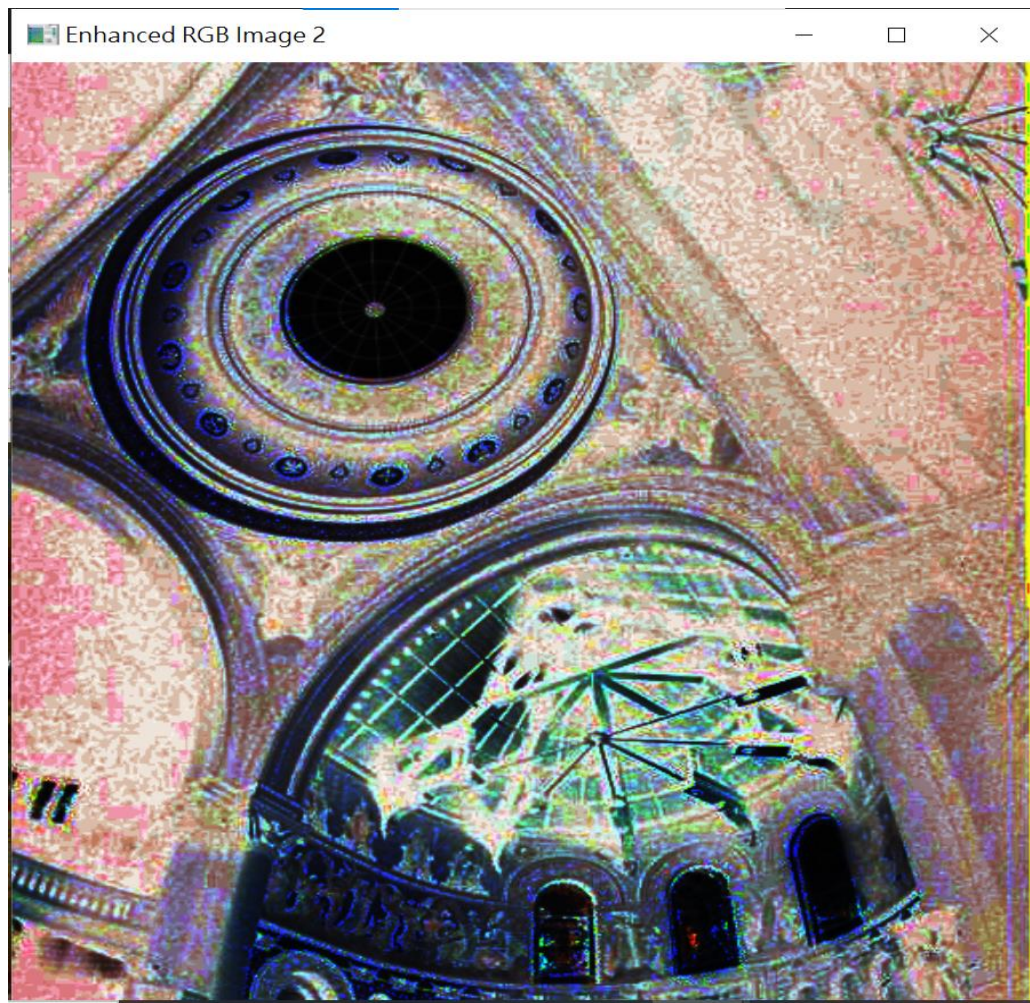

● Experimental results:

