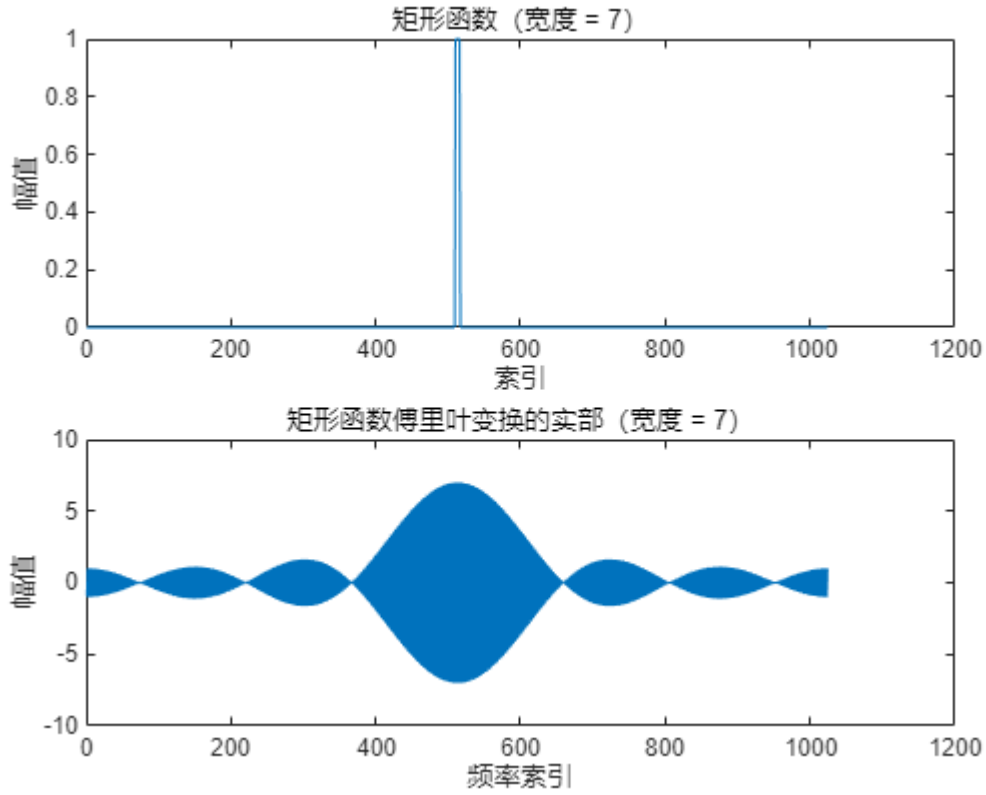
Lab 7 Report

12210309 Li Zhiying

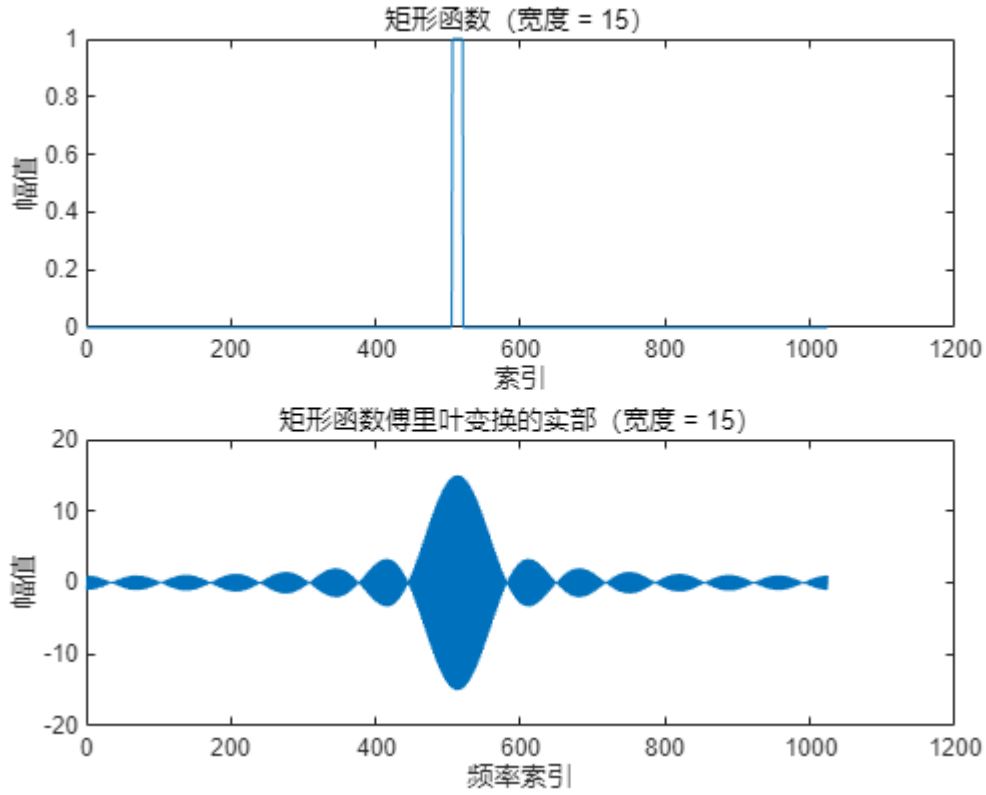
Problem 1 - Some 1-D ffts of simple functions

```
N = 1024;
seq = zeros(1,N);
center = 513;
```

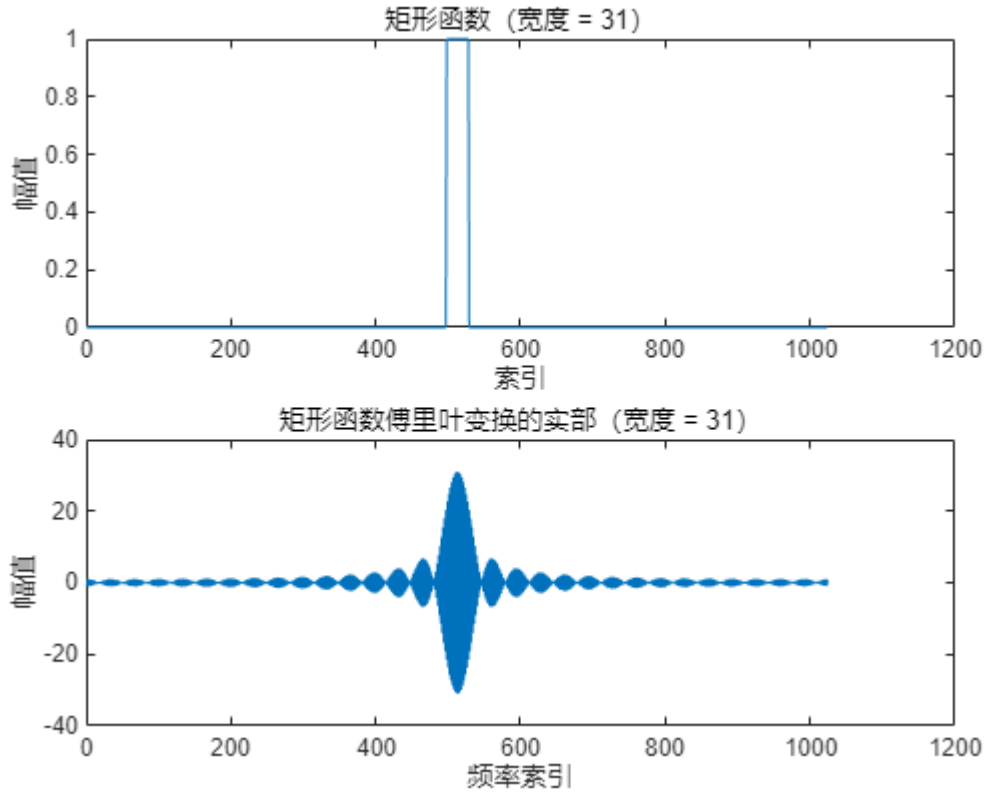
```
% i
width = 7;
half_width = (width - 1) / 2;
seq(center - half_width : center + half_width ) = 1;

Y = fft(seq);
Y_shifted = fftshift(Y);

figure;
subplot(2, 1, 1);
plot(seq);
title('矩形函数（宽度 = 7）');
xlabel('索引');
ylabel('幅值');

subplot(2, 1, 2);
plot(real(Y_shifted));
title('矩形函数傅里叶变换的实部（宽度 = 7）');
xlabel('频率索引');
ylabel('幅值');
```




```
% ii
width = 15;
half_width = (width - 1) / 2;
seq = zeros(1,N);
seq(center - half_width : center + half_width ) = 1;

Y = fft(seq);
Y_shifted = fftshift(Y);

figure;
subplot(2, 1, 1);
plot(seq);
title('矩形函数（宽度 = 15）');
xlabel('索引');
ylabel('幅值');

subplot(2, 1, 2);
plot(real(Y_shifted));
title('矩形函数傅里叶变换的实部（宽度 = 15）');
xlabel('频率索引');
ylabel('幅值');
```

