Homework #1 Hybrid Images

Assigned on September 11, 2023

Due by September 25, 2023

112061535 陳嘉旻

Details

Please refer to README.md in the homework folder to see more details about how to finish your code implementation in this homework.

1 Implementation (45%)

1.1 Image filtering (20%)

Please finish the function **my_imfilter** in the file **my_imfilter.py** and briefly describe your implementation ideas. Noted that you can not use convolution function from any python built-in libraries (eg. numpy, scipy).

```
output = np.zeros_like(image)

R = image[:,:,0]
G = image[:,:,1]
B = image[:,:,2]

H_image = image.shape[0]
W_image = image.shape[1]
H_imfilter = imfilter.shape[0]
W_imfilter = imfilter.shape[1]

# padding size according to kernal, processing boundary issue of the image
v_offset = (H_imfilter-1) // 2
h_offset = (W_imfilter-1) // 2

# padding three channel with vertical and horizontal offset
R_pad = np.pad(R, (v_offset, h_offset), mode='constant')
G_pad = np.pad(G, (v_offset, h_offset), mode='constant')
B_pad = np.pad(B, (v_offset, h_offset), mode='constant')

# do convolution for loop for each pixel in the image
for h in range(H_image):
    for w in range(W_image):
        output[h][w][0] = np.sum(np.multiply(R_pad[h:h+H_imfilter, w:w+W_imfilter], imfilter))
        output[h][w][1] = np.sum(np.multiply(G_pad[h:h+H_imfilter, w:w+W_imfilter], imfilter))
        output[h][w][2] = np.sum(np.multiply(B_pad[h:h+H_imfilter, w:w+W_imfilter], imfilter))
```

首先，根據進來的 filter 大小決定 padding 的長寬，透過 **np.pad()** 對 RGB 三個 channel 做 zero-padding，讓 output image 大小可以和 input image 大小相等。

再來利用 for loop 對於每一個 pixel 做 convolution，將 image 上每個和 filter 大小相等的區塊和 filter 做 multiplication，得到最終 per pixel 的值。

1.2 Extract and combine the high-frequency and low-frequency signals (20%)

Please finish the **TODO** in the file **hw1.py**.

```
low_frequencies = my_imfilter(image1, gaussian_filter)
high_frequencies = image2 - my_imfilter(image2, gaussian_filter)
hybrid_image = normalize(low_frequencies + high_frequencies)
```

使用 gaussian filter (size depend on **cutoff_frequency**)把圖片經過前面一步的 **my_imfilter** function 後，可以得到 low_frequency 的圖片；原本的圖片和經過 fliter 的圖片相減則會得到 high_frequency 圖片；最後將兩張圖片相加並經過 **normalize** 後，就能得到最終的 hybrid image。

1.3 Others (5%)

Please list the additional packages and versions required in your implementation and describe how to run your code. (make sure we can run your code)

(1) Additional packages
No. (all settings are the **same as recommendation**)

(2) How to run my code
python HW1.py

2 Experiments (30%)

2.1 Hybrid Image (10%)

Put your hybrid result from the cat-dog pair and briefly explain your result.

Hybrid image 中，貓是 high frequency，狗是 low frequency，cutoff_frequency 是預設的7。

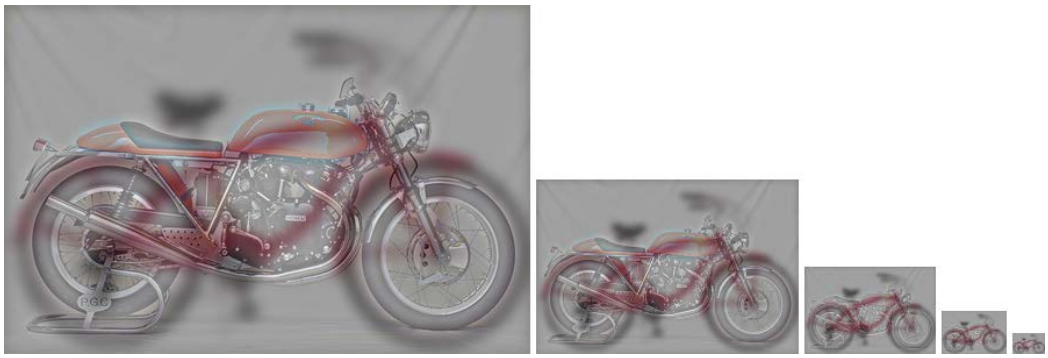
近距離會看到 high frequency的部分，也就是貓；當把距離拉遠，就可以看到 low frequency 的狗。

