HW2: Histogram and Spatial Filtering

1.1 Histogram Equalization

Suppose that you have performed histogram equalization on a digital image. Will a second pass of histogram equalization (on the histogram-equalized image) produce exactly the same result as the first pass? Prove your answer.

Yes, they are the same, let's see an example,

The origin image:

r	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**
n	**5**	**2**	**6**	**0**	**0**	**2**	**0**	**1**
p	**5/16**	**1/8**	**3/8**	**0**	**0**	**1/8**	**0**	**1/16**

Image after one histogram equalization:

r	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**
P(r)	**5/16**	**7/16**	**13/16**	**13/16**	**13/16**	**15/16**	**15/16**	1
r(result)	**35/16**	**49/16**	**91/16**	**91/16**	**91/16**	**105/16**	**105/16**	7
r	**2**	**3**	**6**	**6**	**6**	**7**	**7**	7

r	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**
P(r)	**0**	**0**	**5**	2	**0**	**0**	6	3

(a)

Image after a second histogram equalization:

r	0	1	2	3	4	5	6	7
P(r)	0	0	5/16	7/16	7/16	7/16	13/16	1
r(result)	0	0	35/16	49/16	49/16	49/16	91/16	7
r	0	0	2	3	3	3	6	7

r	0	1	2	3	4	5	6	7
P(r)	0	0	5	2	0	0	6	3

(b)

(a) is the table after one histogram equalization and (b) is the table after a second histogram equalization, they are the same, also, we can get this conclusion from mathematical aspect, the histogram equalization in discrete value is the $\sum$ of the previous value, thus after one $\sum$ the value has already been distributed equally, another $\sum$ will not change the distributed property of it, thus they are the same.

1.2 Spatial Filtering

Consider a 4*4 gray image and a 3*3 filter:

1. Convolve the gray image with the given filter with zero-padding, and show your result(whose size should be 4*4).

2. Discuss the meanings of positive values and negative values in your convolution result respectively.

3. Describe some applications of the given filter based on your own knowledge.

Sol:

1、We can use the thought of zero-padding to expand the image into 6*6 matrix(since we only care about the center 4*4 area), for each point, we grasp the adjacent 9 point and multiply the 9 point with filter to get the output for the point, and we apply this to each point to get the result:

177	420	279	271
74	72	90	-52
-61	-131	-2	-19
-172	-199	-215	-60

2、The filter is used to detect vertical edge, it use the difference between gray-level value, The positive value means that the lower line 's pixel is lighter than the upper line and the negative value means that the lower line 's pixel is darker than the upper line, the abstract of result value, if it is larger than a threshold value T, then it is likely to be an edge.

3、The filter is called Prewitt operator, it is similar to the sharping spatial filter, and it is the first order differential between the upper line the the lower line, thus it can be used to enhance edge and reduce noise.

2 Programming Tasks

2.2 Histogram Equalization

Write a function that applies histogram equalization on a gray scale image. The function prototype is "\equalize hist(input img) ! output img", returning a gray scale image whose histogram is approximately at. For the report, please load your input image and use your program to:

1. Compute and display its histogram. Manually paste the histogram on your report. Note:You must compute the histogram by yourself, but existing APIs can be used for display.

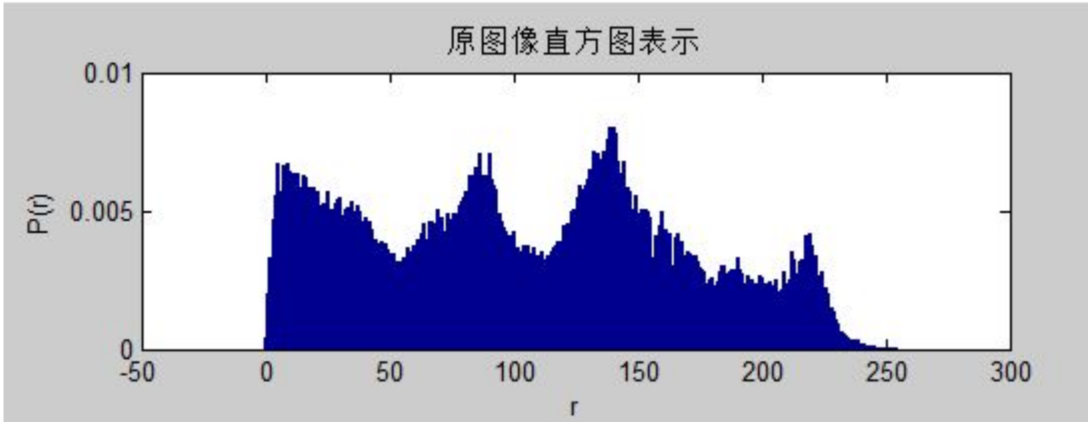
2. Equalize its histogram. Paste the histogram-equalized result and the corresponding histogram on your report.

3. Analyze your histogram-equalized result in less than 1 page.

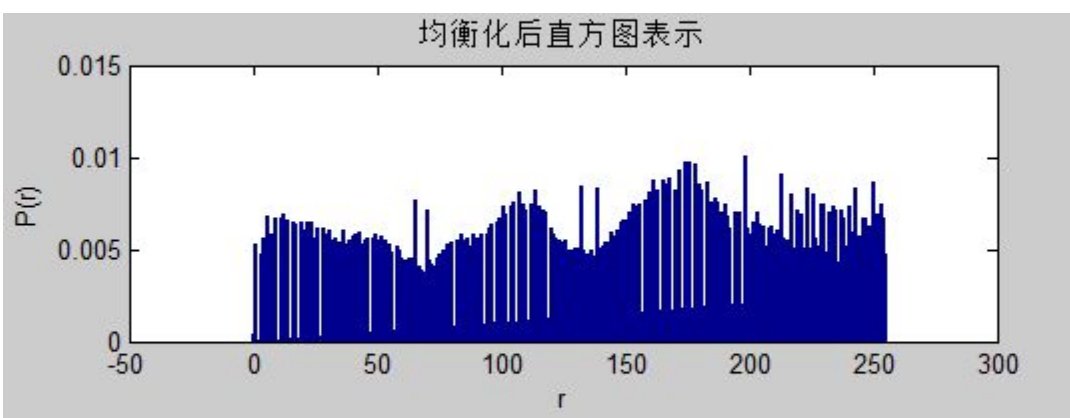
4. Detailedly discuss how you implement the histogram equalization operation,, in less than 2 pages. Please focus on the algorithm part.

Sol:

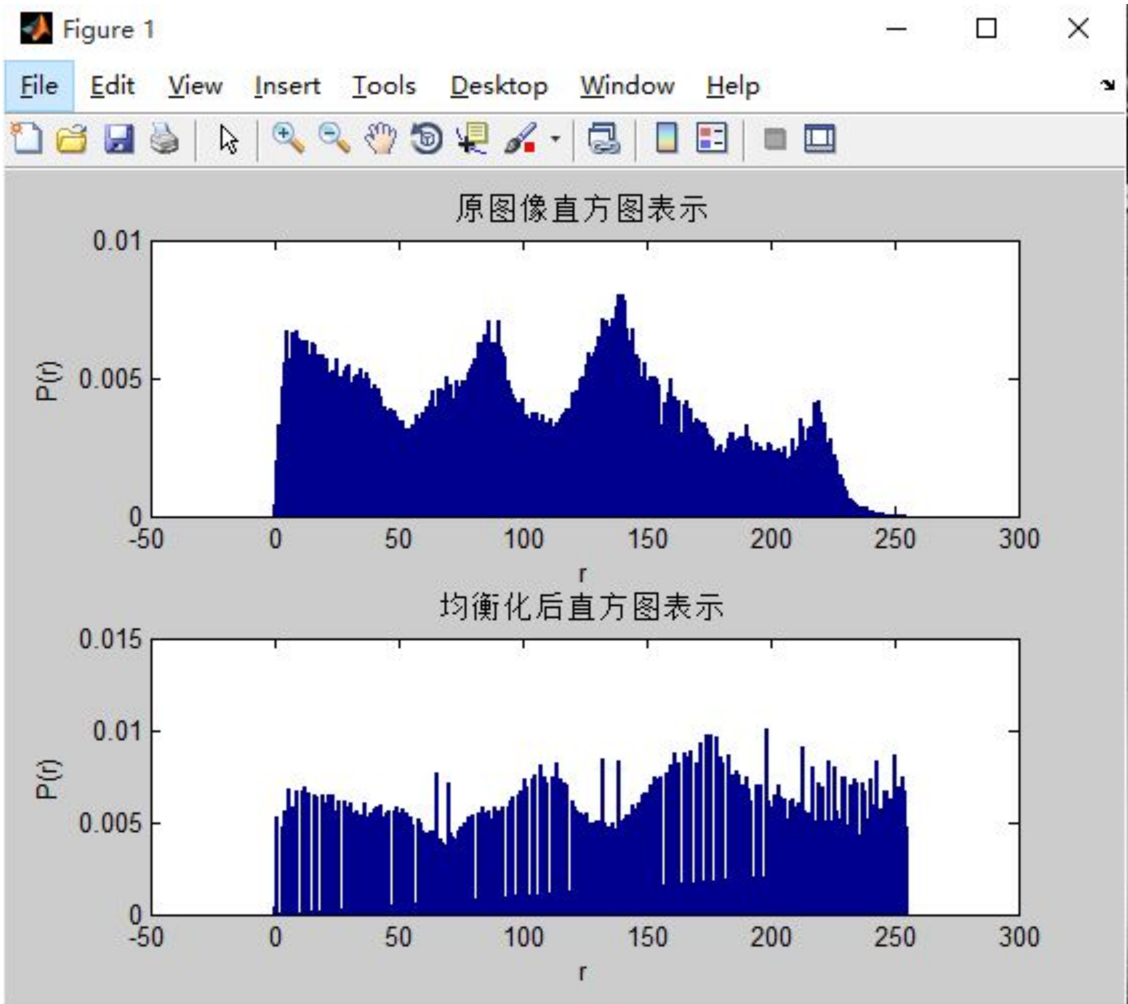
1、The histogram is shown as below, you can use 'for' and 'imread' to do this job in matlab.



2、The figure is shown below:



And we can use subplot to compare the origin histogram and equalized histogram together to compare with each other:



3、As you can see above, the equalized histogram distribute the probability normally, although the probability is not equal at each gray-level, but most of them look just the same height, in the origin histogram, the area between 225 and 250 has small p(r), after equalization, the probability in that area has the same height as other area, there is some maximum point and minimum point in the equalized histogram, because in the origin histogram, the minimum point has very small p(r), thus after $\sum$, the point has merged into other point, in other words, the point(representing one gray-level) doesn't exist in new matrix. Also, the maximum point has merged other points' p(r) into itself, thus the resulting p(r) can be very high.

4、First, we need to visit the whole matrix to calculate the number of each gray-level, using a array whose length is 256 to store the number. Then we divide them by the size of matrix(m*n) to get the p(r).

Second, for discrete point, we apply the $\sum$ function to each point to accumulate the p(r), then we multiply each point by 255 to get the new gray-level.

We use the array as a transformation to map the origin matrix(image) into a new image, for example, if array[128] = 140, which means the gray-level of 128 in origin matrix needs to change into 140, in this way, we apply this algorithm to get a new matrix and output it, and the equalization has been done.

2.3 Spatial Filtering

Write a function that performs spatial filtering on a gray scale image. The function prototype is filter2d(input img, filter) ! output img", where filter" is the given filter. Modify the prototype if necessary.

For the report, please load your input image and use your filter2d" function to:

1. Smooth your input image with 3 3, 7 7 and 11 11 averaging filters respectively. Paste your three results on the report.
2. Sharpen your input image with a 33 Laplacian filter and then paste the result. In addition, discuss why Laplacian filter can be used for sharpening.
3. Perform high-boost filtering on your input image. The averaging part of the process should be done using the filter in Fig. 3.32(a) of the textbook. Choose a k as you see fit. Write down the selected k and paste your result on the report.
4. Detailedly discuss how you implement the spatial filtering operation, i.e., the "filter2d" function, in less than 2 pages.

Sol:

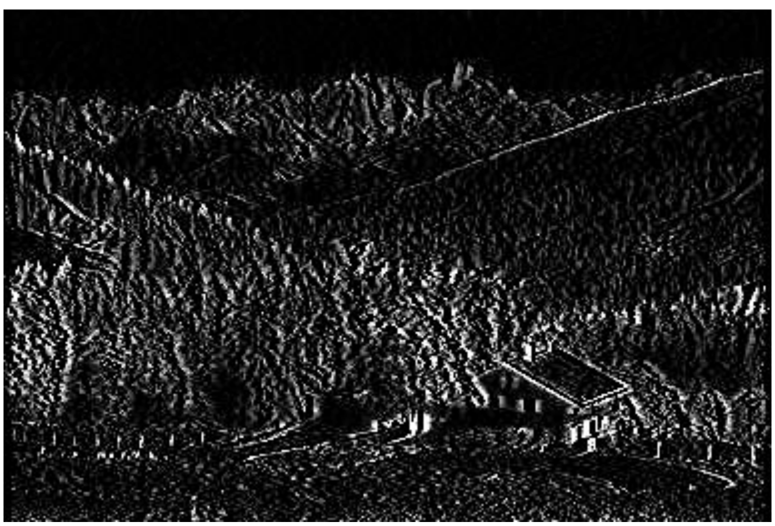
1、The figure is shown as follow:

3*3 averaging filter

7*7 averaging filter

11*11 averaging filter

2、The result image is shown as below:



Laplacian filter

The principal of sharpen is enhancing details, and the second order of derivative is the details for image. The Laplace transformation for f(x,y) is

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

and we can apply partial derivative on this equation to get

$$\frac{\partial^2 f}{\partial^2 x^2} = f(x+1, y) + f(x-1, y) - 2f(x, y)$$

$$\frac{\partial^2 f}{\partial^2 y^2} = f(x, y+1) + f(x, y-1) - 2f(x, y)$$

And we can add them to get the final Laplace operator

$$\nabla^2 f = [f(x+1, y) + f(x-1, y) + f(x, y+1) + f(x, y-1)] - 4f(x, y)$$

Since Laplace is linear transformation, and in matrix form, it can be expressed as:

0	−1	0
−1	4	−1
0	−1	0

Thus is can be used to sharpen image.

3、K = 3， the result is shown as below:

4、The algorithm is based on the convolution in book, we first expand the origin matrix(m*n) into (m+filter.m-1, n+filter.n-1), the center of the expand-matrix is the same as origin, but the edge is padding with 0. Then we visit the central m*n point, for each point, we choose the adjacent points(the number is filter.m*filter.n), convolve it with filter, sum the result up to get a value for that point, after getting the value for each point, we trimmed to get the central m*n matrix. We have already done spiral filtering.