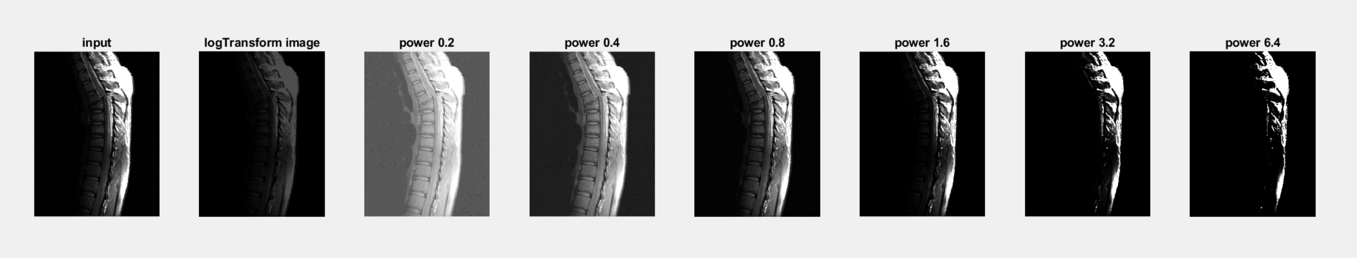
ImageProcess Lab 2 作業

by 108062138 Po-Yu,Wu

Proj3.1: Image Enhancement Using Intensity Transformations

- (1) 放上原圖 & log transformation 的結果 & power-law transformation 至少 5 種不同 r 的結果

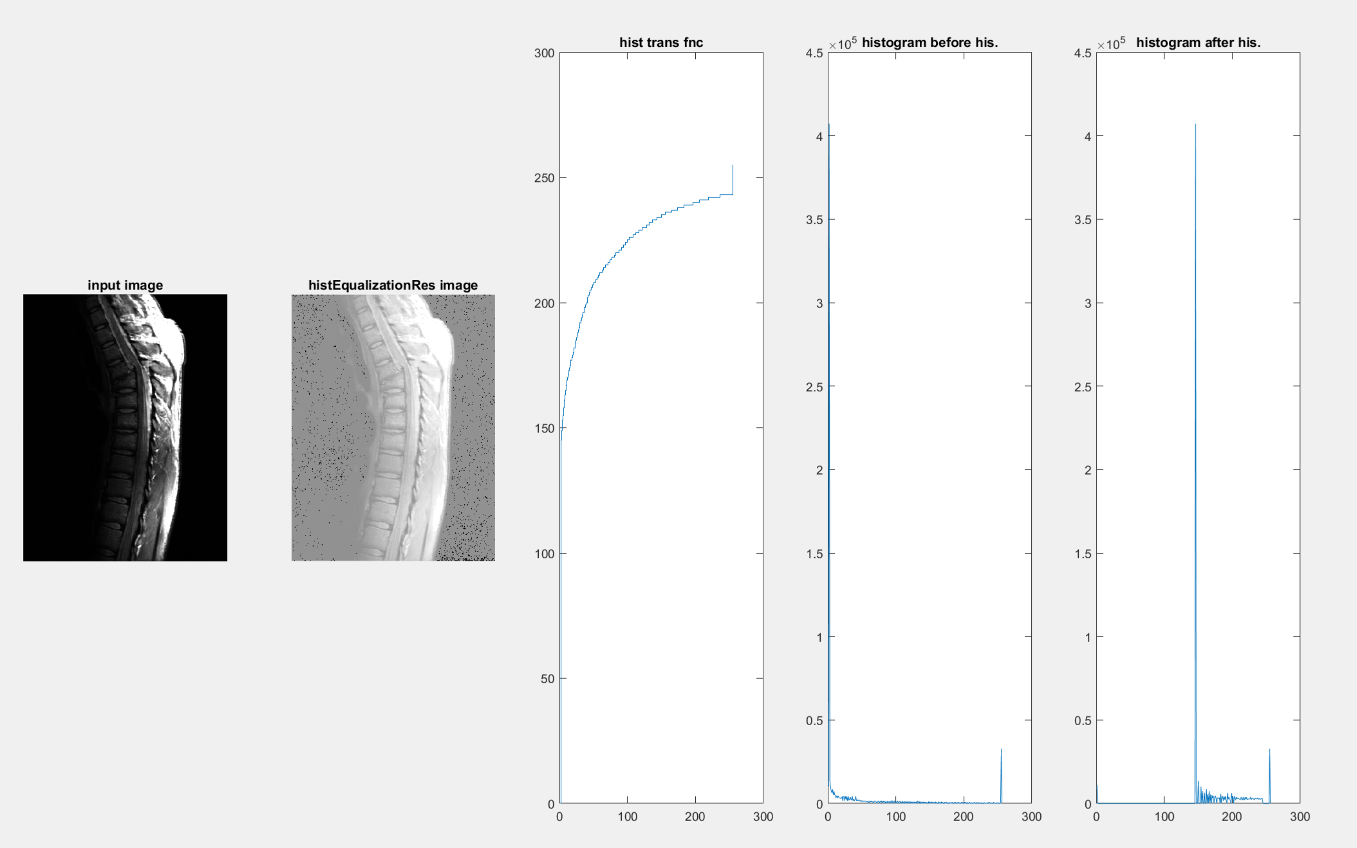


Images from left to right are basic input/log transform/r=0.2/0.4/0.8/1.6/3.2/6.4’s corresponding result.

- (2) 比較分析其不同
In the log transformation, the difference between high intensity and low intensity will be leveled down by the log function. It makes the bright area becomes not that bright while the dark area remains practically the same. Thus, the image will become not that bright, compared to the original image.
As for r=0.2/0.4/0.8/1.6/3.2/6.4, I find that the image’s light and shade contrast becomes larger as long as the growth of power, denoted as r. It seems that way-high power generate more loss information on the produced picture. Also, low power will blur the produced image. Hence, the power should be sophisticatedly selected to prevent the above situation.

Proj3.2: Histogram Equalizatio

- (1) 放上原圖 & 原圖的 histogram & histogram equalization transformation function(c.f. Fig. 3.22©) & enhance 後的圖 & enhance 後的圖的 histogram (共 5 張圖)

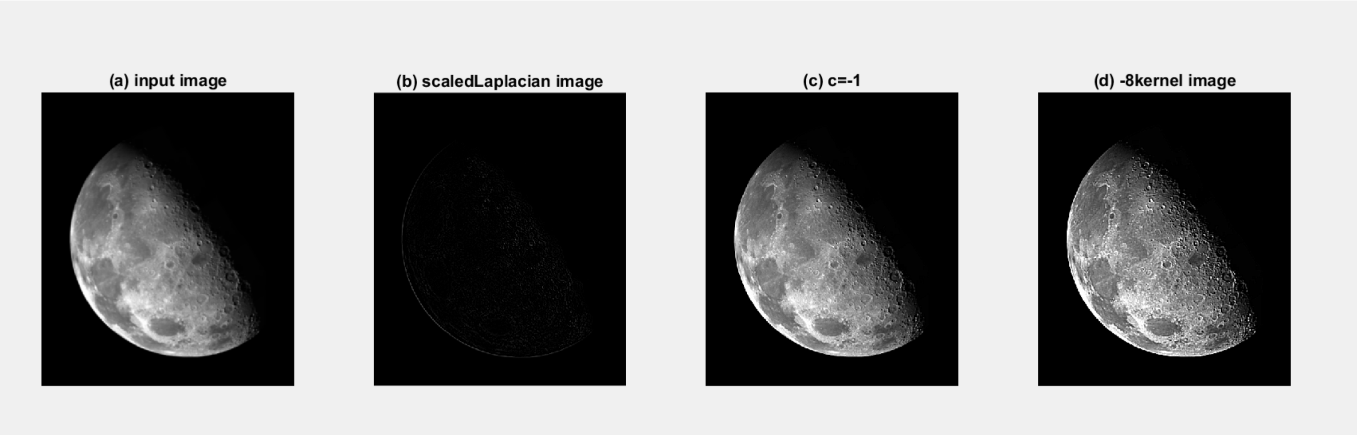


Images from left to right are input/histEqualization/ histogram equalization transformation function/before enhanced’s histogram/ after enhanced histogram.

- (2) 討論分析處理前後的結果
  As you can see, the original image has sheer black background while the histo. picture obtains grey, even white background. The reason comes from the fact that the histo. process tend to bend the intensity distribution into “uniform distribution”. Since the original distribution heavily resides on intensity 0. After histo. process, the overall distribution must be moved to right. That is to make the overall image brighter. The histo. process will make overall-bright image become darker while and the overall-dark image become brighter.

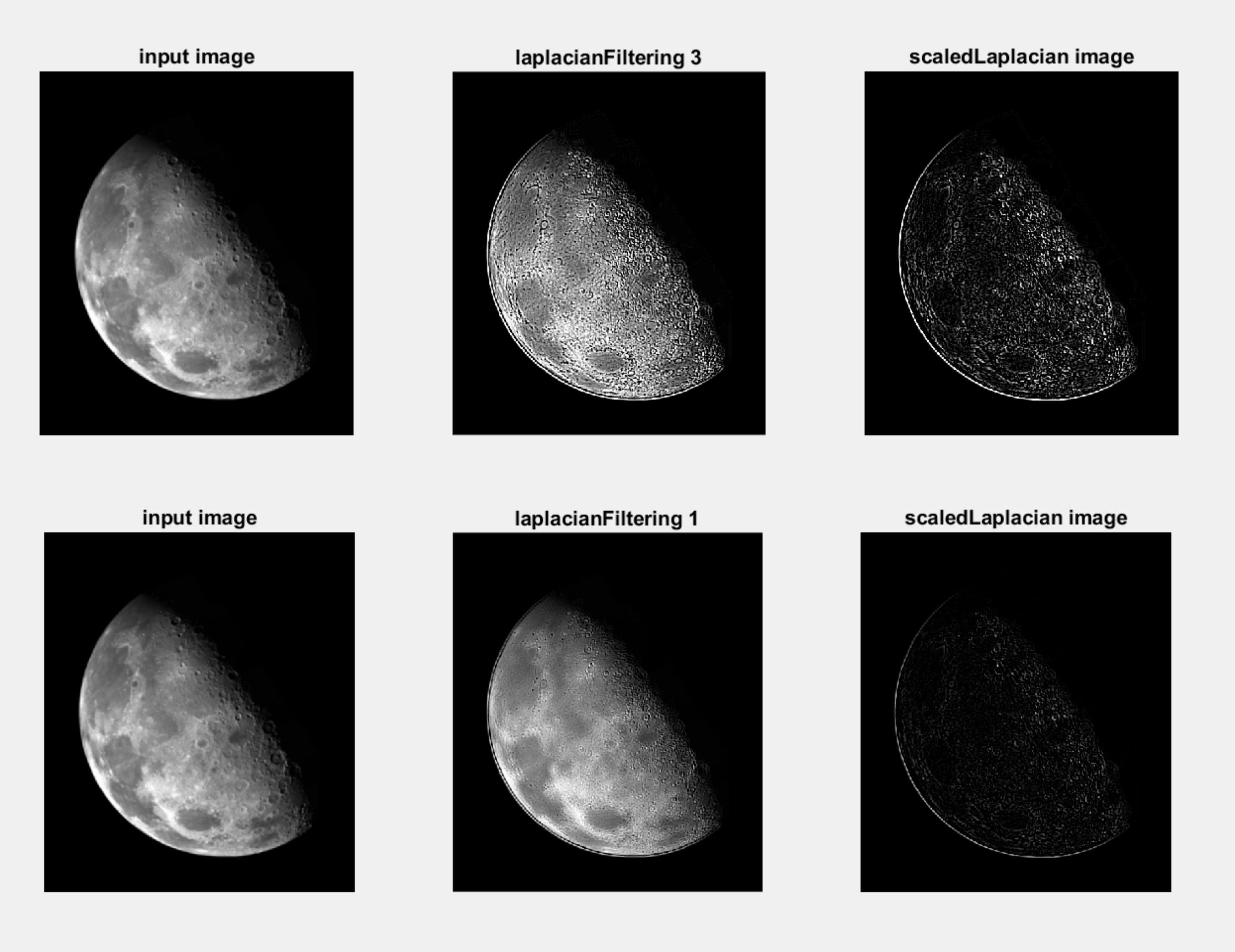
Proj3.3 ~ 3.4: Spatial Filtering, Enhancement Using the Laplacian

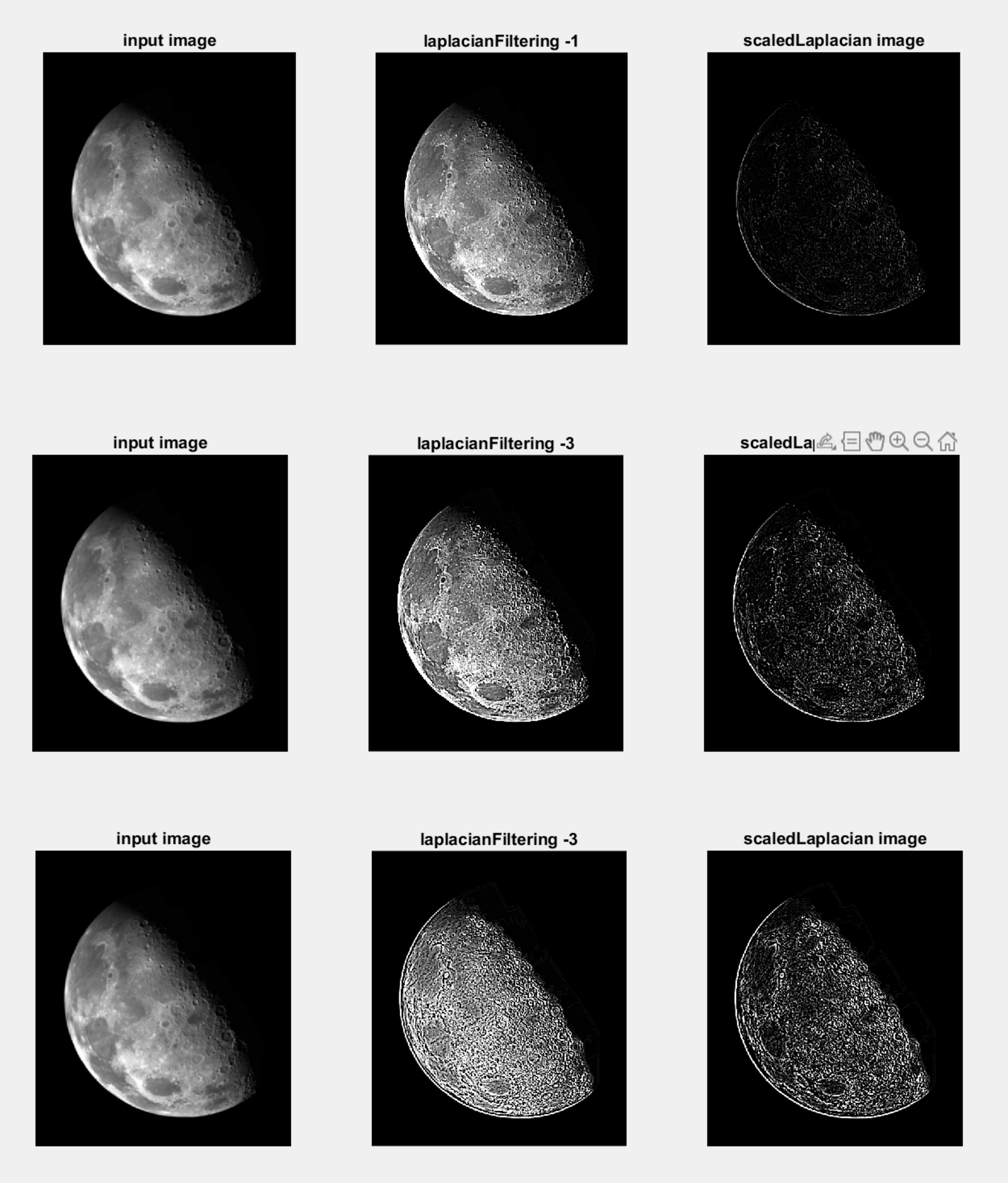
- (1) 用 laplacian 重複 4/e Fig. 3.46 (a,b,c,d) 的結果 (四張圖)



Images from left to right are input/scaledLaplacian/c=-1 with filter a/c=-1 with filter b.

- (2) 分析不同 scale、不同 mask size 的差異 (可以加上多張圖)

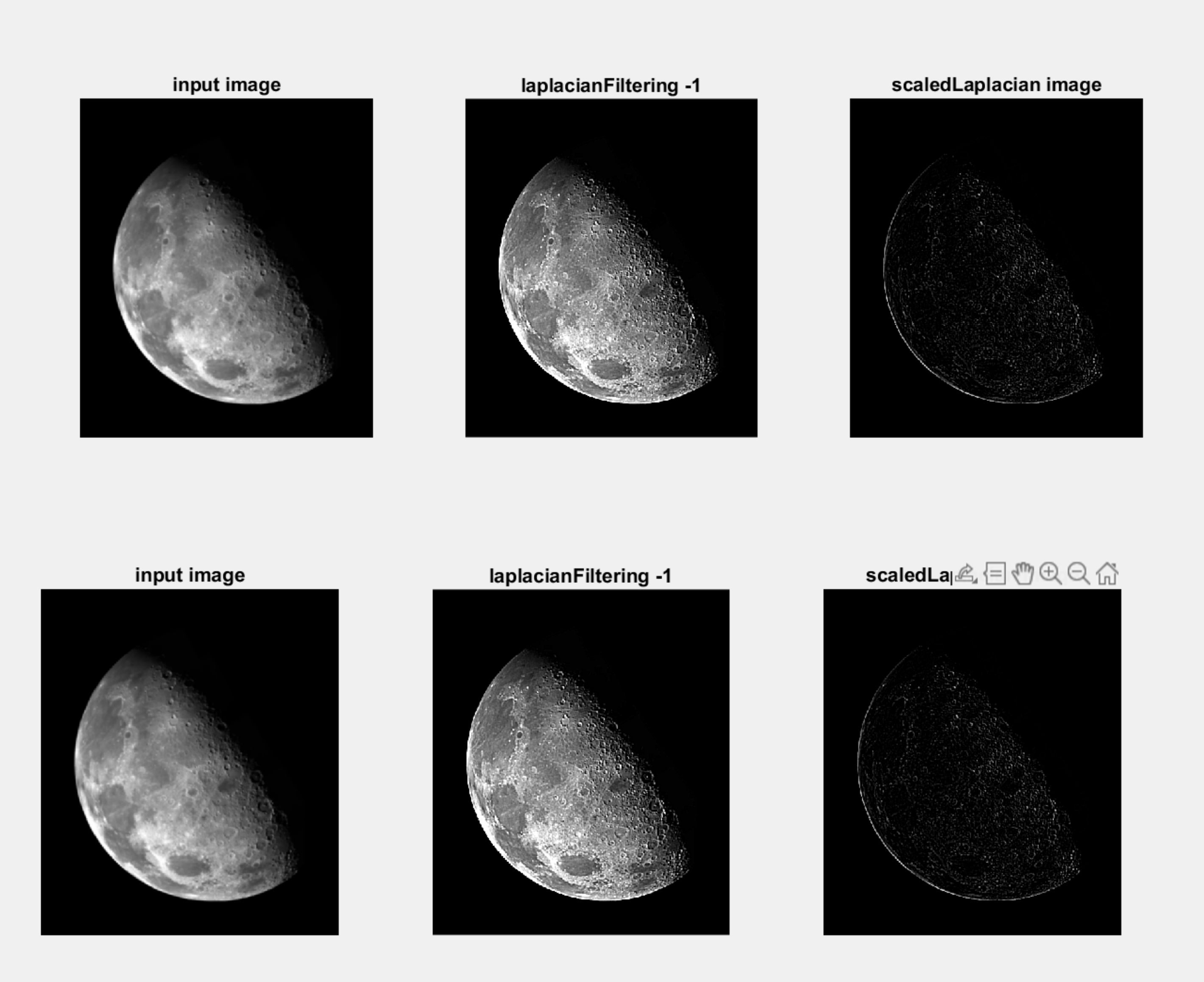



input image
laplacianFiltering -1
scaledLaplacian image
input image
laplacianFiltering -3
scaledLa
input image
laplacianFiltering -3
scaledLaplacian image

input image	laplacianFilter	scaledLaplacian	scale	filter type
Fig0338(a)(blurry_moon).tif	-	-	3	[1 1 1;1 -8 1;1 1 1];
Fig0338(a)(blurry_moon).tif	-	-	1	[1 1 1;1 -8 1;1 1 1];
Fig0338(a)(blurry_moon).tif	-	-	-1	[1 1 1;1 -8 1;1 1 1];
Fig0338(a)(blurry_moon).tif	-	-	-3	[1 1 1;1 -8 1;1 1 1];
Fig0338(a)(blurry_moon).tif	-	-	-3	[0 0 -1 0 0;0 -1 -2 -1 0;-1 -2 16 -2 -1;0 -1 -2 -1 0;0 0 -1 0 0];

From row 0-3, the filter type holds and the scale varies from 3/1/-1/3. As you can see, the image becomes sharper along as the rise of reduction on the scale. And on row 4-5 can show that under 5x5 the texture becomes more rough than 3x3 under the condition that the scale remains the same.

Ps: although TA states that the spatialFilter can use different methods to handle boundry, the moon picture has 4-sided dark background s.t. it is hard to show the effect on the boundry.



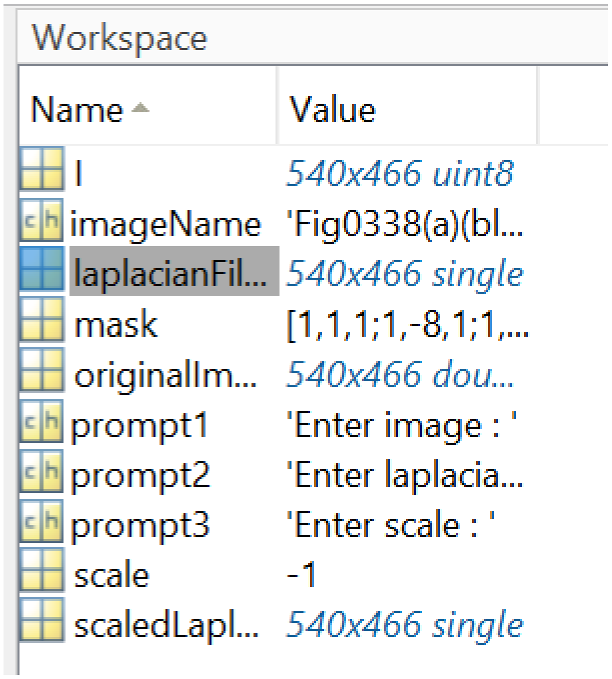
- original boundry method: boundry for 0

```
for covi = nhi:halfmI
    for covj = nhj:halfmJ
        val = 0;
        if i+covi >=1 && i+covi <=numRows && j+covj >=1 && j+covj <=numCols
                        %disp(covi+centermI+covj+centermJ);
                        val = A(i+covi,j+covj)*mask(covi+centermI,covj+centermJ);
        end
        res(i,j) = res(i,j) + val;
        end
end
```

- original boundry method: boundry for 1

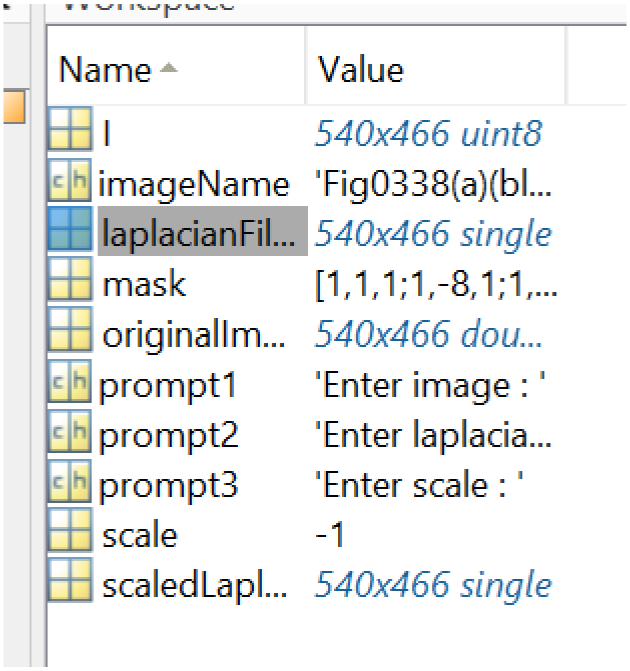
```
for covi = nhi:halfmI
    for covj = nhj:halfmJ
        val = 0;
        if i+covi >=1 && i+covi <=numRows && j+covj >=1 && j+covj <=numCols
                        %disp(covi+centermI+covj+centermJ);
                        val = A(i+covi,j+covj)*mask(covi+centermI,covj+centermJ);
        end
        res(i,j) = res(i,j) + val;
        end
end
```

The main difference lies in that the over-boundry element handling method. The size of the second-method picture seems smaller than the first-method picture. What hides behind the phonomenon is that the second boundry methon directly set the value as 1, which makes the color bright. Thus, the second seems to be smaller than the first picture.



Workspace

Name ▲	Value
I	540x466 uint8
imageName	'Fig0338(a)(bl...
laplacianFil...	540x466 single
mask	[1,1,1;1,-8,1;1,...
originalIm...	540x466 dou...
prompt1	'Enter image : '
prompt2	'Enter laplacia...
prompt3	'Enter scale : '
scale	-1
scaledLapl...	540x466 single



Name ▲	Value
I	540x466 uint8
imageName	'Fig0338(a)(bl...
laplacianFil...	540x466 single
mask	[1,1,1;1,-8,1;1,...
originalIm...	540x466 dou...
prompt1	'Enter image : '
prompt2	'Enter laplacia...
prompt3	'Enter scale : '
scale	-1
scaledLapl...	540x466 single