|  |
| --- |
| 機器視覺作業報告 |
| Homework #1 |
|  |
|  |
| 姓名: 吳秉宸  學號: 00957202  日期: 2022/04/13 |
|  |

|  |
| --- |
|  |

# 方法

* 想法，演算步驟。

這次的報告最一開始是構思要做哪些image operation的組合達成老師所要求的項目，如下：

1. 第一張圖片採用

Gauss + different color coordinate + tensorflow + 雪山youtube影片

1. 第二張圖片採用

Gray-level mapping + 老師的測試影片

1. 第三張圖片採用

Histogram equalization + 老師的測試影片

1. 第四張圖片採用

High-pass + 老師的測試影片

每張圖片都 define 一個獨立的 function。

其中，第一張圖是先轉換到 HSV 的座標系中，因為其很適合調整顏色飽和度，進行顏色飽和度的提升，再轉回BGR座標系，用 opencv生成Gauss kernel，並將其kernal帶入tensorflow做卷積運算。

第二張圖是先將圖片轉到灰階圖再處理，自己實作出Gray-level mapping

(gamma correction of gamma = 0.5)，先建Table再進行對應，此處若將Table建置在function外可以減少每次for loop都要建Table的運算與時間 (因為mapping的function是固定的) ，但為了程式的易讀性寫在裡面。

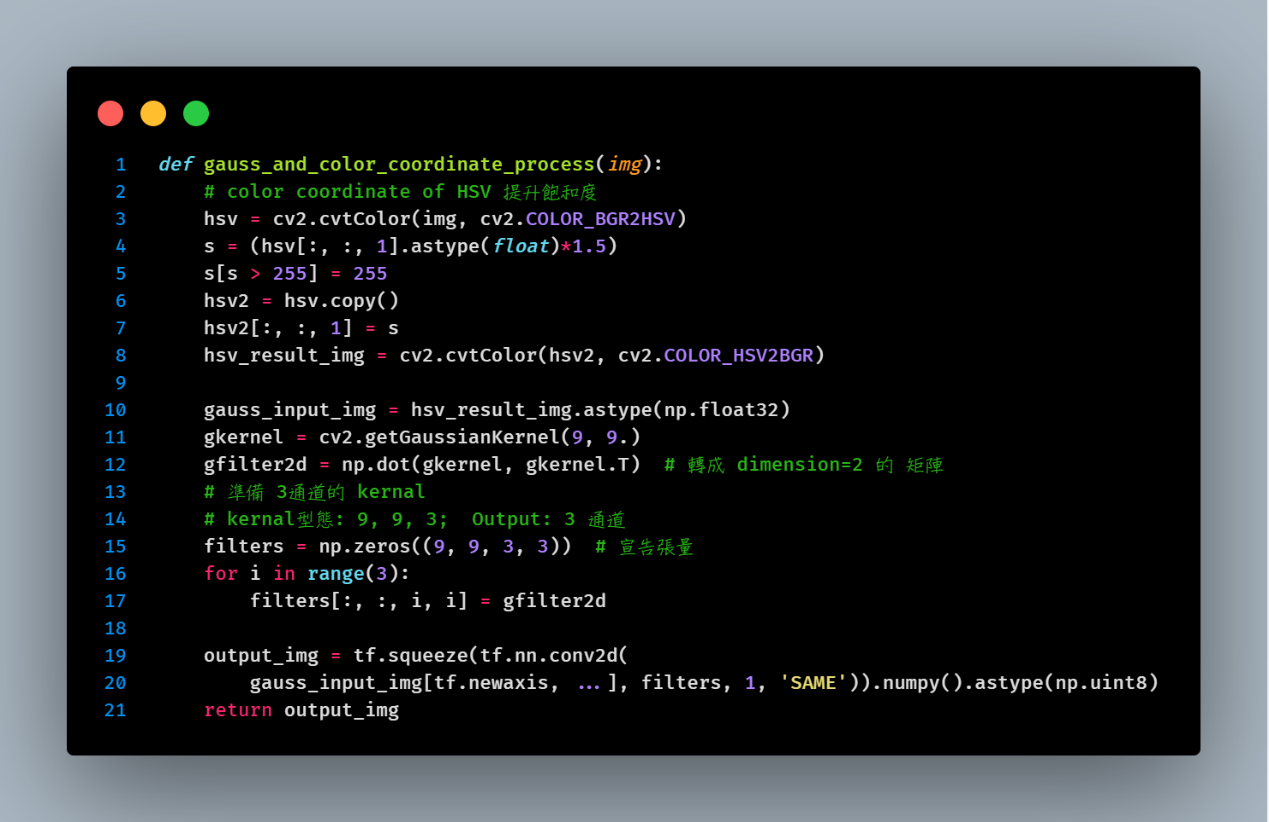
第三張圖也先將圖片轉到灰階圖再處理，而原本是自己實作出histogram equalization，雖然有成功，不過發現程式真的跑太慢了，因為其要依據input的img才能決定mapping的曲線，所以最後去Opencv的網站上看到了cv2.equalizeHist() function，並使用之。

