

# EXPERINMENT 9

**Aim :** Apply Different filter on Fourier transform of image and convert back to spatial domain.

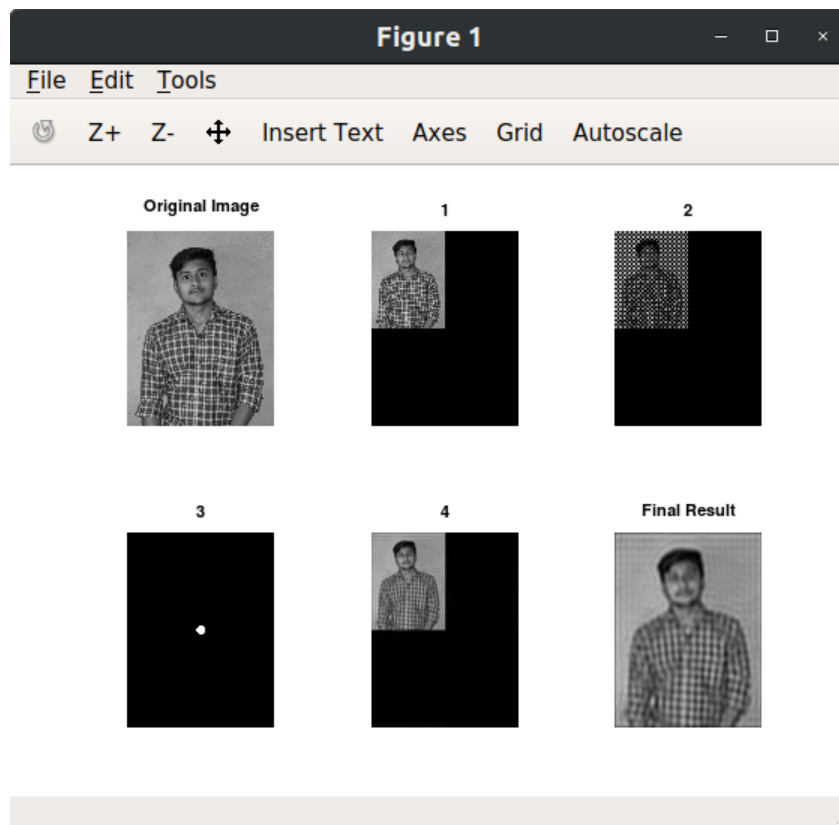
## ❖ Exercises :

### 1. Ideal low pass filter.

Code :

```
1 pkg load image;
2 r = rgb2gray(imread('/home/nihar/Desktop/SEM 7/IP/Lab/Lab6/my.jpeg'));
3
4 subplot(2,3,1);imshow(r);title("Original Image");
5 [m,n]=size(r);
6 M=2*m;
7 N=2*n;
8 pad=zeros(M,N);
9 pad(1:m,1:n)=r;
10 subplot(2,3,2);imshow(uint8(pad));title("1");
11
12 for i=1:M
13     for j=1:N
14         pad(i,j)=pad(i,j)*(-1)^(i-1+j-1);
15     endfor
16 endfor
17
18 subplot(2,3,3);imshow(uint8(pad));title("2");
19
20 F = fft2(pad);
21 H = zeros(M,N);
22 D0=50;
23 for i=1:M
24     for j=1:N
25         D=sqrt((i-M/2)^2+(j-N/2)^2);
26         if(D<=D0)
27             H(i,j)=1;
28         else
29             H(i,j)=0;
30         endif
31     endfor
32 endfor
33
34 subplot(2,3,4);imshow(H);title("3");
35
36 G=H.*F;
37 Gp = real(ifft2(G));
38 for i=1:M
39     for j=1:N
40         st(i,j)=Gp(i,j)*(-1)^(i-1+j-1);
41     endfor
42 endfor
43
44 subplot(2,3,5);imshow(uint8(st));title("4");
45
46 subplot(2,3,6);s=st(1:m,1:n);
47 imshow(uint8(s));title("Final Result");
```

Output :

