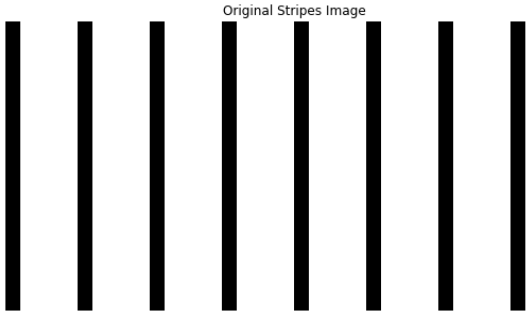
數位影像處理 HW2

電機系控制組碩一 R13921109 陳柏丞

Problem 1:

An image containing vertical bars like the one shown below was blurred using square box kernels of size 23, 25, and 45 pixels on the side, respectively. The vertical bars are 5 pixels wide, 100 pixels high, and their separation is 20 pixels.

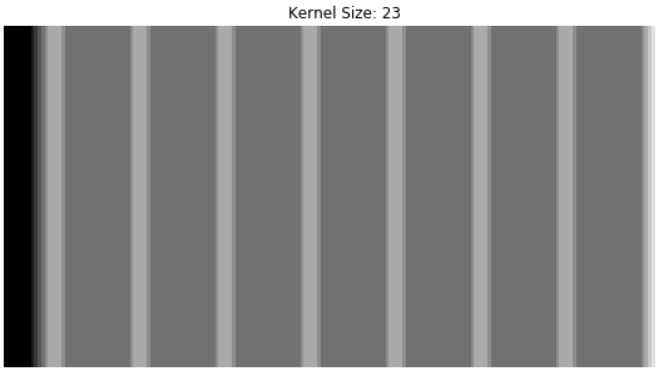
a) Generate the image according to the specification and Implement the lowpass filtering in Matlab



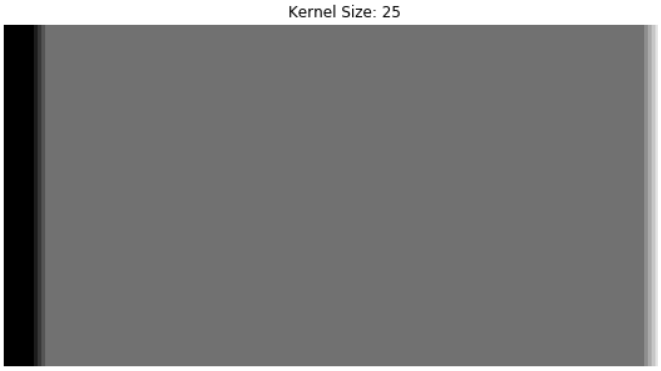
由 python matplotlib 生成條紋圖像，黑線寬度 5 pixels，白邊寬度 20 pixels，圖片寬度 200 pixels。

b) If the filters are implemented correctly, you can see a clear separation between the filtered vertical bars for box kernels of size 23 and 45. However, such separation does not exist for the box kernel of size 25. More specifically, the bars have merged. Explain the reason.

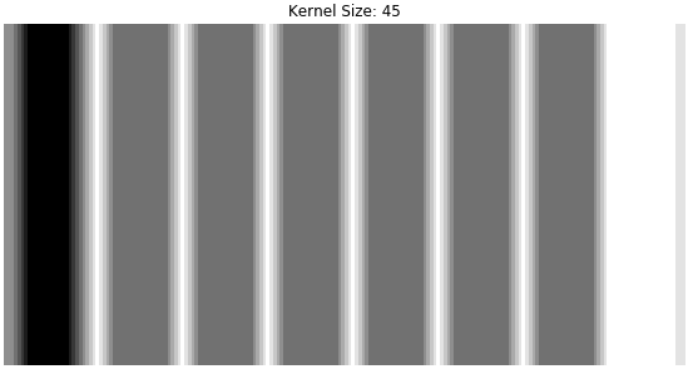
1. 利用 23*23 box kernels 進行低通濾波(模糊濾波)



2. 利用 25*25 box kernels 進行低通濾波(模糊濾波)



3. 利用 45*45 box kernels 進行低通濾波(模糊濾波)

