

程式用Python編寫，於Colab上執行。

網址：

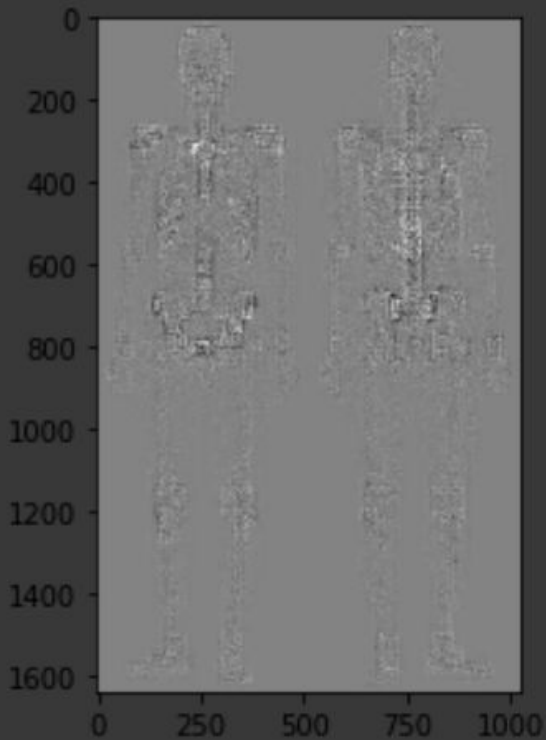
<https://colab.research.google.com/drive/1OYelrl8Vu1p9mL1rDUTxetM2Nx09f35O?authuser=1#scrollTo=gqEamhwsrpvP>

```
1  import skimage.io as io
2  import skimage.filters as fl
3  import skimage.util as ut
4  import skimage.exposure as ex
5  import numpy as np
6  import scipy.ndimage as ndi
7  import matplotlib.pyplot as plt
```

上面是程式碼用的函式庫

```
1  # (b) Laplacian of image
2  img_lap = fl.laplace(img)
3  img_lap -= img_lap.min()
4  img_lap /= img_lap.max()
5  io.imshow(img_lap, cmap=plt.cm.gray)
6  print(img_lap.shape)
```

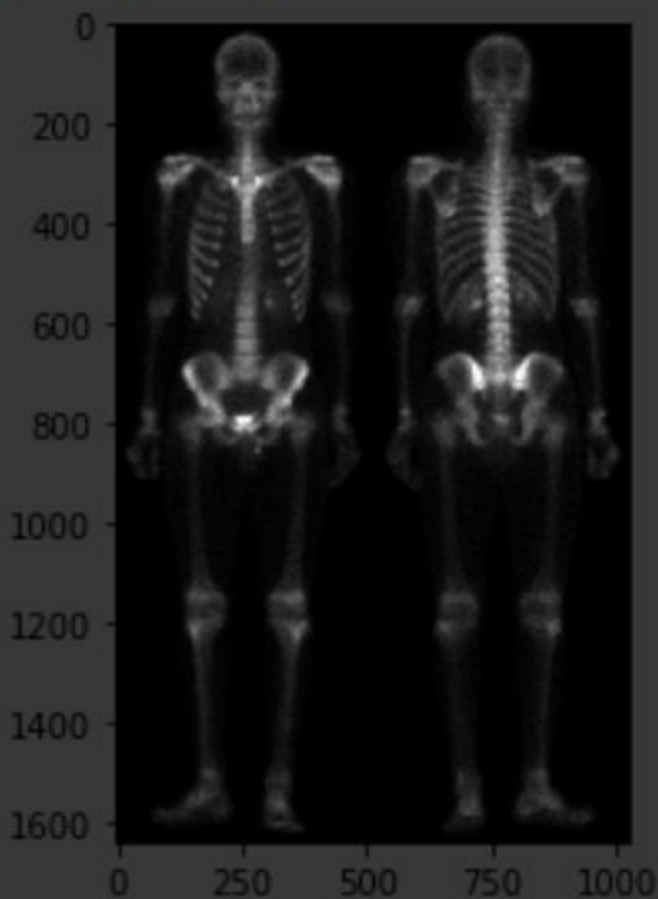
(1641, 1026)