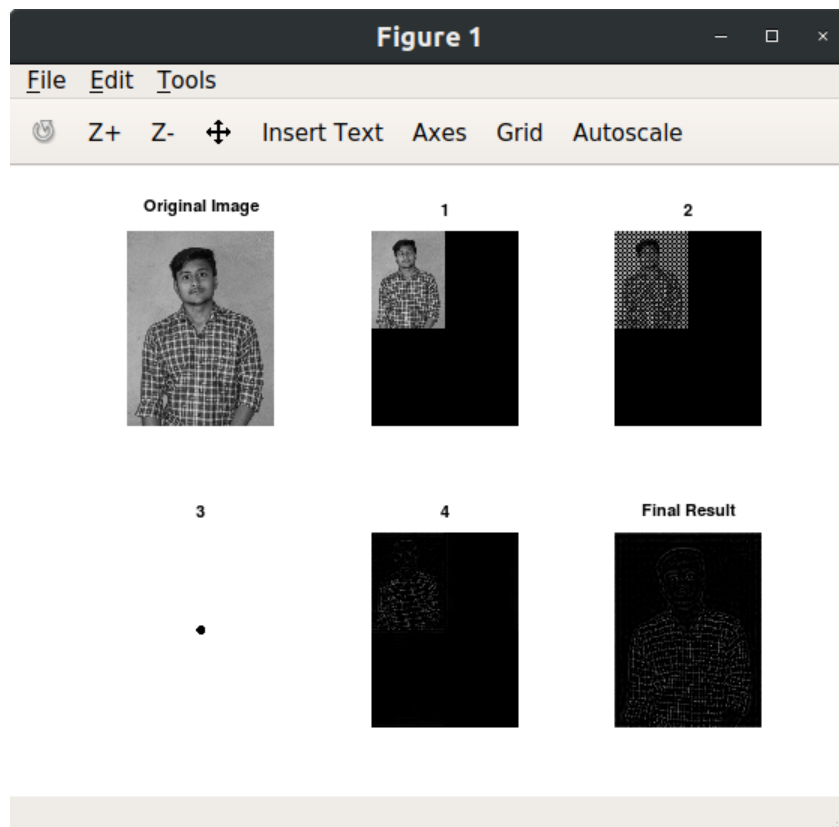


## 2. Ideal high pass filter.

Code :

```
1 pkg load image;
2 r = rgb2gray(imread('/home/nihar/Desktop/SEM 7/IP/Lab/Lab6/my.jpeg'));
3
4 subplot(2,3,1);imshow(r);title("Original Image");
5
6 [m,n]=size(r);
7 M=2*m;
8 N=2*n;
9 pad=zeros(M,N);
10 pad(1:m,1:n)=r;
11
12 subplot(2,3,2);imshow(uint8(pad));title("1");
13
14 for i=1:M
15     for j=1:N
16         pad(i,j)=pad(i,j)*(-1)^(i-1+j-1);
17     endfor
18 endfor
19
20 subplot(2,3,3);imshow(uint8(pad));title("2");
21
22 F = fft2(pad);
23 H = zeros(M,N);
24 D0=50;
25 for i=1:M
26     for j=1:N
27         D=sqrt((i-M/2)^2+(j-N/2)^2);
28         if(D<=D0)
29             H(i,j)=0;
30         else
31             H(i,j)=1;
32         endif
33     endfor
34 endfor
35
36 subplot(2,3,4);imshow(H);title("3");
37
38 G=H.*F;
39 Gp = real(ifft2(G));
40 for i=1:M
41     for j=1:N
42         st(i,j)=Gp(i,j)*(-1)^(i-1+j-1);
43     endfor
44 endfor
45
46 subplot(2,3,5);imshow(uint8(st));title("4");
47
48 subplot(2,3,6);s=st(1:m,1:n);
49 imshow(uint8(s));title("Final Result");
```

Output :

