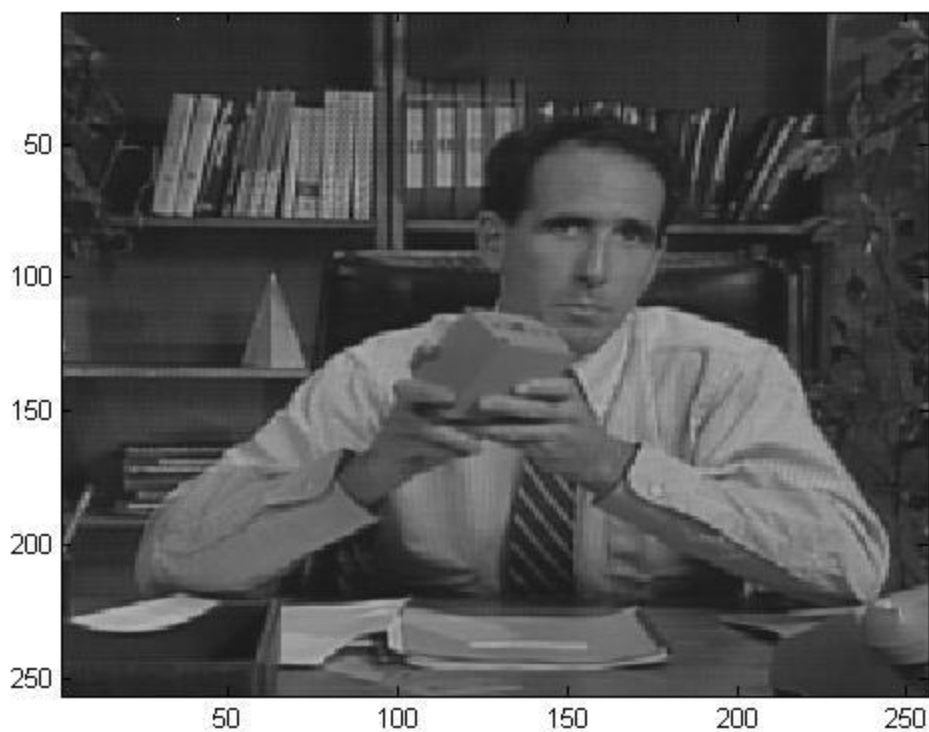
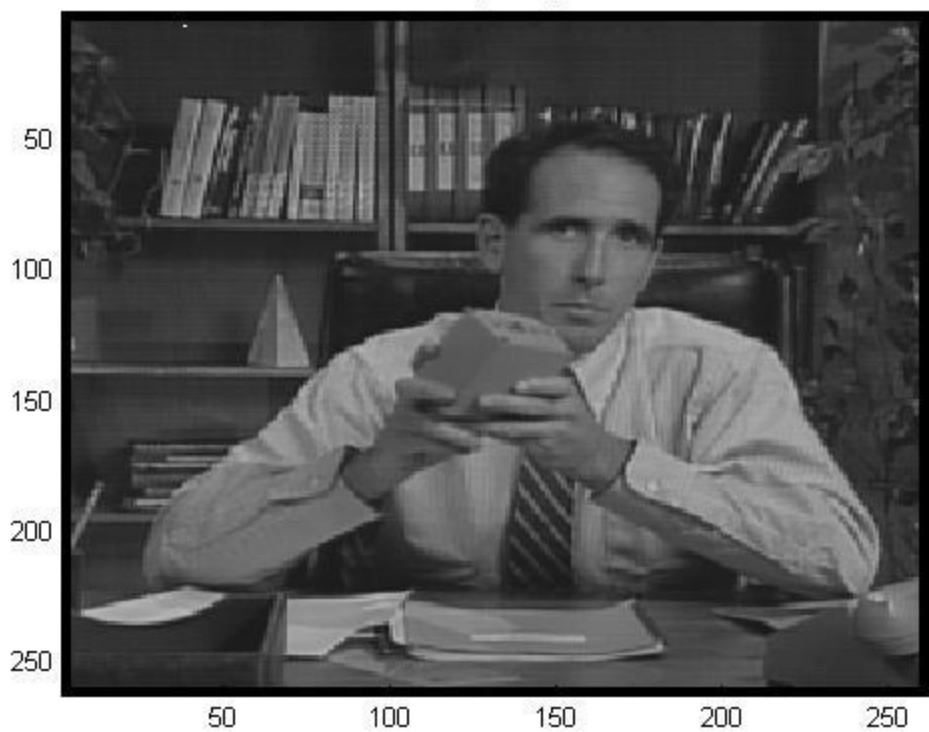


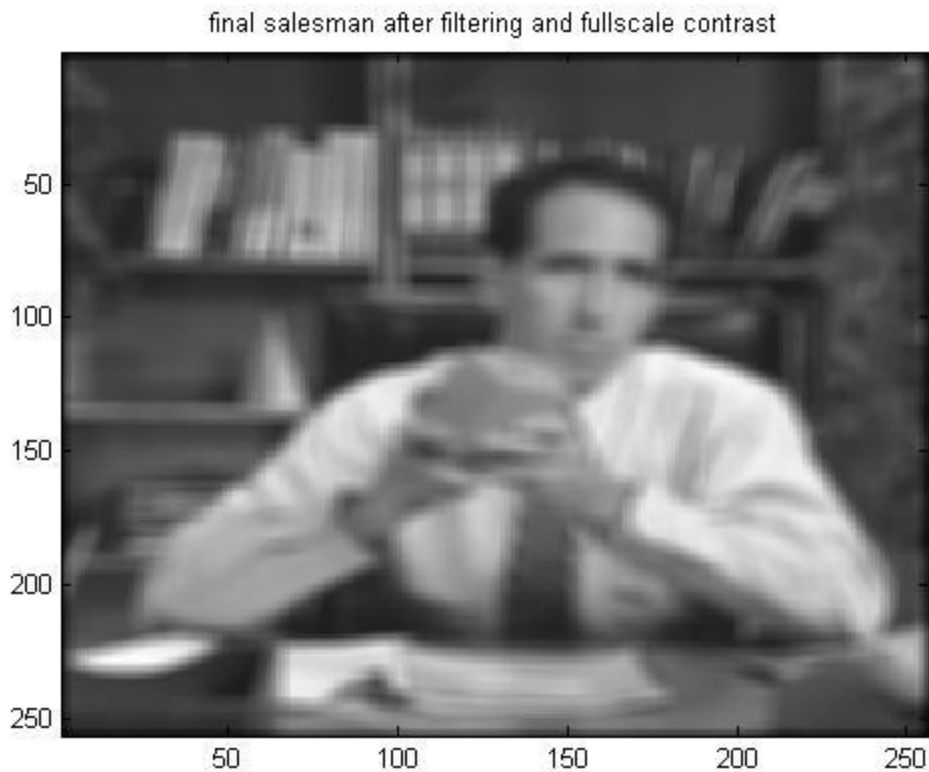
Problem-1 a

```
clc;
clearall;
closeall;
%(1a).Reading and displaying Salesman Image
salesman=fopen('salesman.bin','r');
s=fread(salesman,[256,256],'uchar');
s=s';
figure(1);
image(s);
colormap(gray(256));
fss=contraststretch(s);
figure;
image(fss);
Title('Original salesman Image with full scale Stretch');
padsz=262;
ZPsalesman=zeros(padsz,padsz);
ZPsalesman(4:259,4:259)=s;
figure;
colormap(gray(256));
image(contraststretch(ZPsalesman));
Title('Salesman man after Zeropadding and fullscale stretch');
%final salesman
Finsalesman=zeros(256,256);
fori=1:256
for j=1:256
sum=0;
for a=1:7;
for b=1:7;
sum=sum+ZPsalesman(i+a-1,j+b-1);
end
end
Finsalesman(i,j)=sum/49;
end
end
figure;
colormap(gray(256));
image(contraststretch(Finsalesman));
Title('final salesman after filtering and fullscale contrast');
```



