Roll Number	Name	Teacher Name	Subject Name	Assignment Number
MSDSF21M519	Durrah Khan	Dr Muhammad Ali	Digital Image Processing	2nd

Digital Image Processing Assignment 2

Combining Spatial Enhancement Methods

In [1]:

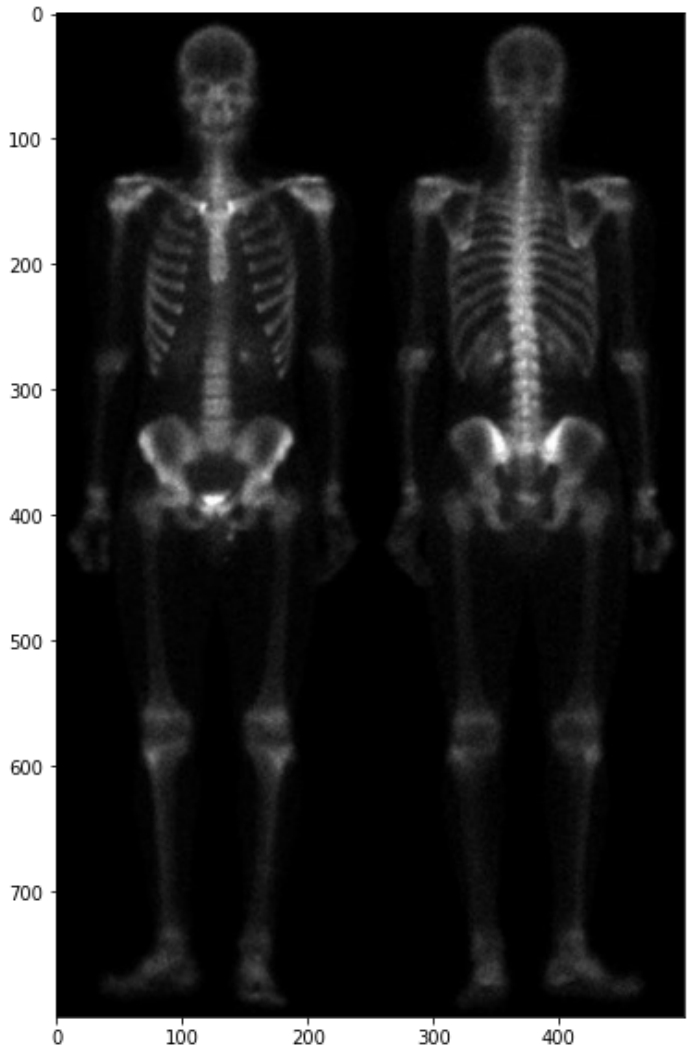
```
#import neccessary modules
import numpy as np #numpy for image matrix manipulation
import cv2 as cv   #opencv for image reading,display & filtering
import matplotlib.pyplot as plt
```

In [2]:

```
orignal_image = cv.imread('dip.tif') #read image from local drive and gray scale
plt.figure(figsize=(20,10))
plt.imshow(orignal_image)    #display image in new window
```

Out[2]:

```
<matplotlib.image.AxesImage at 0x7f3401b37790>
```



In [3]:

```
#laplacian filter 1
lap_filter_1 = np.array([[0, 1, 0],
                         [1, -4, 1],
```

```
                          [0, 1, 0]])

#laplacian filter 2
lap_filter_2 = np.array([[0, -1, 0],
                         [-1, 4, -1],
                         [0, -1, 0]])

#laplacian filter 3
lap_filter_3 = np.array([[0, 1, 0],
                         [1, -8, 1],
                         [0, 1, 0]])

#laplacian filter 4
lap_filter_4 = np.array([[0, -1, 0],
                         [-1, 8, -1],
                         [0, -1, 0]])
```

In [4]:

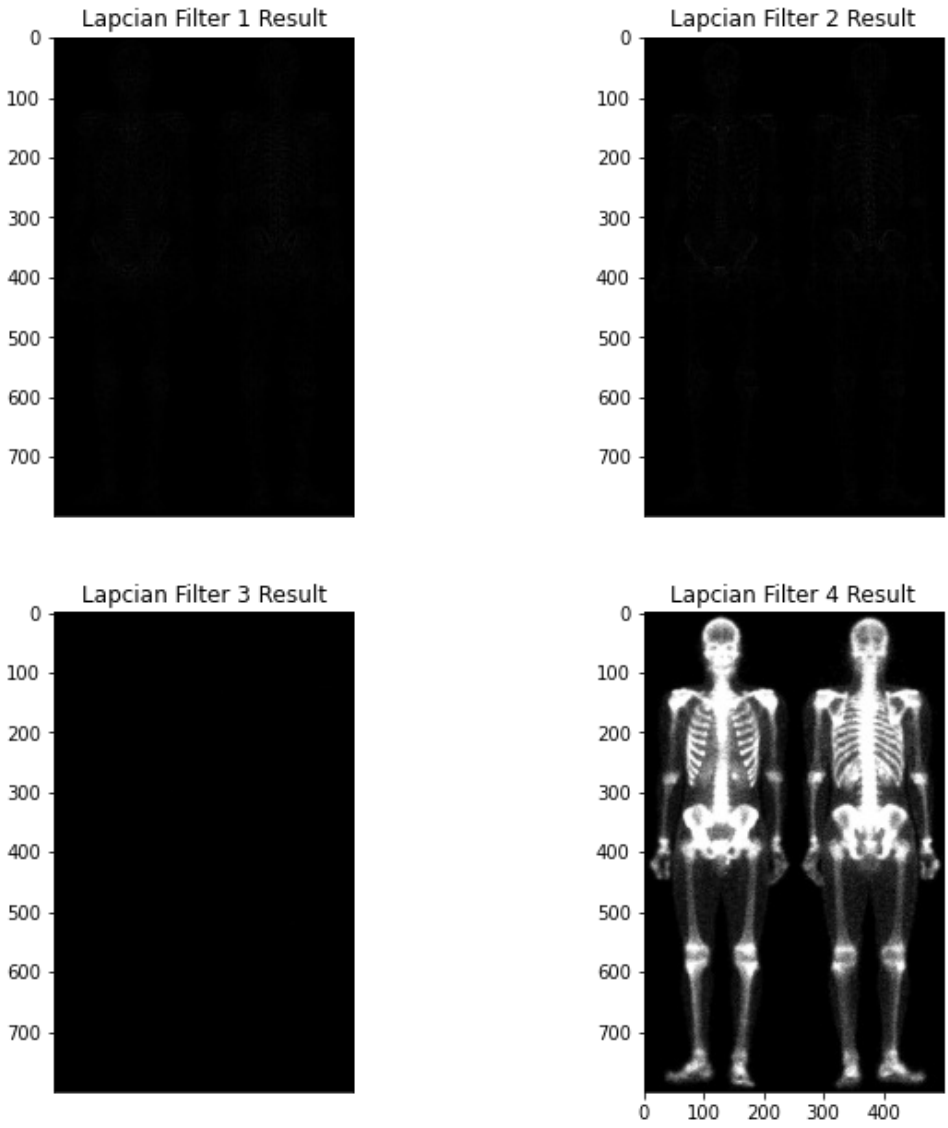
```
lap_res_1 = cv.filter2D(orignal_image, ddepth=-1, kernel=lap_filter_1) #laplacian filter
1
lap_res_2 = cv.filter2D(orignal_image, ddepth=-1, kernel=lap_filter_2) #laplacian filter
2
lap_res_3 = cv.filter2D(orignal_image, ddepth=-1, kernel=lap_filter_3) #laplacian filter
3
lap_res_4 = cv.filter2D(orignal_image, ddepth=-1, kernel=lap_filter_4) #laplacian filter
4
```

In [5]:

```
#display Results
#create subplots with matplot library

# fig, axarr = plt.subplots(2,2)
# fig.set_figheight(12)
# fig.set_figwidth(12)
# axarr[0,0].imshow(lap_res_1)
# axarr[0,1].imshow(lap_res_2)
# axarr[1,0].imshow(lap_res_3)
# axarr[1,1].imshow(lap_res_4)
```

In [6]:

```
fig = plt.figure()
fig.set_size_inches(10.5, 10.5)
ax1 = fig.add_subplot(221)
ax2 = fig.add_subplot(222)
ax3 = fig.add_subplot(223)
ax4 = fig.add_subplot(224)

#display image to each axes
ax1.imshow(lap_res_1)
ax2.imshow(lap_res_2)
ax3.imshow(lap_res_3)
ax4.imshow(lap_res_4)

#set text each subplot && remove x axes
ax1.title.set_text('Lapcian Filter 1 Result')
ax1.get_xaxis().set_visible(False)
ax2.title.set_text('Lapcian Filter 2 Result')
ax2.get_xaxis().set_visible(False)
ax3.title.set_text('Lapcian Filter 3 Result')
ax3.get_xaxis().set_visible(False)
ax4.title.set_text('Lapcian Filter 4 Result')
ax3.get_xaxis().set_visible(False)

#save image to local drive
# plt.savefig('laplacian_result.jpg')

#display figure
plt.show()
```