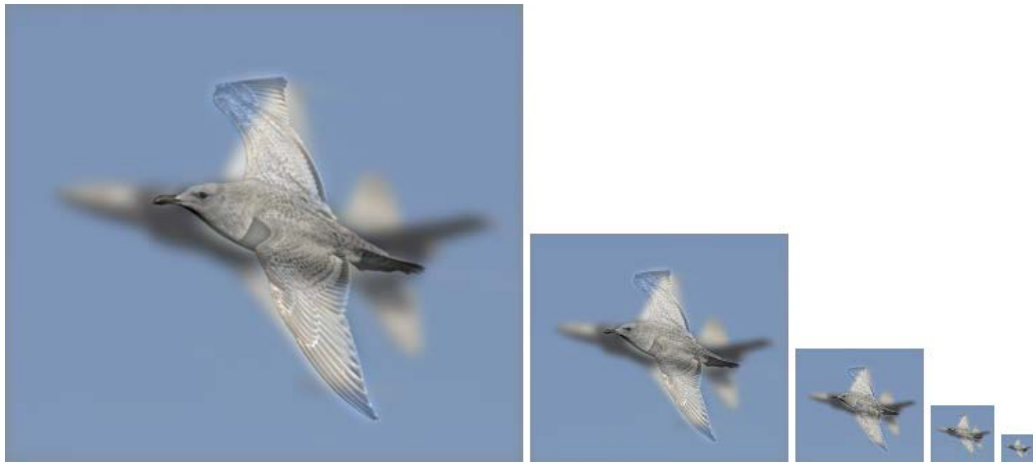
2.2 Other hybrid images (10%)

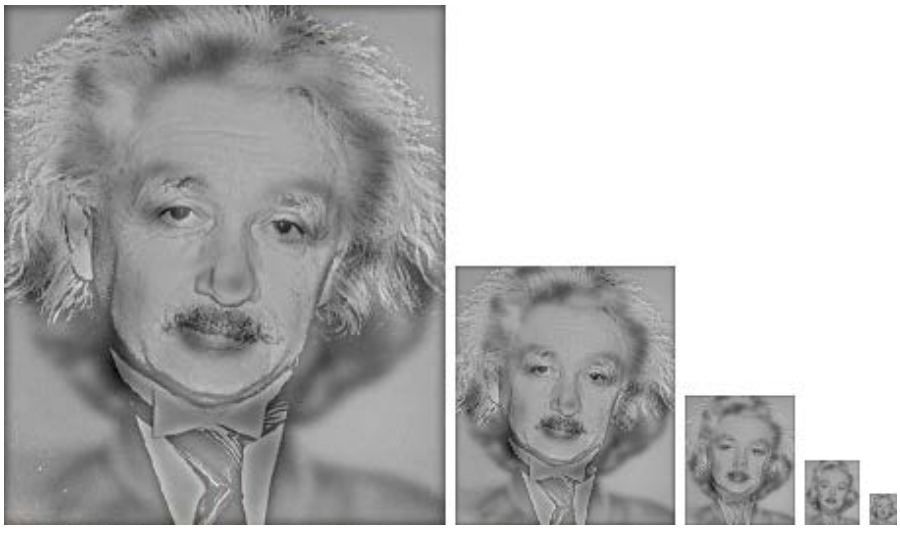
Try different pairs of pictures in the folder **/data** and put your results here. Comparing the result of Problem 2.1, what's the difference?

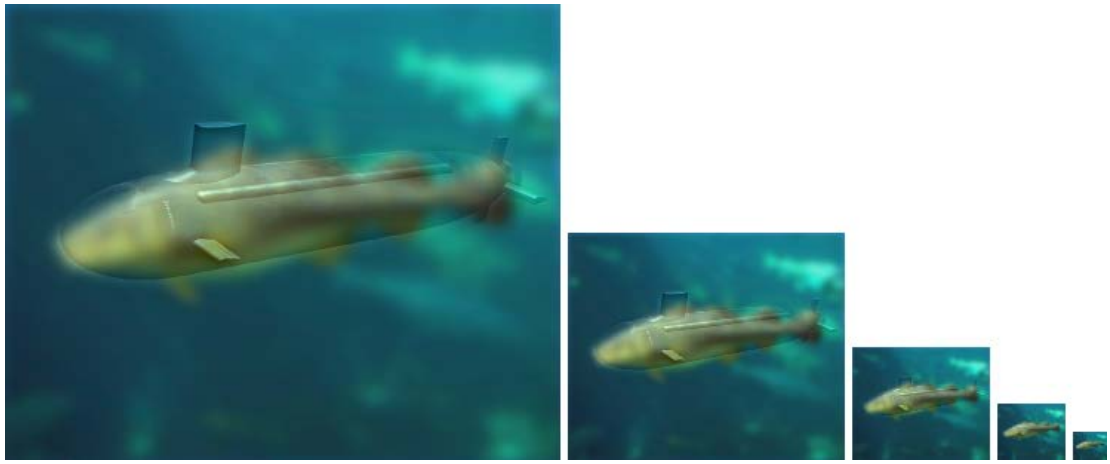
下一頁的四組圖片分別是：腳踏車與機車、鳥與噴射機、Einstein 與 Marilyn、魚與潛水艇的 hybrid image pair。