第四張圖是先將圖片轉到灰階圖再處理，採用的是high\_pass的img operation來處理，先用傅立葉轉換轉換到適合操縱細節的 Fourier domain，並做上下左右的區塊顛倒，以中心點為圓心，將Frequency為50以下的去掉，再做逆傅立葉轉換回原本domain呈現。

最後再將圖片都copy到一張大圖中，做VideoWriter得結果的影片。

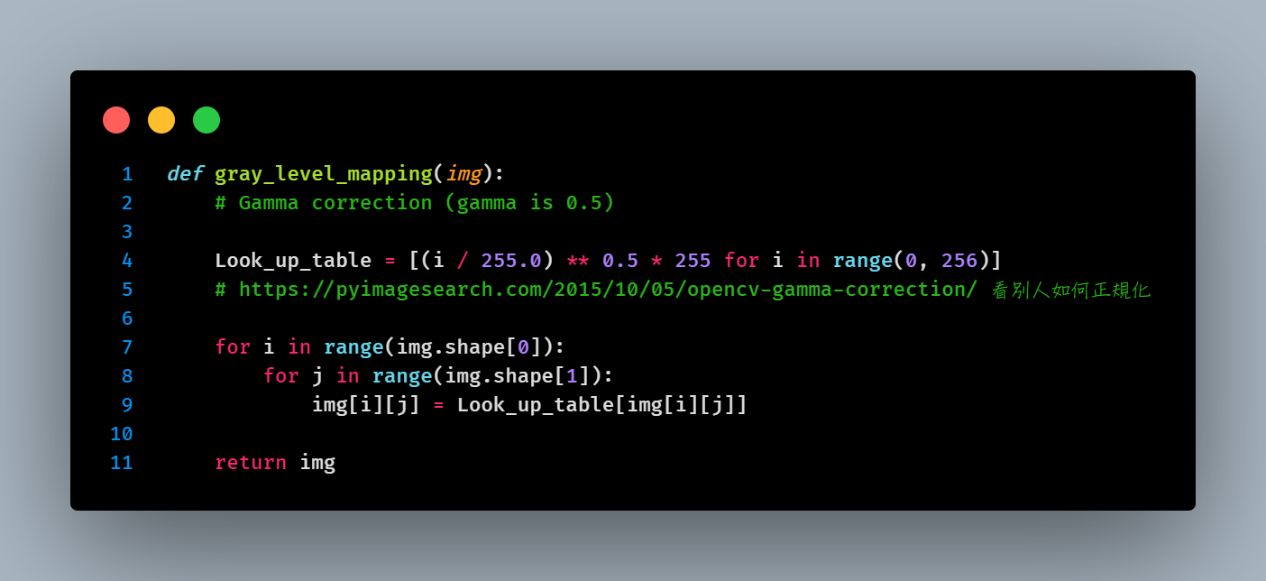
* 重要程式片段說明。

1. Gauss + different color coordinate + tensorflow + 雪山youtube影片



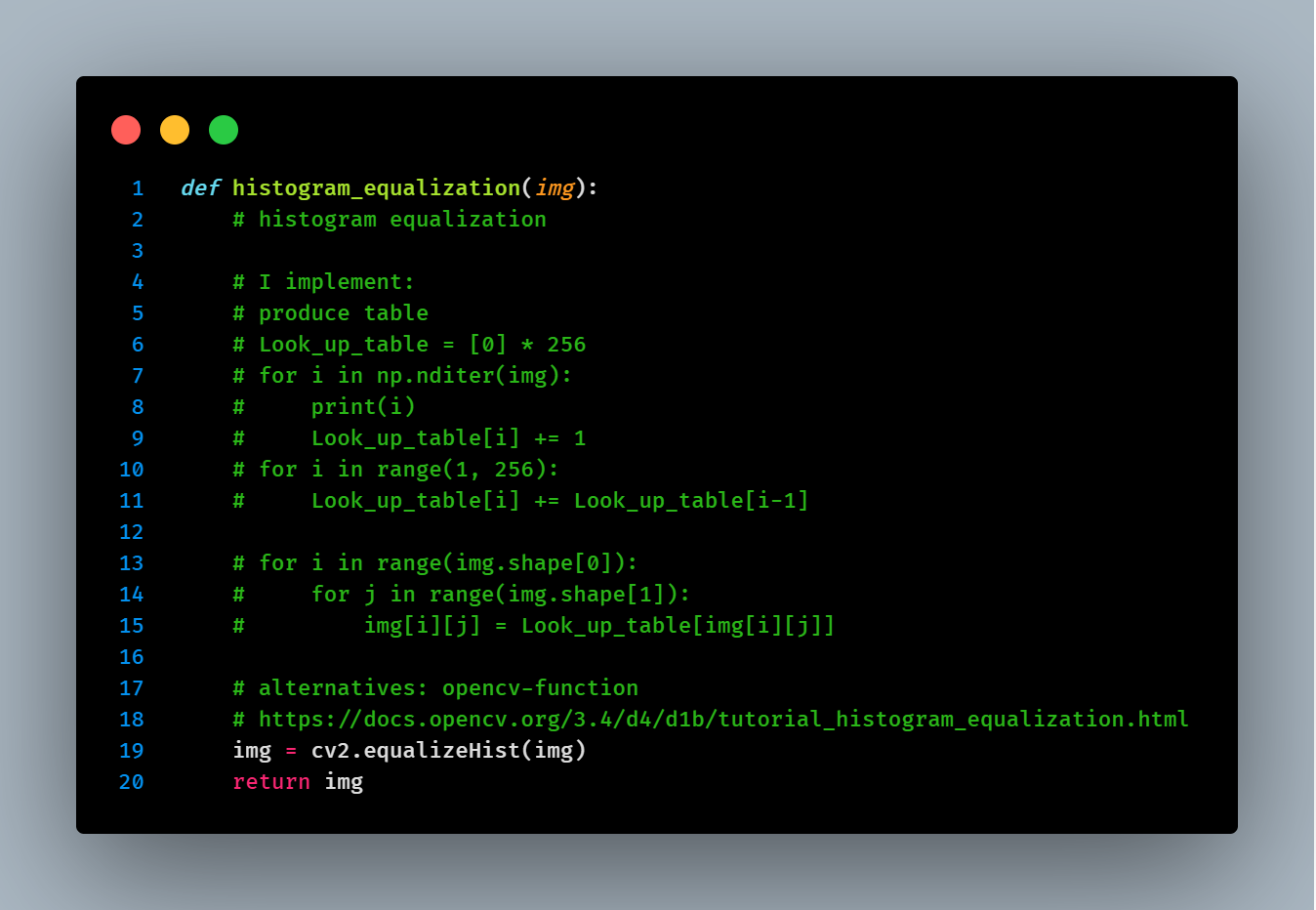
先做色彩座標系的轉換，轉換到HSV，提升飽和度，再轉回BGR座標系，利用OpenCV得到高斯kernal，利用tensorflow把該kernal與原圖片進行卷積運算。