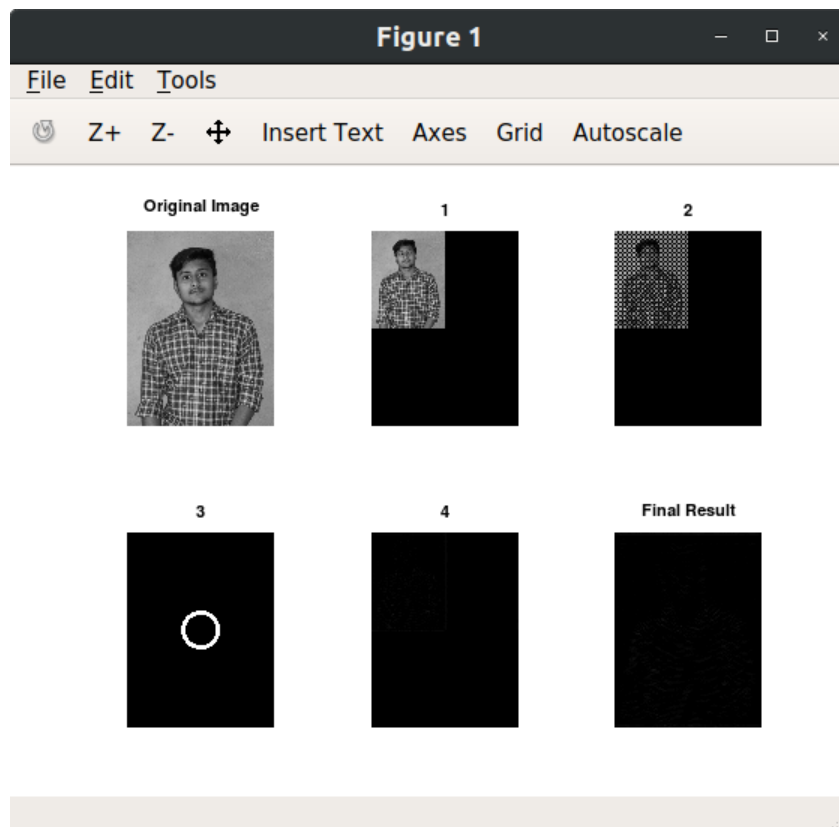


### 3. Ideal band pass filter.

Code :

```
1 pkg load image;
2 r = rgb2gray(imread('/home/nihar/Desktop/SEM 7/IP/Lab/Lab6/my.jpeg'));
3
4 subplot(2,3,1);
5 imshow(r);
6 title("Original Image");
7
8 [m,n]=size(r);
9 M=2*m;
10 N=2*n;
11 pad=zeros(M,N);
12 pad(1:m,1:n)=r;
13
14 subplot(2,3,2);
15 imshow(uint8(pad));
16 title("1");
17
18 for i=1:M
19     for j=1:N
20         pad(i,j)=pad(i,j)*(-1)^(i-1+j-1);
21     endfor
22 endfor
23
24 subplot(2,3,3);imshow(uint8(pad));title("2");
25
26 F = fft2(pad);
27 H = zeros(M,N);
28 D0=200;
29 W=50;
30 for i=1:M
31     for j=1:N
32         D=sqrt((i-M/2)^2+(j-N/2)^2);
33         if(D0-(W/2)<=D && D<=D0+(W/2))
34             H(i,j)=0;
35         else
36             H(i,j)=1;
37         endif
38     endfor
39 endfor
40 H=not(H);
41
42 subplot(2,3,4);imshow(H);title("3");
43 G=H.*F;
44 Gp = real(ifft2(G));
45 for i=1:M
46     for j=1:N
47         st(i,j)=Gp(i,j)*(-1)^(i-1+j-1);
48     endfor
49 endfor
50
51 subplot(2,3,5);imshow(uint8(st));title("4");
52
53 subplot(2,3,6);s=st(1:m,1:n);
54 imshow(uint8(s));title("Final Result");
```

Output :

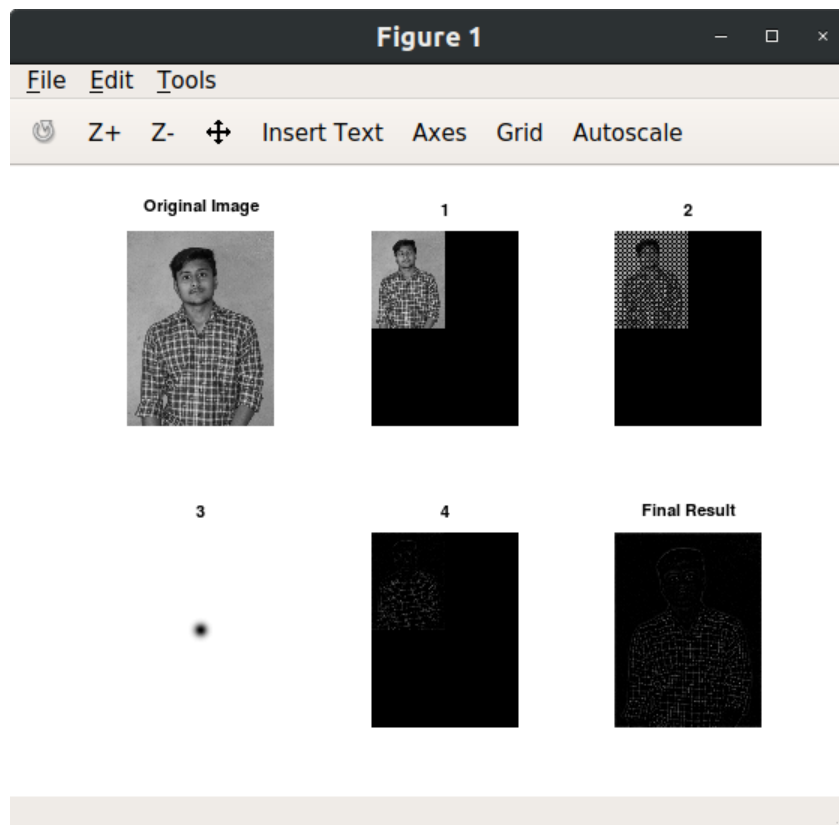


#### 4. Guassian High Pass filter.

Code :

```
1 pkg load image;
2 r = rgb2gray(imread('/home/nihar/Desktop/SEM 7/IP/Lab/Lab6/my.jpeg'));
3
4 subplot(2,3,1);
5 imshow(r);
6 title("Original Image");
7
8 [m,n]=size(r);
9 M=2*m;
10 N=2*n;
11 pad=zeros(M,N);
12 pad(1:m,1:n)=r;
13
14 subplot(2,3,2);
15 imshow(uint8(pad));
16 title("1");
17
18 for i=1:M
19     for j=1:N
20         pad(i,j)=pad(i,j)*(-1)^(i-1+j-1);
21     endfor
22 endfor
23
24 subplot(2,3,3);imshow(uint8(pad));title("2");
25
26 F = fft2(pad);
27 H = zeros(M,N);
28 D0=50;
29 for i=1:M
30     for j=1:N
31         D=sqrt((i-M/2)^2+(j-N/2)^2);
32         term = -((D*D)/(2*D0*D0));
33         H(i,j)=1-(power(e,term));
34     endfor
35 endfor
36
37 subplot(2,3,4);imshow(H);title("3");
38
39 G=H.*F;
40 Gp = real(ifft2(G));
41 for i=1:M
42     for j=1:N
43         st(i,j)=Gp(i,j)*(-1)^(i-1+j-1);
44     endfor
45 endfor
46
47 subplot(2,3,5);imshow(uint8(st));title("4");
48
49 subplot(2,3,6);s=st(1:m,1:n);
50 imshow(uint8(s));title("Final Result");
```

