

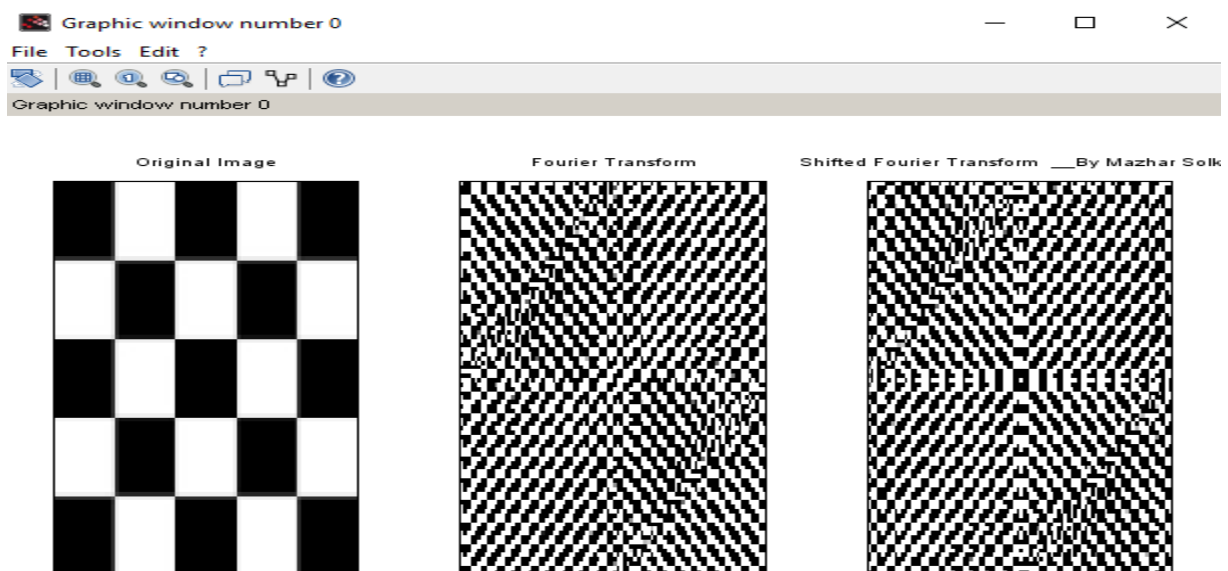
Practical 03

(a) Program to apply Discrete Fourier Transform on an image.

Code :-

```
File Edit Format Options Window Execute ?
pract_03_a.sce (C:\MazharSolkar\pract_03_a.sce) - SciNotes
pract_03_a.sce
1 clc;
2 clear;
3 I = imread('C:\Program-Files-(x86)\scilab-6.0.1\IPCV\images\checkbox.png');
4 subplot(1,3,1)
5 imshow(I)
6 title('Original-Image')
7
8 I = double(I);
9 J = fft2(I);
10 subplot(1,3,2)
11 imshow(J)
12 title('Fourier-Transform')
13
14 L = fftshift(real(J));
15 subplot(1,3,3)
16 imshow(L)
17 title('Shifted-Fourier-Transform')
18
```