該程式的執行結果總共為十二個輸出視窗

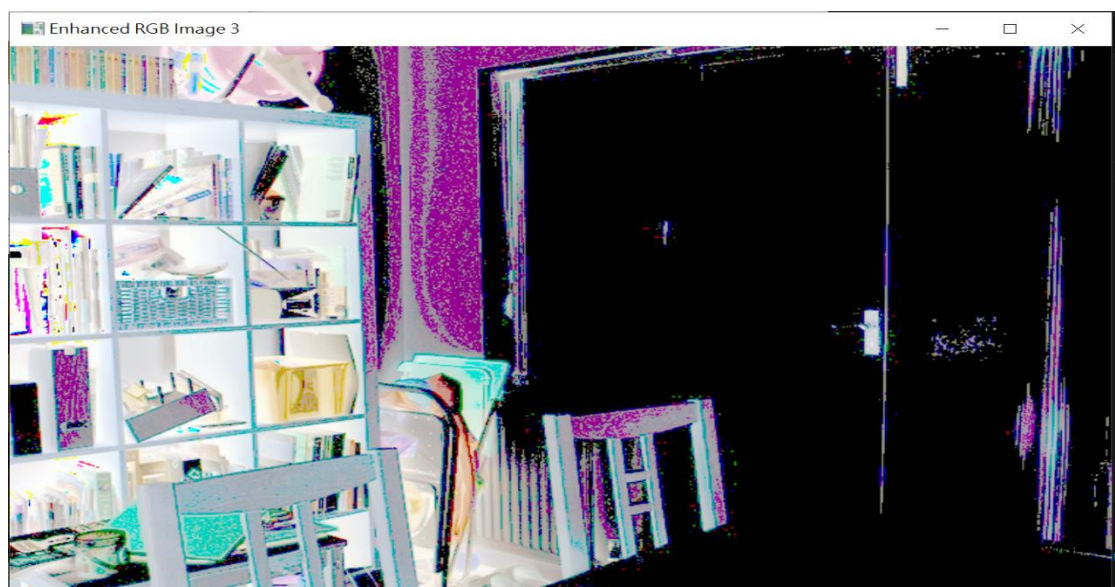
1. Enhanced RGB image1(如下圖)



2. Enhanced RGB image2 (如下圖)



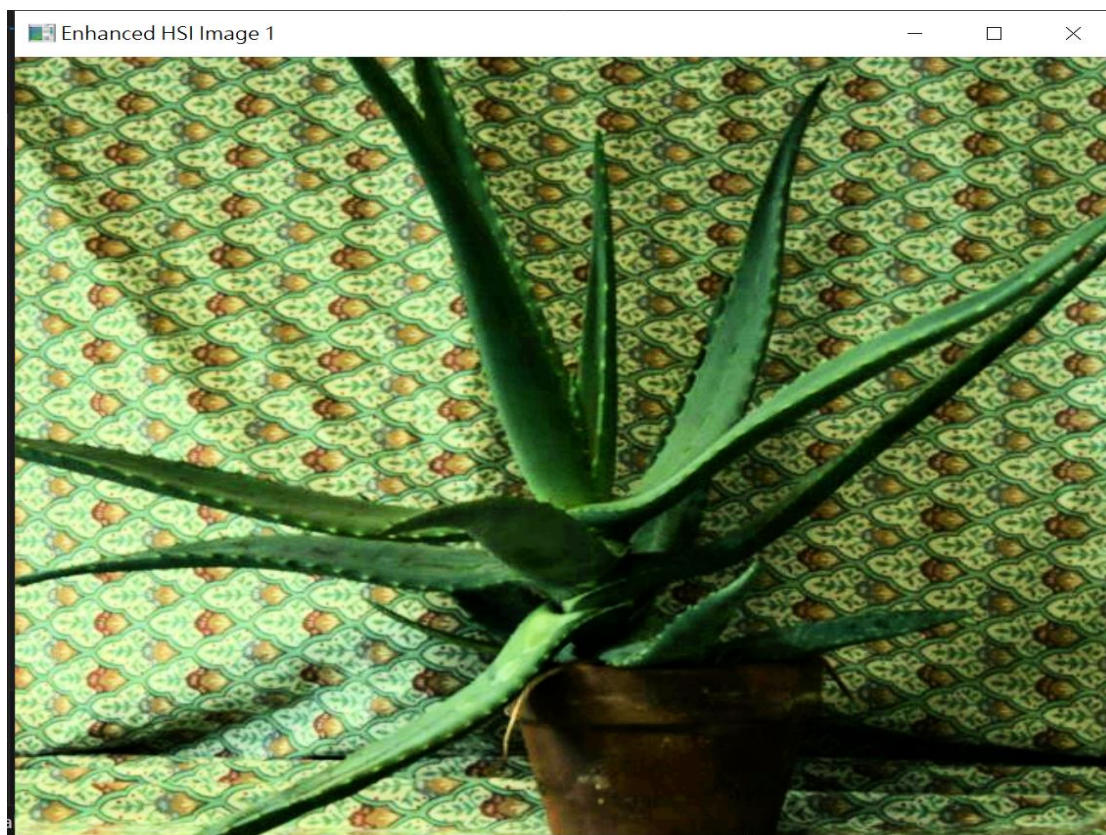
3. Enhanced RGB image3 (如下圖)