Combining laplacian result to original image

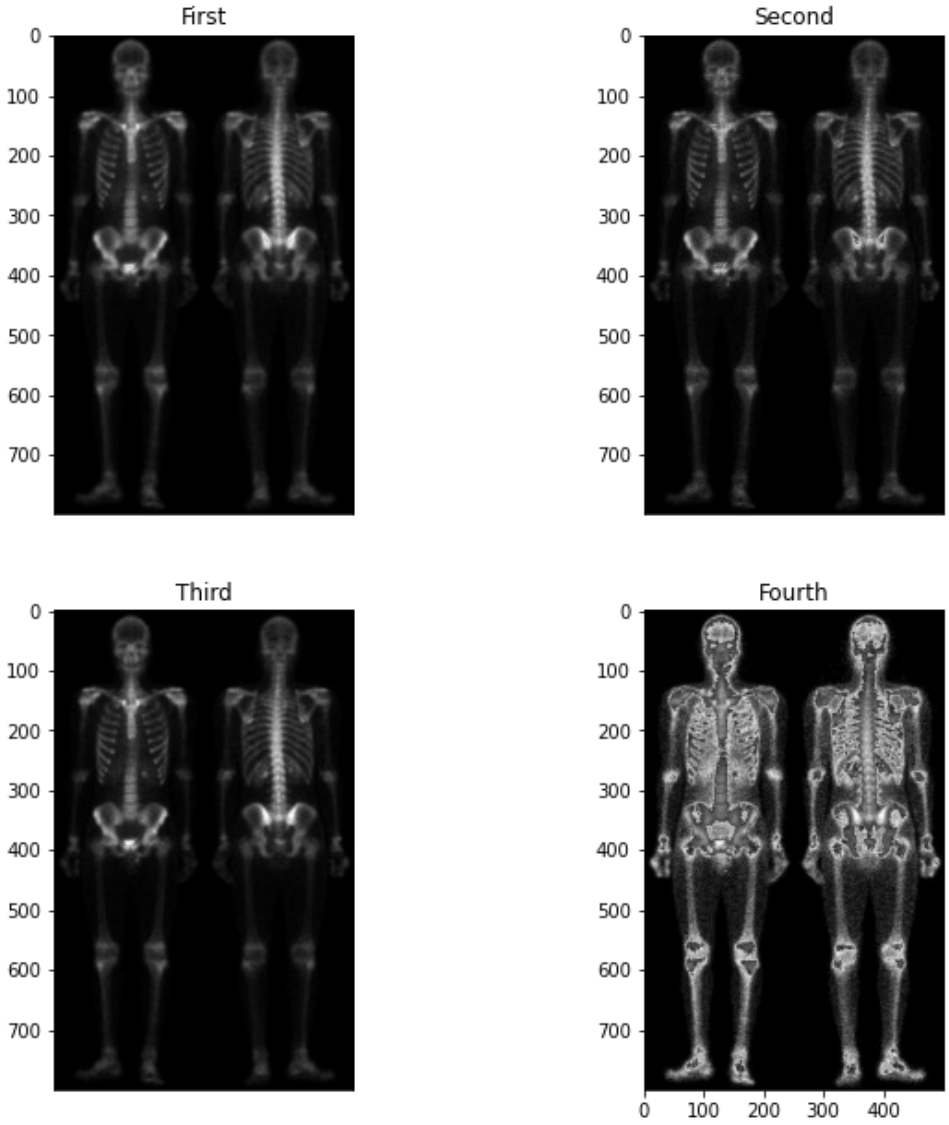
In [7]:

```
#combining result 1 to original image
lap_origional1 = orignal_image + lap_res_1

#combining result 2 to origional image
lap_origional2 = orignal_image + lap_res_2

#combining result 3 to origional image
lap_origional3 = orignal_image + lap_res_3

#combining result 4 to origional image
lap_origional4 = orignal_image + lap_res_4
```