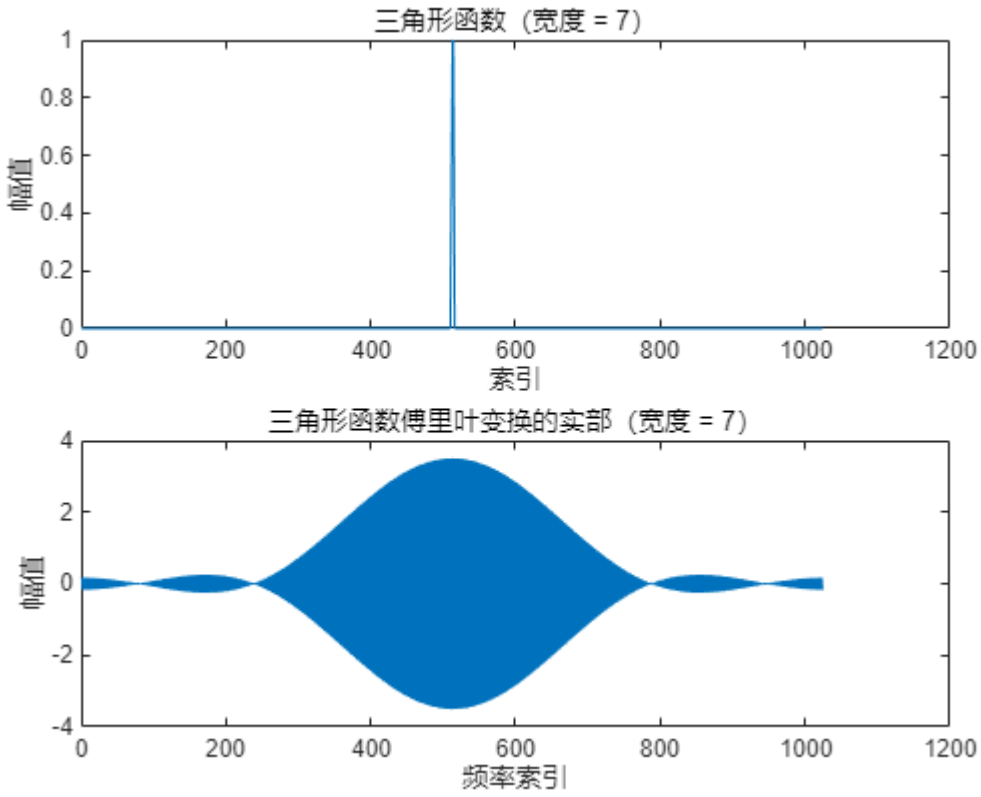



```
% iii
width = 31;
half_width = (width - 1) / 2;
seq = zeros(1,N);
seq(center - half_width : center + half_width ) = 1;

Y = fft(seq);
Y_shifted = fftshift(Y);

figure;
subplot(2, 1, 1);
plot(seq);
title('矩形函数（宽度 = 31）');
xlabel('索引');
ylabel('幅值');

subplot(2, 1, 2);
plot(real(Y_shifted));
title('矩形函数傅里叶变换的实部（宽度 = 31）');
xlabel('频率索引');
ylabel('幅值');
```

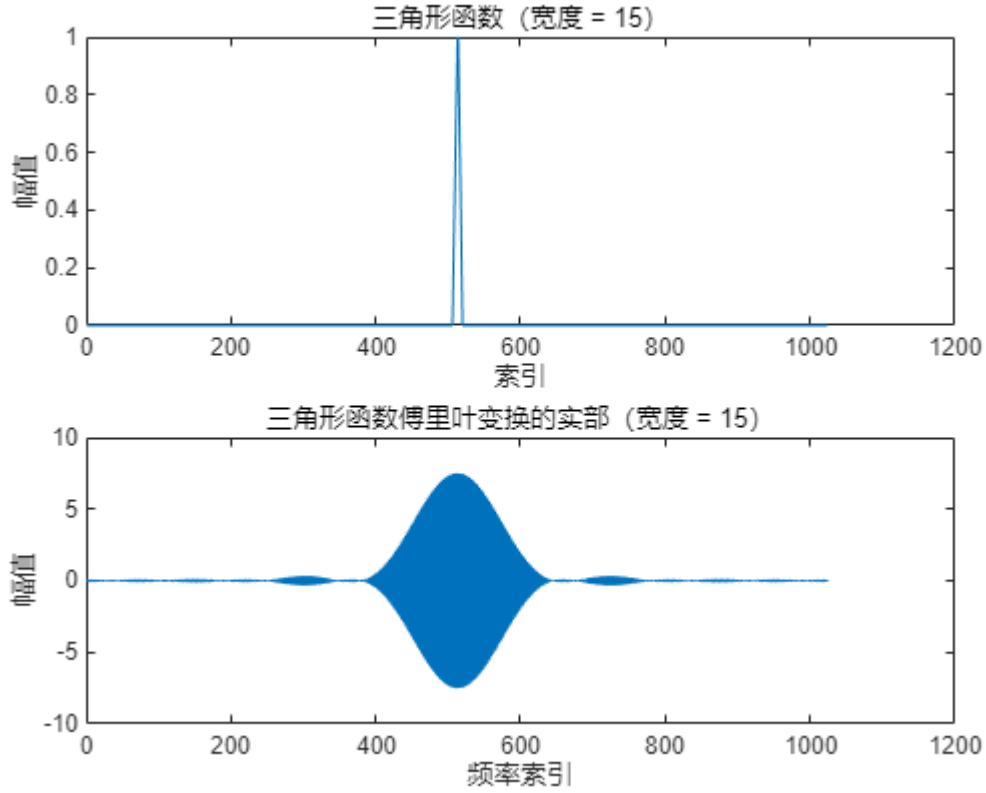



```
% iv
create_triangle = @(width) ...
    [linspace(0, 1, (width + 1)/2), linspace(1, 0, width/2)];

width = 7;
half_width = (width - 1) / 2;
seq = zeros(1, N);
triangle = create_triangle(width);
seq(center - half_width : center + half_width) = triangle;


Y = fft(seq);
Y_shifted = fftshift(Y);

figure;
subplot(2, 1, 1);
plot(seq);
title('三角形函数（宽度 = 7）');
xlabel('索引');
ylabel('幅值');

subplot(2, 1, 2);
plot(real(Y_shifted));
title('三角形函数傅里叶变换的实部（宽度 = 7）');
xlabel('频率索引');
```

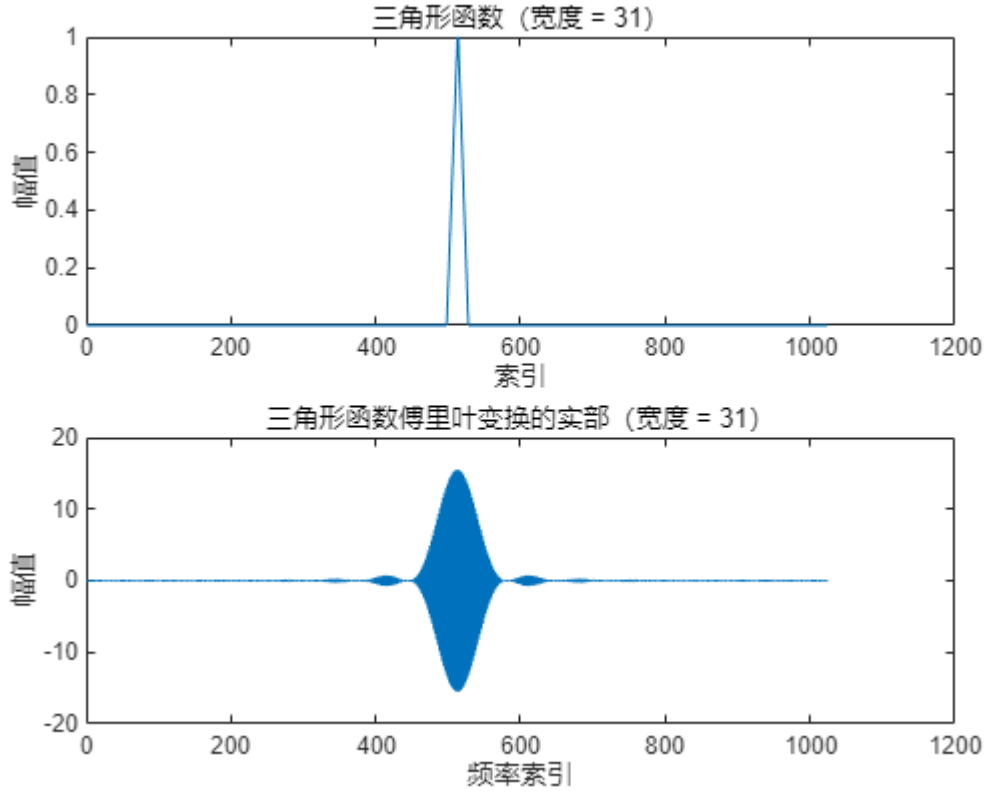
```
ylabel('幅值');
```




```
width = 15;
half_width = (width - 1) / 2;
seq = zeros(1, N);
triangle = create_triangle(width);
seq(center - half_width : center + half_width) = triangle;


Y = fft(seq);
Y_shifted = fftshift(Y);

figure;
subplot(2, 1, 1);
plot(seq);
title('三角形函数（宽度 = 15）');
xlabel('索引');
ylabel('幅值');

subplot(2, 1, 2);
plot(real(Y_shifted));
title('三角形函数傅里叶变换的实部（宽度 = 15）');
xlabel('频率索引');
ylabel('幅值');
```