Output :



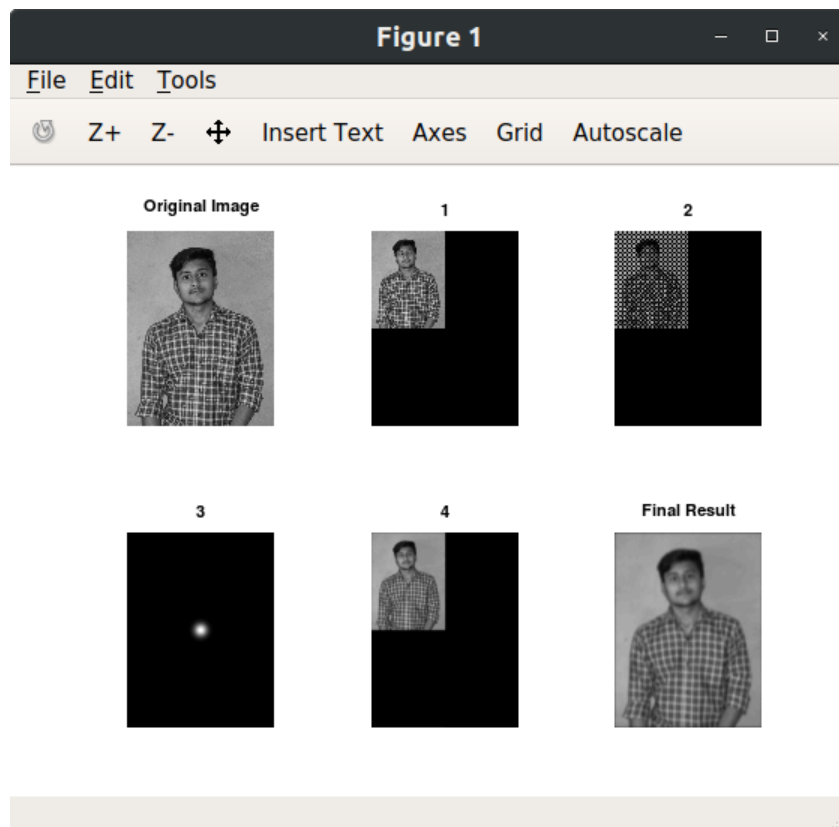


## 5. Guassian Low pass filter.

Code :

```
1 pkg load image;
2 r = rgb2gray(imread('/home/nihar/Desktop/SEM 7/IP/Lab/Lab6/my.jpeg'));
3
4 subplot(2,3,1);
5 imshow(r);
6 title("Original Image");
7
8 [m,n]=size(r);
9 M=2*m;
10 N=2*n;
11 pad=zeros(M,N);
12 pad(1:m,1:n)=r;
13
14 subplot(2,3,2);
15 imshow(uint8(pad));
16 title("1");
17
18 for i=1:M
19     for j=1:N
20         pad(i,j)=pad(i,j)*(-1)^(i-1+j-1);
21     endfor
22 endfor
23
24 subplot(2,3,3);imshow(uint8(pad));title("2");
25
26 F = fft2(pad);
27 H = zeros(M,N);
28 D0=50;
29 for i=1:M
30     for j=1:N
31         D=sqrt((i-M/2)^2+(j-N/2)^2);
32         term = -((D*D)/(2*D0*D0));
33         H(i,j)=power(e,term);
34     endfor
35 endfor
36
37 subplot(2,3,4);imshow(H);title("3");
38
39 G=H.*F;
40 Gp = real(ifft2(G));
41 for i=1:M
42     for j=1:N
43         st(i,j)=Gp(i,j)*(-1)^(i-1+j-1);
44     endfor
45 endfor
46
47 subplot(2,3,5);imshow(uint8(st));title("4");
48
49 subplot(2,3,6);s=st(1:m,1:n);
50 imshow(uint8(s));title("Final Result");
51
```

Output :