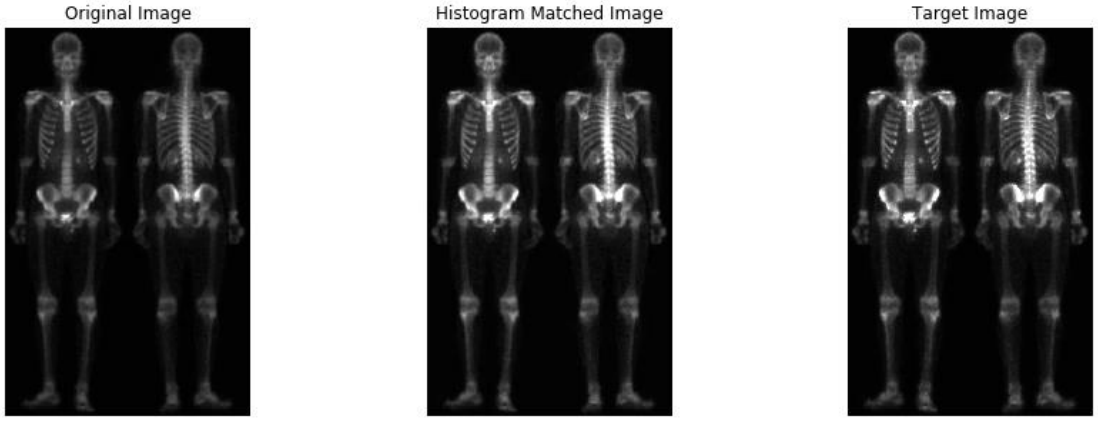


由輸出結果可以看出在 box kernels 尺寸在 23*23 與 45*45 時，濾波後條紋分非常明顯，而在 box kernels 尺寸在 25*25 時，整張圖片經過濾波後看不到條紋的輸出，原因是在濾波時的 box kernels 的尺寸剛好與原圖條紋間終點的距離相等，黑條紋中心與下一條黑條紋中心的距離為 2.5(中心到邊)+20(白色區域)+2.5(第二條黑條紋邊到中心)=25，再將整個 box kernels 模糊平均時顏色剛好會呈現整片灰色，又因為最左側只有黑線濾波後顏色較黑，最右側是白色區域濾波後才呈現白色。

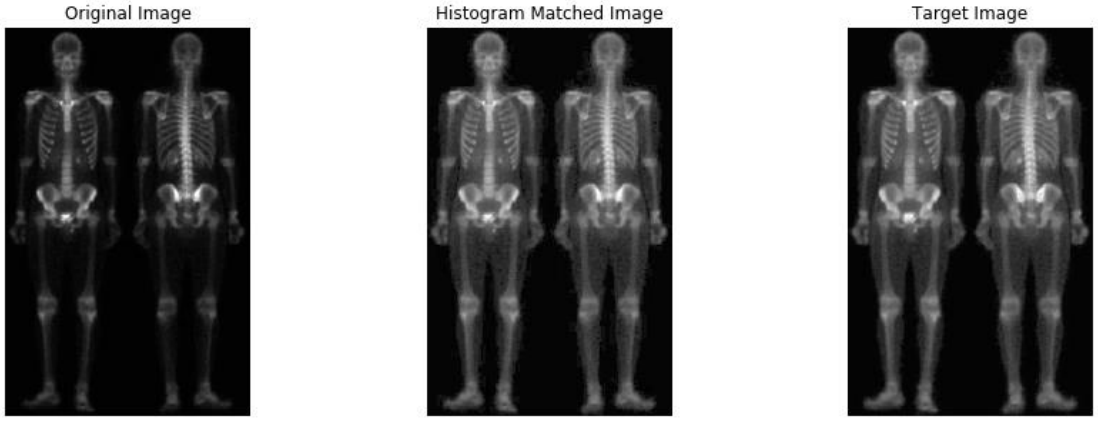
Problem 2:

The images below were obtained using a combination of filters described in Section 3.8. Now, let's see if the results can be obtained by histogram matching.

