影像色彩的轉換與分析

目標任務

對影像進行色彩模型轉換並將其模型通道分離

(1)RGB色彩模型分離

(2)HSV色彩模型分離

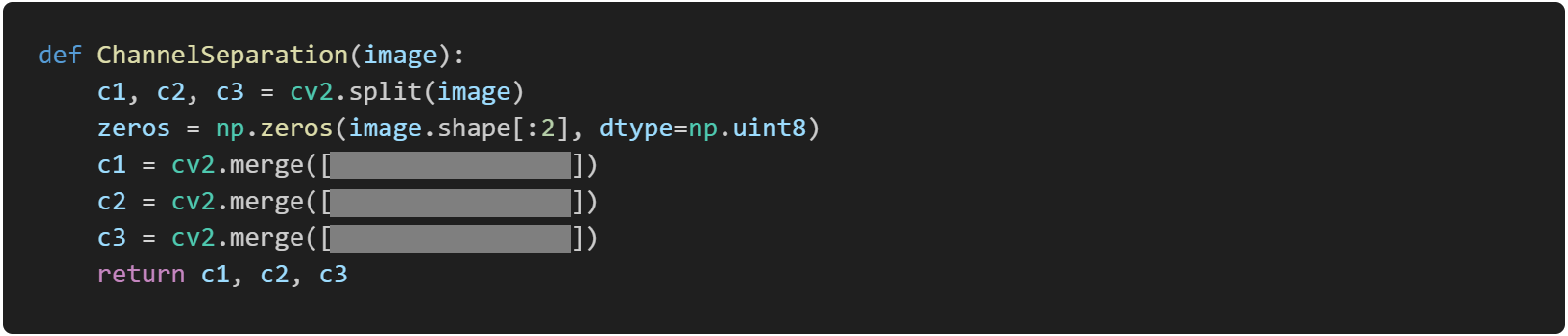
(3)YCrCb色彩模型分離

(4)色彩影像增強使用伽瑪校正(Gamma Correction)

- **gamma1：2**
- **gamma2：0.5**

通道分離

- □ **色彩通道分離用:**
  - ✓ *cv2.split(m[, mv])*
- □ **色彩通道合併用:**
  - ✓ *cv2. merge(mv[, dst])*



```python
def ChannelSeparation(image):
    c1, c2, c3 = cv2.split(image)
    zeros = np.zeros(image.shape[:2], dtype=np.uint8)
    c1 = cv2.merge([[redacted]])
    c2 = cv2.merge([[redacted]])
    c3 = cv2.merge([[redacted]])
    return c1, c2, c3
```