Output :-



(b) Program to apply Low pass and High pass filters in frequency domain

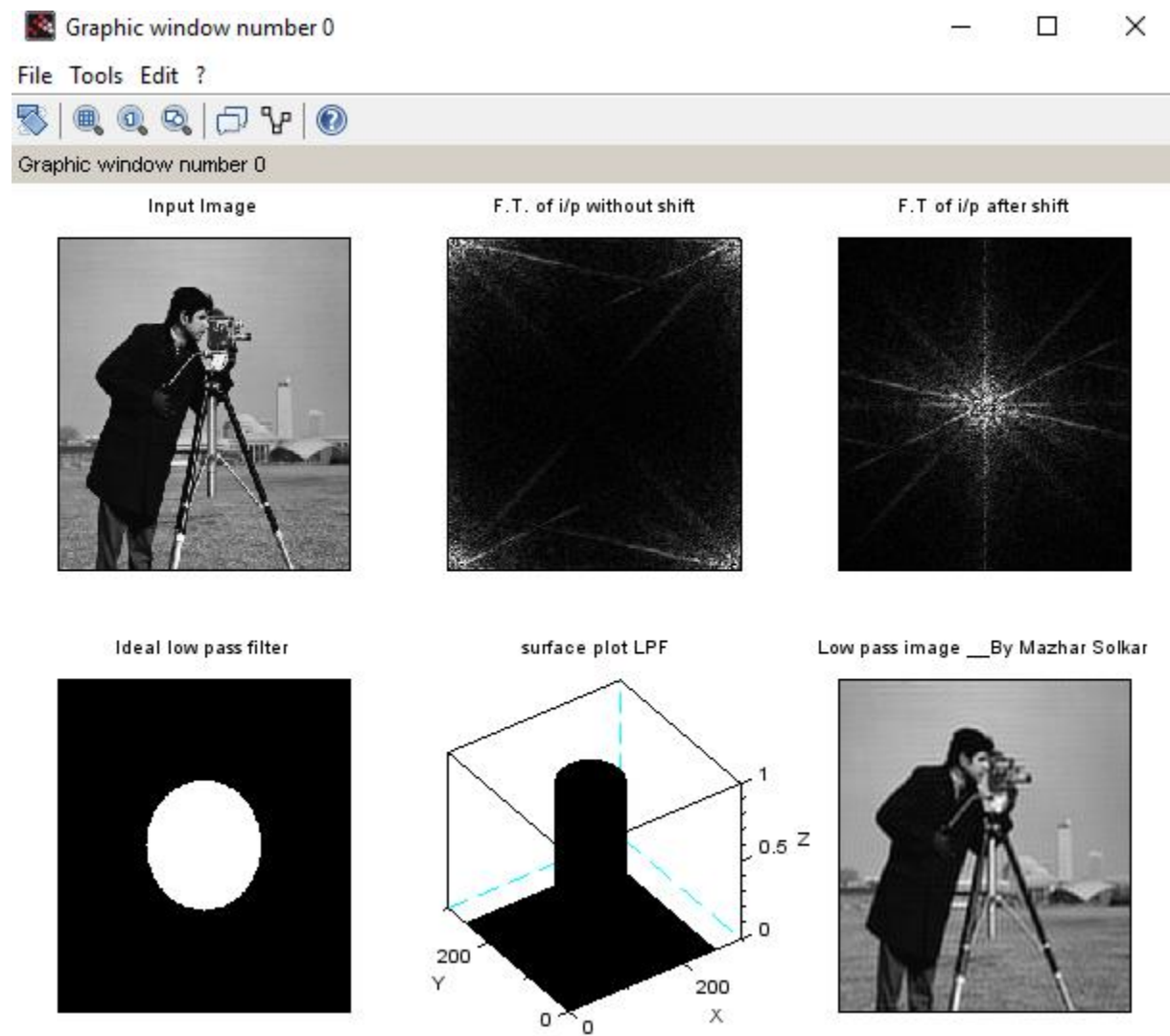
(i) Ideal Low pass Filter (ILPF)

Code :-

```
File Edit Format Options Window Execute ?
practical_03_B_j_ILPF.sce (C:\0_MSc_IT_Notes\image Processing\Practicals\Practical-03\Practical_03_B\practical_03_B_j_ILPF.sce)
practical_03_b_iii_glpf.sce practical_03_B_j_ILPF.sce

1 clc;
2 clear all;
3 a = imread('C:\Program Files\scilab-6.1.1\IPCV\images\cameraman.tif');
4
5 a = im2double(a);
6 subplot(2,3,1), imshow(a), title('Input Image');
7 [m,n] = size(a);
8 D0 = 50;
9 A = fft2(a);
10 subplot(2,3,2), imshow(uint8(abs(A))), title('F.T. of i/p without shift');
11 A_shift = fftshift(A);
12 A_real = abs(A_shift);
13 subplot(2,3,3), imshow(uint8(A_real)), title('F.T. of i/p after shift');
14 A_low = zeros(m,n);
15 for u=1:m
16     for v=1:n
17         d(u,v)=sqrt((u-(m/2))^2+(v-(n/2))^2);
18         if d(u,v) <= D0
19             A_low(u,v)=A_shift(u,v);
20             filt(u,v) = 1;
21         else
22             A_low(u,v)=0;
23             filt(u,v) = 0;
24         end
25     end
26 end
27 subplot(2,3,4), imshow(filt), title('Ideal low pass filter')
28 subplot(2,3,5), mesh(filt), title('surface plot-LPF')
29 B = fftshift(A_low);
30 B_inverse = ifft(B);
31 B_real = abs(B_inverse);
32 subplot(2,3,6), imshow(B_real), title('Low pass image __By Mazhar Solkar')
33
```