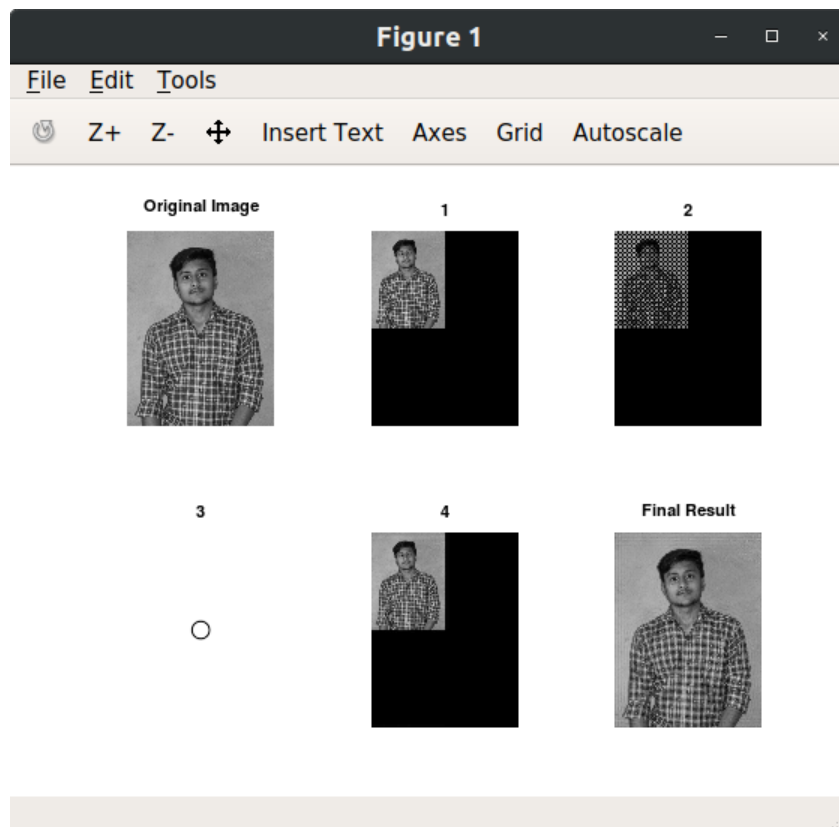


## 6. Gaussian band reject filter.

Code :

```
1 pkg load image;
2 r = rgb2gray(imread('/home/nihar/Desktop/SEM 7/IP/Lab/Lab6/my.jpeg'));
3
4 subplot(2,3,1);
5 imshow(r);
6 title("Original Image");
7
8 [m,n]=size(r);
9 M=2*m;
10 N=2*n;
11 pad=zeros(M,N);
12 pad(1:m,1:n)=r;
13
14 subplot(2,3,2);
15 imshow(uint8(pad));
16 title("1");
17
18 for i=1:M
19     for j=1:N
20         pad(i,j)=pad(i,j)*(-1)^(i-1+j-1);
21     endfor
22 endfor
23
24 subplot(2,3,3);imshow(uint8(pad));title("2");
25
26 F = fft2(pad);
27 H = zeros(M,N);
28 D0=100;
29 W=20;
30 for i=1:M
31     for j=1:N
32         D=sqrt((i-M/2)^2+(j-N/2)^2);
33         term = -power((((D*D)-(D0*D0))/(D*W)),2);
34         H(i,j)=1-power(e,term);
35     endfor
36 endfor
37
38 subplot(2,3,4);imshow(H);title("3");
39
40 G=H.*F;
41 Gp = real(ifft2(G));
42 for i=1:M
43     for j=1:N
44         st(i,j)=Gp(i,j)*(-1)^(i-1+j-1);
45     endfor
46 endfor
47
48 subplot(2,3,5);imshow(uint8(st));title("4");
49
50 subplot(2,3,6);s=st(1:m,1:n);
51 imshow(uint8(s));title("Final Result");
52
```

Output :