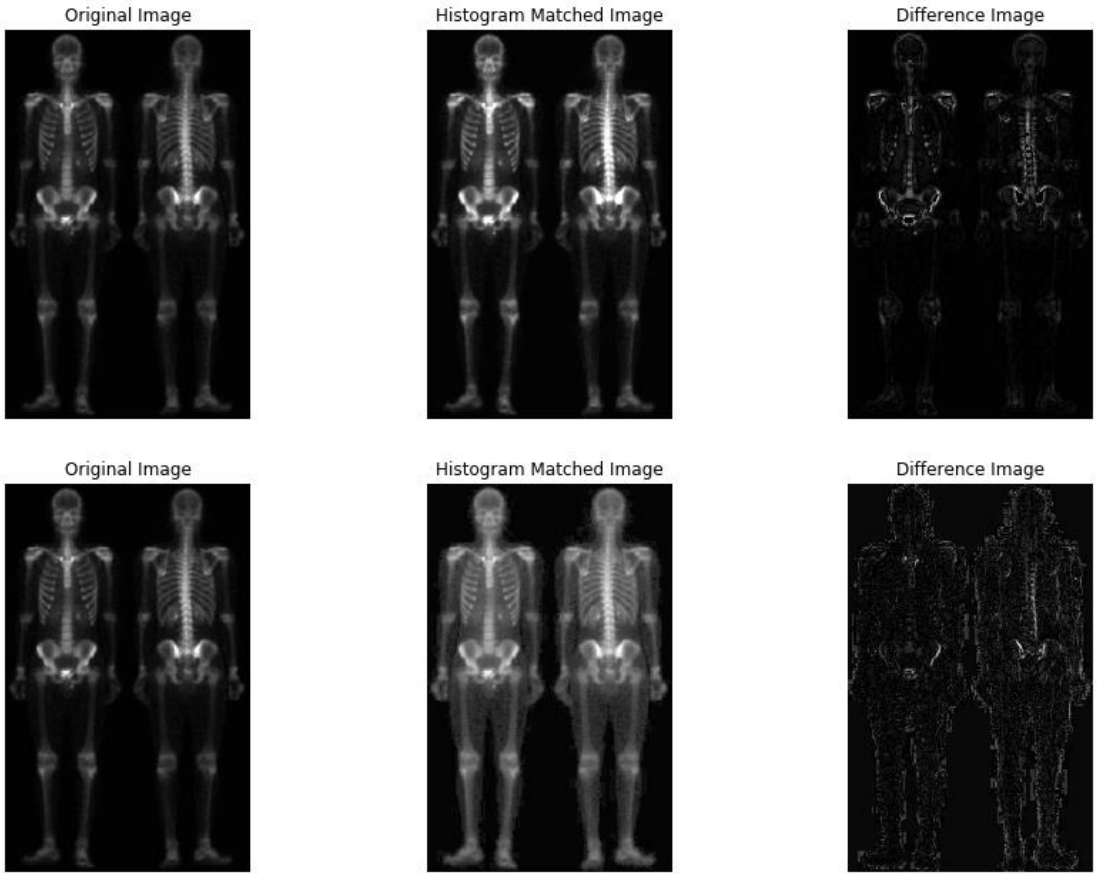
a) Design a histogram matching algorithm to convert the left image to the middle image.



b) Design a histogram matching algorithm to convert the left image to the right image.



Also provide a difference image, probably scaled up to 255, in both cases to show the performance of your algorithm.



繪製 Target image 和 Histogram matching image 誤差結果，先計算兩圖差值，減去最小值(目的在將誤差最小的值設定為 0)，再將其值除上相減後最大的值(目的在進行歸一化)，而後 scale 設定為 255，以便輸出檢查。

公式、代碼如下:

$$g_m = g - \min(g) \ ，g_s = [g_m / \max(g_m)]$$

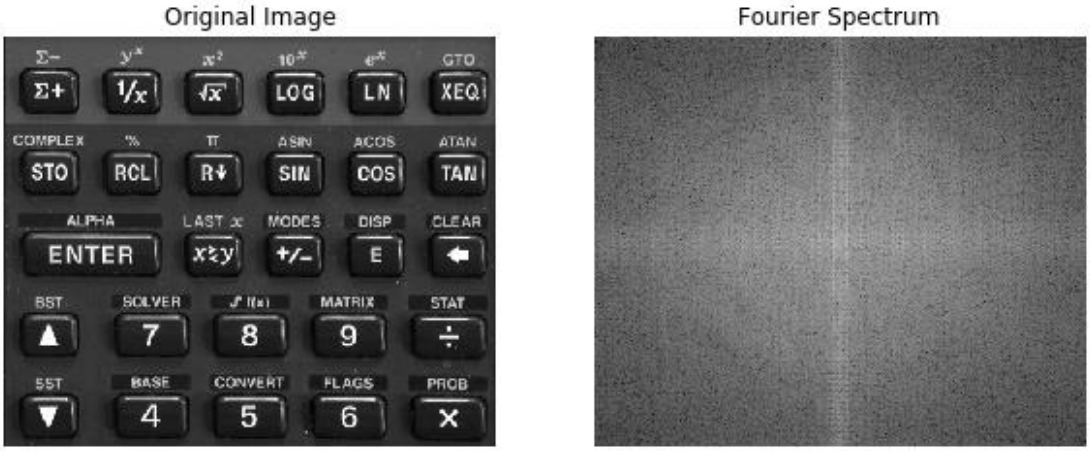
```
def difference_image(matched_image,image2):
    g = abs(matched_image-image2)
    gm = g-np.min(g)
    gs = 255*(gm/np.max(gm))
    return gs
```