Output :-

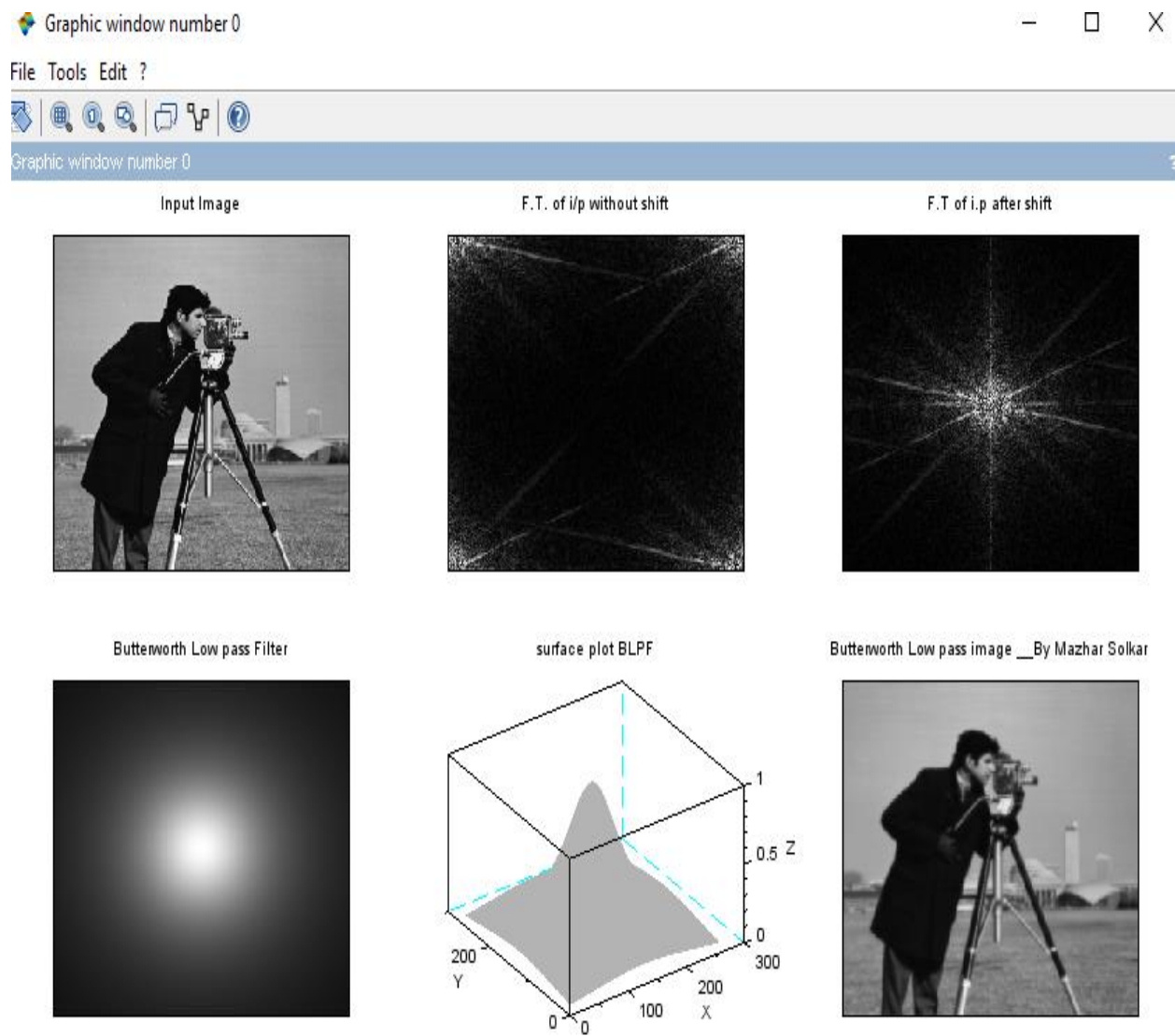


(ii) Butterworth Low pass filter (BLPF)

Code :-

```
File Edit Format Options Window Execute ?
pract_3_blpf.sce (C:\MazharSolkar\pract_3_blpf.sce) - SciNotes
pract_03_a.sce X pract_3_ilpf.sce X pract_3_blpf.sce X
1 clc;
2 clear;
3 a = imread('C:\Program-Files-(x86)\scilab-6.0.1\IPCV\images\cameraman.tif');
4 a = im2double(a);
5 subplot(2,3,1), imshow(a), title('Input-Image');
6
7 [m,n] = size(a);
8
9 A = fft2(a);
10 subplot(2,3,2), imshow(uint8(abs(A))), title('F.T.-of-i/p-without-shift');
11
12 A_shift = fftshift(A);
13 A_real = abs(A_shift);
14 subplot(2,3,3), imshow(uint8(A_real)), title('F.T.-of-i.p.-after-shift');
15
16 D0 = 50;
17 d = zeros(m,n);
18 order = 1;
19 for u = 1:m
20     for v = 1:n
21         d(u,v) = sqrt((u-(m/2))^2 + (v-(n/2))^2);
22         h(u,v) = 1 / (1 + (d(u,v)/D0)^(2*order));
23     end
24 end
25 subplot(2,3,4), imshow(h), title('Butterworth-Low-pass-Filter')
26 subplot(2,3,5), mesh(h), title('surface-plot-BLPF')
27
28 B = A_shift.*h;
29 B_inverse = ifft(B);
30 B_real = abs(B_inverse);
31 subplot(2,3,6), imshow(B_real), title('Butterworth-Low-pass-image-__By-Mazhar-Solkar')
32
```

Output :-



(iii) Gaussian Low pass Filter (GLPS)