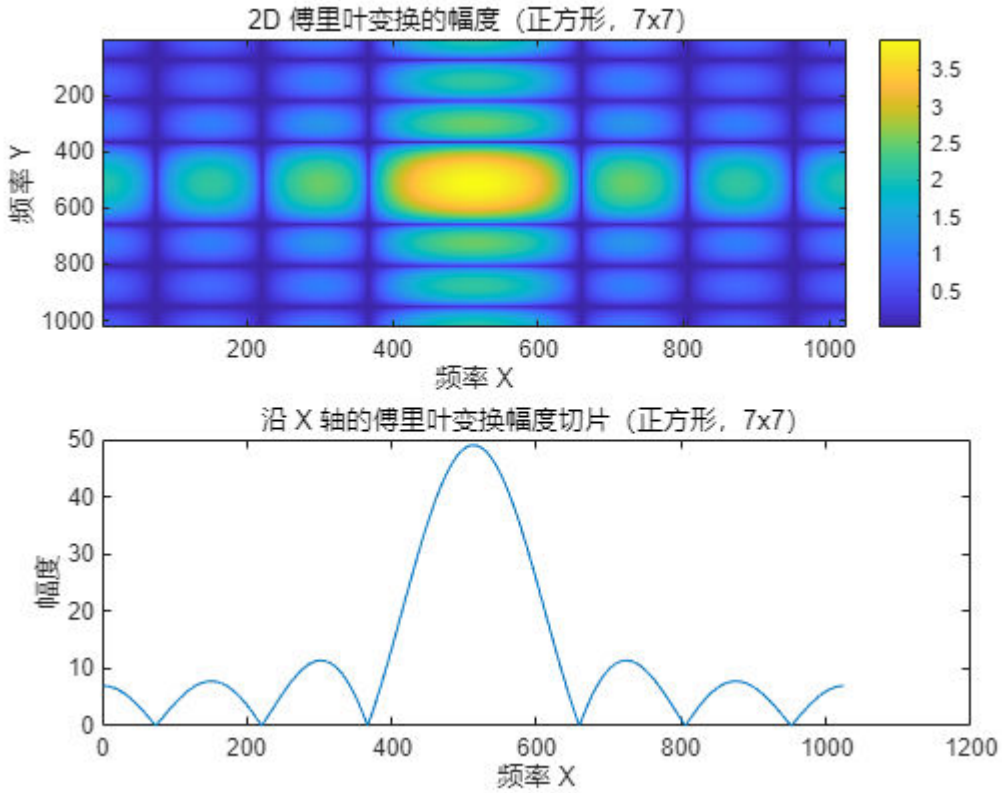

```
width = 31;
half_width = (width - 1) / 2;
seq = zeros(1, N);
triangle = create_triangle(width);
seq(center - half_width : center + half_width) = triangle;

Y = fft(seq);
Y_shifted = fftshift(Y);

figure;
subplot(2, 1, 1);
plot(seq);
title('三角形函数（宽度 = 31）');
xlabel('索引');
ylabel('幅值');


subplot(2, 1, 2);
plot(real(Y_shifted));
title('三角形函数傅里叶变换的实部（宽度 = 31）');
xlabel('频率索引');
ylabel('幅值');
```



Problem 2 - 2-D transforms

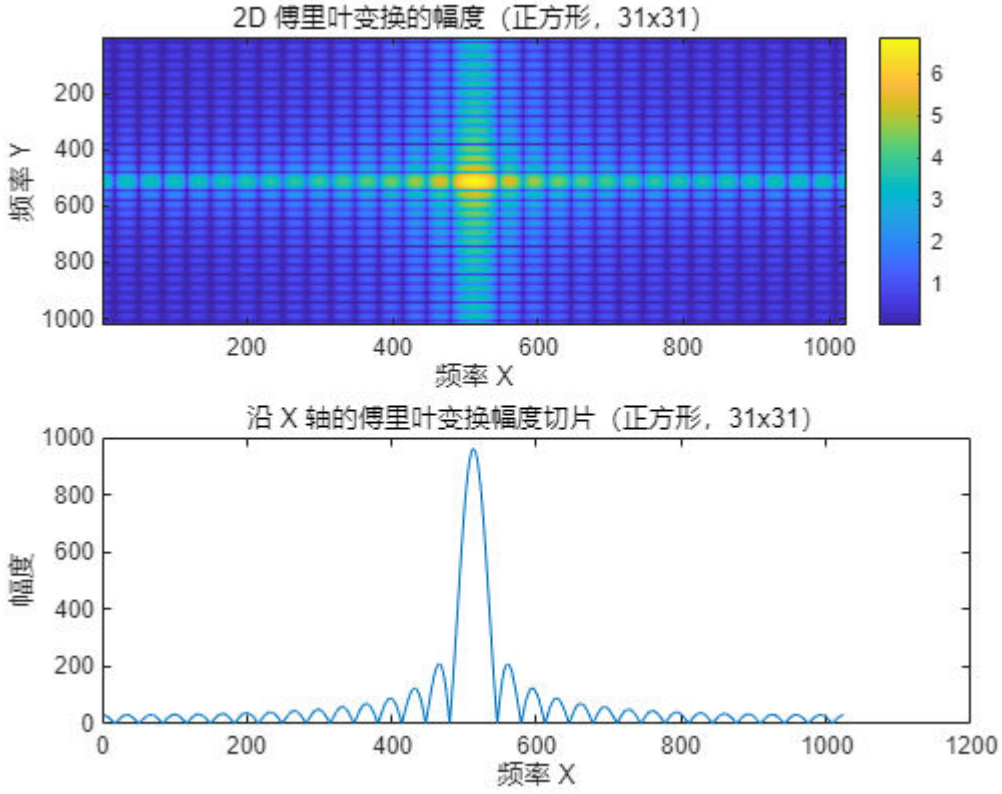
```
N = 1024;
array = zeros(N,N);
center = 513;

% i
width = 7;
half_width = (width - 1) / 2;
array(center - half_width : center + half_width, center - half_width : center +
half_width) = 1;

F = fft2(array);
F_shifted = fftshift(F);

magnitude = abs(F_shifted);

figure;
subplot(2, 1, 1);
imagesc(log(1 + magnitude));
colorbar;
title('2D 傅里叶变换的幅度（正方形，7x7）');
xlabel('频率 X');
ylabel('频率 Y');
```

```
subplot(2, 1, 2);
plot(magnitude(center, :));
title('沿 X 轴的傅里叶变换幅度切片（正方形，7x7）');
xlabel('频率 X');
ylabel('幅度');
```



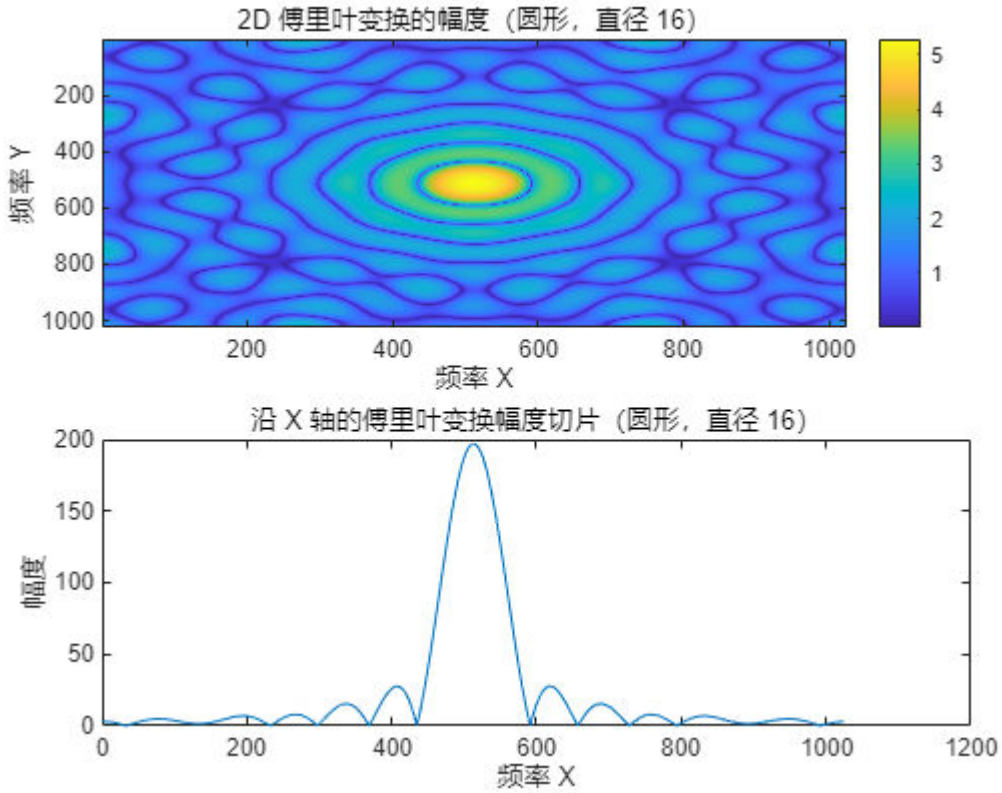

```
% ii
width = 31;
half_width = (width - 1) / 2;
array(center - half_width : center + half_width, center - half_width : center +
half_width) = 1;

F = fft2(array);
F_shifted = fftshift(F);

magnitude = abs(F_shifted);

figure;
subplot(2, 1, 1);
imagesc(log(1 + magnitude));
colorbar;
title('2D 傅里叶变换的幅度（正方形，31x31）');
xlabel('频率 X');
ylabel('频率 Y');

subplot(2, 1, 2);
```

```
plot(magnitude(center, :));
title('沿 X 轴的傅里叶变换幅度切片（正方形，31x31）');
xlabel('频率 X');
ylabel('幅度');
```