Salesman man after Zeropadding and fullscale stretch





Prb-1 b

```
clc;
clearall;
closeall;
%(1b)Display Salesman Image
salesman=fopen('salesman.bin','r');
s=fread(salesman,[256,256],'uchar');
s=s';
figure(1);
colormap(gray(256));
fsstretch=contraststretch(s);
image(fsstretch);
Title('Original salesman Image with full scale Stretch');
```

```
h=zeros(128,128);
h(62:68,62:68)=ones(7,7)/49;
figure(2);
Impresponse=contraststretch(h);
imshow(Impresponse);
Title('Impulse response H');
```

%Zeropadding and creating ZPsalesman and ZPh

```
Padsize = 256+128-1;
ZPsalesman = zeros(Padsize,Padsize);
ZPsalesman(1:256,1:256) = s;
figure(3);
colormap(gray(256));
FZeroPaddedS=contraststretch(ZPsalesman);
image(FZeroPaddedS);
Title('Zeroadded salesman Image');
```

```
ZPh = zeros(Padsize,Padsize);
ZPh(1:128,1:128) = h;
```

```

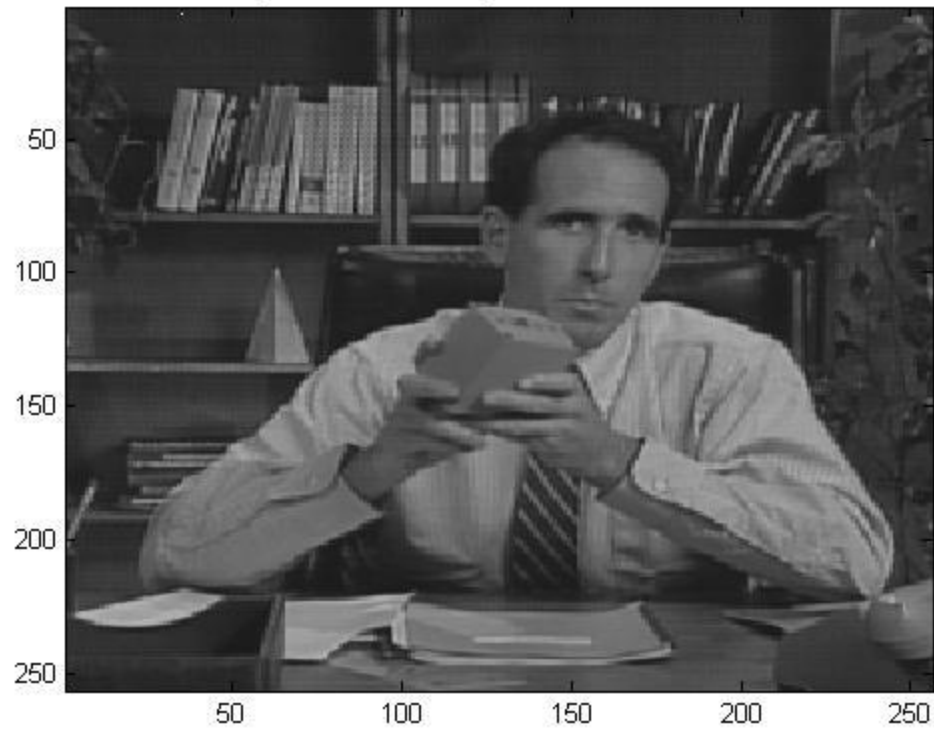
figure(4);
FZPh=contraststretch(ZPh);
colormap(gray(256));
image(FZPh);
Title('Zeropadded H impulse reponse');

%Centered DFT of salesman and Impulse responseH
fSCDFT=fftshift(fft2(ZPsalesman));
SCDFT=(fft2(ZPsalesman));
SSCDFT=contraststretch(fSCDFT);
LOZDS=log(1+abs(SSCDFT));
figure(5);
imshow(LOZDS);
Title('Centered DFT log mag of Zeropadded Salesman Image');
%CODFT of H
fHCDFT= fftshift(fft2(ZPh));
HCDFT= (fft2(ZPh));
figure(6);
SHCDFT=contraststretch(fHCDFT);
LOZDH=log(1+abs(SHCDFT));
%imshow(HCDFT);
imshow(LOZDH);
Title('Centered DFT log mag of Zeropadded H Image');
%convolution or pointwise multiplication
Z1 = real(ifft2(SCDFT .* HCDFT));
FZ1=contraststretch(Z1);
figure(7);
colormap(gray(256));
image(FZ1);
Title(' Zeropadded Output Image');

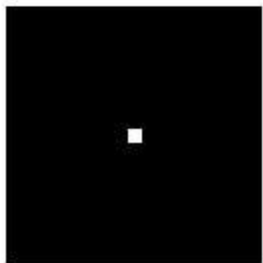
%Final Output Image
Finalsalesman=FZ1(65:320,65:320);
figure(8);
colormap(gray(256));
image(Finalsalesman);
Title('Salesman Final Output image after filtering and cropping');

```

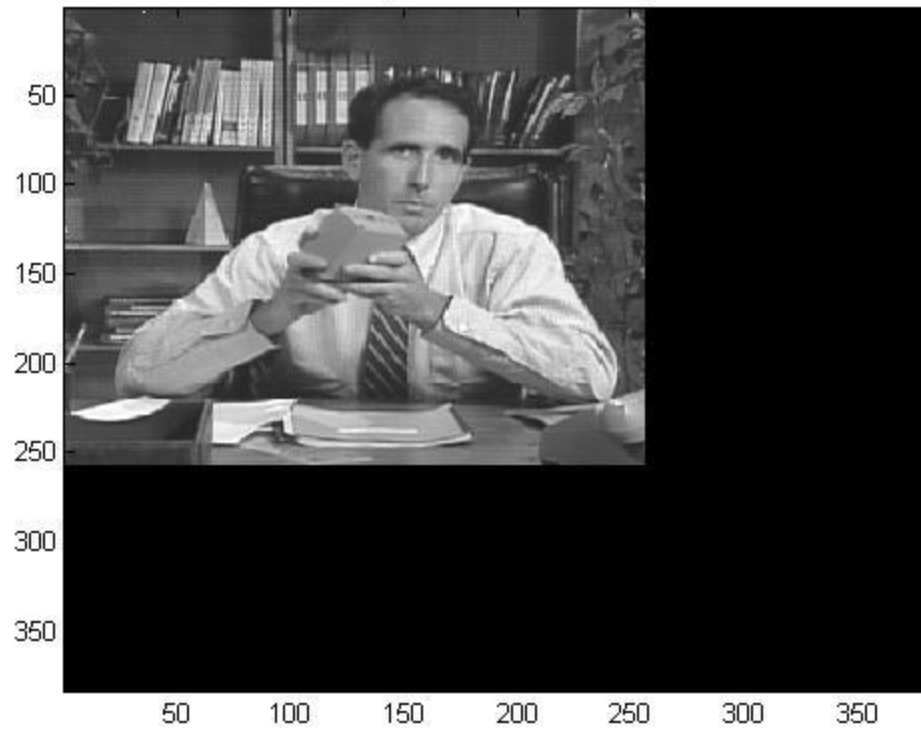
Original salesman Image with full scale Stretch



