

Project2

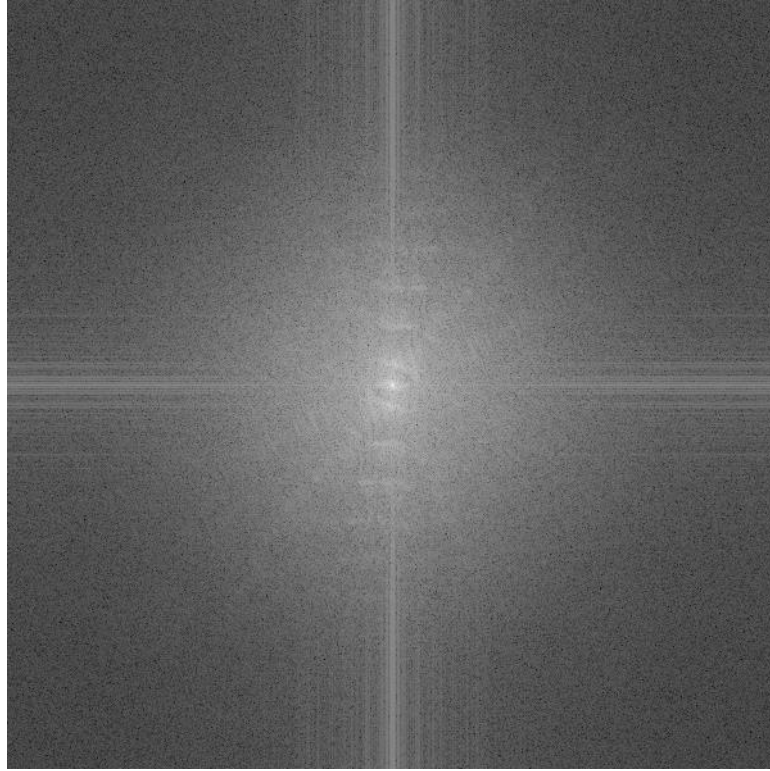
0811562 何祁恩

(a) Source code

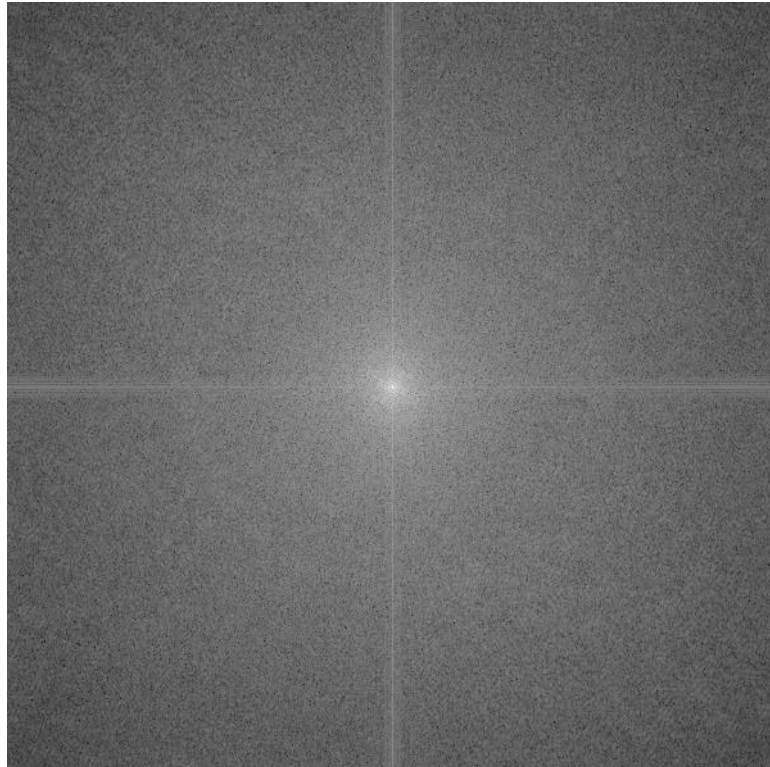
```
1  clc;
2  clear;
3  close all;
4
5  filename = 'fruit';
6  f = imread([filename, '.tif']);
7  [M, N] = size(f);
8
9  %(b)
10 F = fft2(double(f));
11 F_shift = fftshift(F);
12
13 %padding to avoid wraparound error
14 F_pad = fft2(double(f), 2*M, 2*N);
15 F_pad_shift = fftshift(F_pad);
16
17 %create Gaussian LPF and HPF
18 LPF = zeros(2*M, 2*N);
19 D0 = 200;
20
21 for u = 1:2*M
22     for v = 1:2*N
23         D2 = (u-M)^2 + (v-N)^2;
24         LPF(u,v) = exp(-1*D2/(2*D0*D0));
25     end
26 end
27
28 HPF = 1 - LPF;
29
30 %Image pass LPF
31 G_LPF_shift = F_pad_shift .* LPF;
32 G_LPF = ifftshift(double(G_LPF_shift));
33 g_LPF = ifft2(double(G_LPF));
34 Re_g_LPF = real(g_LPF(1:M, 1:N));
35
36
37 %Image pass HPF
38 G_HPF_shift = F_pad_shift .* HPF;
39 G_HPF = ifftshift(double(G_HPF_shift));
40 g_HPF = ifft2(double(G_HPF));
41 Re_g_HPF = real(g_HPF(1:M, 1:N));
42
43 %get top25 abs(Fshift)
44 Abs_F_shift = abs(F_shift);
45 Abs_F_shift = Abs_F_shift(1:M, 1:N/2);
46 max_abs = zeros(25, 1);
47 u = zeros(25, 1);
48 v = zeros(25, 1);
49 idx = 1;
50
51 while idx ≤ 25
52     sort_abs = sort(Abs_F_shift(:));
