Combining Spatial Enhancement Methods

1 Introduction

Prepare to use the Laplace method to highlight the small details of the image, then add them to the original image to increase the details, then use the Sobel gradient to process the original image, and use a 5*5 mean filter for smoothing, and finally use Laplace The processed sharpened image is multiplied by the smoothed image to form a masked image, which is then added to the original image to form a sharpened image, and a power-law transformation is used to obtain the final result.

2 Methods

2.1 Laplacian sharpening

The Laplacian sharpening image is related to the degree of mutation of the surrounding pixels of a certain pixel of the image to this pixel, that is to say, it is based on the degree of change of the image pixels. We know that the first-order differential of a function describes where the image of the function changes, that is, growing or decreasing; while the second-order differential describes the speed of image change, whether it increases sharply or decreases gently. Based on this, we can guess that the transition degree of the pigment of the image can be found according to the second-order differential. For example, the transition from white to black is relatively sharp.

```
img1 = cv2.imread("a.png")
cv2.imshow("original",img1)
```

```
# # 经过拉普拉斯处理
# img2 = cv2.boxFilter(img1,-1,(3,3),normalize=False)
img2 =cv2.Laplacian(img1, -1, ksize=3)
# 拉普拉斯图像和原图相叠加
img3 = img1 + img2
cv2.imshow("ori+lap",img3)
cv2.imshow("laplacian",img2)
```

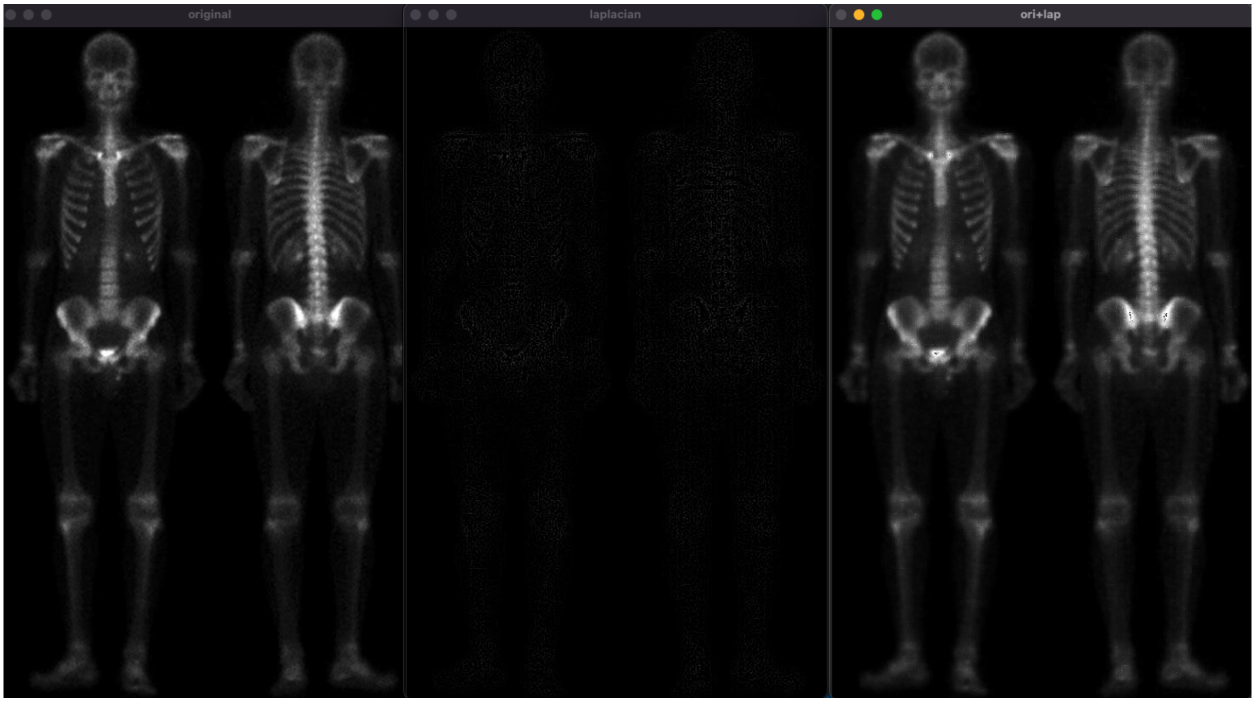

Figure1 → Figure2 → Figure3

the first picture is original picture, the middle is the picture after laplacian sharpening , the last is the addition of the first and the middle .