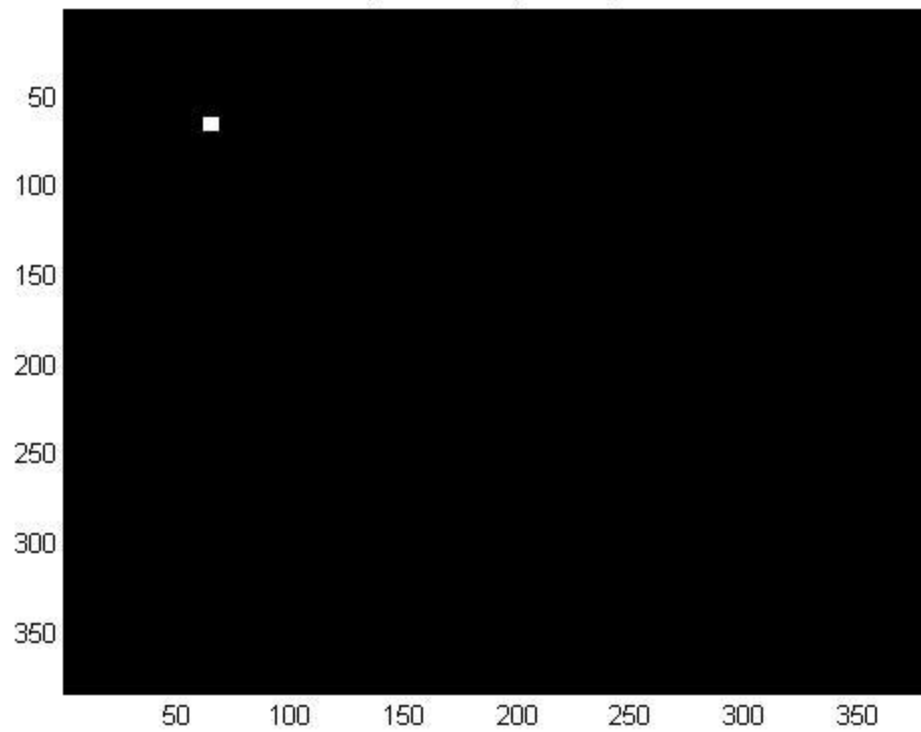
Impulse response H



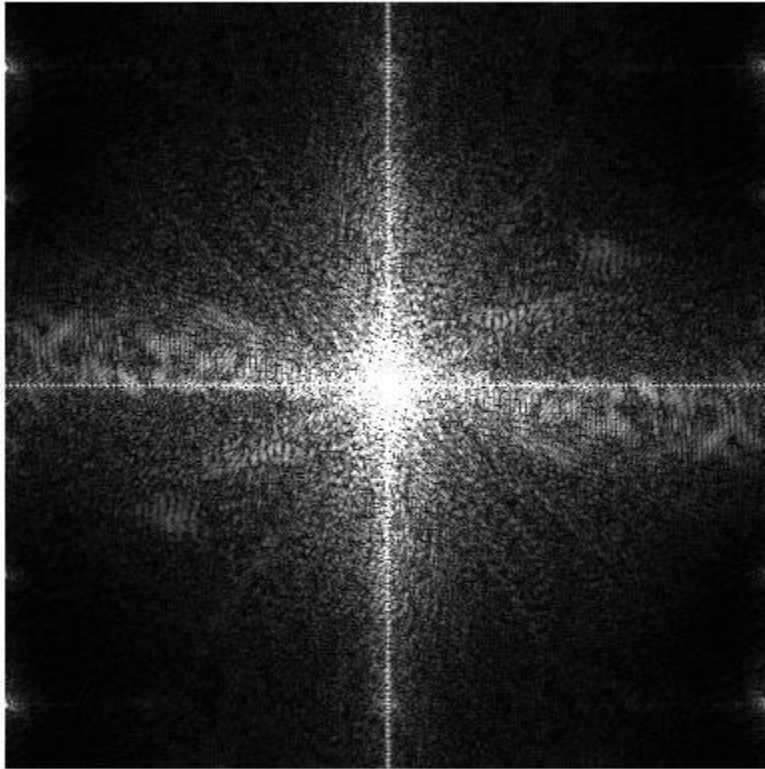
Zeropadded salesman Image



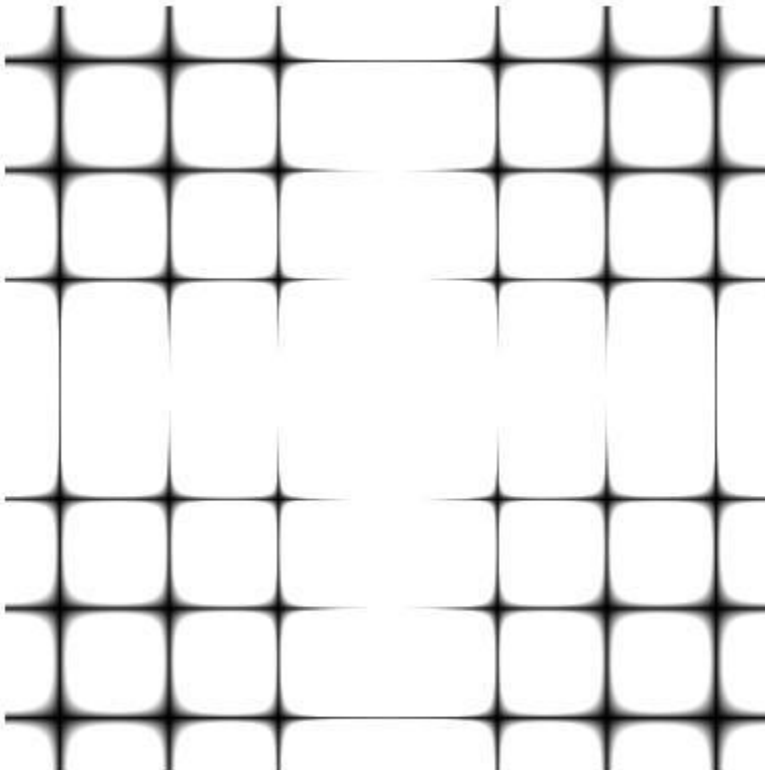
Zeropadded H impulse reponse



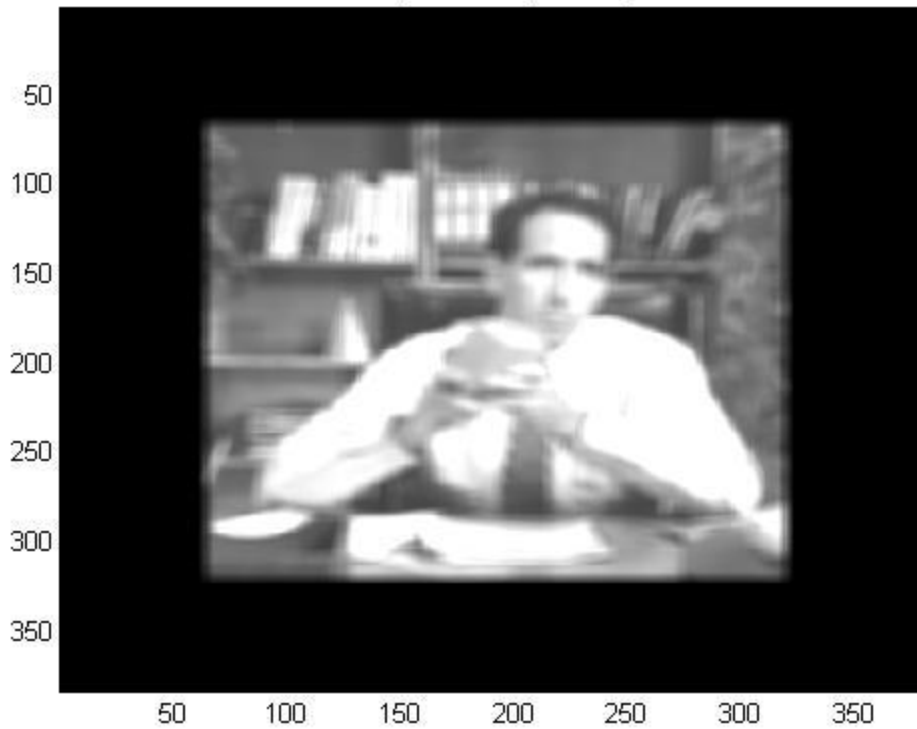
Centered DFT log mag of Zeropadded Salesman Image



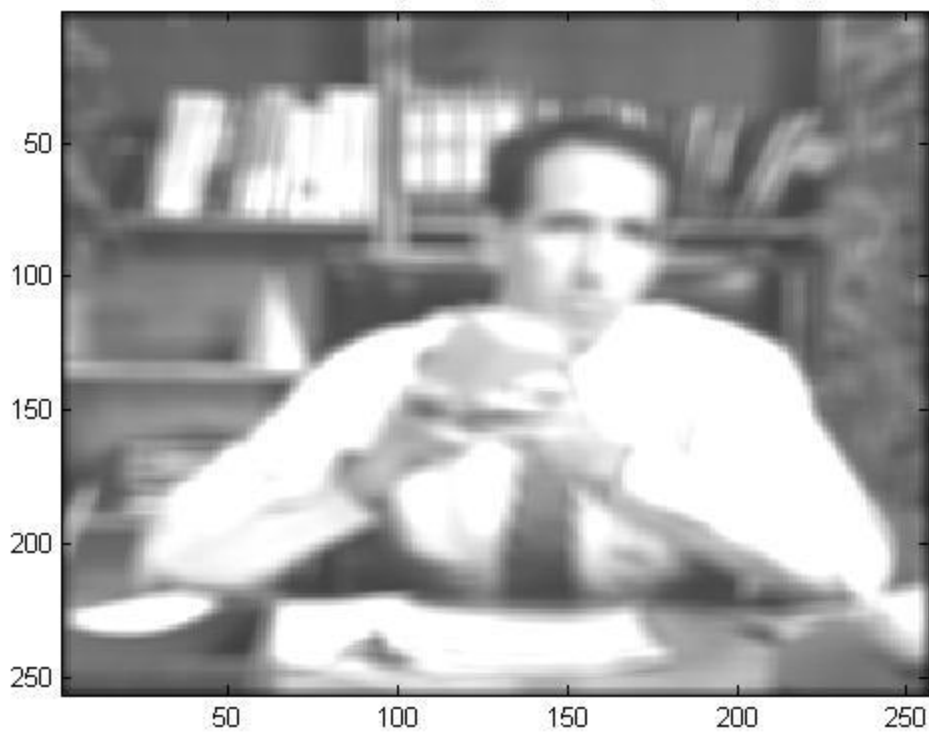
Centered DFT log mag of Zeropadded H Image



Zeropadded Output Image



Salesman Final Output image after filtering and cropping



Prb-1 c

```
clearall;  
closeall;  
%(1c).Reading and displaying Salesman Image  
salesman=fopen('salesman.bin','r');  
s=fread(salesman,[256,256],'uchar');  
s=s';  
figure(1);
```



```

colormap(gray(256));
fin=contraststretch(s);
image(fin);
Title('Original salesman Image with full scale Stretch');

ZPsalesman = zeros(512,512);
ZPsalesman(1:256,1:256) = s;
FZPsalesman=contraststretch(ZPsalesman);
figure(2);
colormap(gray(256));
image(FZPsalesman);