1. Gray\_level\_mapping function



自己實作出gray\_level\_mapping的function，因為上課在講解常數c的時候沒有很懂，所以上網看別人如何正規化。

1. Histogram\_equalization



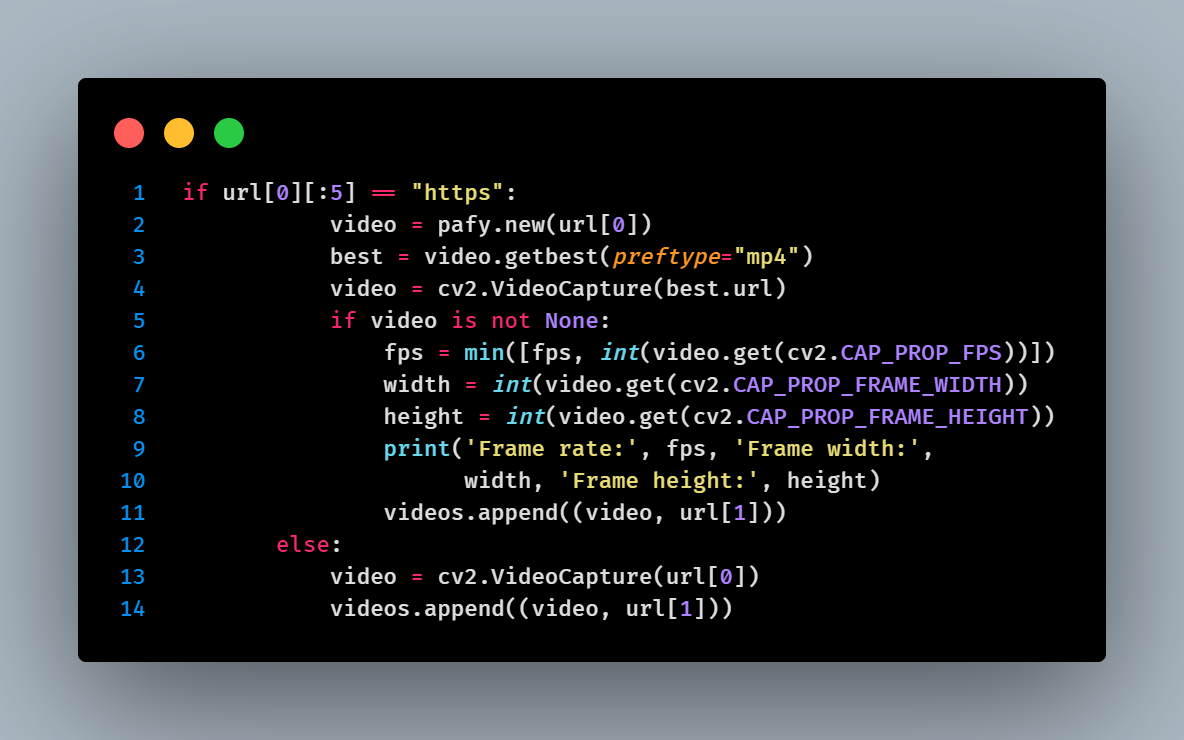