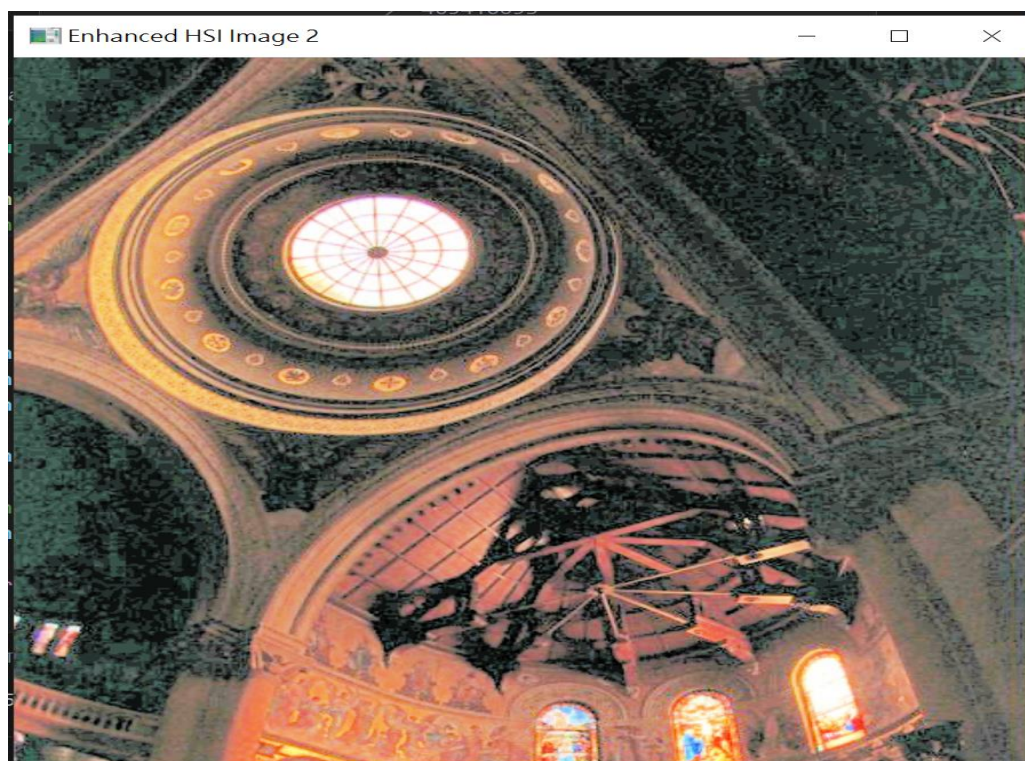
4. Enhanced RGB image4 (如下圖)



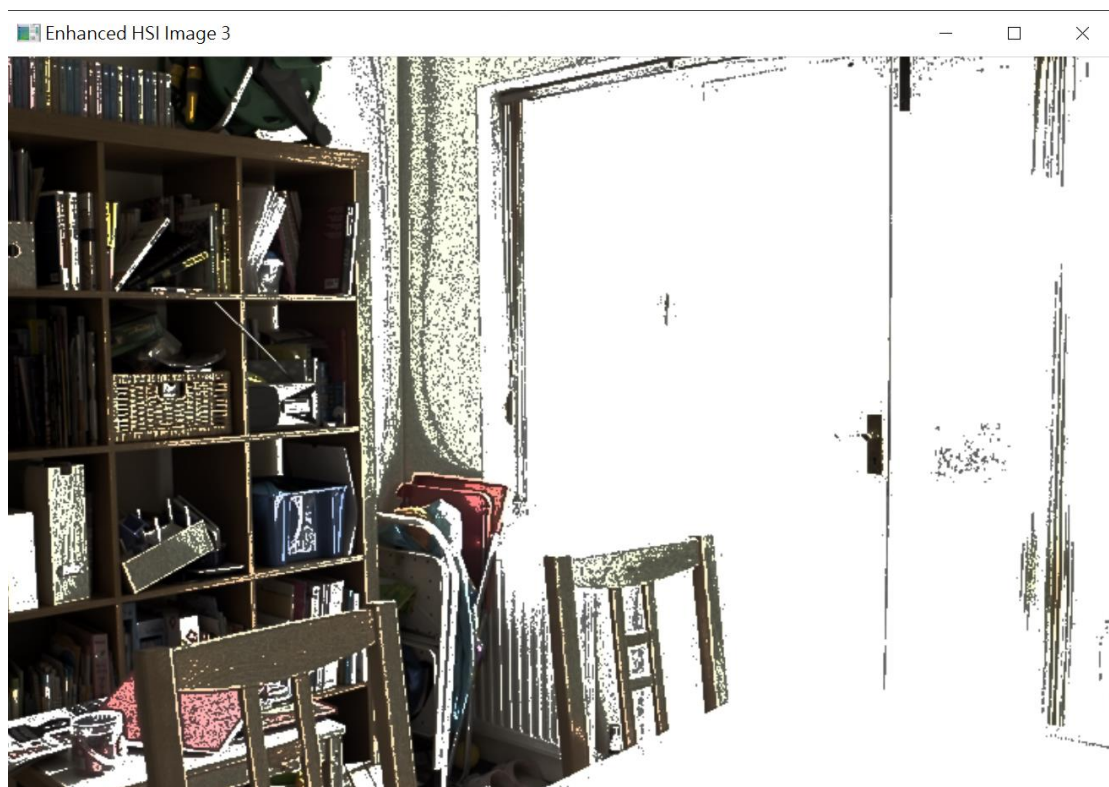
5. Enhanced HSI image1 (如下圖)