Title('Zeropadded salesman image after fullscale contrast stretch');

%impulse response
H= zeros(256,256);
H(126:132,126:132)=ones(7,7)/49;
figure(3);
colormap(gray(256));
image(contraststretch(H));
Title('Filter H after Zeropadding and full scale stretch');

H2=fftshift(H);
FH2=contraststretch(H2);
figure(4);
colormap(gray(256));
image(FH2);
Title('H after fullscale stretch and shifting');

ZPH2 = zeros(512,512);
ZPH2(1:128,1:128) = H2(1:128,1:128);
ZPH2(1:128,385:512) = H2(1:128,129:256);
ZPH2(385:512,1:128) = H2(129:256,1:128);
ZPH2(385:512,385:512) = H2(129:256,129:256);

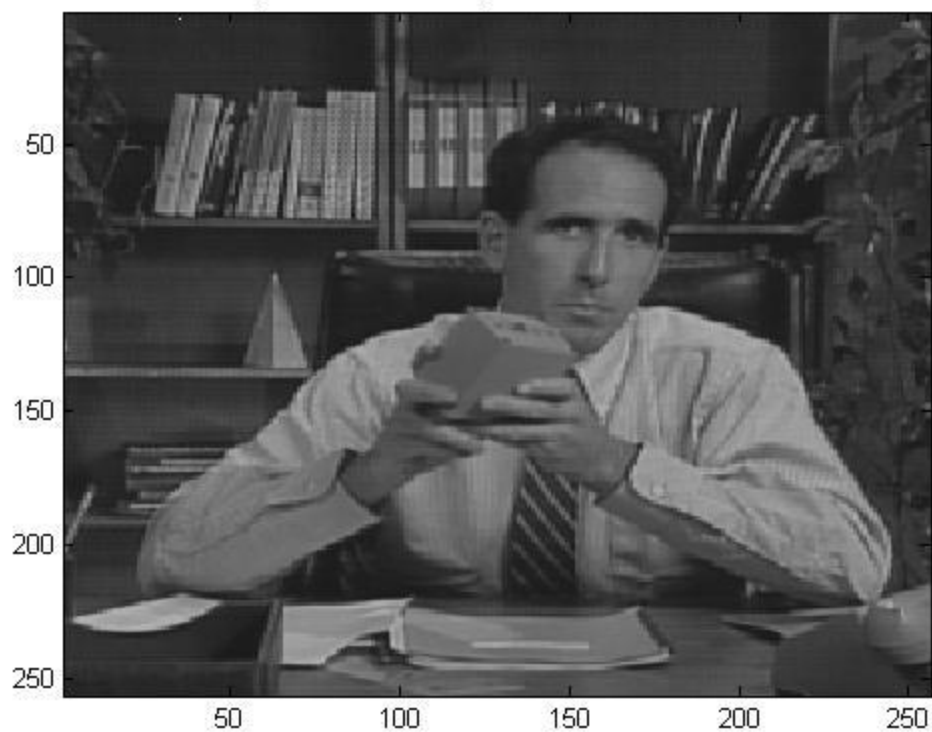
FZPH2= contraststretch(ZPH2);
figure(5);
colormap(gray(256));
image(FZPH2);
Title('Final H');

Z1 = ifft2(fft2(ZPsalesman).*fft2(ZPH2));
Z1 = Z1(1:256,1:256);
figure(6);
colormap(gray(256));
image(contraststretch(Z1));
Title('Final 256*256 Output image ');

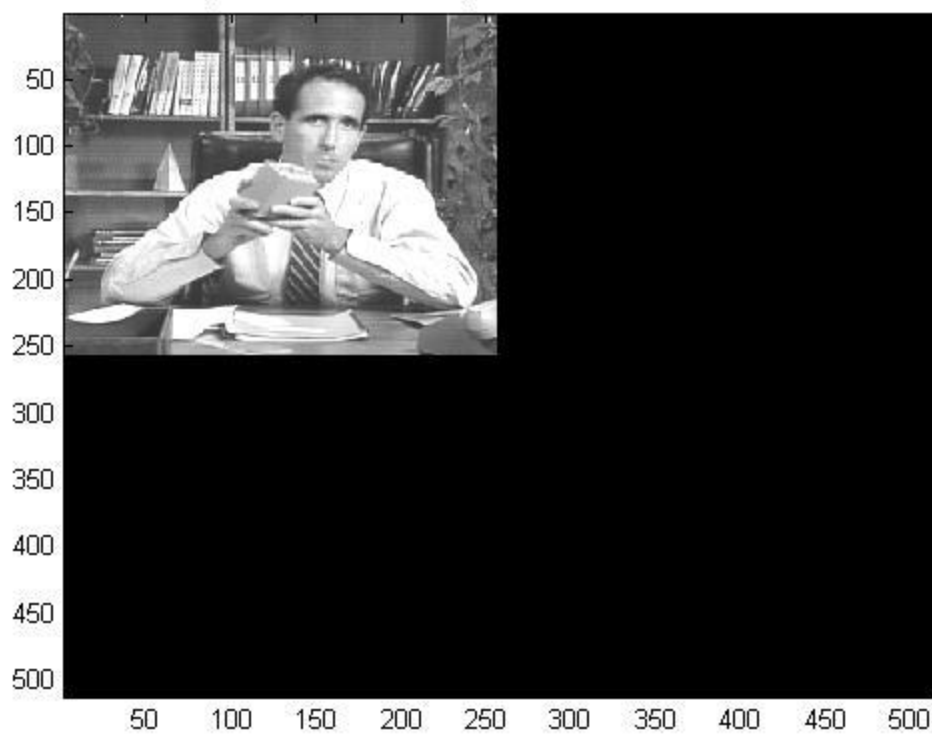
```