使用cv2.equalizeHist()，進行直方圖等化，增加對比。

1. High\_pass



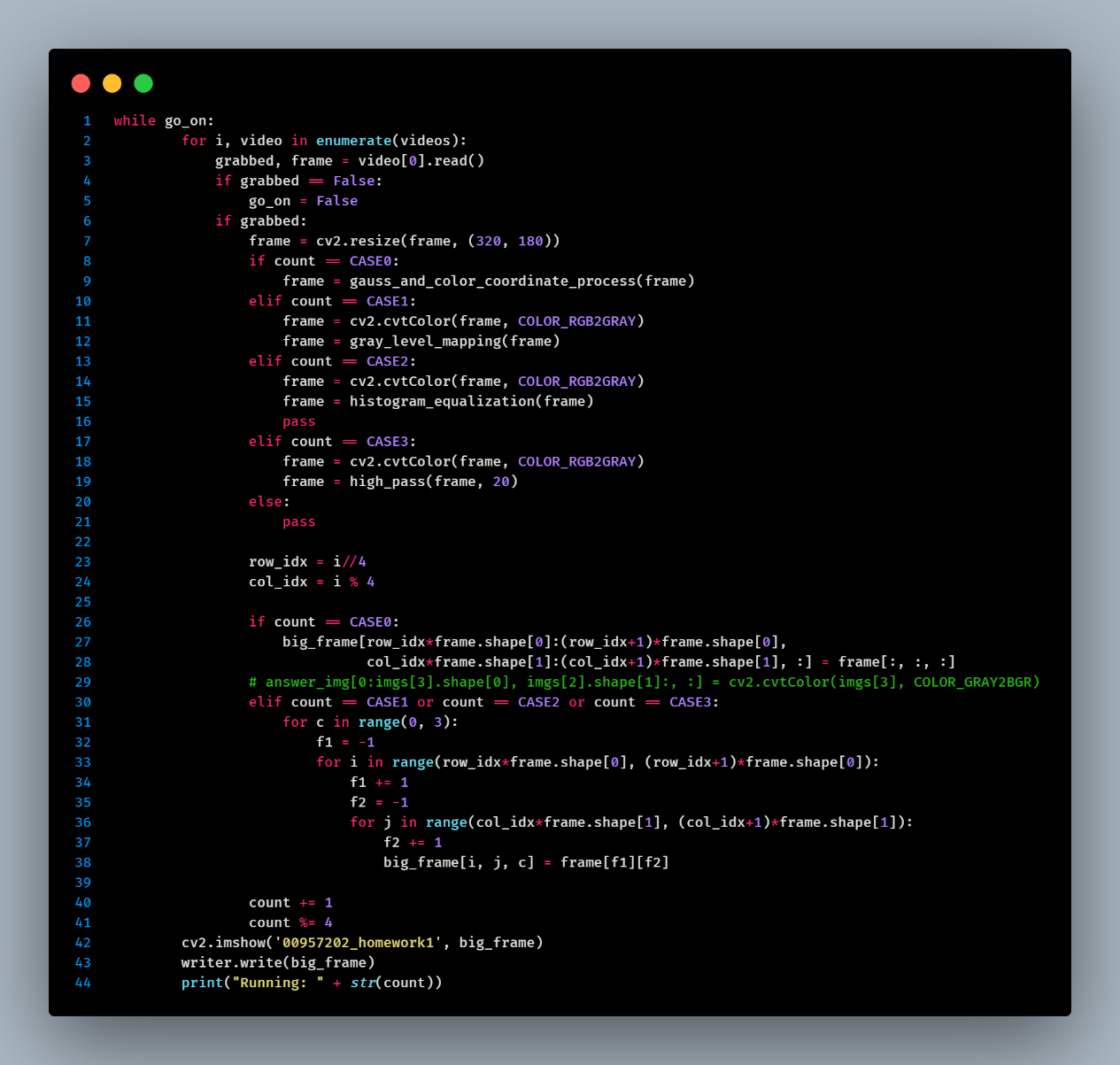
透過傅立葉轉換到 Fourier domain，並將四角對調，使低頻在中間，將低頻部分小於50 frequency 的移除，再四角對調做逆傅立葉轉換，回到原本domain。