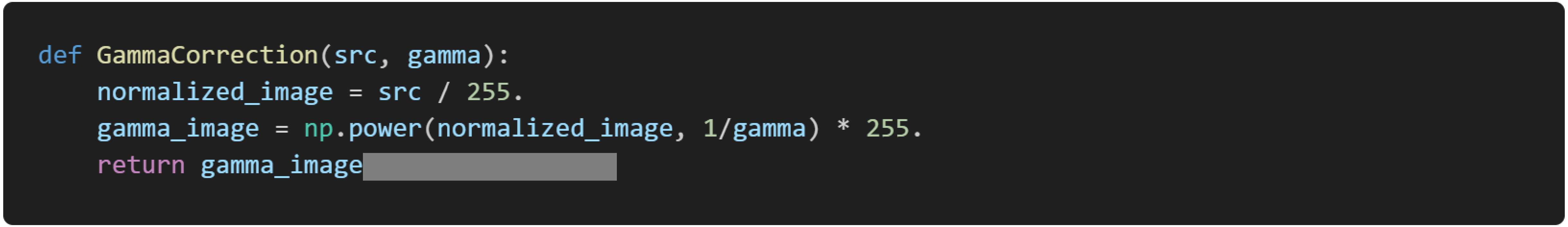
Gamma校正

Gamma > 1：亮度減少

Gamma < 1：亮度提升

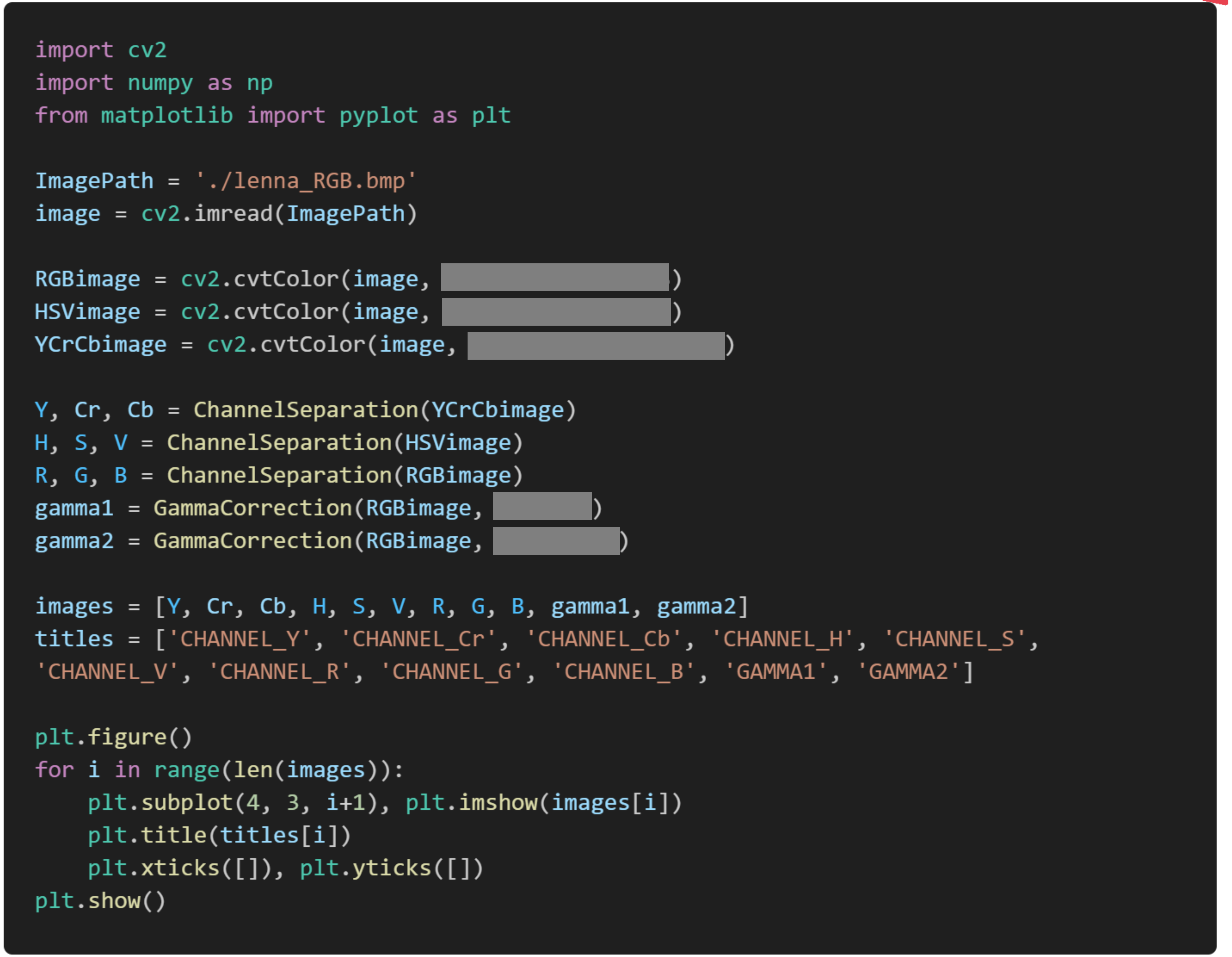
□ **Gamma校正用:**

- ✓ ***np.power(歸一化影像, 1/gamma) * 255***