In [8]:

```
fig = plt.figure()
fig.set_size_inches(10.5, 10.5)
ax1 = fig.add_subplot(221)
ax2 = fig.add_subplot(222)
ax3 = fig.add_subplot(223)
ax4 = fig.add_subplot(224)

#display image to each axes
ax1.imshow(lap_origional1)
ax2.imshow(lap_origional2)
ax3.imshow(lap_origional3)
ax4.imshow(lap_origional4)

#set text each subplot && remove x axes
```

```
ax1.title.set_text('First')
ax1.get_xaxis().set_visible(False)
ax2.title.set_text('Second')
ax2.get_xaxis().set_visible(False)
ax3.title.set_text('Third')
ax3.get_xaxis().set_visible(False)
ax4.title.set_text('Fourth')
ax3.get_xaxis().set_visible(False)

#save image to local drive
# plt.savefig('combined_result.jpg')

#display figure
plt.show()
```



In [9]:

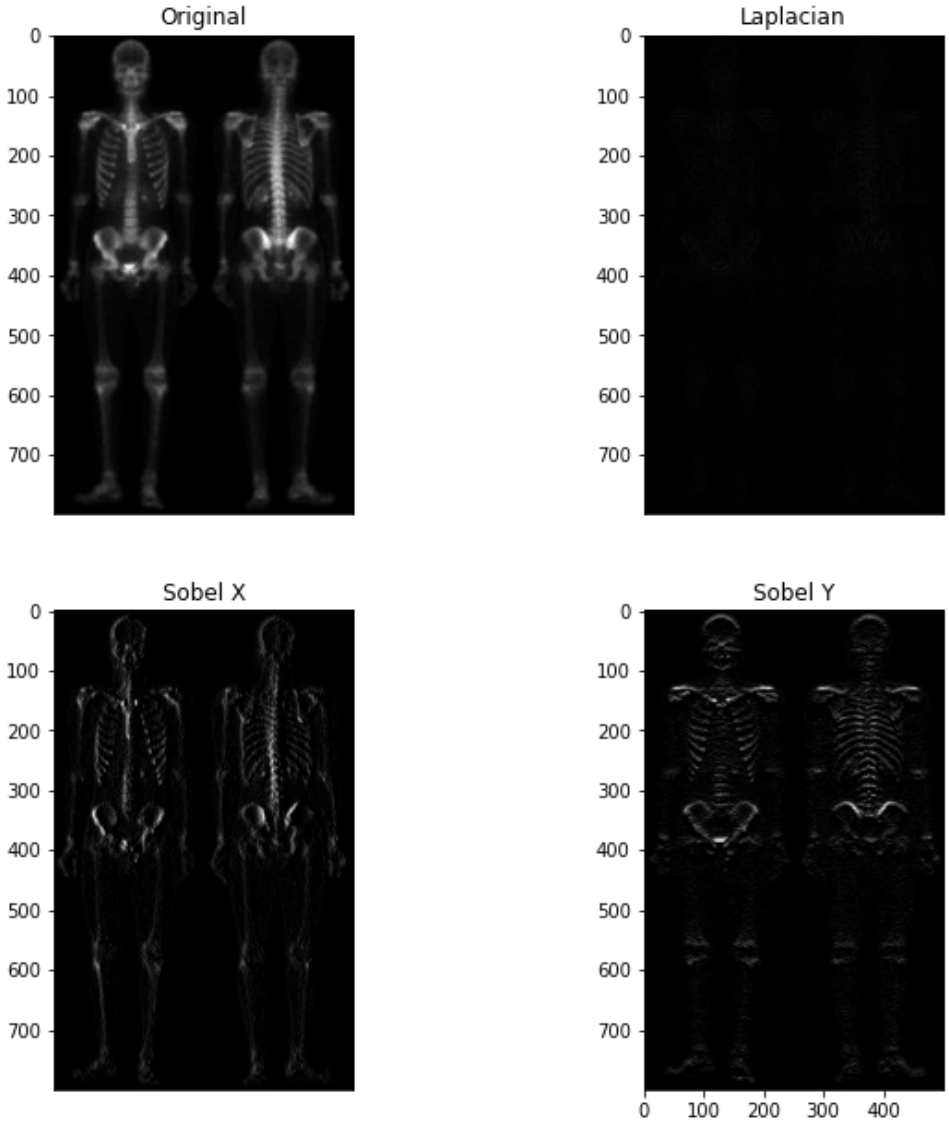
```
#using built in functions in opencv
laplacian = cv.Laplacian(orignal_image,cv.CV_16UC4)
sobelx = cv.Sobel(orignal_image,cv.CV_16U,1,0,ksize=3)
sobely = cv.Sobel(orignal_image,cv.CV_16U,0,1,ksize=3)
```