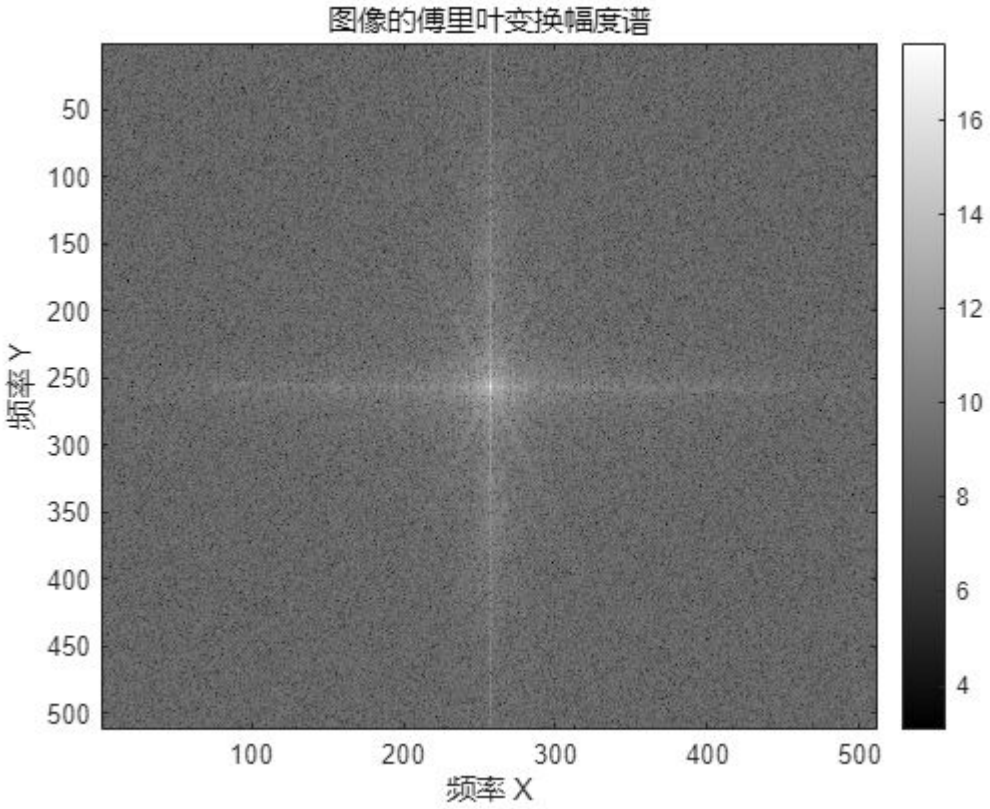

```
% iii
radius = 8;
[rows, cols] = meshgrid(1:N, 1:N);
distance = sqrt((rows - center).^2 + (cols - center).^2);

array = zeros(N, N);
array(distance <= radius) = 1;

F = fft2(array);
F_shifted = fftshift(F);

magnitude = abs(F_shifted);

figure;
subplot(2, 1, 1);
imagesc(log(1 + magnitude));
colorbar;
title('2D 傅里叶变换的幅度（圆形，直径 16）');
xlabel('频率 X');
ylabel('频率 Y');
```

```
subplot(2, 1, 2);
plot(magnitude(center, :));
title('沿 X 轴的傅里叶变换幅度切片（圆形，直径 16）');
xlabel('频率 X');
ylabel('幅度');
```



Problem 3 - A low-pass filter

```
f = fopen('lab7prob3data','r');
img = fread(f,[512,512],'uint8');
fclose(f);

img = img';

img = double(img);

F = fft2(img);
F_shifted = fftshift(F);

figure;
imagesc(log(1 + abs(F_shifted)));  % 使用对数尺度显示傅里叶变换的幅度谱
colormap('gray');
colorbar;
title('图像的傅里叶变换幅度谱');
xlabel('频率 X');
ylabel('频率 Y');
```




```
% i
mask_width = 255;

[N, M] = size(F_shifted);

mask = zeros(N, M);
centerX = round(N/2);
centerY = round(M/2);

mask(centerX - mask_width:centerX + mask_width, centerY - mask_width:centerY +
mask_width) = 1;

F_filtered = F_shifted .* mask;

img_filtered = ifft2(ifftshift(F_filtered));

figure;
imshow(abs(img_filtered), []);
title('低通滤波后的图像（掩模宽度 = 255）');
```

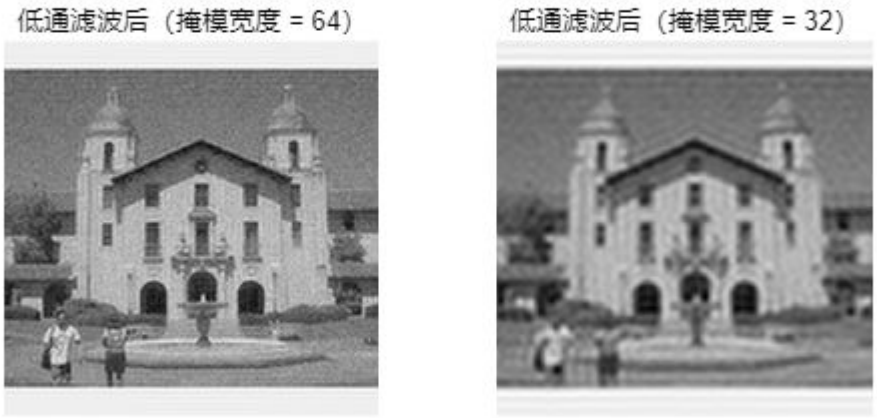
低通滤波后的图像（掩模宽度 = 255）

```
% ii
mask_width = 127;

mask = zeros(N, M);
mask(centerX - mask_width:centerX + mask_width, centerY - mask_width:centerY +
mask_width) = 1;

F_filtered = F_shifted .* mask;

img_filtered = ifft2(ifftshift(F_filtered));

figure;
imshow(abs(img_filtered), []);
title('低通滤波后的图像（掩模宽度 = 127）');
```

低通滤波后的图像（掩模宽度 = 127）

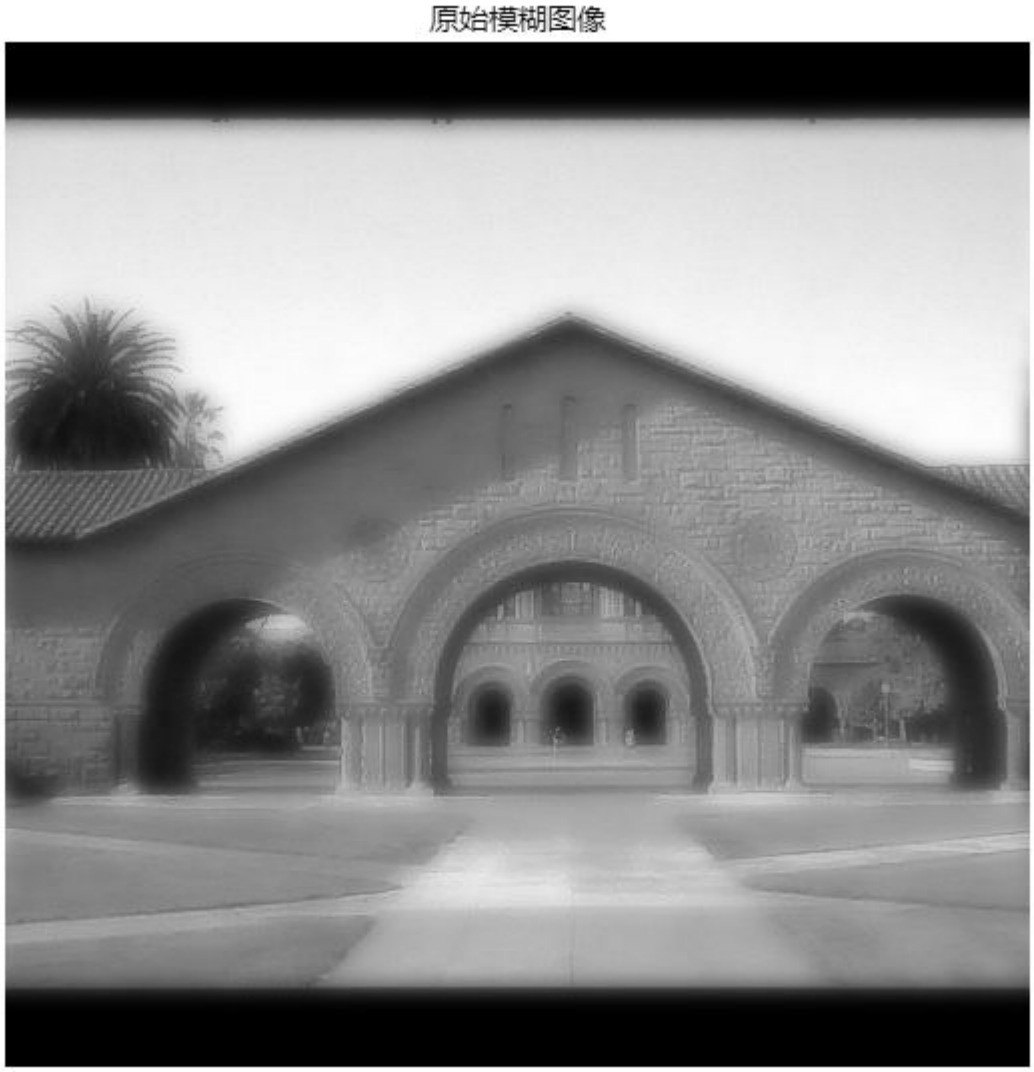
```
% iii
mask_width = 64;
mask = zeros(N, M);
mask(centerX - mask_width:centerX + mask_width, centerY - mask_width:centerY +
mask_width) = 1;
F_filtered = F_shifted .* mask;
img_filtered_64 = ifft2(ifftshift(F_filtered));

mask_width = 32;
mask = zeros(N, M);
mask(centerX - mask_width:centerX + mask_width, centerY - mask_width:centerY +
mask_width) = 1;
F_filtered = F_shifted .* mask;
img_filtered_32 = ifft2(ifftshift(F_filtered));
```

```
figure;
subplot(1,2, 1);
imshow(abs(img_filtered_64), []);
title('低通滤波后（掩模宽度 = 64）');

subplot(1, 2, 2);
imshow(abs(img_filtered_32), []);
title('低通滤波后（掩模宽度 = 32）');
```