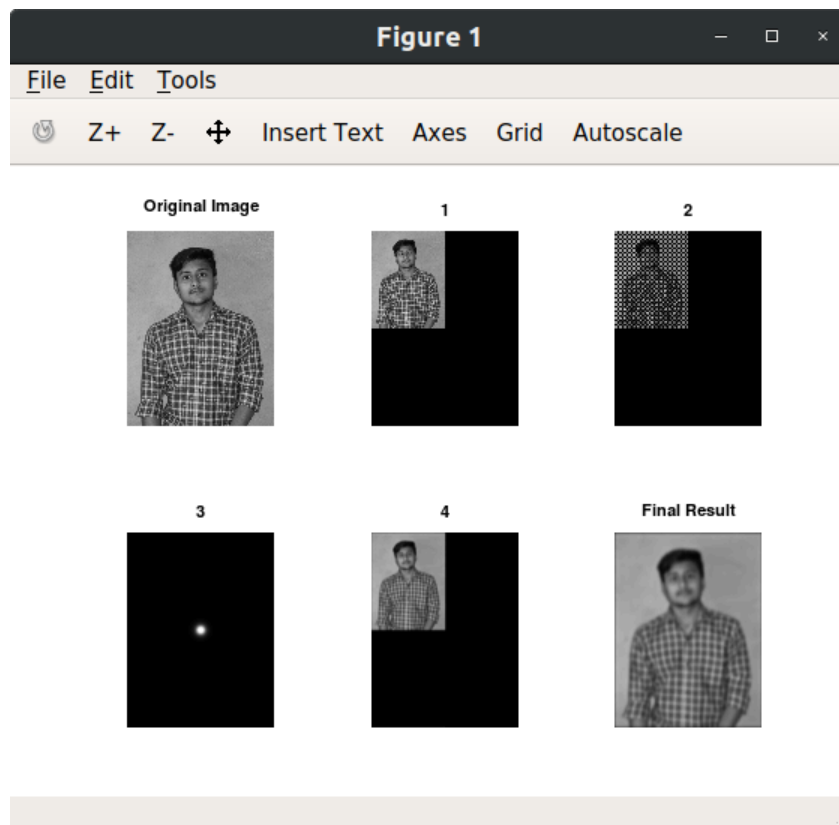


## 7. Butterworth low pass filter.

Code :

```
1 pkg load image;
2 r = rgb2gray(imread('/home/nihar/Desktop/SEM 7/IP/Lab/Lab6/my.jpeg'));
3 subplot(2,3,1);
4 imshow(r);
5 title("Original Image");
6
7 [m,n]=size(r);
8 M=2*m;
9 N=2*n;
10 pad=zeros(M,N);
11 pad(1:m,1:n)=r;
12
13 subplot(2,3,2);
14 imshow(uint8(pad));
15 title("1");
16
17 for i=1:M
18     for j=1:N
19         pad(i,j)=pad(i,j)*(-1)^(i-1+j-1);
20     endfor
21 endfor
22
23 subplot(2,3,3);imshow(uint8(pad));title("2");
24
25 F = fft2(pad);
26 H = zeros(M,N);
27 D0=50;
28 order=2;
29 for i=1:M
30     for j=1:N
31         D=sqrt((i-M/2)^2+(j-N/2)^2);
32         H(i,j)=1/(1+power((D/D0),2*order));
33     endfor
34 endfor
35
36 subplot(2,3,4);imshow(H);title("3");
37
38 G=H.*F;
39 Gp = real(ifft2(G));
40 for i=1:M
41     for j=1:N
42         st(i,j)=Gp(i,j)*(-1)^(i-1+j-1);
43     endfor
44 endfor
45
46 subplot(2,3,5);imshow(uint8(st));title("4");
47
48 subplot(2,3,6);s=st(1:m,1:n);
49 imshow(uint8(s));title("Final Result");
50
```

Output :

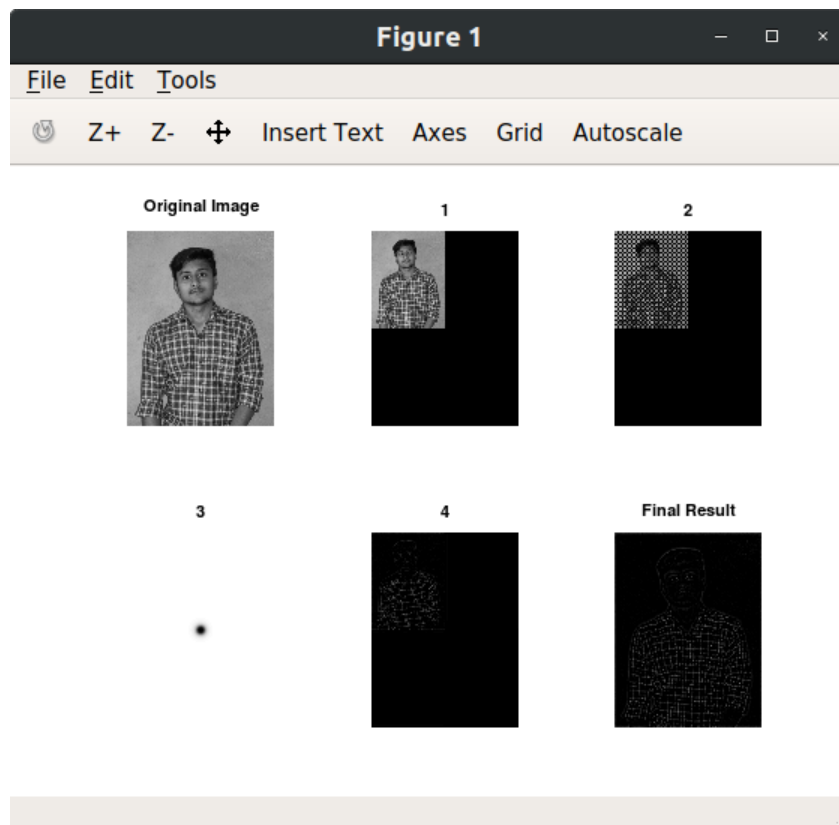


## 8. Butterworth high pass filter.

Code :

```
1 pkg load image;
2 r = rgb2gray(imread('/home/nihar/Desktop/SEM 7/IP/Lab/Lab6/my.jpeg'));
3
4 subplot(2,3,1);
5 imshow(r);
6 title("Original Image");
7
8 [m,n]=size(r);
9 M=2*m;
10 N=2*n;
11 pad=zeros(M,N);
12 pad(1:m,1:n)=r;
13
14 subplot(2,3,2);
15 imshow(uint8(pad));
16 title("1");
17
18 for i=1:M
19     for j=1:N
20         pad(i,j)=pad(i,j)*(-1)^(i-1+j-1);
21     endfor
22 endfor
23
24 subplot(2,3,3);imshow(uint8(pad));title("2");
25
26 F = fft2(pad);
27 H = zeros(M,N);
28 D0=50;
29 order=2;
30 for i=1:M
31     for j=1:N
32         D=sqrt((i-M/2)^2+(j-N/2)^2);
33         H(i,j)=1/(1+power((D0/D),2*order));
34     endfor
35 endfor
36
37 subplot(2,3,4);imshow(H);title("3");
38
39 G=H.*F;
40 Gp = real(ifft2(G));
41 for i=1:M
42     for j=1:N
43         st(i,j)=Gp(i,j)*(-1)^(i-1+j-1);
44     endfor
45 endfor
46
47 subplot(2,3,5);imshow(uint8(st));title("4");
48
49 subplot(2,3,6);s=st(1:m,1:n);
50 imshow(uint8(s));title("Final Result");
51
```