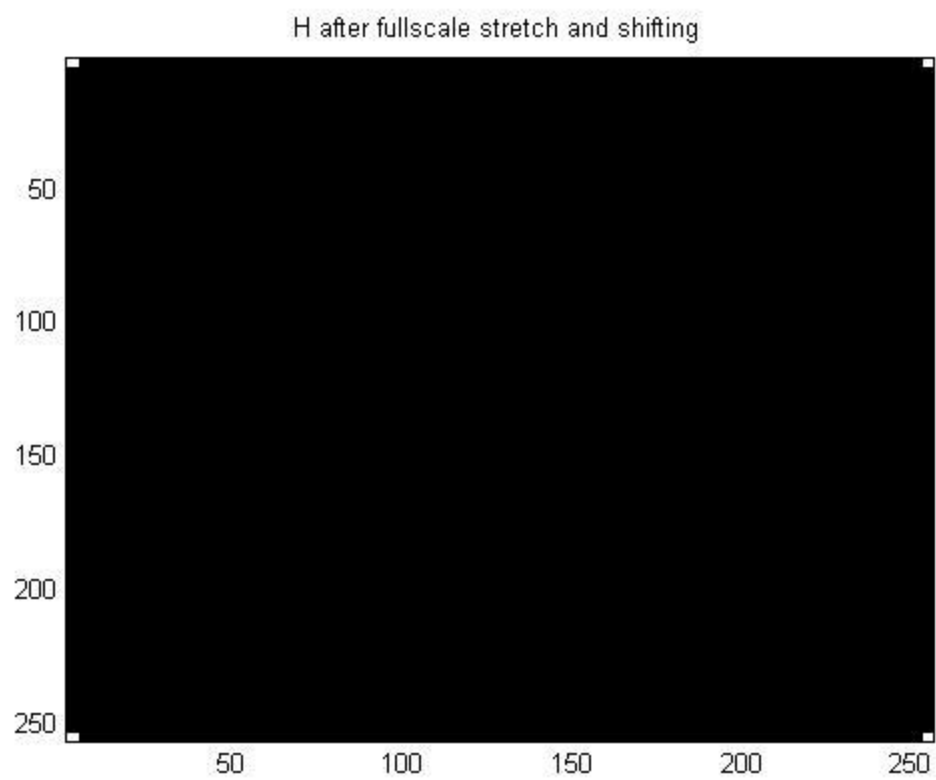
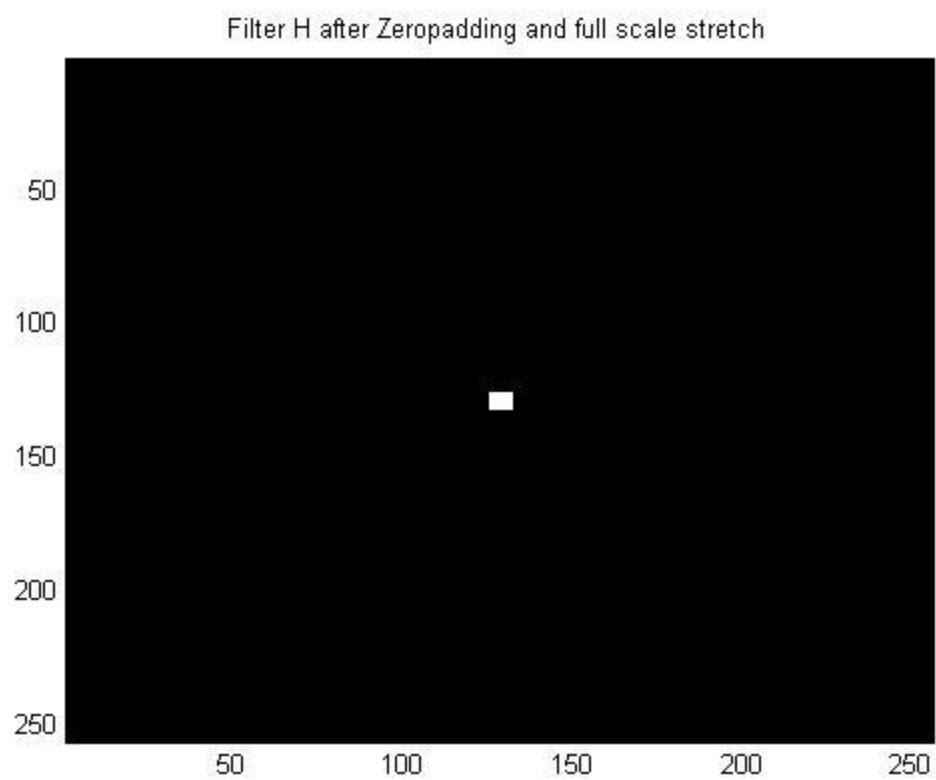
Output:

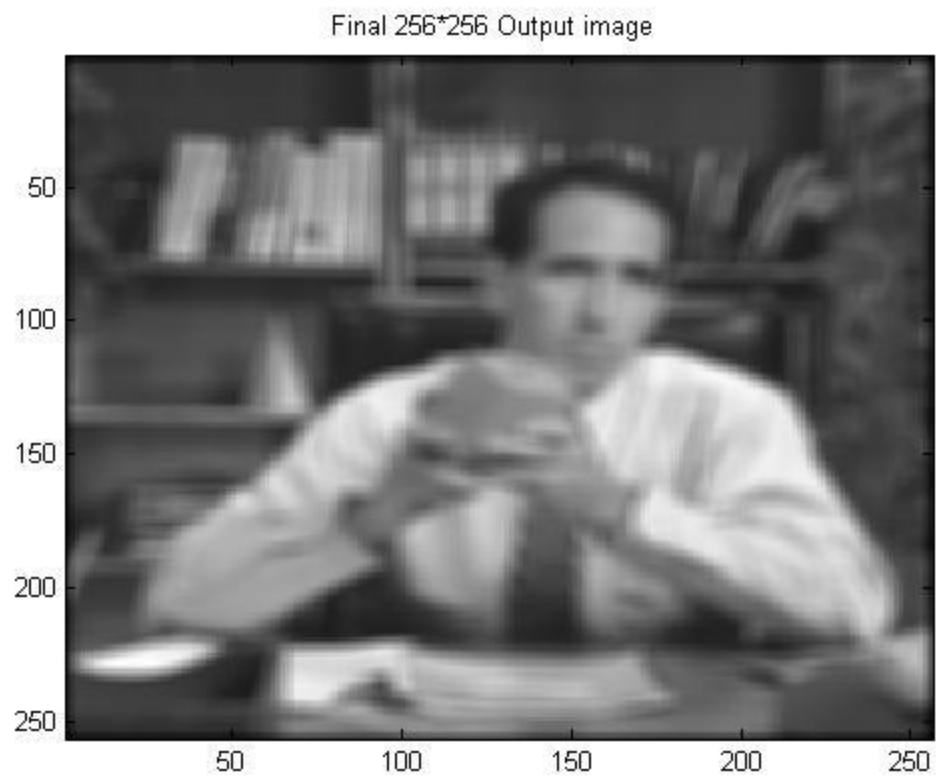
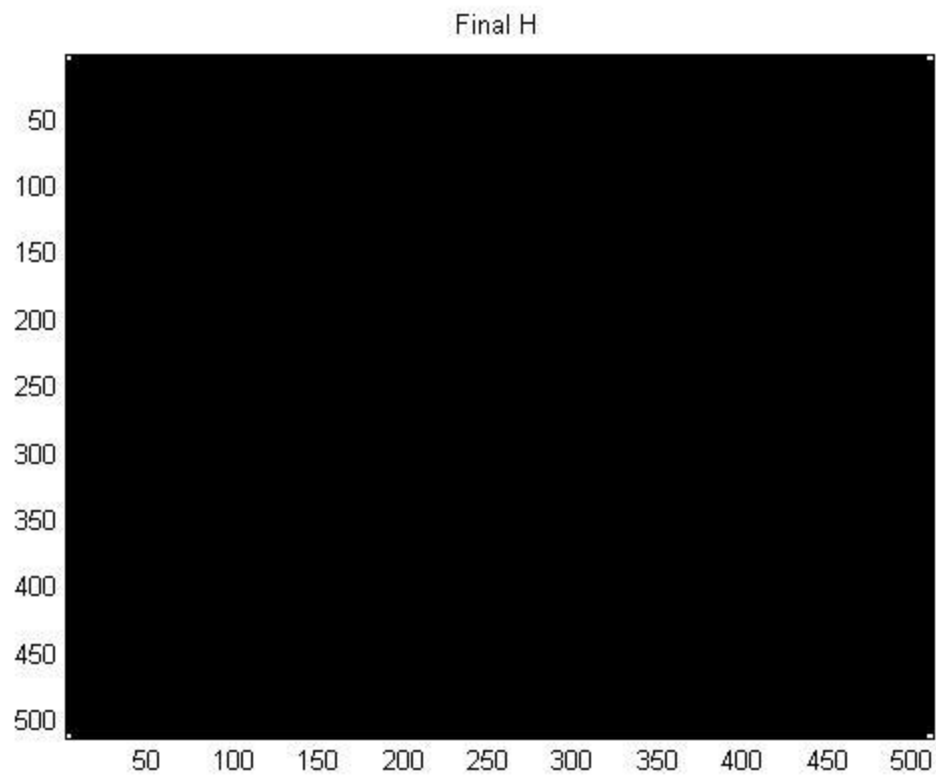
Original salesman Image with full scale Stretch