Code :-

```
File Edit Format Options Window Execute ?
practical_03_b_iii_glpf.sce (C:\0_MSc_IT_Notes\Image Processing\Practicals\Practical-03\practical_03_b_iii_glpf.sce) - SciNotes
*practical_03_b_iii_glpf.sce

1 clc;
2 clear;
3 a = imread('C:\Program Files\scilab-6.1.1\IPCV\images\cameraman.tif');
4 a = im2double(a);
5 subplot(2,3,1), imshow(a), title('Input Image');
6
7 [m,n] = size(a);
8
9 A = fft2(a);
10 subplot(2,3,2), imshow(uint8(abs(A))), title('F.T. of i/p without shift');
11
12 A_shift = fftshift(A);
13 A_real = abs(A_shift);
14 subplot(2,3,3), imshow(uint8(A_real)), title('F.T. of i.p after shift');
15
16 D0 = 50;
17 d = zeros(m,n);
18 order = 1;
19 for u=1:m
20     for v=1:n
21         d = sqrt((u-(m/2)).^2 + (v-(n/2)).^2);
22         h(u,v) = exp(-(d^2)/(2*D0.^2));
23     end
24 end
25 subplot(2,3,4), imshow(h), title('Gaussian Low pass Filter')
26 subplot(2,3,5), mesh(h), title('surface plot-BLPF')
27
28 H_low = A_shift.*h;
29 H_low_shift = fftshift(H_low);
30 H_low_shift = ifft(H_low_shift);
31 B_real = abs(H_low_shift);
32 subplot(2,3,6), imshow(B_real), title('Gaussian Low pass image __By Mazhar Solkar')
23
```

Output :-

File Tools Edit ?



Graphic window number 0

?