Output :



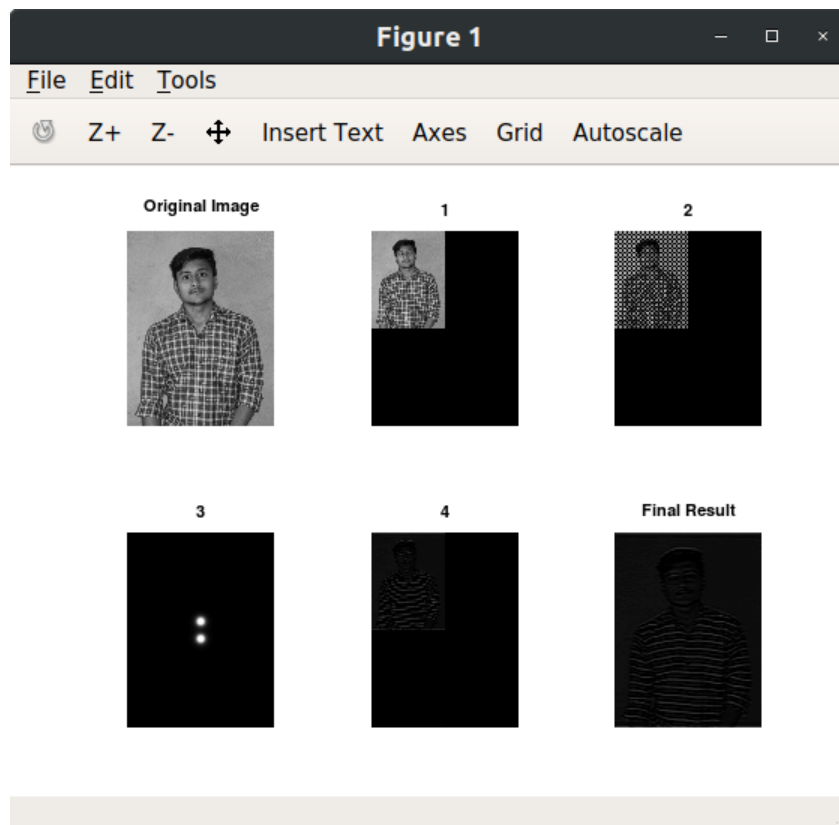


## 9. Notch pass filter.

Code :

```
1 pkg load image;
2 r = rgb2gray(imread('/home/nihar/Desktop/SEM 7/IP/Lab/Lab6/my.jpeg'));
3 subplot(2,3,1);
4 imshow(r);
5 title("Original Image");
6
7 [m,n]=size(r);
8 M=2*m;
9 N=2*n;
10 pad=zeros(M,N);
11 pad(1:m,1:n)=r;
12
13 subplot(2,3,2);
14 imshow(uint8(pad));
15 title("1");
16
17 for i=1:M
18     for j=1:N
19         pad(i,j)=pad(i,j)*(-1)^(i-1+j-1);
20     endfor
21 endfor
22
23 subplot(2,3,3);imshow(uint8(pad));title("2");
24
25 F = fft2(pad);
26 H = zeros(M,N);
27 D0=40;
28 order=2;
29 u=[100,-100];
30 v=[0,0];
31 for i=1:M,
32     for j=1:N,
33         H(i,j)=1;
34         for k =1:2,
35             D1=sqrt((i-(M/2)-u(k))^2+(j-(N/2)-v(k))^2);
36             D2=sqrt((i-(M/2)+u(k))^2+(j-(N/2)+v(k))^2);
37             t1=1/(1+power(D0/D1,2*order));
38             t2=1/(1+power(D0/D2,2*order));
39             H(i,j)=H(i,j)*t1*t2;
40         endfor
41     endfor
42 endfor
43 H = 1-H;
44
45 subplot(2,3,4);imshow(H);title("3");
46
47 G=H.*F;
48 Gp = real(iff2(G));
49 for i=1:M,
50     for j=1:N,
51         st(i,j)=Gp(i,j)*(-1)^(i-1+j-1);
52     endfor
53 endfor
54
55 subplot(2,3,5);imshow(uint8(st));title("4");
56
57 subplot(2,3,6);s=st(1:m,1:n);
58 imshow(uint8(s));title("Final Result");
59
```

Output :