這裡主要會根據不同的 pair 調整 **cutoff_frequency** ，當發現高頻的圖案太過明顯、蓋過低頻的圖時，就會調低 cutoff_frequency ，反之亦然。逐步調整得到最合適的 **cutoff_frequency**，讓兩張圖片的結合不至於太過突兀。









2.3 Customized hybrid images (10%)

Gather your own picture pairs and show your results of hybrid results. Briefly explain the difference between customized results and results from Problem 2.1 and 2.2.

這裡我有特別挑選了**顏色、輪廓分布較接近**的兩張圖片做 hybrid image，所以結果大致和 2.1、2.2 相近，Benedict 是高頻的圖，otter 則是低頻。近距離時會看到 Benedict ，遠距離就會看到 otter 慢慢浮現。

(1) Original picture pairs

(2) High / Low frequencies

High frequency Low frequency

(3) Hybrid image

3 Discussion (25%)

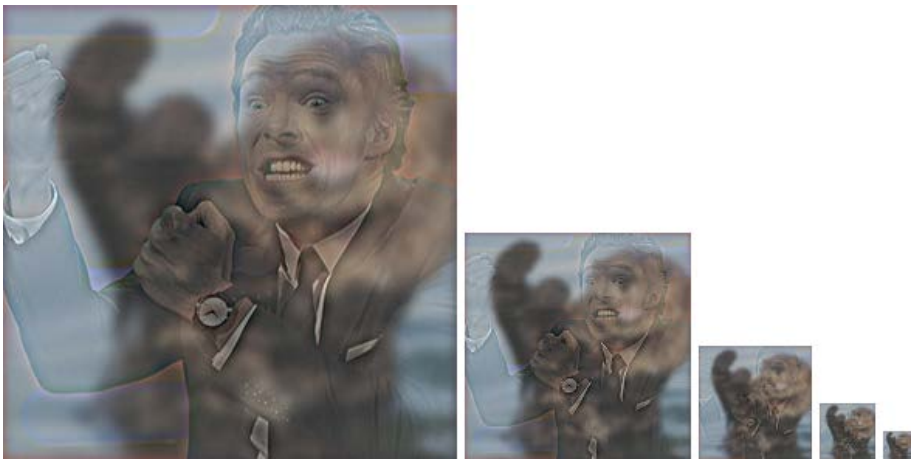
Do you discover anything special in your experimental results?

What applications do you think this technology can be used for?

從自己實作還有原先的 pair 發現，不是所有的 image pair 都適合這個 hybrid image的方式，要位置和顏色分布大致對的上，並且通常兩張圖片的背景都盡量是比較乾淨或是blur的，才能成功。以下放上兩組失敗的例子：

1. Yet another Benedict Cumberbatch-otter pair again
   A. Original picture pairs


   B. Hybrid image



2. 柯達鴨
   A. Original picture pairs


B. Hybrid image

在第一組圖片中，如果單看原始圖片兩者其實非常相似，但是因為手的位置還有比例整體不太對，很難有 2.3 裡面的效果。

第二組圖片也是類似的問題，尤其可達鴨的顏色較突兀，而且整體造型比較偏卡通的比例，和真人的身形比例還是有差距，原本有想過要把柯文哲的圖片放大，讓兩張圖的比例變得接近一點，但這樣兩張圖片的 pixel 值就對不上，在最後要得到 hybrid image 的相加階段會有問題，要另外再做 upsampling 成相同的畫素才有辦法做。

從以上兩組失敗的例子，可以得到結論：圖片的顏色分布、輪廓等等越 align，hybrid image 的結果會越成功。

而題目原先給的都是一些比較經典的 hybrid image pair，像是 einstein 和 marilyn，兩張圖片整體輪廓和色彩分布都非常接近，所以疊圖之後也很成功。

這個技術感覺可以用在未來小孩的預測，將兩個夫妻的臉部照片做成 hybrid image，再透過夫妻兩個高頻低頻的2種(夫高頻/妻低頻、夫低頻/妻高頻)組合，綜合得到未來小孩可能的長相。