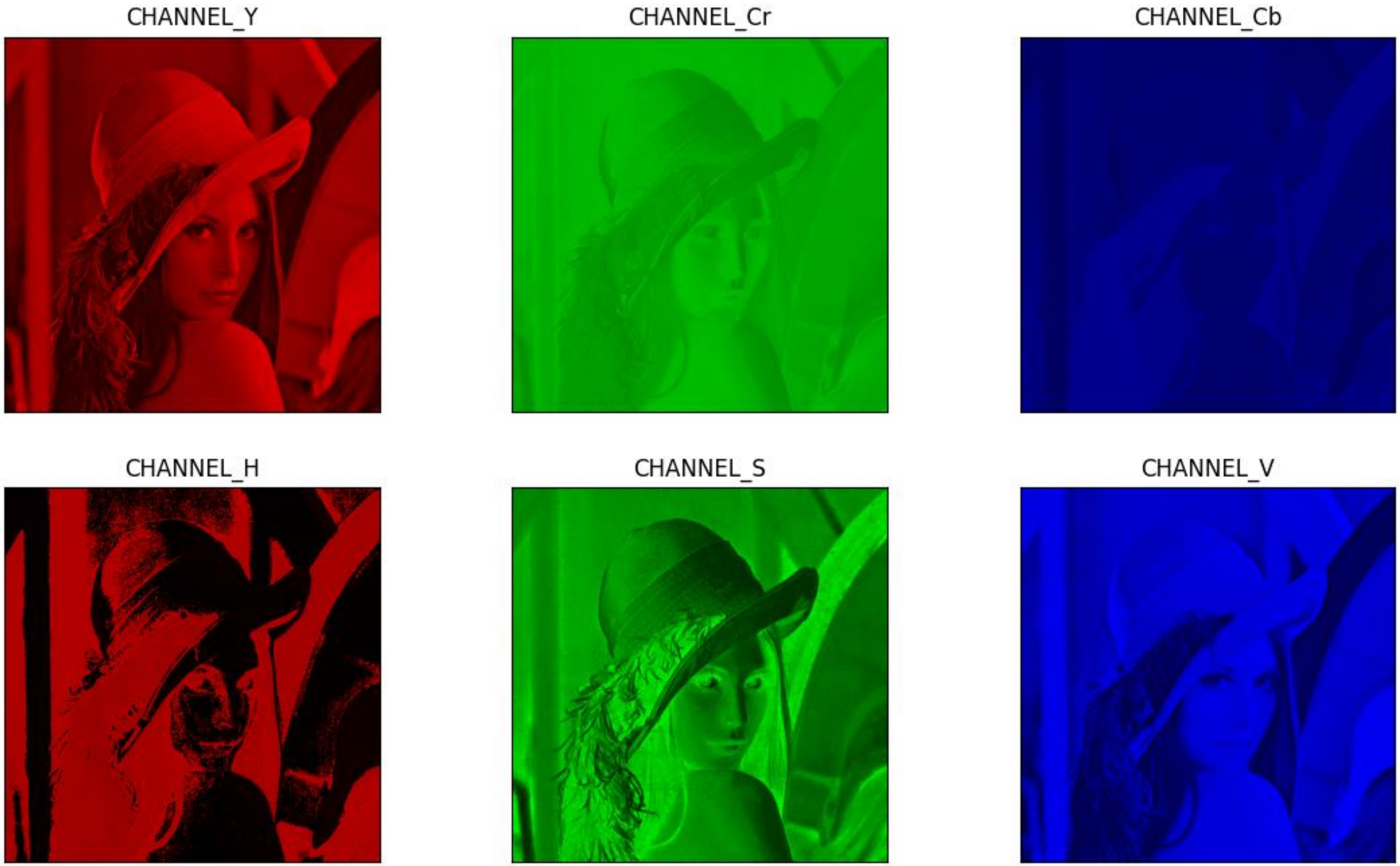
```python
def GammaCorrection(src, gamma):
    normalized_image = src / 255.
    gamma_image = np.power(normalized_image, 1/gamma) * 255.
    return gamma_image[redacted]
```