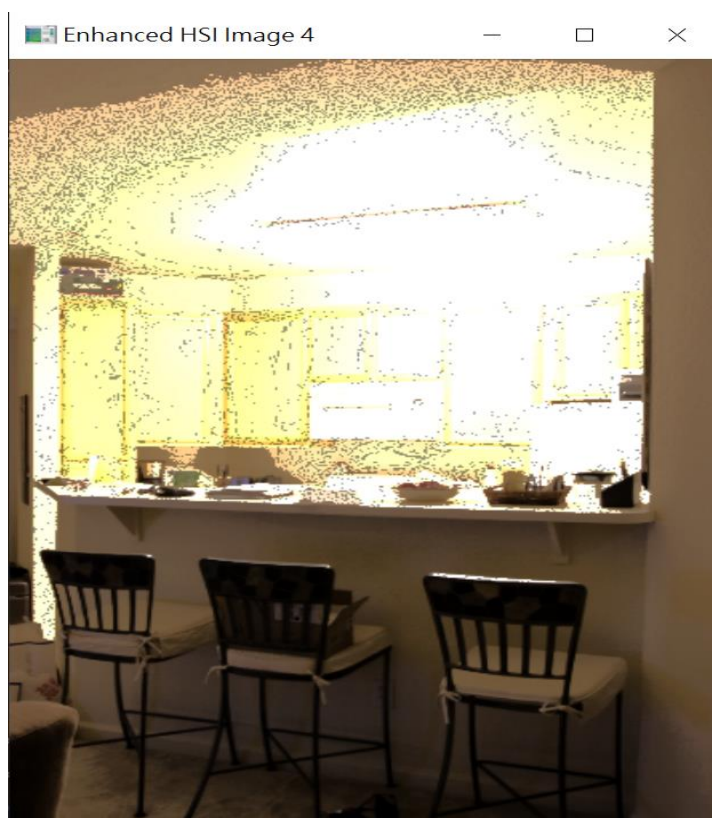
6. Enhanced HSI image2(如下圖)