1. Video 的抓取與 youtube影片的抓取



改寫張老師的檔案，將url前面字串與”https”判斷是否為video或是 youtube影片網址。

1. 主要程式碼



主要程式碼即將圖片進行轉換再input進去各自處理的function，做完後再統整成一張大圖輸出，最後的31行for迴圈是因為採用的是channel last結構，無法使用python簡潔的語法插入灰階(由2通道插入到3通道)，所以自己實作for迴圈將灰階得的RGB三通道都填相同數字。

# 結果

成果影片由左而右為

1. Gauss + different color coordinate(提升飽和) + tensorflow + 雪山youtube影片
2. Gray-level mapping + 老師的測試影片
3. Histogram equalization + 老師的測試影片
4. High-pass + 老師的測試影片

**YouTube成果連結:**

[**https://www.youtube.com/watch?v=p40P\_\_Xq9cs**](https://www.youtube.com/watch?v=p40P__Xq9cs)

**Github 程式連結:**

[**https://github.com/evan20010126/code/tree/main/python/Machine\_vision/Homework1**](https://github.com/evan20010126/code/tree/main/python/Machine_vision/Homework1)

**<裡面的homework\_gogogo.py>**

# 結論

在這次的作業中第一次碰到影像處理相關的問題，雖然處理的流程一開始就很清楚，而自己從頭到尾寫了一個檔案，不過在最後的排版上出了很多問題，所以最後改用老師的講義來改寫。

另一個問題是在資料結構的不熟悉，因為opencv灰階圖為2通道，原本查了很多資料看要怎麼寫成三通道的，網路上有看到可以用轉座標系的方式轉到BGR，不過應該是因為我的python版本關係，所以這個方法會報錯，於是自己寫了一個簡易的複製陣列元素的for迴圈。

覺得再這次的作業中對影像的資料結構與處理有更進一步的認識，希望以後也可以繼續努力！

# 參考文獻

<https://www.w3schools.com/python/numpy/trypython.asp?filename=demo_numpy_array_iterating4>

<https://docs.opencv.org/3.4/d4/d1b/tutorial_histogram_equalization.html>

<https://shengyu7697.github.io/python-opencv-gray-to-rgb/>

<https://docs.opencv.org/3.4/d4/d1b/tutorial_histogram_equalization.html>

<https://ithelp.ithome.com.tw/articles/10266785>

<https://stackoverflow.com/questions/68144185/cv2-error-opencv4-5-2-error-5bad-argument-in-function-cvtcolor>