Zeropadded salesman image after fullscale contrast stretch







Prb-2 a

```
clc;
girl2=fopen('girl2.bin','r');
g=fread(girl2,[256,256],'uchar');
g=g';
figure(1);
colormap(gray(256));
image(g);
```

```

Title('Original girl2 Image ');

girl2Noise32Hi=fopen('girl2Noise32Hi.bin','r');
gnh=fread(girl2Noise32Hi,[256,256],'uchar');
gnh=gnh';
figure(2);
colormap(gray(256));
image(gnh);
Title('Original girl2Noise32Hi Image ');

girl2Noise32=fopen('girl2Noise32.bin','r');
gn=fread(girl2Noise32,[256,256],'uchar');
gn=gn';
figure(3);
colormap(gray(256));
image(gn);
Title('Original girl2Noise32 Image ');

M=256;
N=256;
sum2=0;
for m=1:256
    sum1=0;
    for n=1:256
        NM=(g(m,n)-gnh(m,n))^2;
        sum1=sum1+NM;
    end
    sum2=sum2+sum1;
end
MSE1=sum2/(M*N);
display(MSE1);

M=256;
N=256;
sum2=0;
for m=1:256
    sum1=0;
    for n=1:256
        NM=(g(m,n)-gn(m,n))^2;
        sum1=sum1+NM;
    end
    sum2=sum2+sum1;
end
MSE2=sum2/(M*N);
display(MSE2);

output:
MSE1 =

    692.5050

MSE2 =

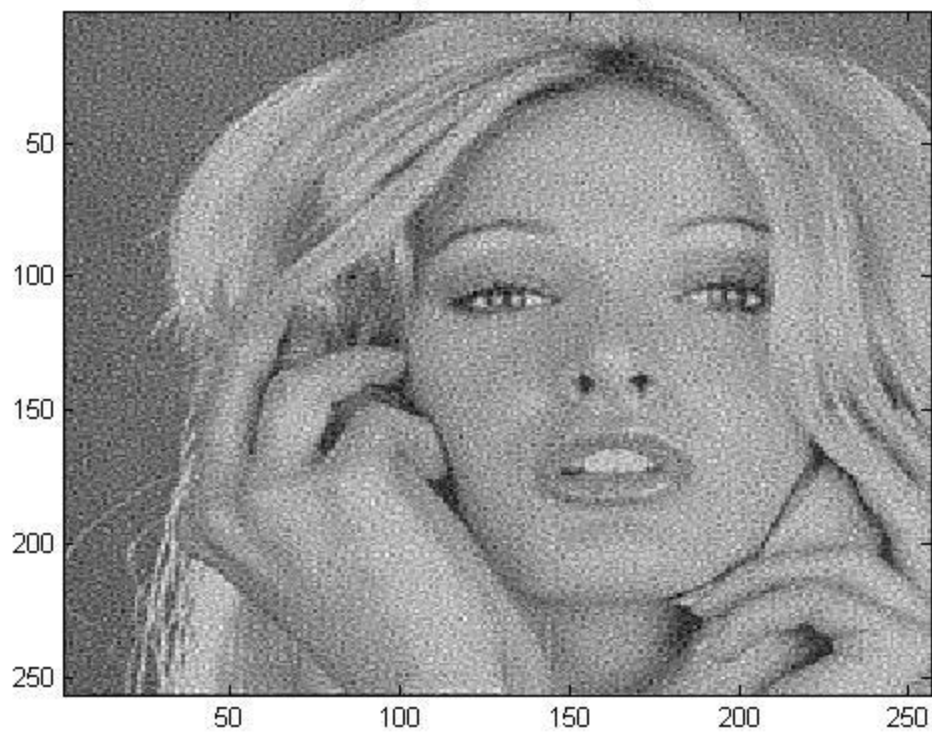
    744.4679

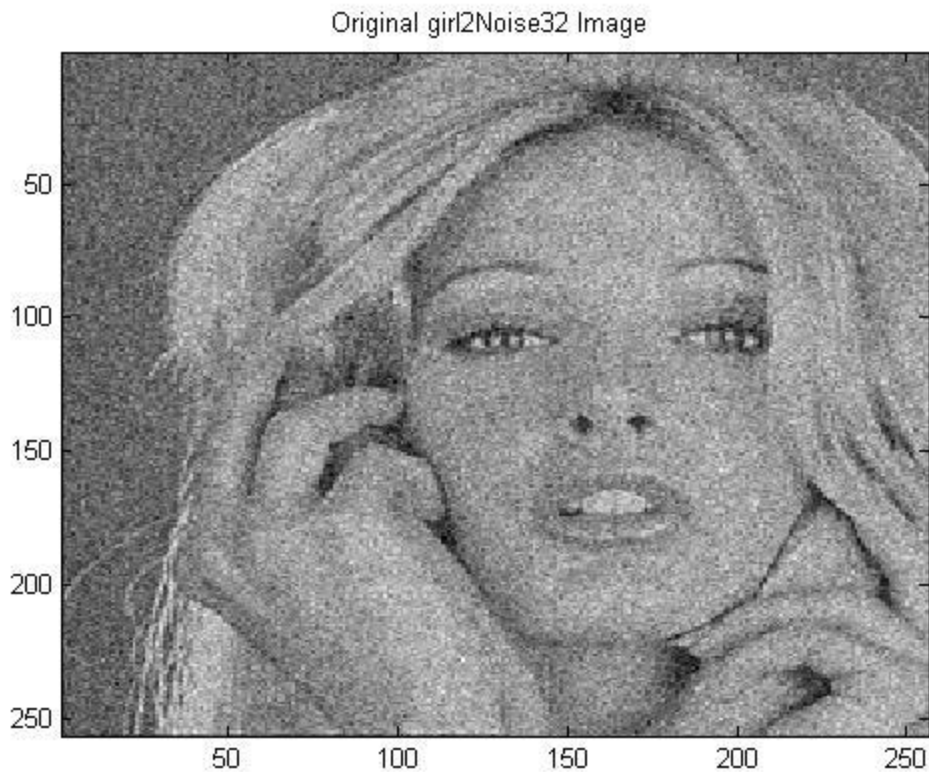
```

Original girl2 Image



Original girl2Noise32Hi Image





Prb-2 b

```
clc;
closeall;
clearall;
girl2=fopen('girl2.bin','r');
g=fread(girl2,[256,256],'uchar');
g=g';
figure(1);
colormap(gray(256));
image(g);
Title('Original girl2 Image ');
%
girl2Noise32Hi=fopen('girl2Noise32Hi.bin','r');
gnh=fread(girl2Noise32Hi,[256,256],'uchar');
gnh=gnh';
figure(2);
colormap(gray(256));
image(gnh);
Title('Original girl2Noise32Hi Image ');
%
girl2Noise32=fopen('girl2Noise32.bin','r');
gn=fread(girl2Noise32,[256,256],'uchar');
gn=gn';
figure(3);
colormap(gray(256));
image(gn);
Title('Original girl2Noise32 Image ');

%Computing the MSE
P=256;
Q=256;
sum2=0;
for m=1:256
    sum1=0;
    for n=1:256
```

```

        QP=(g(m,n)-gnh(m,n))^2;
        sum1=sum1+QP;
    end
    sum2=sum2+sum1;
end
MSE1=sum2/(P*Q);
display('MSE between girl2 and girl2Noise32Hi ');
display(MSE1);

M=256;
N=256;
sum2=0;
for m=1:256
    sum1=0;
    for n=1:256
        NM=(g(m,n)-gn(m,n))^2;
        sum1=sum1+NM;
    end
    sum2=sum2+sum1;
end
MSE2=sum2/(M*N);
display('MSE between girl2 image and girl2Noise32 image ');
display(MSE2);
%Filter
U_cutoff = 64;

[U,V] = meshgrid(-128:127,-128:127);
HLtildeCenter = double(sqrt(U.^2 + V.^2) <= U_cutoff);
HLtilde = fftshift(HLtildeCenter);
figure(4);
imshow(HLtilde);
Title('HLtilde');

%Pointwise multiplication
%girl2
g1 = ifft2(HLtilde.*fft2(g));
figure(5);
colormap(gray(256));
image(contraststretch(g1));
Title('girl2 image after pointwise multiplication');
%girl2Noise32Hi
g2 = ifft2(HLtilde.*fft2(gnh));
figure(6);
colormap(gray(256));
image(contraststretch(g2));
Title('girl2Noise32Hi image after pointwise multiplication');
%girl2Noise32
g3 = ifft2(HLtilde.*fft2(gn));
figure(7);
colormap(gray(256));
image(contraststretch(g3));
Title('girl2Noise32 image after pointwise multiplication');

%computing MSE

P=256;
Q=256;
sum2=0;
for m=1:256
    sum1=0;
    for n=1:256
        QP=(g(m,n)-g1(m,n))^2;
        sum1=sum1+QP;
    end

```



```

        sum2=sum2+sum1;
    end
    FinalMSE=sum2/(P*Q);
    display('Final MSE after filtering  between original girl2 image and filtered
    girl2 image');
    display(FinalMSE);

P=256;
Q=256;
sum2=0;
for m=1:256
    sum1=0;
    for n=1:256
        QP=(g(m,n)-g2(m,n))^2;
        sum1=sum1+QP;
    end
    sum2=sum2+sum1;
end
FinMSE1=sum2/(P*Q);
display('Final MSE after filtering  between original girl2 and girl2Noise32Hi
');

display(FinMSE1);

M=256;
N=256;
sum2=0;
for m=1:256
    sum1=0;
    for n=1:256
        NM=(g(m,n)-g3(m,n))^2;
        sum1=sum1+NM;
    end
    sum2=sum2+sum1;
end
FMSE2=sum2/(M*N);
display('Final MSE between original girl2 and girl2Noise32 ');
display(FMSE2);

%computing ISNR
ISNR1=10*log(MSE1/FMSE1);
display('ISNR before and after filtering of girl2Noise32Hi ');
disp('ISNR1=');
disp(ISNR1);
ISNR2=10*log(MSE2/FMSE2);
display('ISNR before and after filtering of girl2Noise32 ');
disp('ISNR2=');
disp(ISNR2);