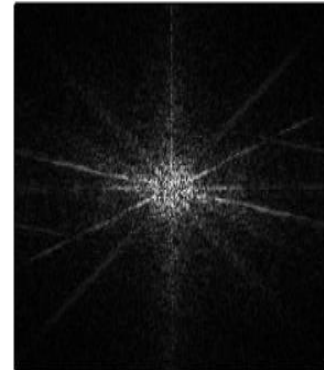
Input Image



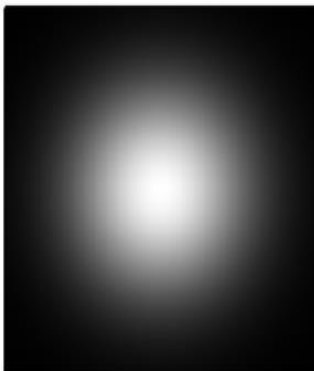
F.T. of i/p without shift



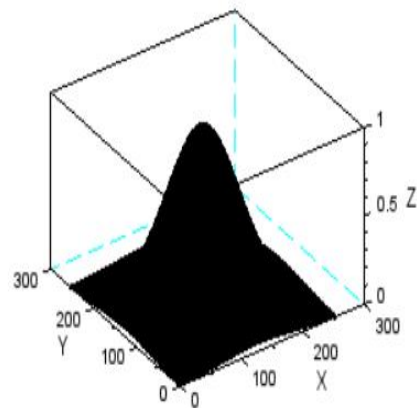
F.T of i.p after shift



Gaussian Low pass Filter



surface plot BLPF



Gaussian Low pass image __By Mazhar Solkar



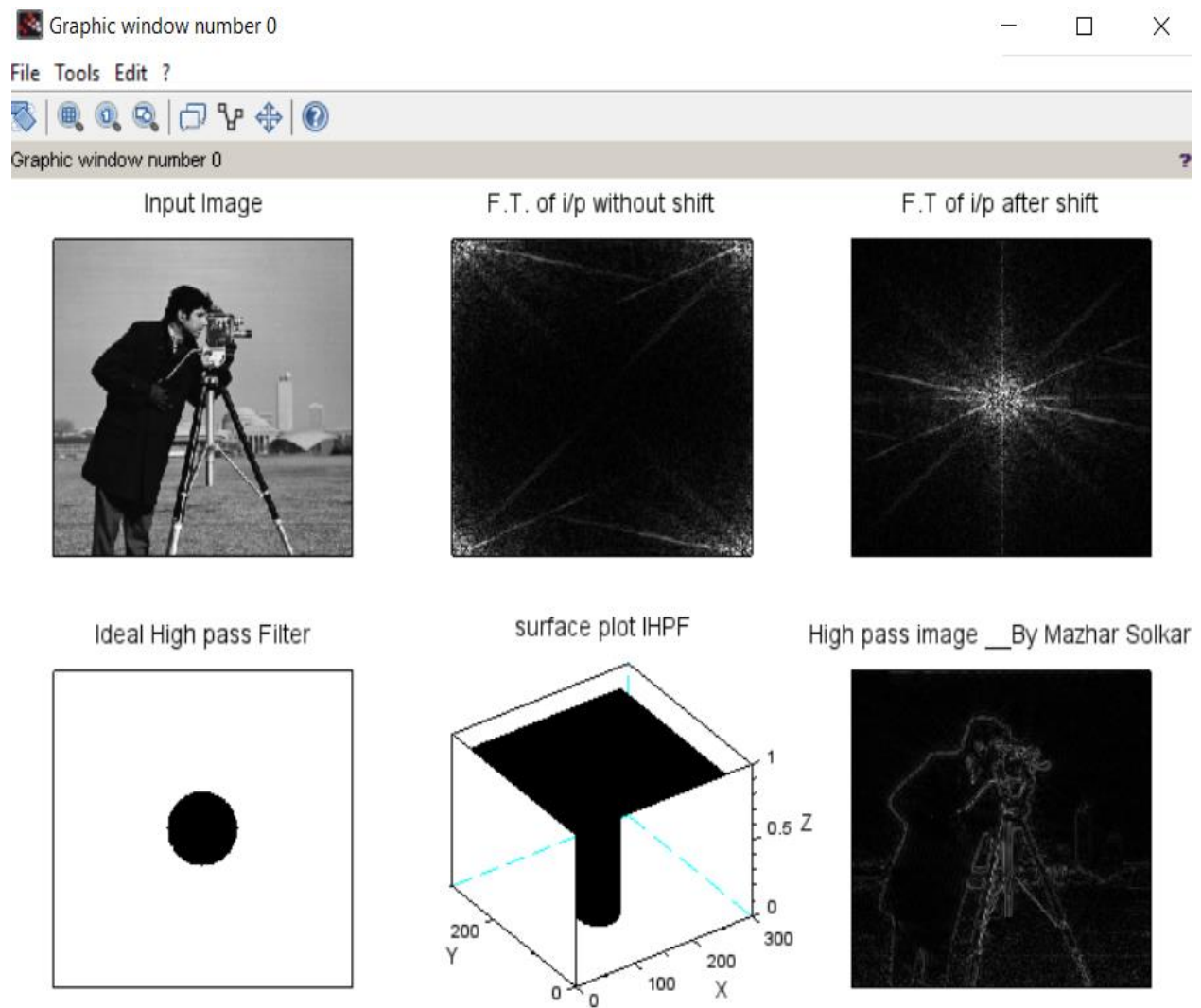
(iv) Ideal High pass Filter (IHPF)

Code :-

```
File Edit Format Options Window Execute ?
practical_03_B_iv_IHPF.sce (C:\0_MSc_IT_Notes\Image Processing\Practicals\Practical-03\Practical_03_B\practical_03_B_iv_IHPF.s
practical_03_b_iii_glpf.sce *practical_03_B_iv_IHPF.sce

1 clc;
2 clear;
3 a = imread('C:\Program Files\scilab-6.1.1\IPCV\images\cameraman.tif');
4 a = im2double(a);
5 subplot(2,3,1), imshow(a), title('Input Image');
6
7 [m,n] = size(a);
8 D0 = 30;
9 A = fft2(a);
10 subplot(2,3,2), imshow(uint8(abs(A))), title('F.T. of i/p without shift');
11
12 A_shift = fftshift(A);
13 A_real = abs(A_shift);
14 subplot(2,3,3), imshow(uint8(A_real)), title('F.T. of i/p after shift');
15 A_low = zeros(m,n);
16 d = zeros(m,n);
17 for u = 1:m
18     for v = 1:n
19         d(u,v) = sqrt((u-(m/2))^2 + (v-(n/2))^2);
20         if d(u,v) <= D0
21             A_high(u,v) = 0;
22             H(u,v) = 0;
23         else
24             A_high(u,v) = A_shift(u,v);
25             H(u,v) = 1;
26         end
27     end
28 end
29 subplot(2,3,4), imshow(H), title('Ideal High-pass Filter')
30 subplot(2,3,5), mesh(H), title('surface plot IHPF')
31 B = fftshift(A_high);
32 B_inverse = ifft(B);
33 B_real = abs(B_inverse);
34 subplot(2,3,6), imshow(B_real), title('High-pass image __By Mazhar Solkar')
```