主程式

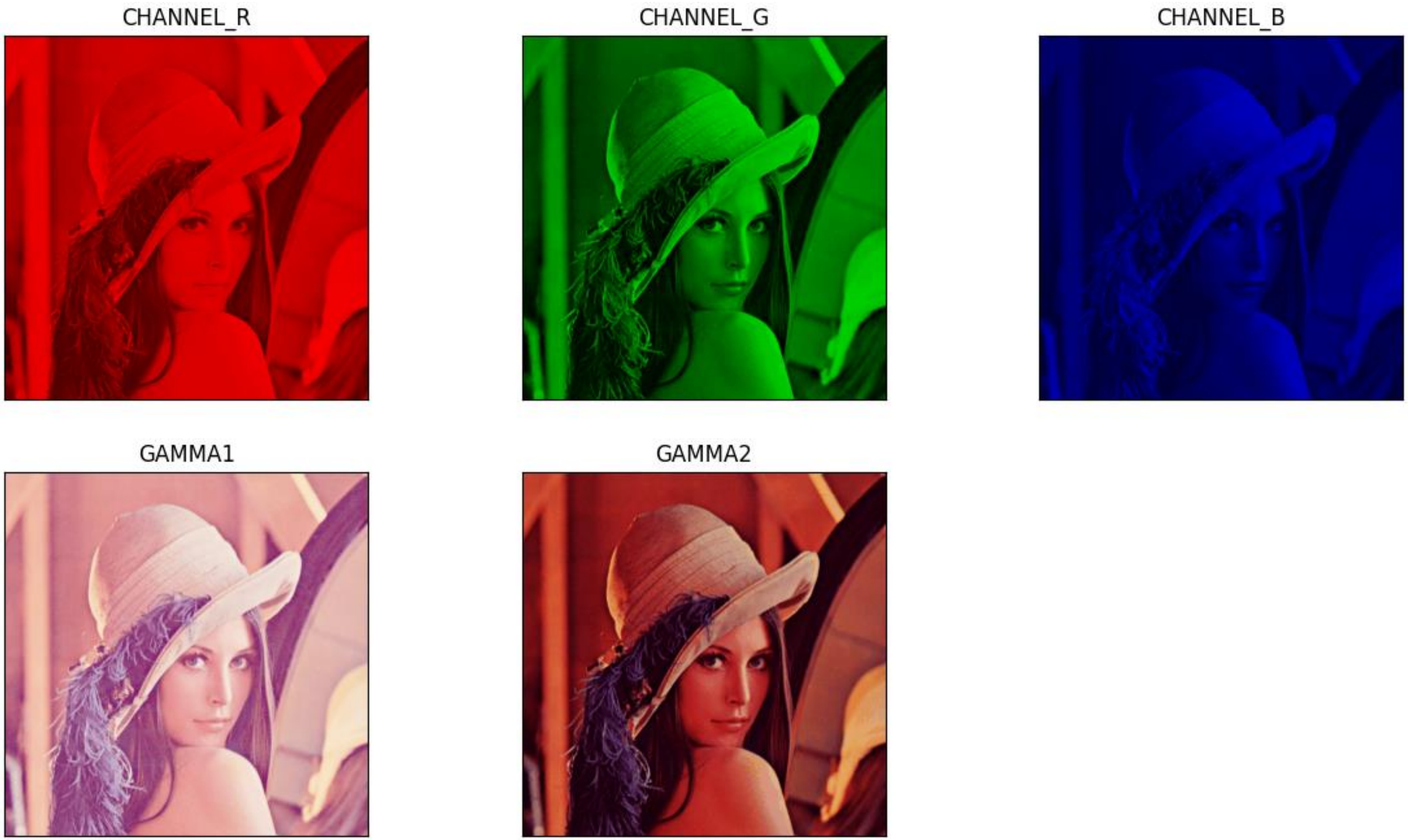


```
import cv2
import numpy as np
from matplotlib import pyplot as plt

ImagePath = './lenna_RGB.bmp'
image = cv2.imread(ImagePath)

RGBimage = cv2.cvtColor(image, [redacted])
HSVimage = cv2.cvtColor(image, [redacted])
YCrCbimage = cv2.cvtColor(image, [redacted])

Y, Cr, Cb = ChannelSeparation(YCrCbimage)
H, S, V = ChannelSeparation(HSVimage)
R, G, B = ChannelSeparation(RGBimage)
gamma1 = GammaCorrection(RGBimage, [redacted])
gamma2 = GammaCorrection(RGBimage, [redacted])

images = [Y, Cr, Cb, H, S, V, R, G, B, gamma1, gamma2]
titles = ['CHANNEL_Y', 'CHANNEL_Cr', 'CHANNEL_Cb', 'CHANNEL_H', 'CHANNEL_S',
'CHANNEL_V', 'CHANNEL_R', 'CHANNEL_G', 'CHANNEL_B', 'GAMMA1', 'GAMMA2']

plt.figure()
for i in range(len(images)):
    plt.subplot(4, 3, i+1), plt.imshow(images[i])
    plt.title(titles[i])
    plt.xticks([]), plt.yticks([])
plt.show()
```