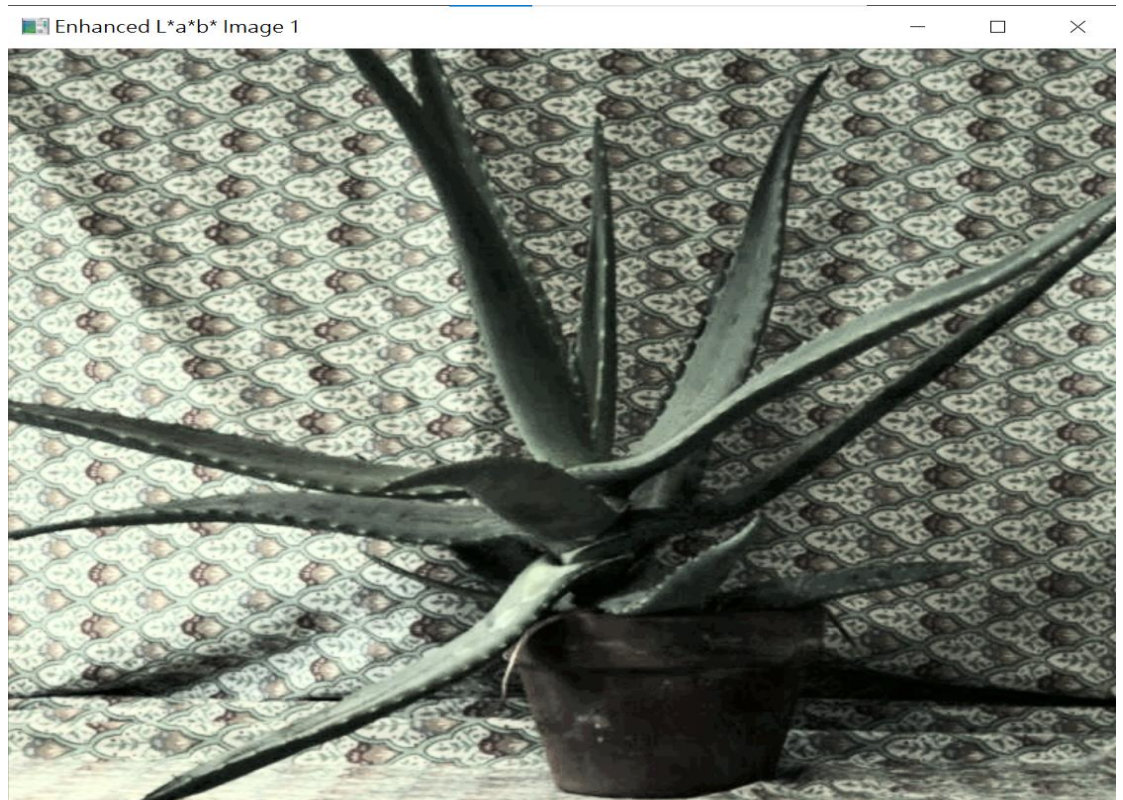
7. Enhanced HSI image3 (如下圖)



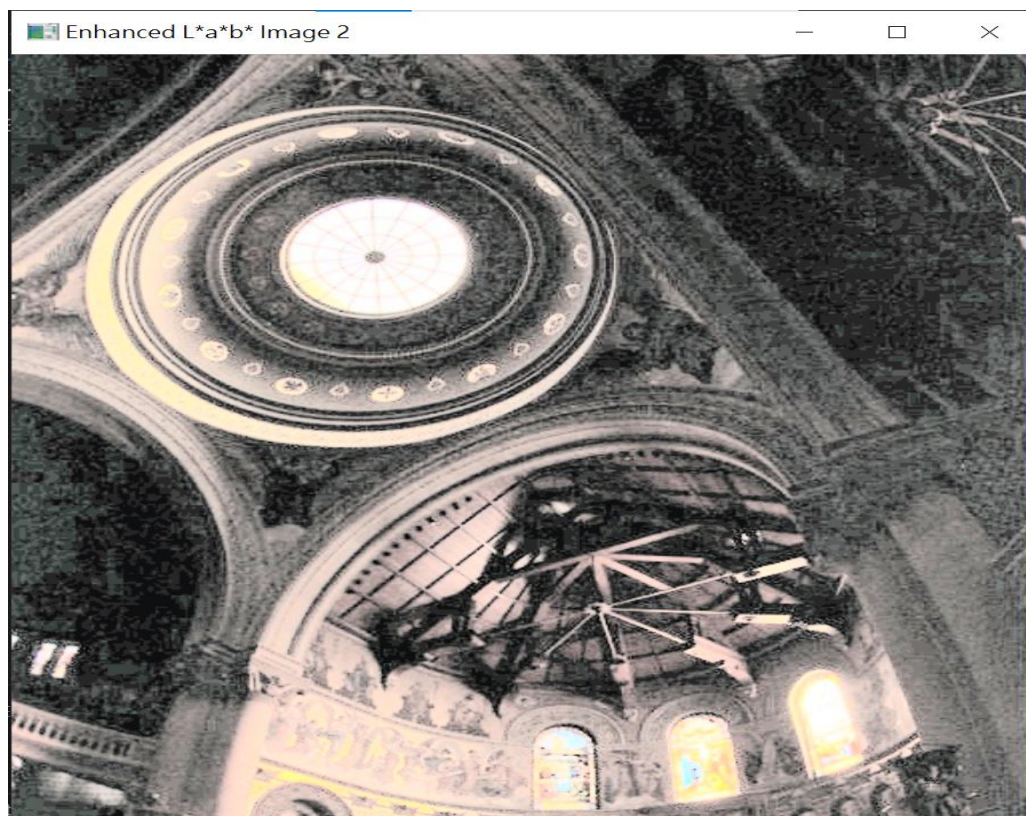
8. Enhanced HSI image4(如下圖)