53     max_value = max(sort_abs);
54     [row, col] = find(Abs_F_shift == max_value);
55     [length, a] = size(row);
56     for j = 0:length - 1
57         u(idx+j, 1) = row(j+1, 1) - 1;
58         v(idx+j, 1) = col(j+1, 1) - 1;
59         max_abs(idx+j, 1) = max_value;
60         Abs_F_shift(row(j+1, 1), col(j+1, 1)) = -inf;
61     end
62     idx = idx + length;
63 end
64
65 figure(1)
66 imshow(mat2gray(log10(1+ abs(F_shift))));
67 img1 = getimage(gcf);
68 imwrite(img1, ['result/', filename, '_ (600x600_DFT).tiff'], 'tiff', 'Resolution', 150)
69
70 figure(2)
71 imshow(LPF);
72 img2 = getimage(gcf);
73 imwrite(img2, ['result/', filename, '_ (1200x1200_LPF).tiff'], 'tiff', 'Resolution', 150)
74
75 figure(3)
76 imshow(HPF);
77 img3 = getimage(gcf);
78 imwrite(img3, ['result/', filename, '_ (1200x1200_HPF).tiff'], 'tiff', 'Resolution', 150)
79
80 figure(4)
81 imshow(mat2gray(Re_g_LPF));
82 img4 = getimage(gcf);
83 imwrite(img4, ['result/', filename, '_ (600x600_LPF_output).tiff'], 'tiff', 'Resolution', 150)
84
85 figure(5)
86 imshow(mat2gray(Re_g_HPF));
87 img5 = getimage(gcf);
88 imwrite(img5, ['result/', filename, '_ (600x600_HPF_output).tiff'], 'tiff', 'Resolution', 150)
```

(b) Fourier magnitude spectra (Log scale)