Output :-



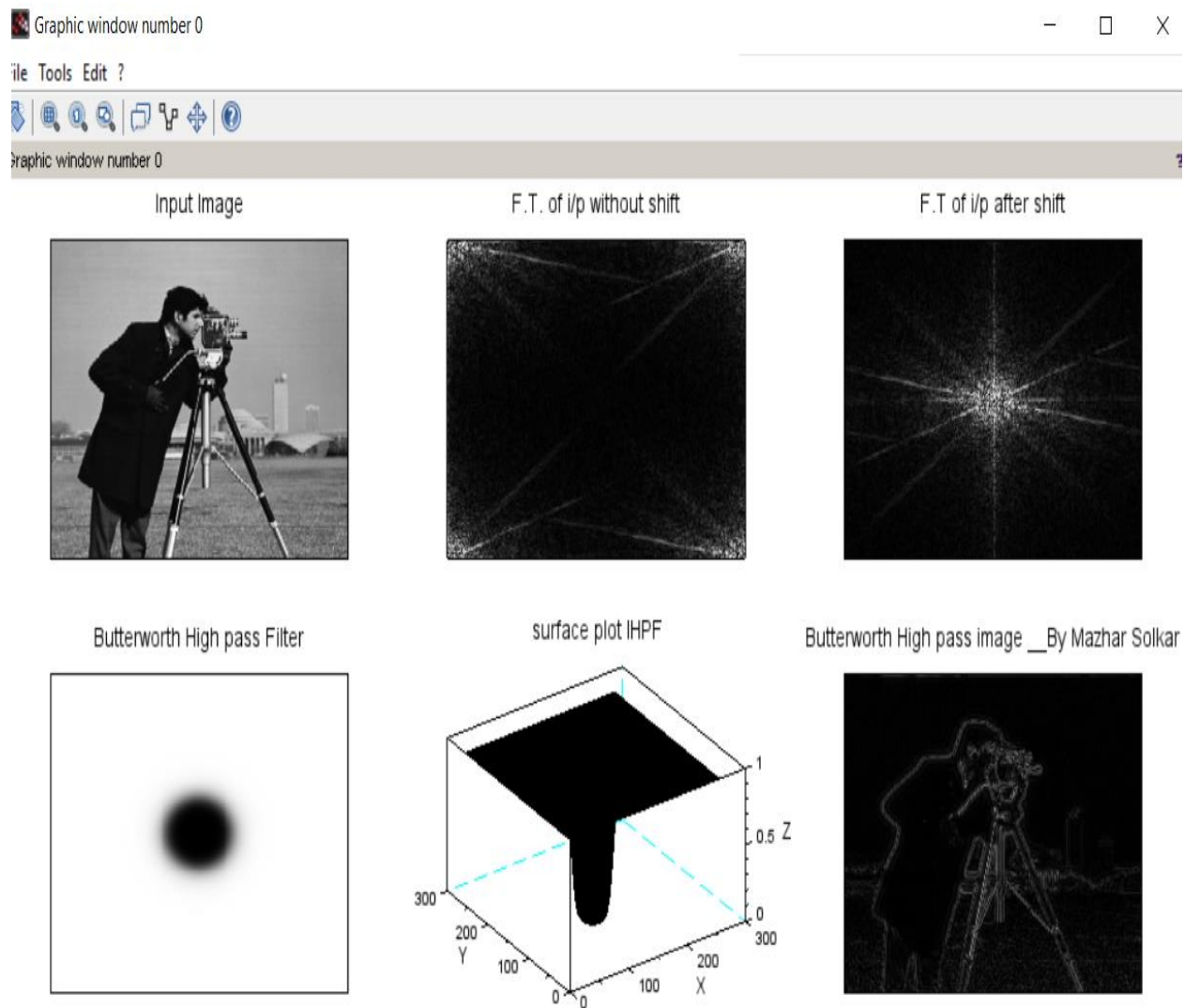
(v) Butterworth High pass Filter (BHPF)

Code :-

```
File Edit Format Options Window Execute ?
practical_03_B_v_BHPF.sce (C:\0_MSc_IT_Notes\Image Processing\Practicals\Practical-03\Practical_03_B\practical_03_B_v_BHPF.sce) - SciNotes
practical_03_B_iv_IHPF.sce practical_03_B_v_BHPF.sce

1 clc;
2 clear;
3 a = imread('C:\Program Files\scilab-6.1.1\IPCV\images\cameraman.tif');
4 a = im2double(a);
5 subplot(2,3,1), imshow(a), title('Input Image');
6
7 [m,n] = size(a);
8
9 A = fft2(a);
10 subplot(2,3,2), imshow(uint8(abs(A))), title('F.T. of i/p without shift');
11
12 A_shift = fftshift(A);
13 A_real = abs(A_shift);
14 subplot(2,3,3), imshow(uint8(A_real)), title('F.T. of i/p after shift');
15
16 D0 = 30;
17 d = zeros(m,n);
18 order = 4;
19 for u = 1:m
20     for v = 1:n
21         d(u,v) = sqrt((u-(m/2))^2 + (v-(n/2))^2);
22         h(u,v) = 1 / (1 + (D0/d(u,v))^(2*order));
23     end
24 end
25 subplot(2,3,4), imshow(h), title('Butterworth High-pass Filter')
26 subplot(2,3,5), mesh(h), title('surface plot IHPF')
27
28 B = A_shift.*h;
29 B_inverse = ifft(B);
30 B_real = abs(B_inverse);
31 subplot(2,3,6), imshow(B_real), title('Butterworth High-pass image __By Mazhar Solkar')
32
```