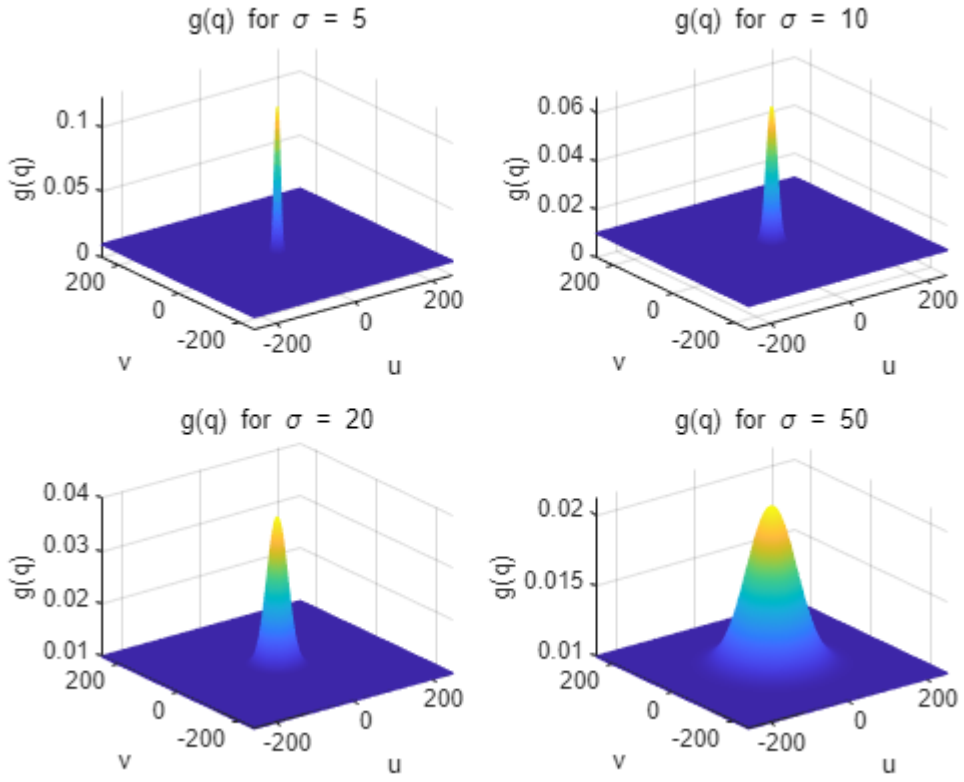
Problem 4 – Sharpening a blurred image

```
f = fopen('lab7prob4data','r');
img = fread(f,[512,512],'uint8');
fclose(f);

img = img';

%i
figure;
imshow(img,[]);
title('原始模糊图像');
```




```
% ii
[u, v] = meshgrid(-256:255, -256:255);

q = sqrt(u.^2 + v.^2);

sigma_values = [5, 10, 20, 50];

figure;
for i = 1:length(sigma_values)
    sigma = sigma_values(i);
    g_q = exp(-q.^2 / (2 * sigma^2)) / (sigma * sqrt(pi)) * 0.99 + 0.01;  % 计算 g(q)

    subplot(2, 2, i);
    mesh(u, v, g_q);  % 绘制 3D 图像
    title(['g(q) for \sigma = ', num2str(sigma)]);
```

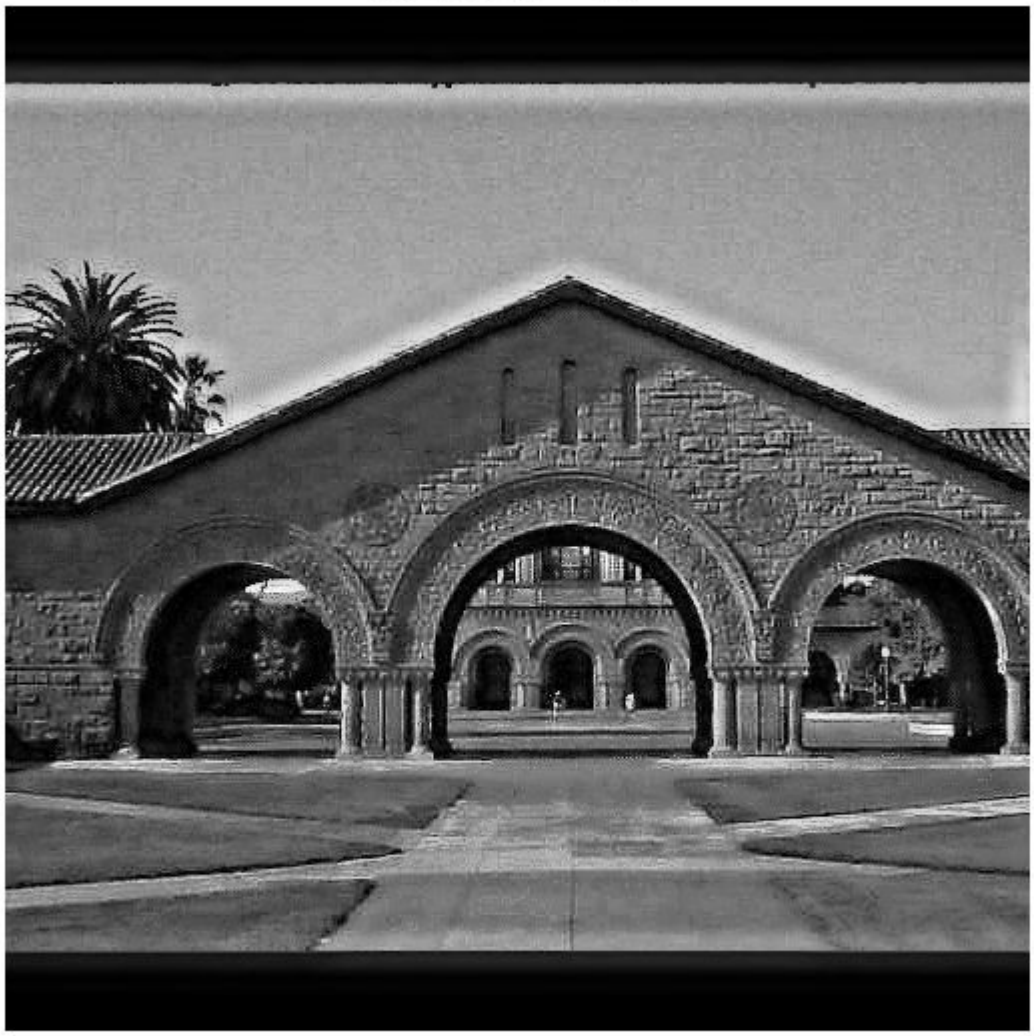
```
    xlabel('u');
    ylabel('v');
    zlabel('g(q)');
end
```