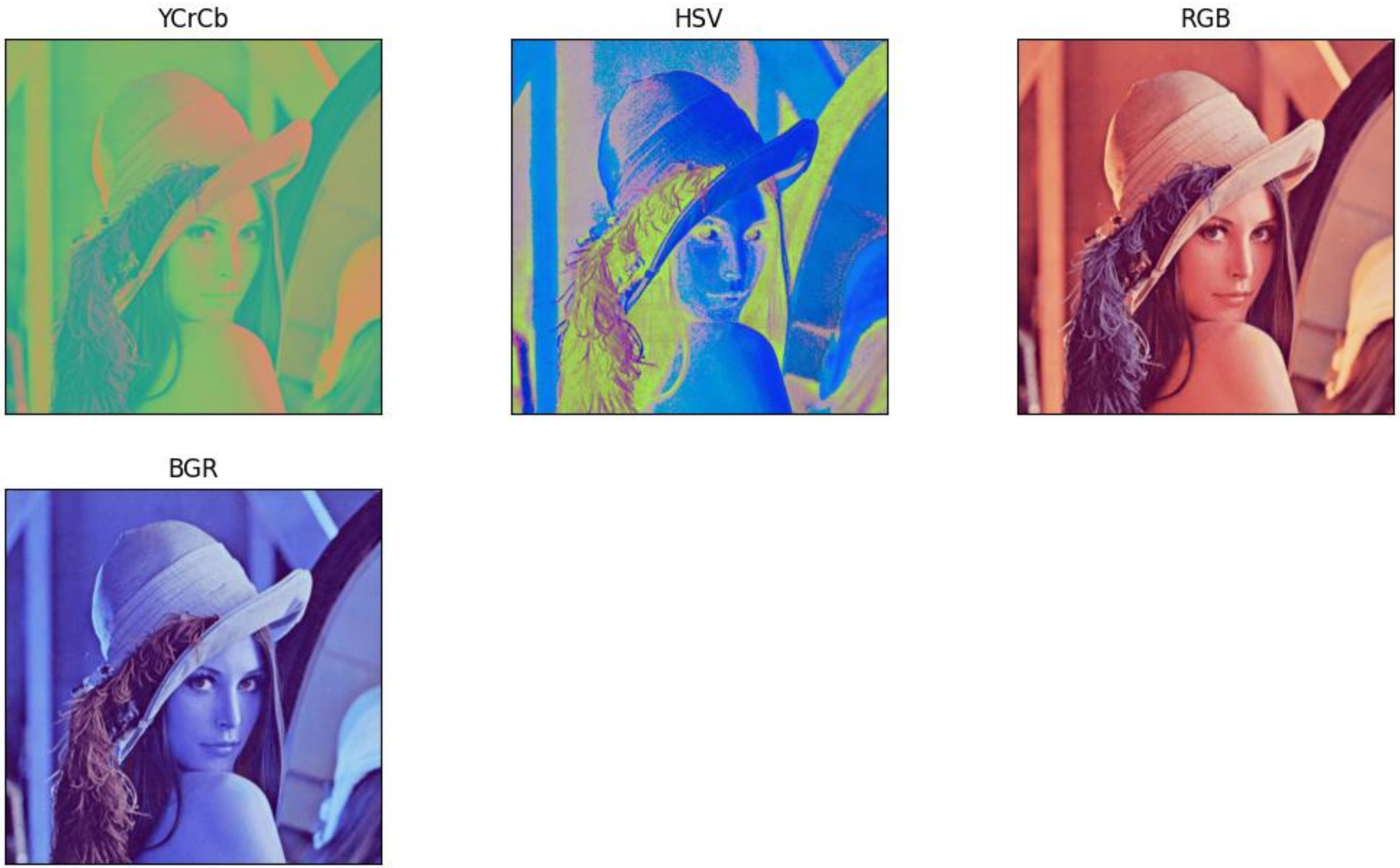
實作結果


CHANNEL_Y
CHANNEL_Cr
CHANNEL_Cb
CHANNEL_H
CHANNEL_S
CHANNEL_V

實作結果



哪怕轉換色彩空間後，影像不也應該一樣嗎?



Thanks for listening