Output :-

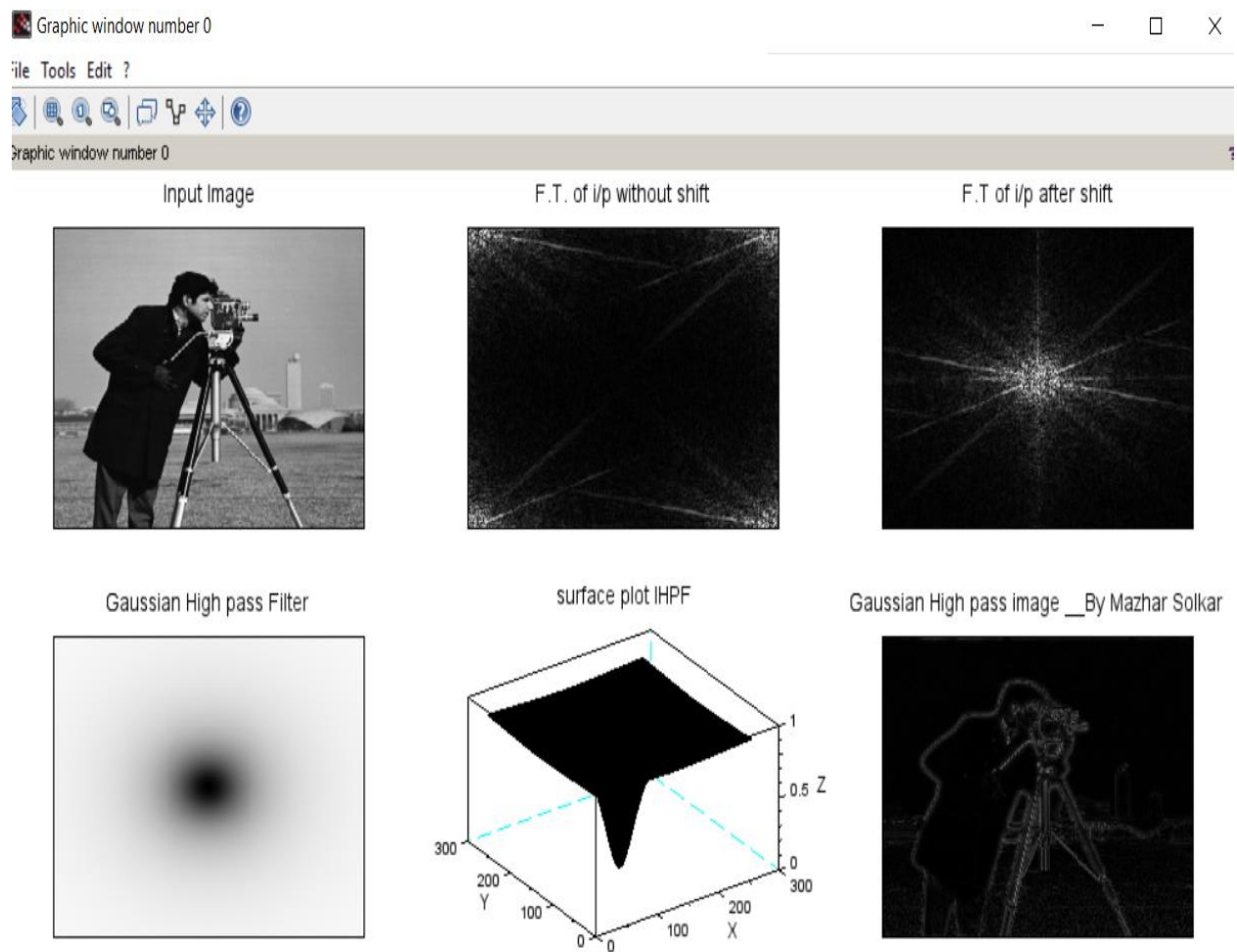


(vi) Gaussian High pass Filter

Code :-

```
File Edit Format Options Window Execute ?
practical_03_B_vi_GHPF.sce (C:\0_MSc_IT_Notes\Image Processing\Practicals\Practical-03\Practical_03_B\practical_03_B_vi_GHPF.sce) - SciN
practical_03_B_iv_IHPF.sce practical_03_B_v_BHPF.sce practical_03_B_vi_GHPF.sce
1 clc;
2 clear;
3 a = imread('C:\Program Files\scilab-6.1.1\IPCV\images\cameraman.tif');
4 a = im2double(a);
5 subplot(2,3,1), imshow(a), title('Input Image');
6
7 [m,n] = size(a);
8
9 A = fft2(a);
10 subplot(2,3,2), imshow(uint8(abs(A))), title('F.T. of i/p without shift');
11
12 A_shift = fftshift(A);
13 A_real = abs(A_shift);
14 subplot(2,3,3), imshow(uint8(A_real)), title('F.T. of i/p after shift');
15
16 D0 = 30;
17 d = zeros(m,n);
18 order = 1;
19 for u = 1:m
20     for v = 1:n
21         d(u,v) = sqrt((u-(m/2))^2 + (v-(n/2))^2);
22         h(u,v) = 1 / (1 + (D0/d(u,v))^(2*order));
23     end
24 end
25 subplot(2,3,4), imshow(h), title('Gaussian High-pass Filter')
26 subplot(2,3,5), mesh(h), title('surface plot IHPF')
27
28 H_high = A_shift.*h;
29 H_high_shift = fftshift(H_high);
30 H_high_shift = ifft(H_high_shift);
31 B_real = abs(H_high_shift);
32 subplot(2,3,6), imshow(B_real), title('Gaussian High-pass image __By Mazhar Solkar')
33
```