2.2 Sobel

The Sobel operator is not a real derivative, but a differential approximation of the function, or a local fitting, and the higher the order of the differential approximation, the more accurate the fitting. Therefore, a large kernel can better approximate the derivative, which can eliminate part of the noise effect. However, when the derivative changes sharply, too large a kernel will lead to a large deviation of the results.

```
# sobel 梯度处理
def Sobel():
    img = cv2.imread('a.png', cv2.IMREAD_GRAYSCALE)

    # 使用cv2.Sobel(src, cv2.CV_64F, 1, 0, ksize=3)  对x轴方向进行sobel算子相乘操作
    sobelx = cv2.Sobel(img, cv2.CV_64F, 1, 0, ksize=3)
    # 由于会出现负值的情况，因此使用cv2.convertScalerAbs() 转换为绝对值的形式
    sobelx = cv2.convertScaleAbs(sobelx)


   # 计算y轴方向上的sobel算子
    sobely = cv2.Sobel(img, cv2.CV_64F, 0, 1, ksize=3)
    sobely = cv2.convertScaleAbs(sobely)

   # 使用cv2.addWeighted 将x轴方向的sobel算子的结果和y轴方向上的sobel算子的结果结合
    sobelxy = cv2.addWeighted(sobelx, 0.5, sobely, 0.5, 0)
    cv2.imshow("sobel",sobelxy)
```

Figure 4

2.3 filter processing

Image filtering, that is, suppressing the noise of the target image while preserving the details of the image as much as possible, is an indispensable operation in image preprocessing, and its processing effect will directly affect the effectiveness of subsequent image processing and analysis. reliability.

```
img5 = cv2.blur(sobelxy,(5,5))
cv2.imshow("5*5 filter processing",img5)
```

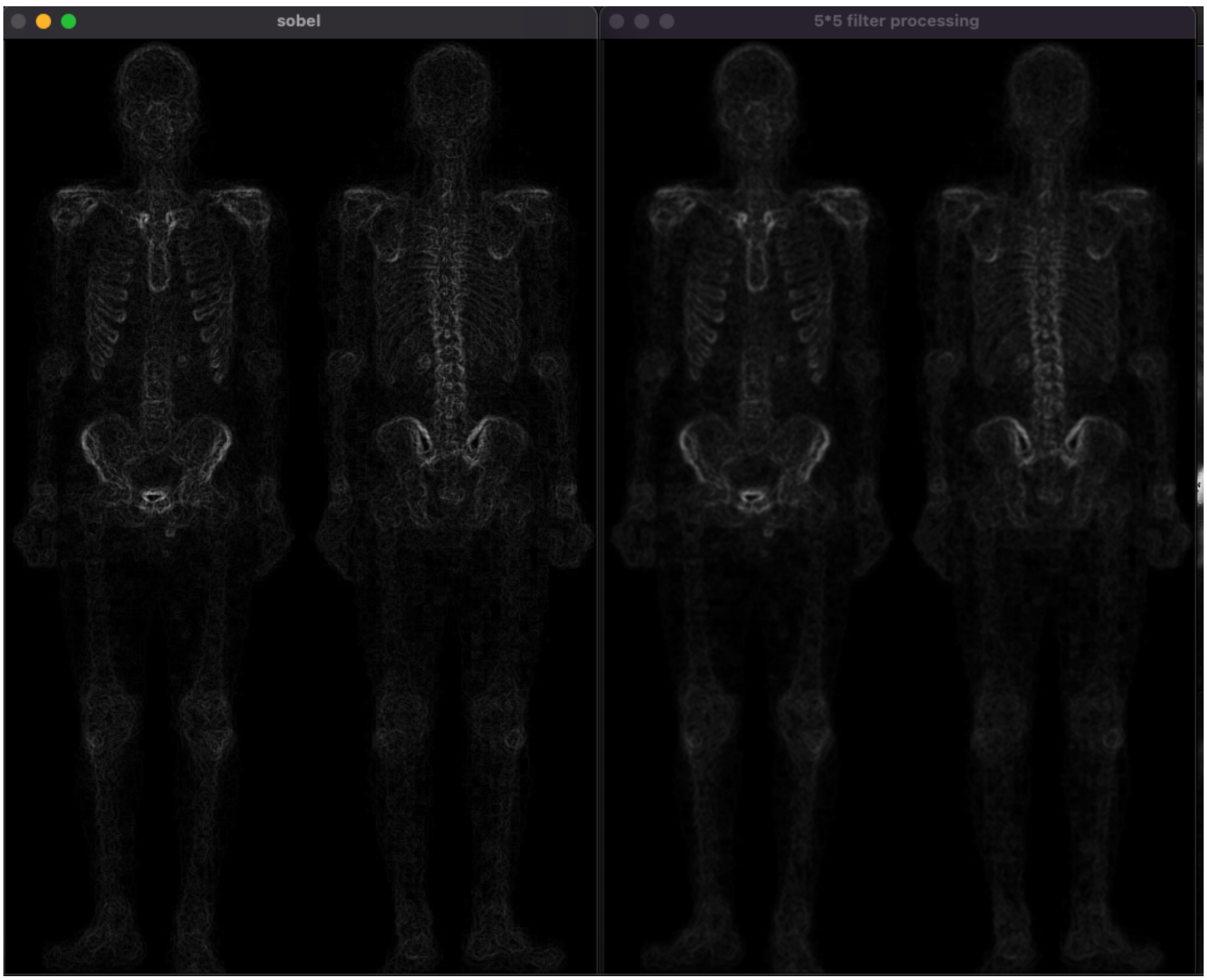

Figure4 Figure5

2.4 Mask Picture

Multiply the sharpened image processed by Laplace and the Sobel image processed by 5*5 filtering to obtain a masked image, which can highlight the details of the image and offset the noise in the image.

```
img6 = cv2.multiply(img3,img5,scale=0.01)
cv2.imshow("mask",img6)
img7 = img1 +img6
cv2.imshow("mask+ori",img7)
```

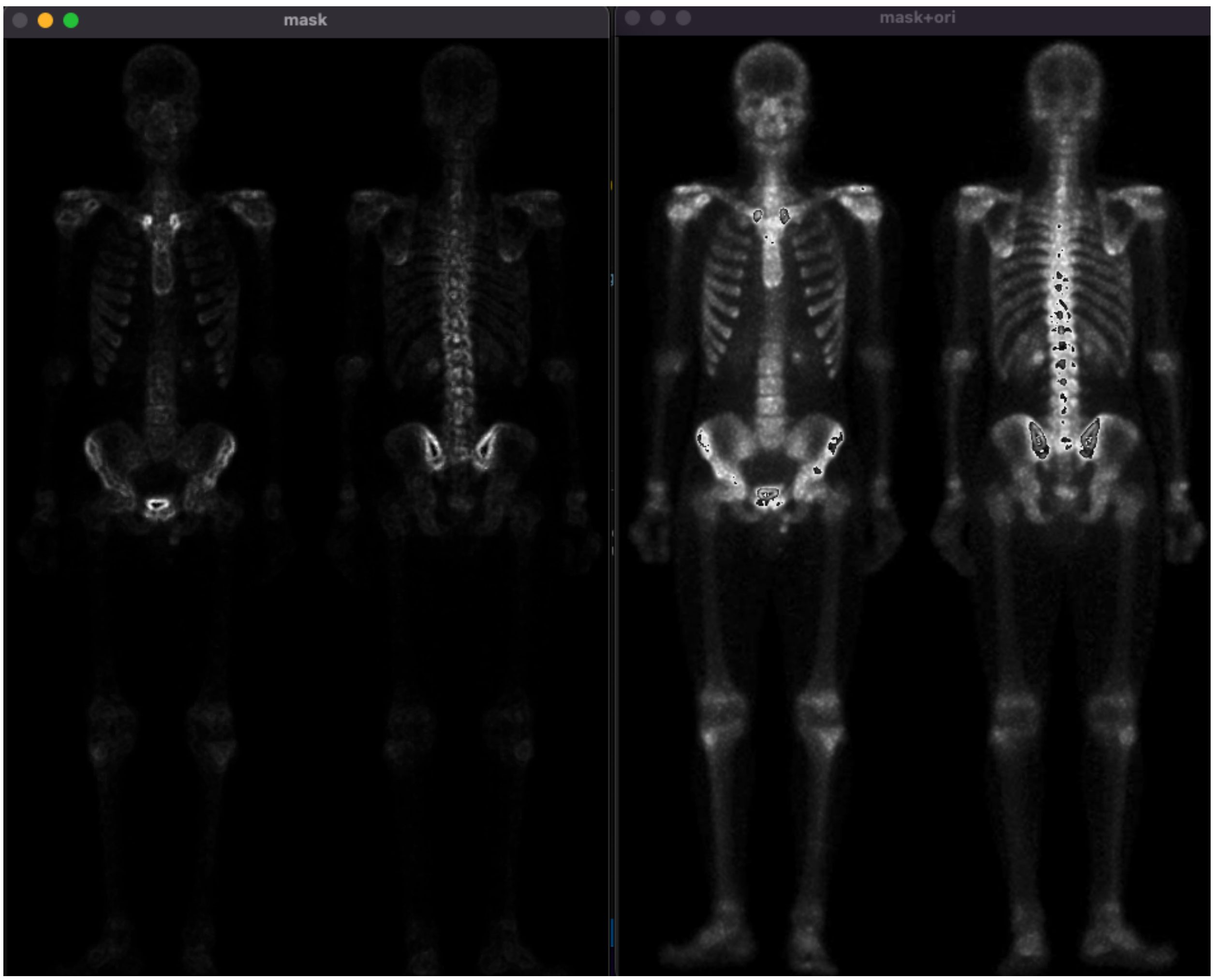

Figure6 → Figure7

2.5 power law transform

Grayscale stretching is also called contrast stretching. It is the most basic grayscale transformation and uses the simplest piecewise linear transformation function. Its main idea is to improve the dynamic range of grayscale during image processing. It can selectively stretch a certain grayscale interval to improve the output image.

Figure7→ Figure 8

```
def adjust_gamma(image, gamma=1.0):
    invgamma = 1/gamma
    brighter_image = np.array(np.power((image/255), invgamma)*255, dtype=np.uint8)
    return brighter_image
img8 = adjust_gamma(img7,gamma=1.5)
cv2.imshow("power law transformation",img8)
```

3 Conclusion

1. This is Figure 2. Use the filter with template [-1,-1,-1;-1,8,-1;-1,-1,-1] to perform Laplace operation on the original image, In order to facilitate the display, the image is calibrated. In this step, a preliminary sharpening filter is performed on the image.

2. This is Figure 3. Since the template used is as above, let the constant c=1, simply add the original image and Figure 2 to get a sharpened image.(At this time, seeing the noise level of Figure 2, there will inevitably be a lot of noise after adding Figure 1 and Figure 2. Laplace operation, as a second-order differential operator, can enhance the details very well, But also produces more noise)

3. This is Figure 4. Try the Sobel gradient operation on the original image, the component gx is [-1,-2,-1;0,0,0;1,2,1], and the component gy is [-1,0 ,1;-2,0,2;-1,0,1] template.

   (The average response of the gradient transformation on gray slopes or steps is stronger than that of the Laplace operation, and the response to noise and small details is weaker than that of the Laplace operation, and can be adjusted by the mean filter. Smoothing can be further reduced, and the edges in the image are much more prominent than those in the Laplacian image (ie Figure 3)

4. This is Figure 5. A mean filter with a size of 5*5 is used to obtain the smoothed Sobel gradient image.

   (Figures 4 and 5 are brighter than Figure 2 indicating that gradient images with significant edge content generally have higher values than Laplacian images)

5. This is Figure 6, where the Laplacian image (ie, Figure 3) is dot-multiplied with the smoothed gradient image (ie, Figure 5).

   (At this time, seeing the advantages of strong edges and the relative reduction of visible noise, the purpose of masking the Laplacian image with the smoothed gradient image is achieved)

6. This is Figure 7. Adding the product image (ie, Figure 6) to the original image produces a desired sharpened image.

(Compared with the original image, the sharpness of most of the details in the image has increased significantly, so we need to combine a variety of methods to process the image. It is impossible to achieve such a good effect by using one method alone. Just look at the corresponding images for comparison)

7. This is Figure 8. We hope to expand the grayscale range and perform power-law transformation on Figure 7, r=0.5, c=1, and then power-law transformation can be performed on the image

   (At this time, the dynamic range of the sharpened image needs to be increased. Even if there are many grayscale transformation functions with this effect, it is better to use power-law transformation, and the effect of histogram equalization and specification is not very good)

   (At this time, although the definition of the outline of the human body is still not high, because the enlarged gray dynamic range also increases the noise, but it is still greatly improved compared to the original image. See the original image and the final image below. image comparison)