In [10]:

```
fig = plt.figure()
fig.set_size_inches(10.5, 10.5)
ax1 = fig.add_subplot(221)
ax2 = fig.add_subplot(222)
ax3 = fig.add_subplot(223)
ax4 = fig.add_subplot(224)

#display image to each axes
ax1.imshow(orignal_image)
```

```
ax2.imshow(laplacian)
ax3.imshow(sobelx)
ax4.imshow(sobely)

#set text each subplot && remove x axes
ax1.title.set_text('Original')
ax1.get_xaxis().set_visible(False)
ax2.title.set_text('Laplacian')
ax2.get_xaxis().set_visible(False)
ax3.title.set_text('Sobel X')
ax3.get_xaxis().set_visible(False)
ax4.title.set_text('Sobel Y')
ax3.get_xaxis().set_visible(False)

#save image to local drive
# plt.savefig('combined(lap,sobel).jpg')

#display figure
plt.show()
```

```
Clipping input data to the valid range for imshow with RGB data ([0..1] for floats or [0.
.255] for integers).
Clipping input data to the valid range for imshow with RGB data ([0..1] for floats or [0.
.255] for integers).
```



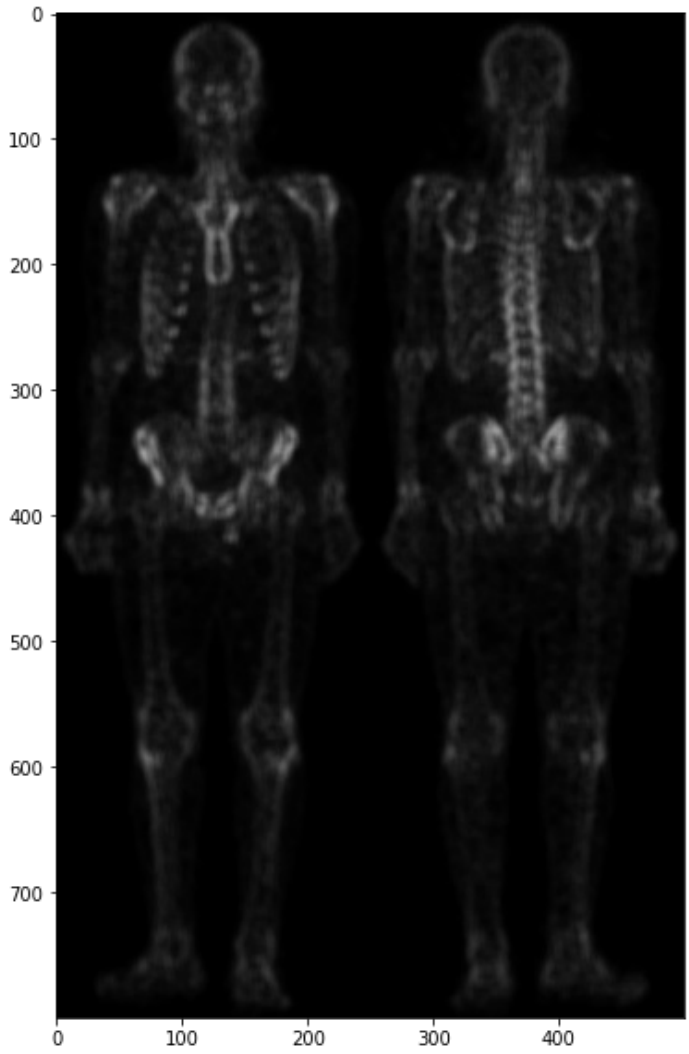
In [11]:

```
sobelx8u = cv.Sobel(orignal_image,cv.CV_8U,2,0,ksize=5)
sobelx64f = cv.Sobel(orignal_image,cv.CV_64F,1,0,ksize=3)
abs_sobel64f = np.absolute(sobelx64f)
sobel_8u = np.uint8(abs_sobel64f)
plt.figure(figsize=(20,10))
blur = cv.blur(sobel_8u,(7,7))
plt.imshow(blur)
```

```
# plt.savefig('sobel.jpg')
```

Out[11]:

```
<matplotlib.image.AxesImage at 0x7f33fdda1280>
```



In [ ]:

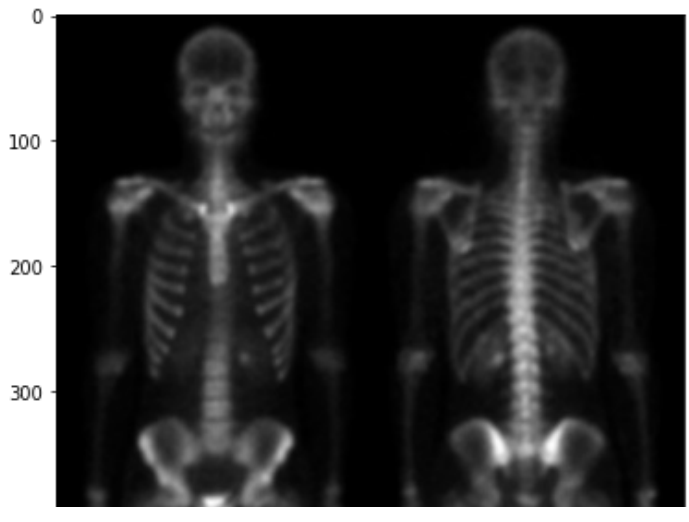
```
```

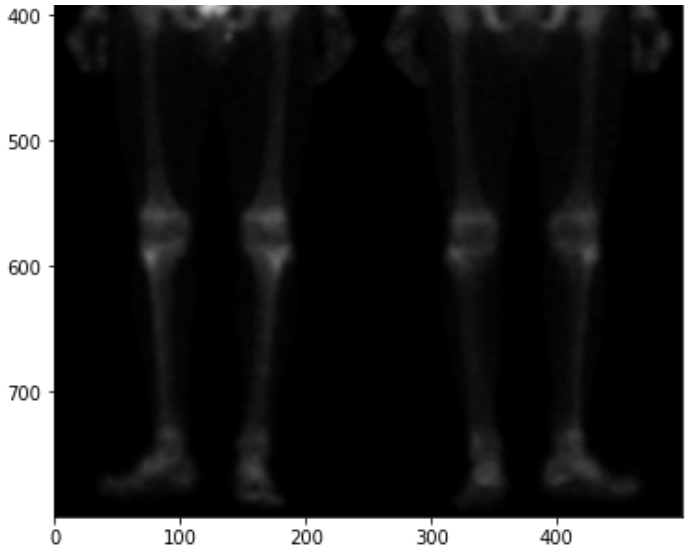
In [12]:

```
blur = cv.blur(orignal_image, (5,5))
plt.figure(figsize=(20,10))
plt.imshow(blur)
# plt.savefig('blur.jpg')
```

Out[12]:

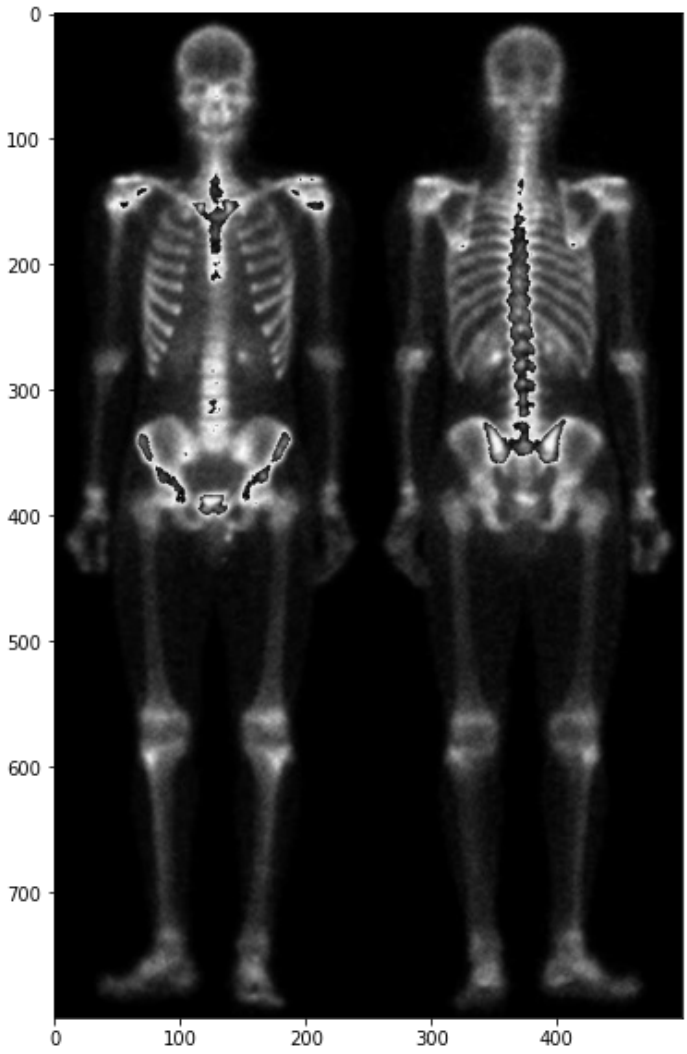
```
<matplotlib.image.AxesImage at 0x7f33fcac1250>
```