其中，g 為 Target image 和 Histogram matching image 誤差，g_s為輸出。

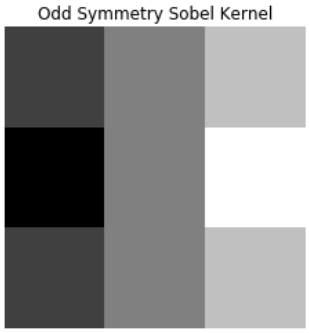
Problem 3:

Follow the steps outlined in Section 4.7 to repeat Example 4.15, pp. 271-273, using the vertical Sobel kernel shown in Fig. 4.38(a) and the test image "keyboard.tif." You may use any existing library to compute Fourier transform.

a) Show the Fourier spectrum of the test image "keyboard."

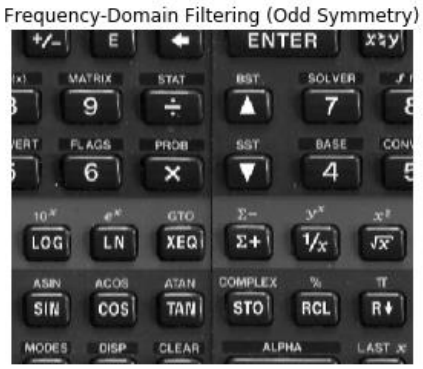


b) Enforce odd symmetry on the kernel. Show the kernel.

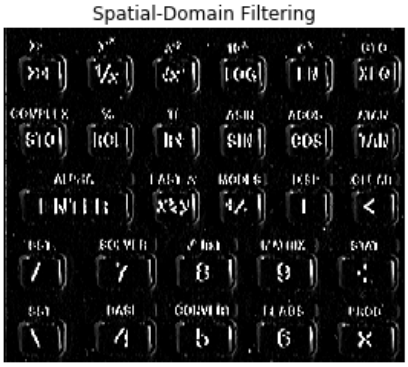


$$, \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$

c) Show the result of frequency-domain filtering of the test image using the vertical Sobel kernel.

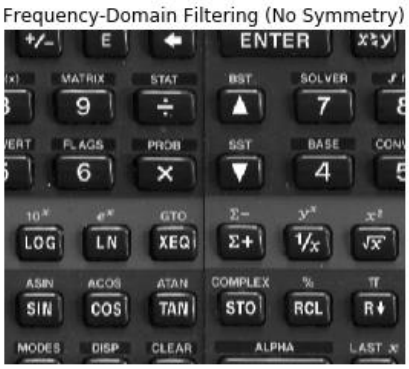


d) Compare your result in c) with the result of space-domain filtering.



Space-domain filtering 強化了圖像中的邊緣，尤其凸顯在垂直邊緣。

e) Show the result of frequency-domain filtering without enforcing odd symmetry on the kernel.



Problem 4:

Embedding a 2-D array of even (odd) dimensions into a larger array of zeros of even (odd) dimensions keeps the symmetry of the original array, provided that the centers coincide. Show that this is true also for the following 1-D arrays. That is show that the larger arrays have the same symmetry as the smaller arrays. For arrays of even length, use arrays of 0's ten elements long. For arrays of odd lengths, use arrays of 0's nine elements long.

a) {0, − b, − c, 0, c, b}

b) {a, b, c, d, c, b}

c) {0, − b, − c, c, b}

奇函數定義: $f(x) = f(len - x), f(0)$ is immaterial

偶函數定義: $f(x) = -f(len - x), , f(0)$ is must be 0

判斷式如下:

```python
def is_symmetric(array):
    """檢查數組是否為偶對稱或奇對稱"""

    # 檢查是否為偶對稱
    is_even = True
    for i in range(1, len(array) // 2):
        if array[i] != array[len(array) - i]:
            is_even = False
            break

    # 檢查是否為奇對稱
    is_odd = (array[0] == 0)  # 第一個元素必須為0
    if is_odd:
        for i in range(1, len(array) // 2):
            if array[i] != -array[len(array) - i]:
                is_odd = False
                break

    if is_even:
        return 'even'
    elif is_odd:
        return 'odd'
    else:
        return 'False'
```

結果如下:

```
In [42]: runfile('C:/My/台大/DIP/HW2/R13921109/p4.py', wdir
Is the array1 Even or Odd Array Symmetric?: odd
Is the array2 Even or Odd Array Symmetric?: False
Is the array3 Even or Odd Array Symmetric?: odd

In [43]:
```

(a)、(c)擴張完仍然是奇函數

(b)擴張完後並不是偶函數