1. Kid.tif



2. Fruit.tif

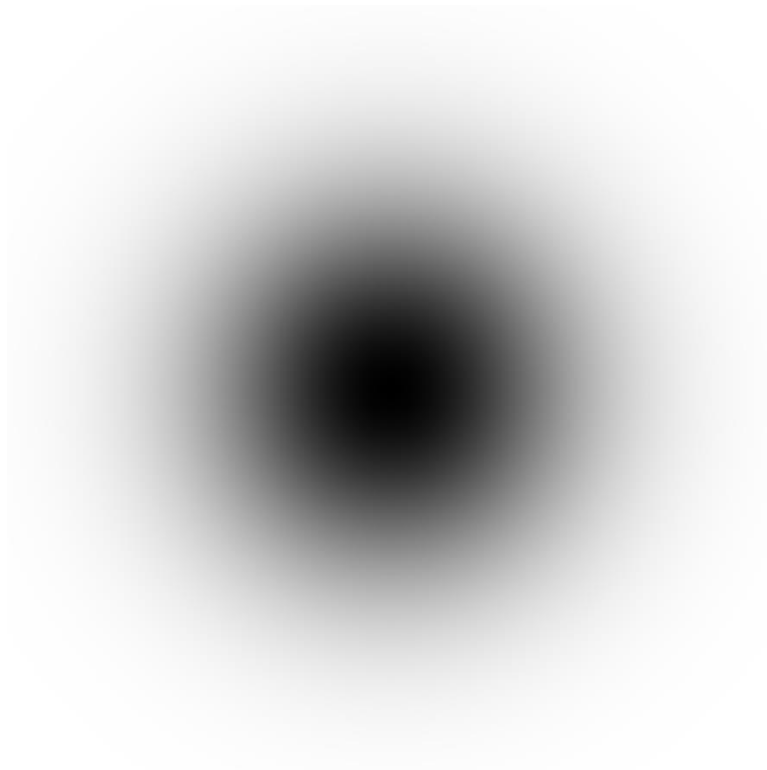


(c) Magnitude response of Gaussian LPF and HPF

1. LPF



2. HPF



(d) Output image

1. Kid LPF output



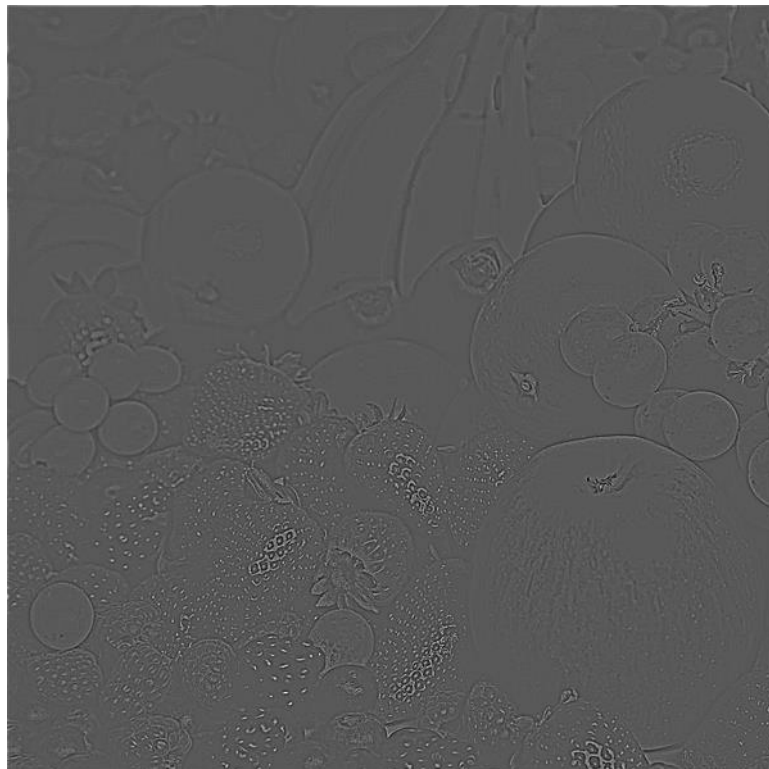
2. Kid HPF output



3. Fruit LPF output



4. Fruit HPF output



(e) Top25 DFT frequency on the left half of (b)

1. Kid

u	v
301	299
300	299
299	299
298	299
297	299
299	297
302	298
298	298
298	294
302	299
302	296
299	298
304	298
316	298
299	294
301	296
317	298
296	296
296	298
316	297
300	294
298	292
297	296
298	297
301	297

2. Fruit

u	v
300	299
301	297
300	298
296	299
303	297
300	297
299	299
295	299
302	297
297	298
301	294
298	299
300	295
302	299
304	299
303	299
296	294
299	298
303	298
299	296
296	296
306	299
297	296
299	297
302	295