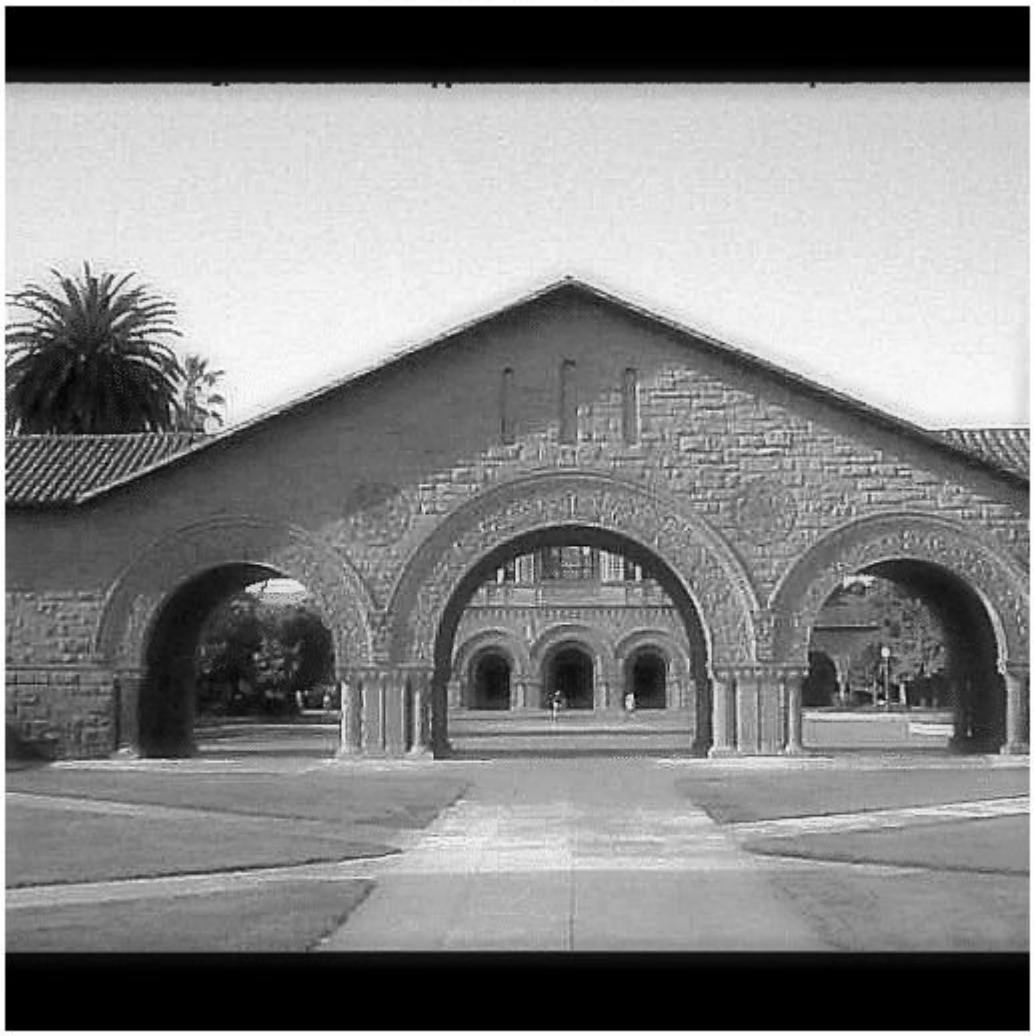

```
% iii
img = double(img);
F = fft2(img);

F_shifted = fftshift(F);

for i = 1:length(sigma_values)
    sigma = sigma_values(i);

    g_q = exp(-q.^2 / (2 * sigma^2)) / (sigma * sqrt(pi)) * 0.99 + 0.01;

    F_filtered = F_shifted ./ g_q;
    img_filtered = ifft2(ifftshift(F_filtered));

    figure;
    imshow(abs(img_filtered), []);
    title(['锐化图像（\sigma = ', num2str(sigma), '）']);
end
```