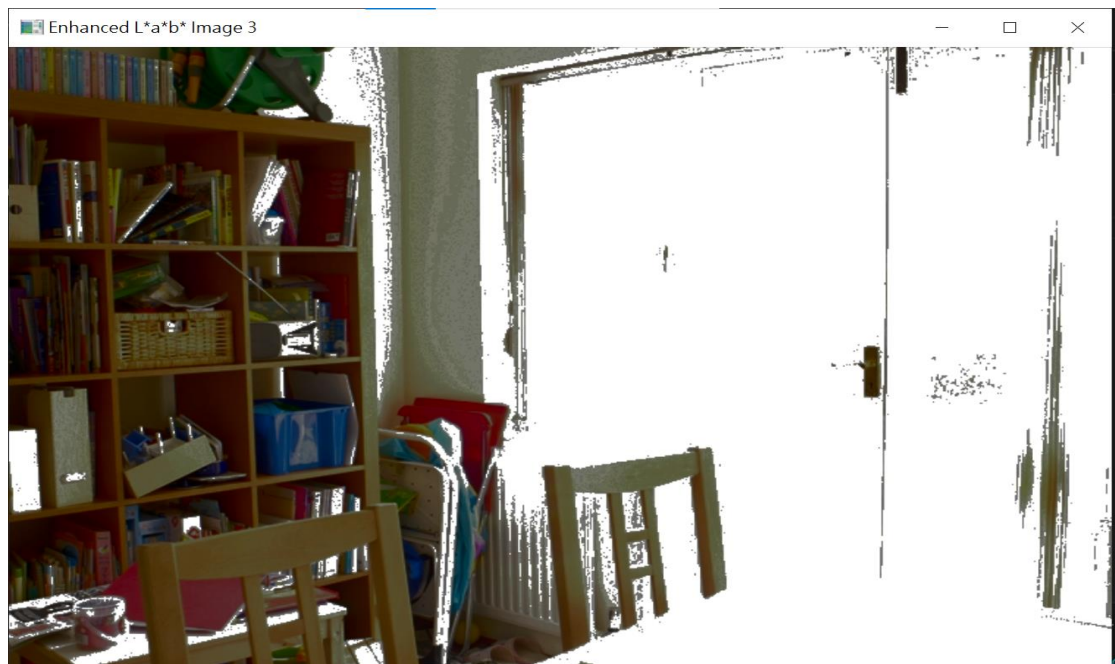
9. Enhanced L*a*b image1 (如下圖)



10. Enhanced L*a*b image2 (如下圖)



11. Enhanced L*a*b image3(如下圖)



12. Enhanced L*a*b image4(如下圖)



● Discussions:

RGB 色彩空間增強：RGB 是最常用的色彩空間，直接使用圖像的紅、綠、藍通道進行處理。通過直方圖均衡化或其他增強技術，可以增強圖像的對比度和亮度。RGB 色彩空間增強通常會對整體圖像的色調和飽和度進行調整，可能會使圖像的色彩更鮮豔或更暗淡。

HSI 色彩空間增強：HSI 是一種基於色調 (Hue)、飽和度 (Saturation) 和亮度 (Intensity) 的色彩表示方式。在 HSI 色彩空間中，色調保持不變，只對飽和度和亮度進行增強。通過調整飽和度和亮度的值，可以改變圖像的飽和度和亮度層次，使圖像顯得更鮮豔或更柔和。

Lab* 色彩空間增強：Lab* 是一種基於亮度 (Luminance) 和色度 (Chrominance) 的色彩表示方式。在 Lab* 色彩空間中，亮度 (L) 保持不變，只對色度 (a^* 和 b^*) 進行增強。通過調整色度通道的值，可以調整圖像的顏色平衡和對比度，使圖像的色彩更鮮豔或更柔和。

透過這次 homework 讓我更能夠了解經過這些處理輸出結果的差異和其運作方法。

● References and Appendix:

--references:

1. <http://ieeexplore.ieee.org/document/6473493/>
2. <https://www.sciencedirect.com/science/article/pii/S0923596522000352>
3. <https://www.mdpi.com/2313-433X/7/8/150>
4. https://link.springer.com/chapter/10.1007/978-3-319-95957-3_47
5. <https://github.com/topics/hsi?l=python>