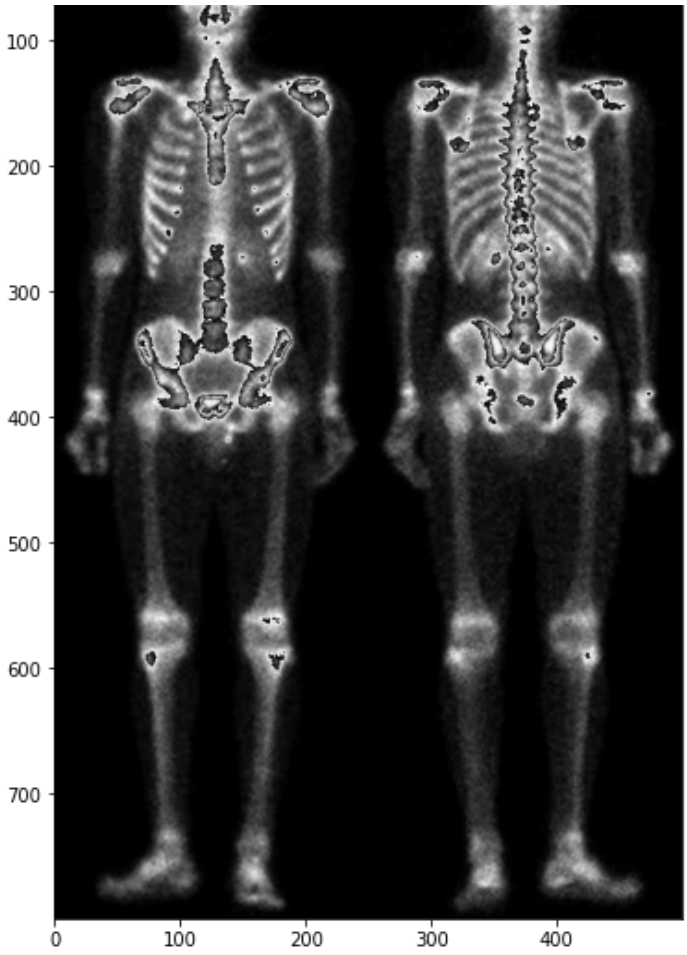
In [13]:

```
combined_lp = blur + lap_origional3
plt.figure(figsize=(20,10))
plt.imshow(combined_lp)
plt.savefig('blur&lap3.jpg')
```



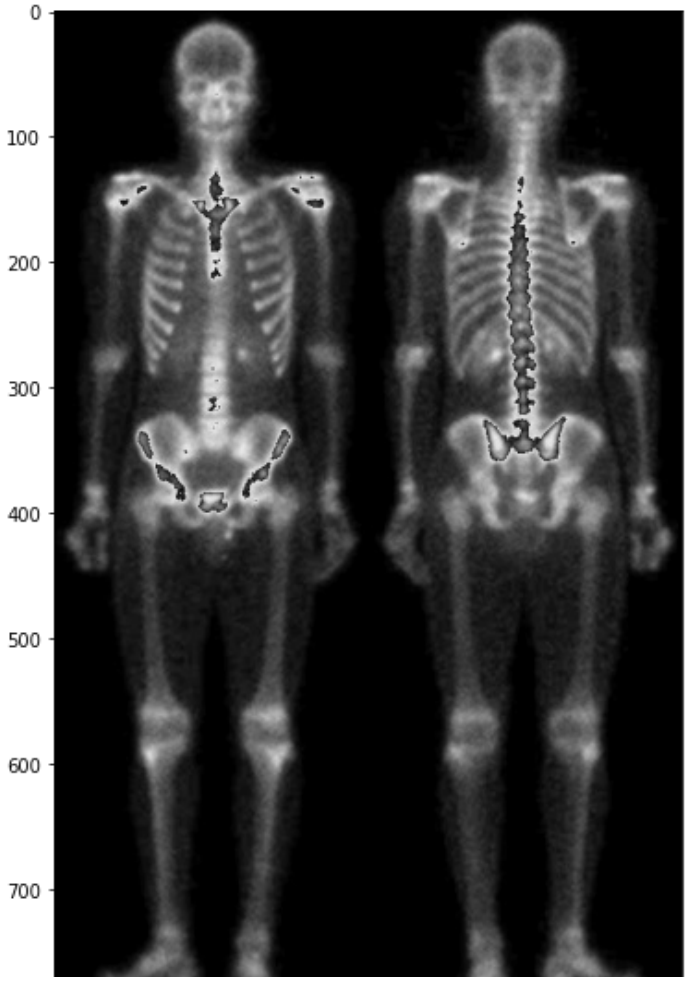
In [14]:

```
averagWeighted = orignal_image + combined_lp
plt.figure(figsize=(20,10))
plt.imshow(averagWeighted)
plt.savefig('blur&lap3.jpg')
```