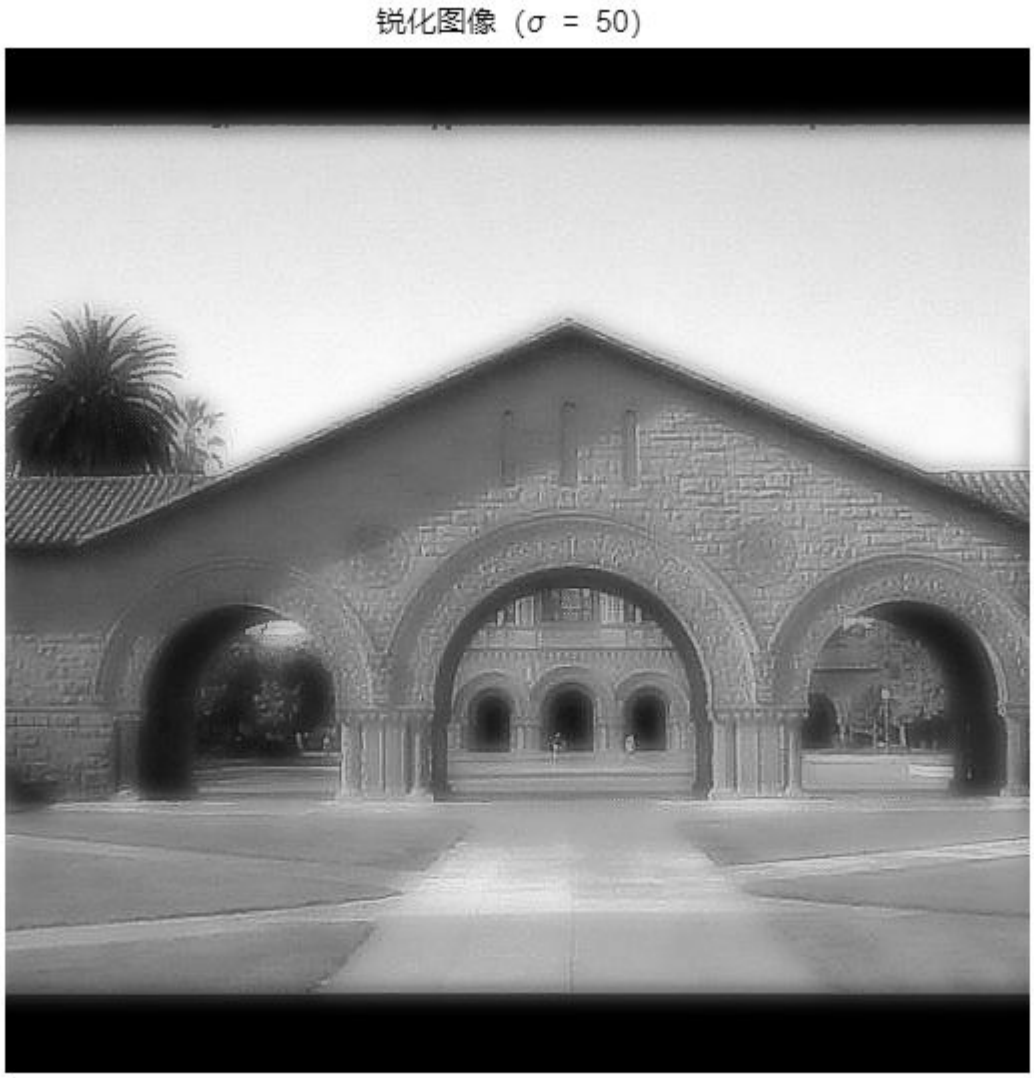
锐化图像（$\sigma = 5$）

锐化图像（σ = 10）



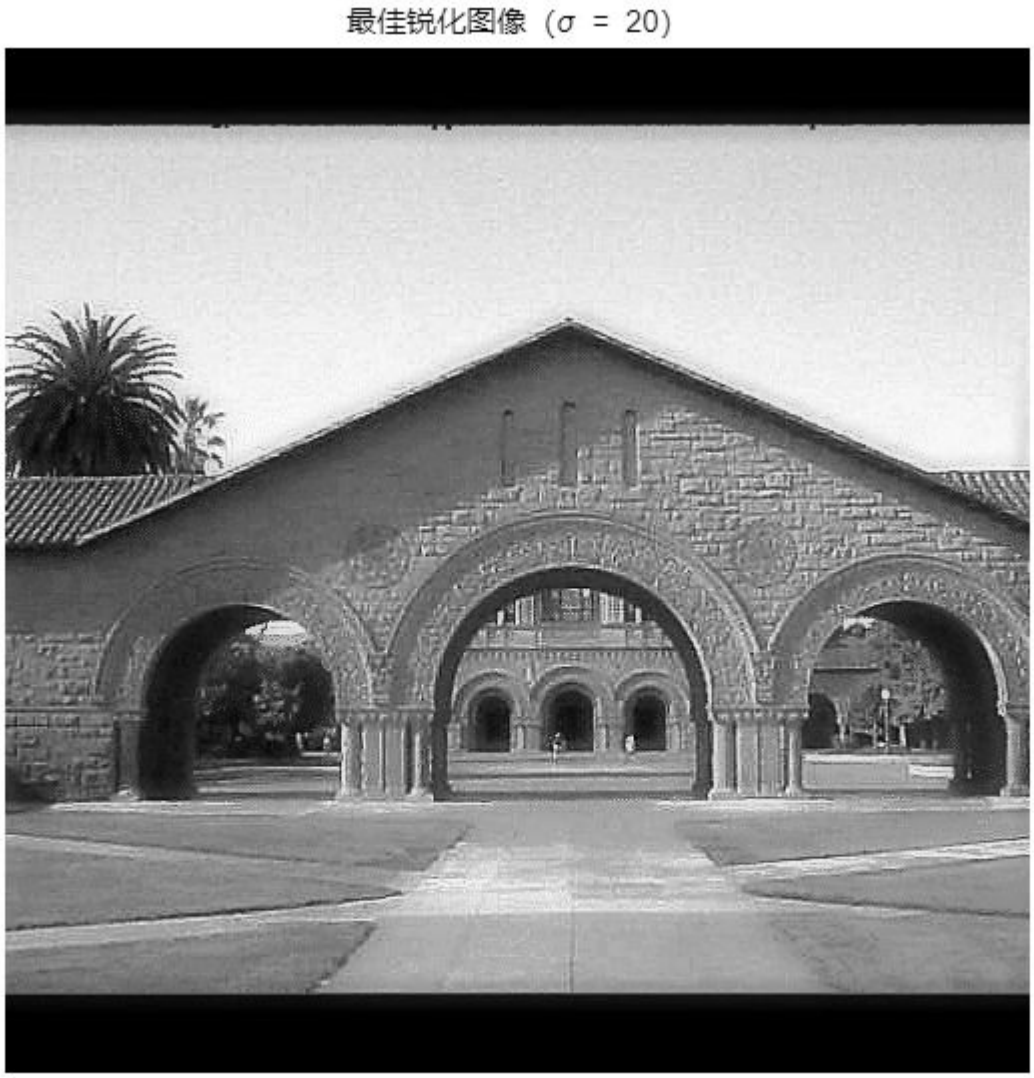
锐化图像（σ = 20）





Answer: 最好的 sigma = 20；

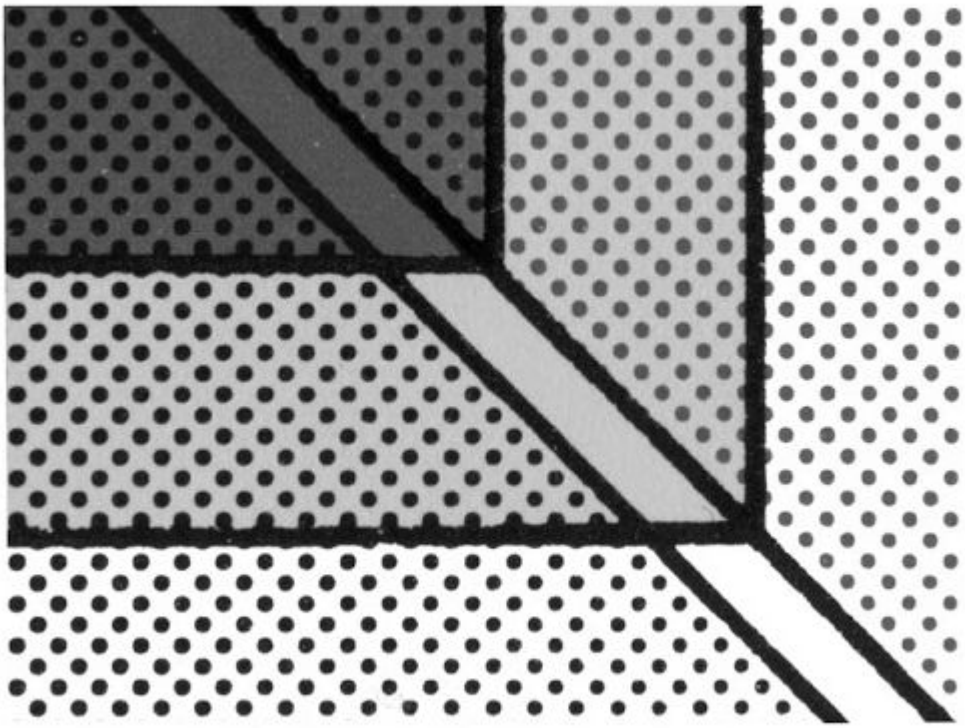
```
sigma_best = 20;

g_q_best = exp(-q.^2 / (2 * sigma_best^2)) / (sigma_best * sqrt(pi)) * 0.99 + 0.01;
F_filtered_best = F_shifted ./ g_q_best;
img_filtered_best = ifft2(ifftshift(F_filtered_best));


figure;
imshow(abs(img_filtered_best), []);
title(['最佳锐化图像（\sigma = ', num2str(sigma_best), '）']);
```



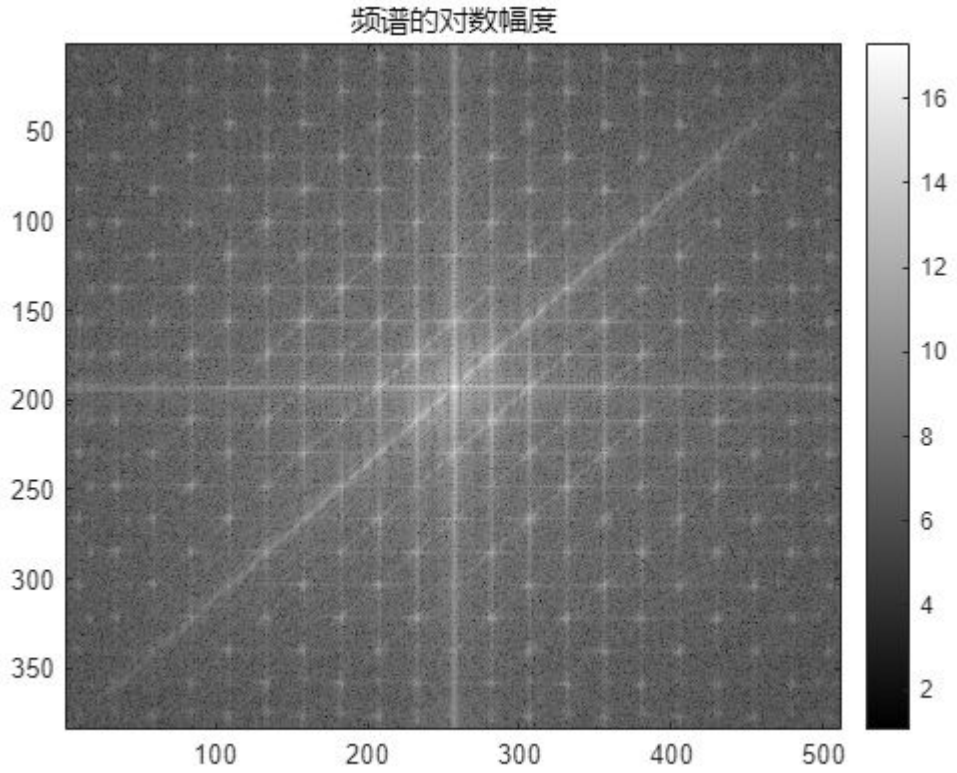
Problem 5 – Image Manipulation in the Frequency Domain

```
% i
img = imread('lab7prob5data.jpg');
img = rgb2gray(img);
figure;
imshow(img, []);
title('原始图像');
```

原始图像



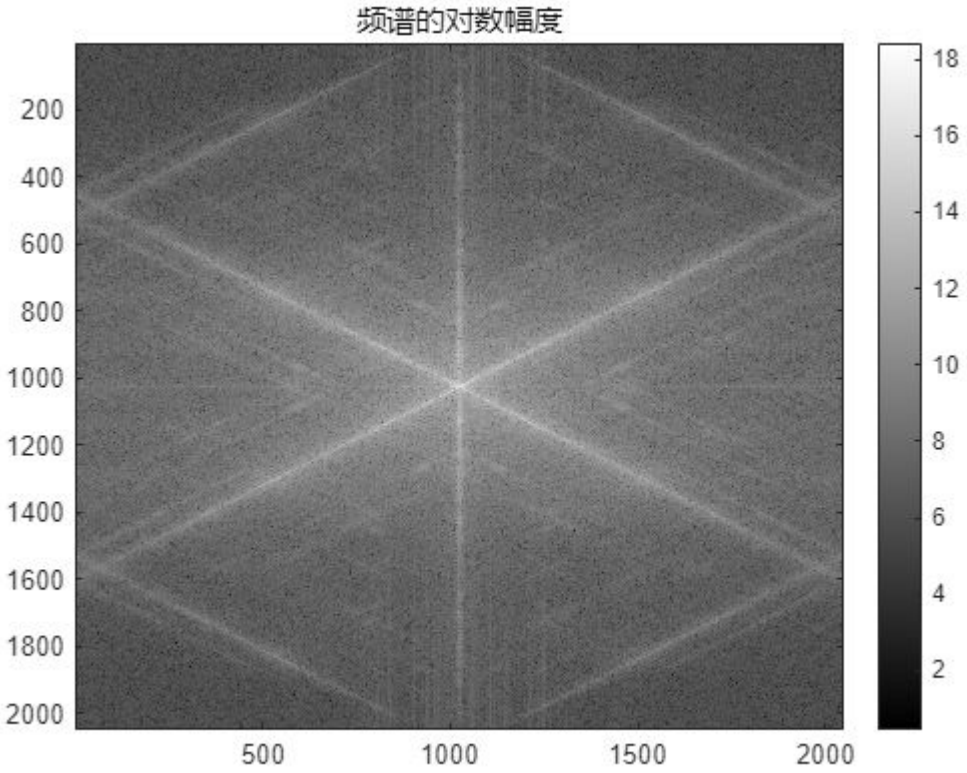
```
% ii
img = double(img);
F = fft2(img);

F_shifted = fftshift(F);

log_magnitude = log(1 + abs(F_shifted));

figure;
imagesc(log_magnitude);
colormap('gray');
colorbar;
title('频谱的对数幅度');
```