這是圖片(b)及其程式碼，將圖片套入skimage.filters的laplace，並減去他的最小值（為負數），接著除以最大值，令圖片的數值處於0到1之間

```
1  # (c) = (a) + (b)
2  sharpened_img = img + img_lap
3  sharpened_img /= sharpened_img.max()
4  io.imshow(sharpened_img, cmap=plt.cm.gray)
5  print(sharpened_img.shape)
```

(1641, 1026)



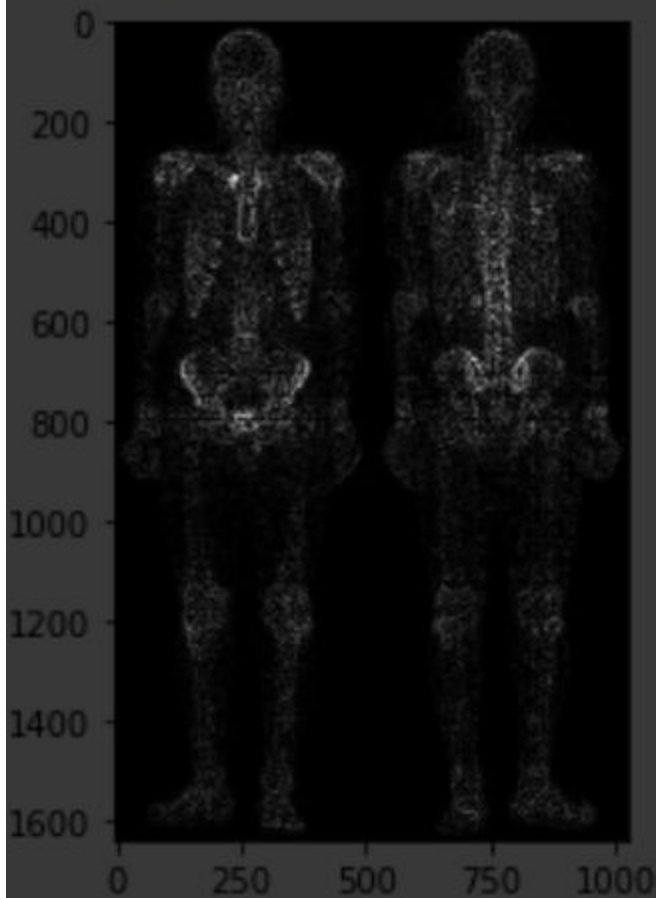
這是圖片(c)及其程式碼，將圖片(b)和原圖相加，然後做了跟圖片(b)類似的處理。

```

1  # (d) Sobel gradient of (a)
2  img_sobel_h = np.absolute(fl.sobel_h(img))
3  img_sobel_v = np.absolute(fl.sobel_v(img))
4  img_sobel = img_sobel_h + img_sobel_v
5  img_sobel /= img_sobel.max()
6  io.imshow(img_sobel, cmap=plt.cm.gray)
7  print(img_sobel.shape)

```

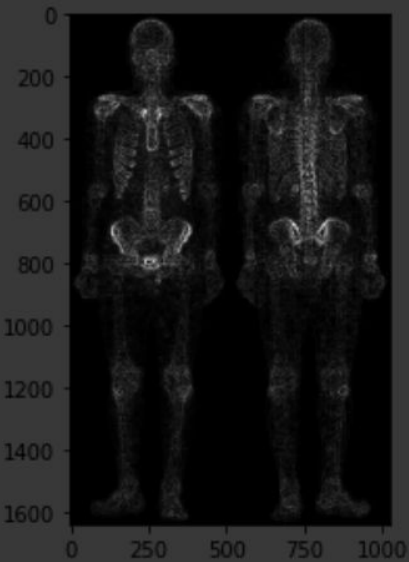
1641, 1026)



這是圖片(d)及其程式碼，使用的是skimage.filters的sobel_h、sobel_v，將原圖做了課本上描述的處理後（sobel gradient），然後除以圖片裡的最大值，就是上圖那樣的處理。

```
1 # (e) Sobel image smoothed with a 5x5 box filter
2 img_e = ndi.convolve(img_sobel, np.ones((5, 5)) / 25, mode = 'constant')
3 img_e /= img_e.max()
4 io.imshow(img_e, cmap=plt.cm.gray)
5 print(img_e.shape)
```