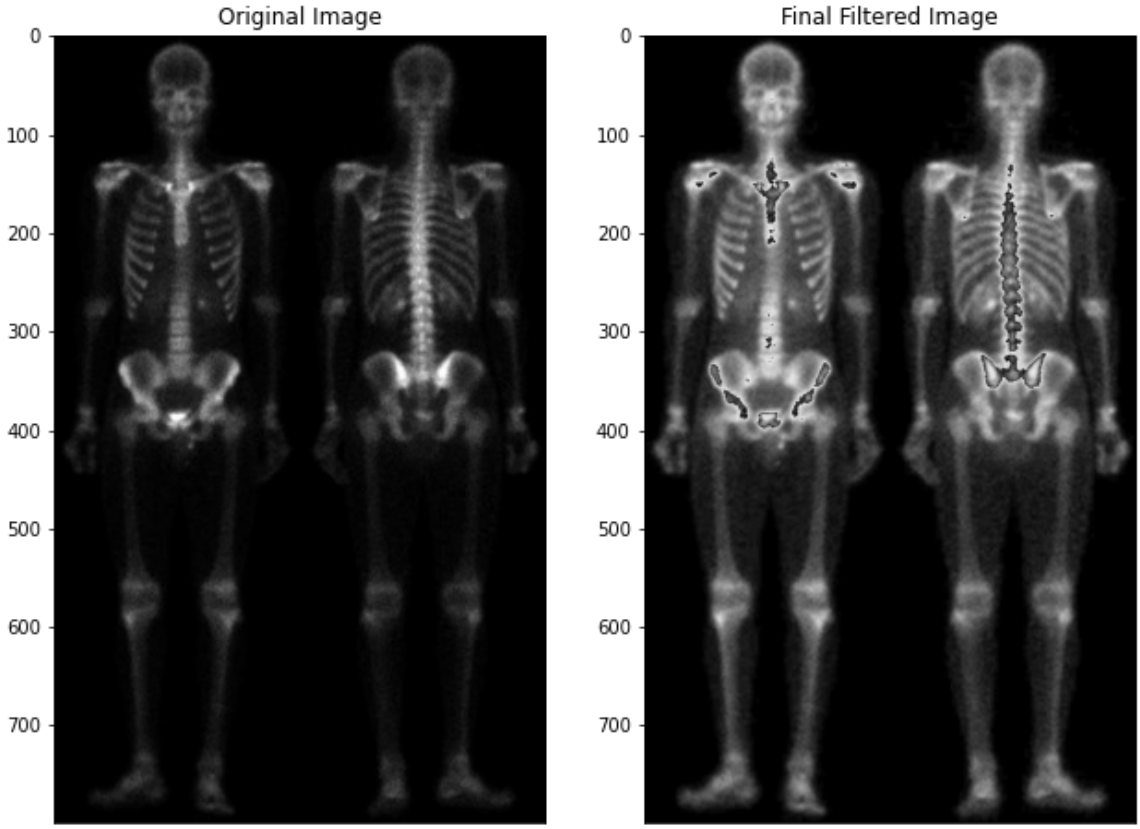
In [15]:

```
# Apply gamma correction.
gamma_corrected = np.array(190*(combined_lp / 190) ** 0.7, dtype = 'uint8')
plt.figure(figsize=(20,10))
plt.imshow(gamma_corrected)
plt.savefig('final.jpg')
```



In [16]:

```
#final Results =>
fig, (ax1, ax2) = plt.subplots(1, 2)
fig.set_size_inches(10.5, 10.5)

#display image to each axes
ax1.imshow(orignal_image)
ax2.imshow(gamma_corrected)


#set text each subplot && remove x axes
ax1.title.set_text('Original Image')
ax1.get_xaxis().set_visible(False)
ax2.title.set_text('Final Filtered Image')
ax2.get_xaxis().set_visible(False)

#save image to local drive
# plt.savefig('combined(lap,sobel).jpg')

#display figure
# plt.savefig('final result.jpg')
plt.show()
```



In [ ]: