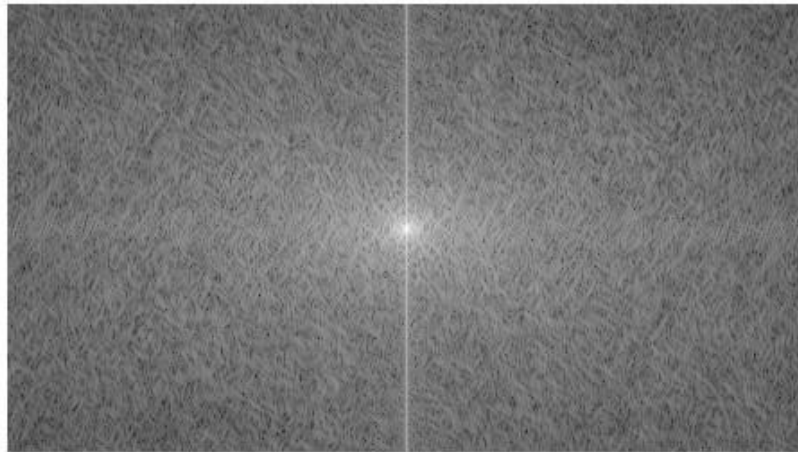


## 1.程序

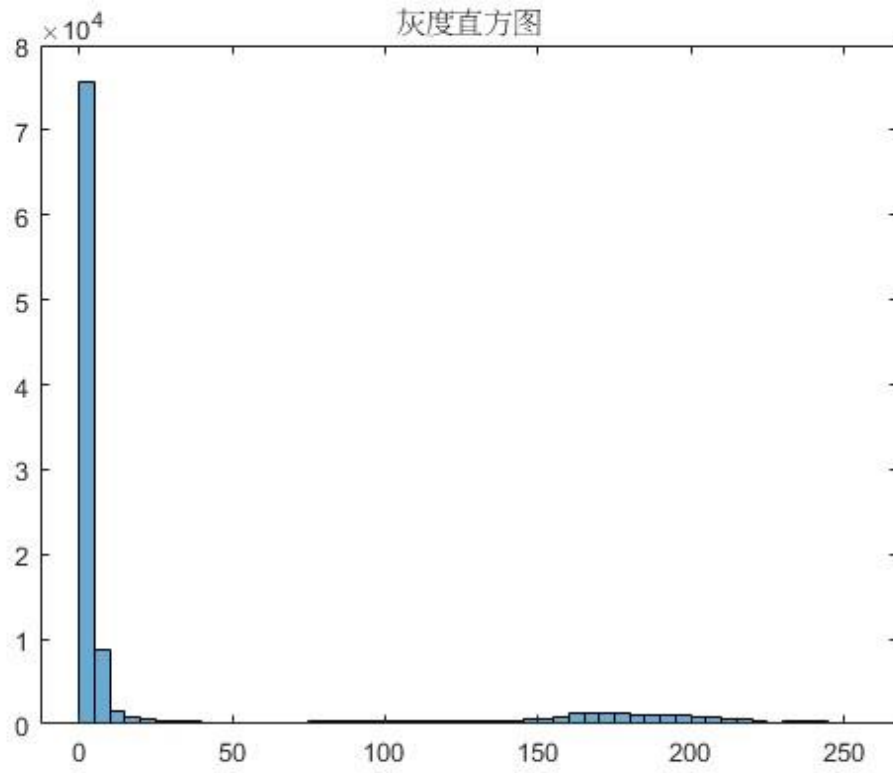
```
1 % 清除命令行和工作空间
2 clear;
3 clc;
4
5 % 读取低照度图像
6 img = imread('1.jpg');
7
8 % 将图像转换为灰度图像
9 gray_img = rgb2gray(img);
10
11 % 计算并显示灰度直方图
12 figure;
13 histogram(gray_img);
14 title('灰度直方图');
15
16 % 计算并显示离散傅里叶变换频谱幅度图
17 dft_img = fft2(double(gray_img));
18 dft_img_shift = fftshift(dft_img);
19 dft_magnitude = abs(dft_img_shift);
20 figure;
21 imshow(log(1 + dft_magnitude), []);
22 title('DFT 频谱幅度图');
23
24 % 对图像进行直方图均衡化
25 equalized_img = histeq(gray_img);
26
27
28 % 对图像进行同态滤波
29 % 首先，对图像进行对数变换
30 log_img = log(double(gray_img) + 1);
31 % 然后，进行高斯滤波
32 gaussian_filter = fspecial('gaussian', [5 5], 2);
33 filtered_log_img = imfilter(log_img, gaussian_filter, 'replicate');
34 % 接着，进行指数变换
35 filtered_img = exp(filtered_log_img) - 1;
36 % 最后，进行直方图均衡化
37 homomorphic_img = histeq(uint8(filtered_img));
38
39
40 % 显示并比较两种算法的效果
41 figure;
42 subplot(1,2,1);
43 imshow(equalized_img);
44 title('直方图均衡化');
45
46 subplot(1,2,2);
47 imshow(homomorphic_img);
48 title('同态滤波');
```

## 2.结果

DFT 频谱幅度图



灰度直方图



直方图均衡化



同态滤波



理论作业：

1. 均值

$$(1+2+1+1+10+2+5+2+6)/9 = 4$$

$$(2+1+4+6+2+5+2+6+1)/9 = 3.8$$

...

2. 中位

1 1 1 2 2 5 6 10  $\Rightarrow 2$

1 2 2 2 3 6 8 10  $\Rightarrow 3$

...

1	2	17	4	3	1	2	1	4	5
1	3	3.8	2.9	4	1	3.5	4.2	4.3	6
5	4.3	4.5	4.6	8	5	4.7	4.7	5.1	8
8	4.8	4.9	6.1	8	5	5.2	5.4	6.7	8
5	6	7	8	9	5	6	7	8	9

1	2	1	6	3
1	2	3	4	4
5	5	5	6	8
5	5	6	8	8
5	6	7	8	9

凌江丁健