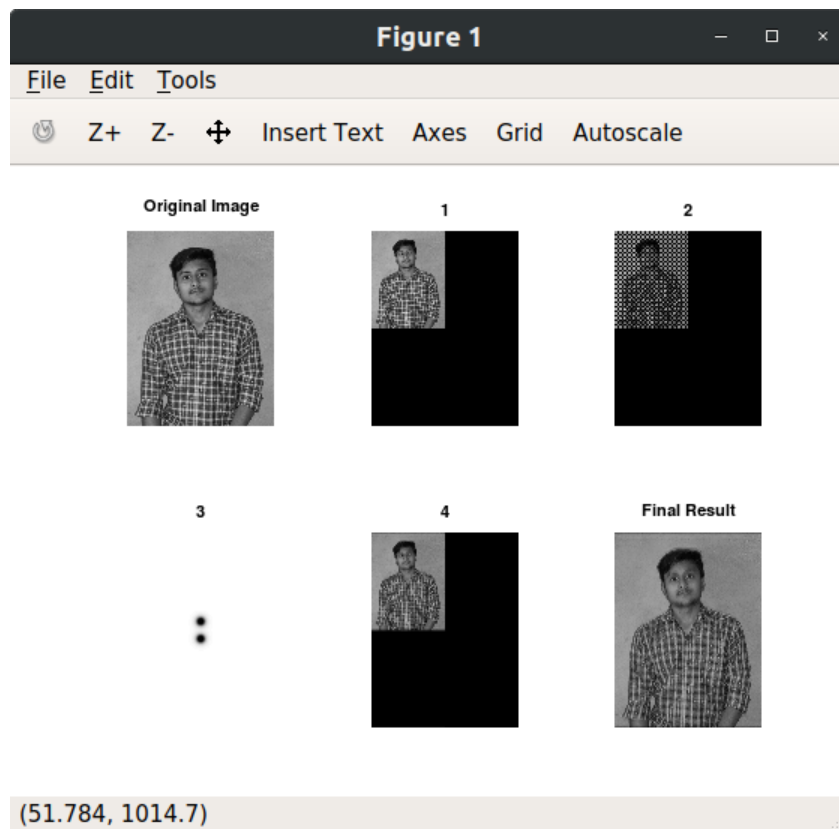


## 10. Notch reject filter.

Code :

```
1 pkg load image;
2 r = rgb2gray(imread('/home/nihar/Desktop/SEM 7/IP/Lab/Lab6/my.jpeg'));
3
4 subplot(2,3,1);
5 imshow(r);
6 title("Original Image");
7
8 [m,n]=size(r);
9 M=2*m;
10 N=2*n;
11 pad=zeros(M,N);
12 pad(1:m,1:n)=r;
13
14 subplot(2,3,2);
15 imshow(uint8(pad));
16 title("1");
17
18 for i=1:M
19     for j=1:N
20         pad(i,j)=pad(i,j)*(-1)^(i-1+j-1);
21     endfor
22 endfor
23
24 subplot(2,3,3);imshow(uint8(pad));title("2");
25
26 F = fft2(pad);
27 H = zeros(M,N);
28 D0=40;
29 order=2;
30 u=[100,-100];
31 v=[0,0];
32 for i=1:M,
33     for j=1:N,
34         H(i,j)=1;
35         for k =1:2,
36             D1=sqrt((i-(M/2)-u(k))^2+(j-(N/2)-v(k))^2);
37             D2=sqrt((i-(M/2)+u(k))^2+(j-(N/2)+v(k))^2);
38             t1=1/(1+power(D0/D1,2*order));
39             t2=1/(1+power(D0/D2,2*order));
40             H(i,j)=H(i,j)*t1*t2;
41         endfor
42     endfor
43 endfor
44
45 subplot(2,3,4);imshow(H);title("3");
46
47 G=H.*F;
48 Gp = real(iff2(G));
49 for i=1:M
50     for j=1:N
51         st(i,j)=Gp(i,j)*(-1)^(i-1+j-1);
52     endfor
53 endfor
54
55 subplot(2,3,5);imshow(uint8(st));title("4");
56
57 subplot(2,3,6);s=st(1:m,1:n);
58 imshow(uint8(s));title("Final Result");
59
```

Output :



## ❖ Final Conclusion :

For Ideal Low Pass Filter:

- Blurring effect which decreases as the cutoff frequency increases.
- Ringing effect which also decreases as the cutoff frequency increases.

For Ideal High Pass Filter:

- Ringing effect.
- Edge distortion (i.e. distorted, thickened object boundaries).
- Both effects are decreased as the cutoff frequency increases.

For Gaussian Low Pass Filter:

- Smooth transition in blurring as a function of increasing cutoff frequency.
- No ringing effect.

For Gaussian High Pass Filter:

- No ringing effect.
- Less edge distortion.
- The results are smoother than those obtained by IHPF.

For Butterworth Low Pass Filter:

- Smooth transition in blurring as a function of increasing cutoff frequency.
- Ringing effect in the BLPF becomes significant as the nth order increases.

For Butterworth High Pass Filter:

- Results are much smoother than obtained with an IHPF.