(1641, 1026)



這是圖片(e)及其程式碼，用5x5的box filter來smooth得到。

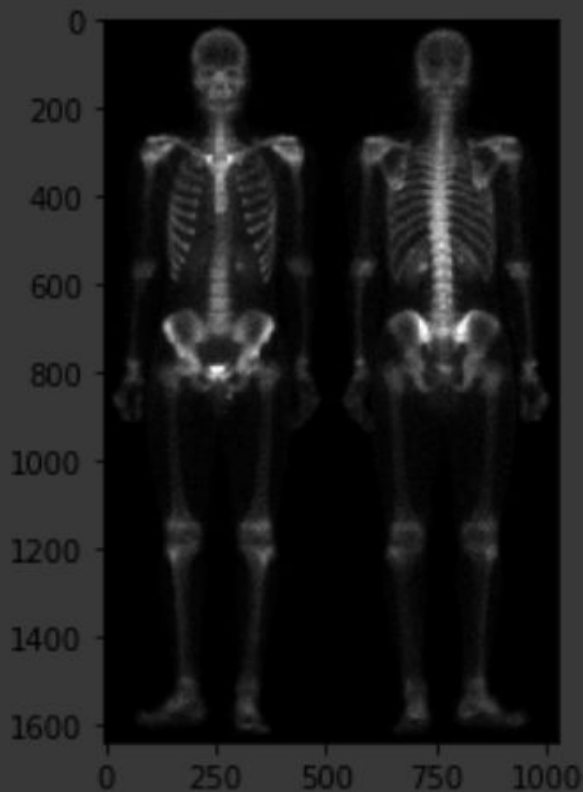
```
1 # (f) (b) product (e)
2 img_f = np.empty(img_e.shape)
3 height = img_f.shape[0]
4 width = img_f.shape[1]
5 for i in range(height):
6     for j in range(width):
7         img_f[i][j] = img_lap[i][j] * img_e[i][j]
8 io.imshow(img_f, cmap=plt.cm.gray)
9 print(img_f.shape)
```



這是圖片(f)及其程式碼（幾乎看不到了），由圖片(b)和圖片(e)相乘得到

```
1  # (g) = (a) + (f)
2  img_g = img + img_f
3  io.imshow(img_g / img_g.max(), cmap=plt.cm.gray)
4  print(img_g.shape)
```

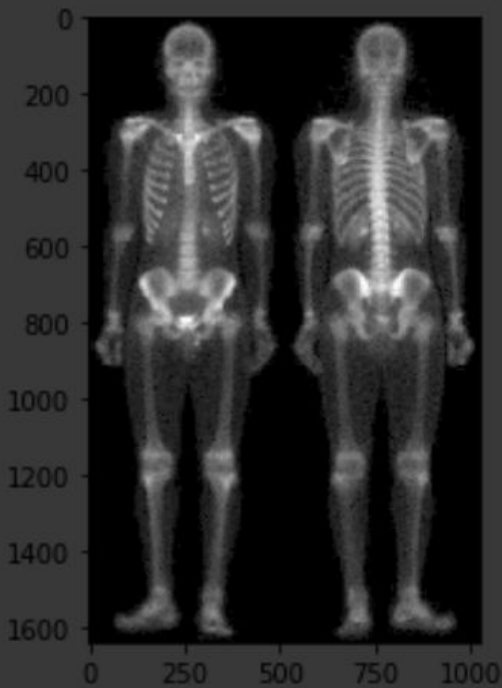
(1641, 1026)



這是圖片(g)及其程式碼（跟原圖到底有哪裡不同啊），由圖片(f)和原圖相加得到。

```
1 # (h)
2 img_h = ex.adjust_gamma(img, gamma = 0.5, gain = 1)
3 io.imshow(img_h, cmap=plt.cm.gray)
4 print(img_h.shape)
```

(1641, 1026)



這是圖片(h)及其程式碼，使用skimage.exposure的adjust_gamma，gamma值設為0.5，gain設成1。

以下放置製作好的圖。

a b

c d

e f

g h