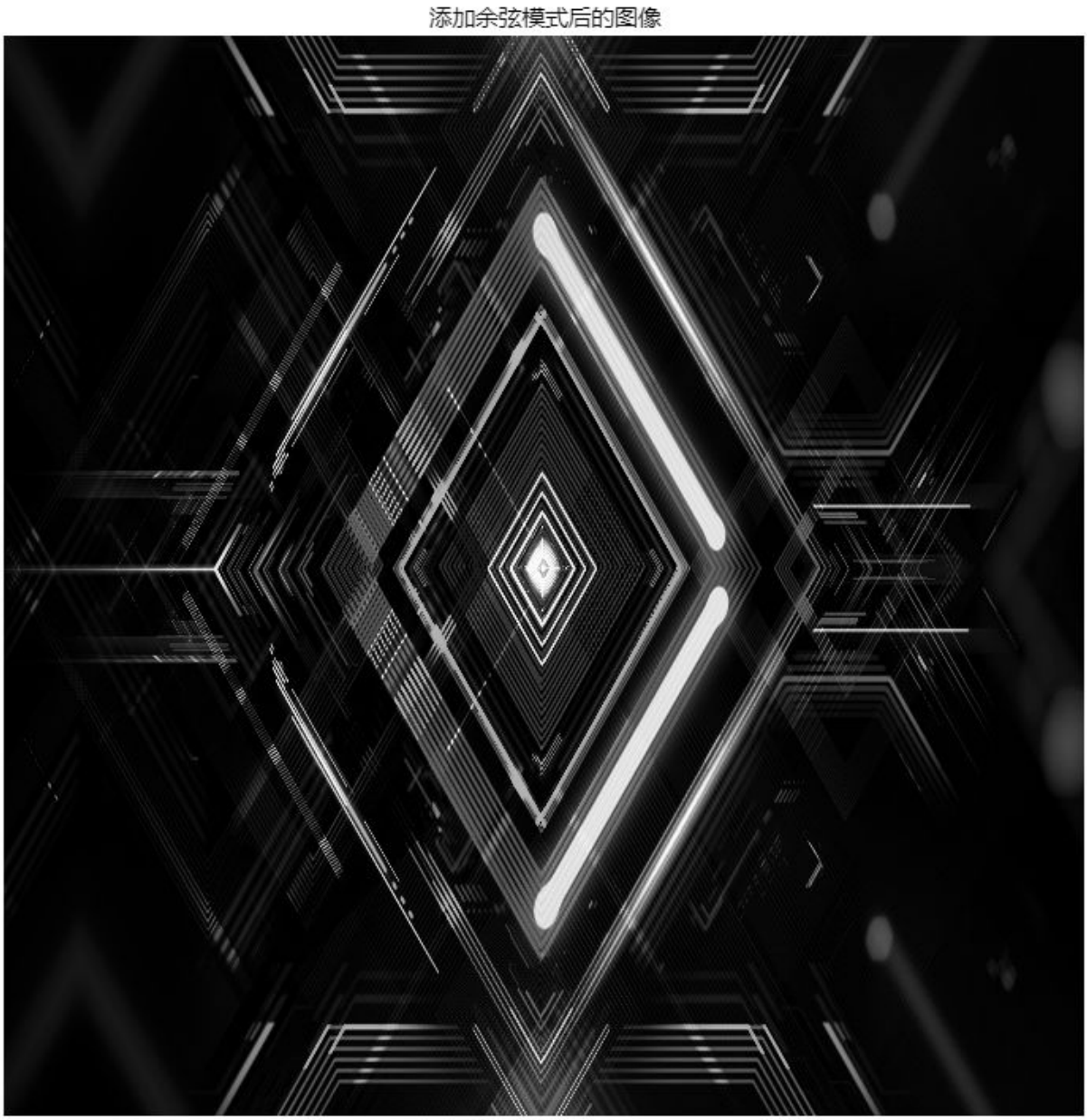

```
% iii
img2 = imread("Default.jpg");

img2 = rgb2gray(img2);

[M, N] = size(img2);
M_new = 2^nextpow2(M);
N_new = 2^nextpow2(N);
img2 = imresize(img2, [M_new, N_new]);

figure;
imshow(img2, []);
title('裁剪后的图像');
```

裁剪后的图像

```
img2 = double(img2);
F2 = fft2(img2);
F2_shifted = fftshift(F2);
log_magnitude2 = log(1 + abs(F2_shifted));

figure;
imagesc(log_magnitude2);
colormap('gray');
colorbar;
title('频谱的对数幅度');
```



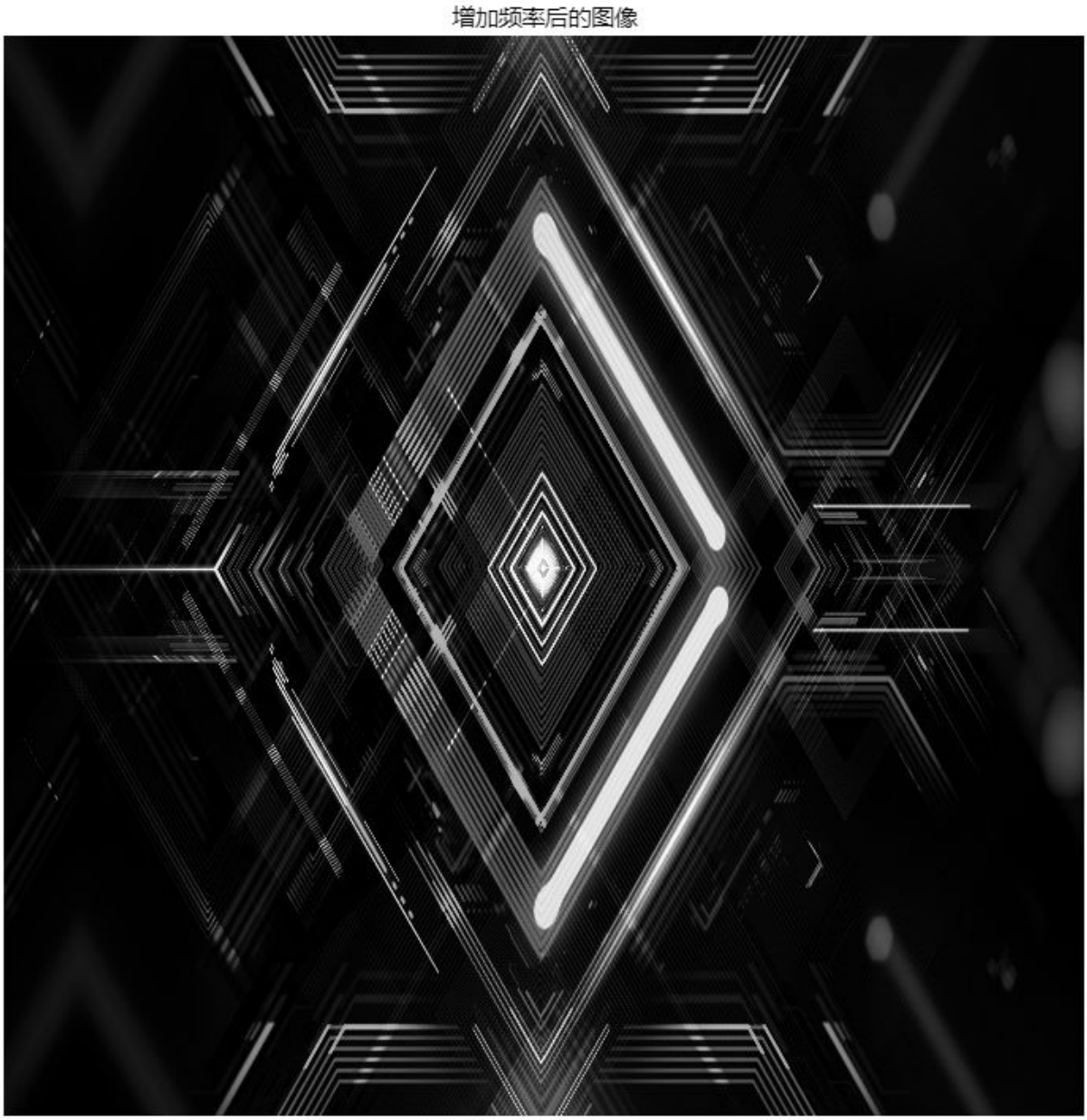

```
% iv
u = linspace(-M_new/2, M_new/2-1, M_new);
v = linspace(-N_new/2, N_new/2-1, N_new);
[U, V] = meshgrid(u, v);

f0 = 10;
cos_pattern = cos(2 * pi * f0 * U) + cos(2 * pi * f0 * V);

F2_shifted = F2_shifted + cos_pattern;

img_patterned = ifft2(ifftshift(F2_shifted));

figure;
imshow(abs(img_patterned), []);
title('添加余弦模式后的图像');
```



Answer: 通过将一个二维余弦模式添加到频域中，影响图像的高频成分

```
% v
f0_new = 20;
cos_pattern_new = cos(2 * pi * f0_new * U) + cos(2 * pi * f0_new * V);

F2_shifted_new = F2_shifted + cos_pattern_new;

img_patterned_new = ifft2(ifftshift(F2_shifted_new));

figure;
imshow(abs(img_patterned_new), []);
```

```
title('增加频率后的图像');
```



Answer: 增加了频率，使得图像中出现更细致的纹理。