Output :-



(c) Program to apply Laplacian filter and high boost in frequency domain.

```
File Edit Format Options Window Execute ?
practical_03_c.sce (C:\O_MSc_IT_Notes\image Processing\Practicals\Practical-03\Practical_03_C\practical_03_c.sce) - SciNotes
practical_03_c.sce

1  clc;
2  clear all;
3  a = imread('C:\Program Files\scilab-6.1.1\IPCV\images\moon.tif');
4  a = im2double(a);
5  subplot(2,2,1), imshow(a), title('Input Image');
6  [m,n] = size(a);
7  D0 = 50;
8  A = fft2(a);
9  A_shift = fftshift(A);
10 A_real = abs(A_shift);
11 H = zeros(m,n);
12 D = zeros(m,n);
13 for u=1:m
14     for v=1:n
15         D(u,v)=sqrt((u-(m/2))^2+(v-(n/2))^2);
16         if D(u,v) <= D0
17             H(u,v) = 0;
18         else
19             H(u,v) = 1;
20         end
21     end
22 end
23
24 AHB = 2.0;
25 H1 = (AHB-1)+H;
26 X = A_shift.*H;
27 X1 = A_shift.*H;
28 XA = abs(iffth(X));
29 XB = abs(iffth(X1));
30
31 subplot(2,2,2), imshow(XA), title('High-pass image');
32 subplot(2,2,3), imshow(XB), title('High-boost image');
33 subplot(2,2,4), imshow(a+XA), title('Input + High-pass = Laplacian Image')
34
```


Output :-

File Tools Edit ?

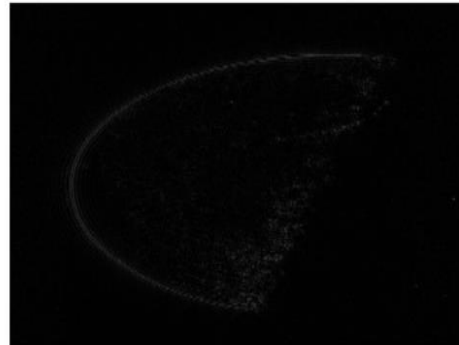


Graphic window number 0

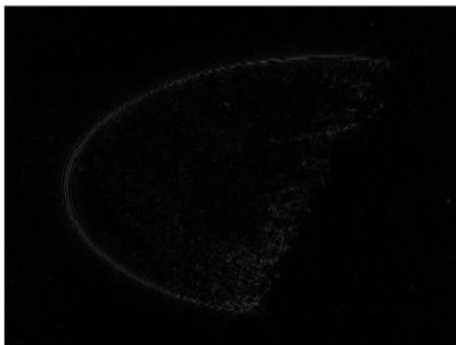
Input Image



High pass image



High boost image



Input + High pass = Laplacian Image __By Mazhar Solkar



(d) Program for Harmonic Filtering.

Code :-

```
File Edit Format Options Window Execute ?
practical_03_Homomorphic.sce (C:\0_MSc_IT_Notes\Image Processing\Practicals\Practical-03\Practical_03_B\practical_03_Homomorphic.sce)
practical_03_c.sce practical_03_Homomorphic.sce
1 //
2 clear all;
3 a = imread('C:\Program Files\scilab-6.1.1\IPCV\images\moon.tif');
4 subplot(2,3,1), imshow(a), title('Input Image');
5 a = double(a);
6 b = a;
7
8 D0 = 50;
9 GL = 0.9;
10 GH = 1.9;
11 [m,n] = size(a);
12
13 b = b+1;
14 log_b = log(b);
15 subplot(2,3,2), imshow(log_b), title('Natural Logarithm');
16 c = fft2(log_b);
17 subplot(2,3,3), imshow(uint8(c), title('Fourier Transform'));
18 dd = fftshift(c);
19 for u=1:m
20     for v=1:n
21         H(u,v) = (GH - GL) * (1 - exp(-1 * (sqrt((u-m/2)^2 + (v-n/2)^2))^2 / D0)^2) + GL;
22     end
23 end
24 subplot(2,3,4);
25 mesh(H);
26 title('Homomorphic filter');
27
28 x = dd.*H;
29 real_x = abs(iff2(x));
30 subplot(2,3,5), imshow(real_x), title('Inverse fourier transform');
31
32 Final = exp(real_x);
33 subplot(2,3,6), imshow(uint8(Final)), title('Filtered Image - __By Mazhar Solkar');
34
```

Output :-