```

Output:
MSE between girl2 and girl2Noise32Hi

MSE1 =

692.5050

MSE between girl2 and girl2Noise32

MSE2 =

744.4679

Final MSE after filtering between original girl2 and filtered girl2

FMSE =

127.7481

Final MSE after filtering between original girl2 and girl2Noise32Hi

FMSE1 =

398.9978

final MSE between original girl2 and girl2Noise32

FMSE2 =

550.8787

ISNR before and after filtering of girl2Noise32Hi

ISNR1=

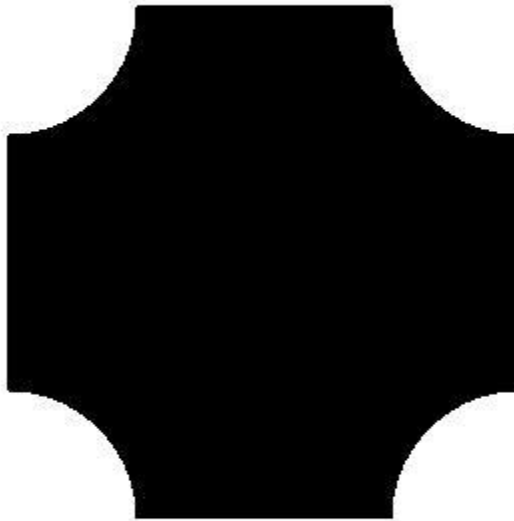
5.5136

ISNR before and after filtering of girl2Noise32

ISNR2=

3.0116

HLtilde



girl2 image after pointwise multiplication



girl2Noise32Hi image after pointwise multiplication



girl2Noise32 image after pointwise multiplication



```
Prb-2 c
clc;
closeall;
clearall;
girl2=fopen('girl2.bin','r');
g=fread(girl2,[256,256],'uchar');
g=g';
figure(1);
colormap(gray(256));
image(contraststretch(g));
```

```

Title('Original girl2 Image ');
%
girl2Noise32Hi=fopen('girl2Noise32Hi.bin','r');
gnh=fread(girl2Noise32Hi,[256,256],'uchar');
gnh=gnh';
figure(2);
colormap(gray(256));
image(contraststretch(gnh));
Title('Original girl2Noise32Hi Image ');
%
girl2Noise32=fopen('girl2Noise32.bin','r');
gn=fread(girl2Noise32,[256,256],'uchar');
gn=gn';
figure(3);
colormap(gray(256));
image(contraststretch(gn));
Title('Original girl2Noise32 Image ');

%Computing the MSE
M=256;
N=256;
sum2=0;
for m=1:256
    sum1=0;
    for n=1:256
        NM=(g(m,n)-gnh(m,n))^2;
        sum1=sum1+NM;
    end
    sum2=sum2+sum1;
end
MSE1=sum2/(M*N);
display('MSE between girl2 and girl2Noise32Hi ');
display(MSE1);

M=256;
N=256;
sum2=0;
for m=1:256
    sum1=0;
    for n=1:256
        NM=(g(m,n)-gn(m,n))^2;
        sum1=sum1+NM;
    end
    sum2=sum2+sum1;
end
MSE2=sum2/(M*N);
display('MSE between girl2 and girl2Noise32 ');
display(MSE2);

U_cutoff_H = 34;
SigmaH = 0.19 * 256 / U_cutoff_H;
[U,V] = meshgrid(-128:127,-128:127);
HtildeCenter = exp((-2*pi*SigmaH^2)/(256.^2)*(U.^2 + V.^2));
Htilde = fftshift(HtildeCenter);
H = ifft2(Htilde);
H2 = fftshift(H);
ZPH2 = zeros(512,512);
ZPH2(1:256,1:256) = H2;

ZPg1=zeros(512,512);
ZPg1(1:256,1:256)=g;
ZPgnh=zeros(512,512);
ZPgnh(1:256,1:256)=gnh;

```

```

ZPgn=zeros(512,512);
ZPgn(1:256,1:256)=gn;

g1 = real(ifft2(fft2(ZPH2).*fft2(ZPg1)));
g2 = real(ifft2(fft2(ZPH2).*fft2(ZPgnh)));
g3 = real(ifft2(fft2(ZPH2).*fft2(ZPgn)));

cg1=g1(129:384,129:384);
figure(5);
colormap(gray(256));
image(contraststretch(cg1));
Title('Original girl2 image after filtering and cropping ');
cg2=g2(129:384,129:384);
figure(6);
colormap(gray(256));
image(contraststretch(cg2));
Title('Original girl2Noise32Hi image after filtering and cropping ');
cg3=g3(129:384,129:384);
figure(7);
colormap(gray(256));
image(contraststretch(cg3));
Title('Original girl2Noise32 image after filtering and cropping ');

M=256;
N=256;
sum2=0;
for m=1:256
    sum1=0;
    for n=1:256
        NM=(g(m,n)-cg1(m,n))^2;
        sum1=sum1+NM;
    end
    sum2=sum2+sum1;
end
FMSE=sum2/(M*N);
display('Final MSE after filtering between original girl2 and filtered girl2');
display(FMSE);

M=256;
N=256;
sum2=0;
for m=1:256
    sum1=0;
    for n=1:256
        NM=(g(m,n)-cg2(m,n))^2;
        sum1=sum1+NM;
    end
    sum2=sum2+sum1;
end
FMSE1=sum2/(M*N);
display('Final MSE after filtering between original girl2 and girl2Noise32Hi');
display(FMSE1);

M=256;
N=256;
sum2=0;

```

```

for m=1:256
    sum1=0;
for n=1:256
    NM=(g(m,n)-cg3(m,n))^2;
    sum1=sum1+NM;
end
    sum2=sum2+sum1;
end
FMSE2=sum2/(M*N);
display('final MSE between original girl2 and girl2Noise32 ');
display(FMSE2);

ISNR1=10*log(MSE1/FMSE1);
display('ISNR before and after filtering of girl2Noise32Hi ');
disp('ISNR1='); disp(ISNR1);
ISNR2=10*log(MSE2/FMSE2);
display('ISNR before and after filtering of girl2Noise32 ');
disp('ISNR2='); disp(ISNR2);

```

Output:

MSE between girl2 and girl2Noise32Hi

MSE1 =

692.5050

MSE between girl2 and girl2Noise32

MSE2 =

744.4679

Final MSE after filtering between original girl2 and filtered girl2

FMSE =

21.9643

Final MSE after filtering between original girl2 and girl2Noise32Hi

FMSE1 =

442.4524

final MSE between original girl2 and girl2Noise32

FMSE2 =

555.0764

ISNR before and after filtering of girl2Noise32Hi

ISNR1=

4.4798

ISNR before and after filtering of girl2Noise32

ISNR2=

2.9356

Original girl2 image after filtering and cropping



Original girl2Noise32Hi image after filtering and cropping



Original girl2Noise32 image after filtering and cropping



Prb-2 d

Program same as prb-2 c

U_cutoff=34

Output:

MSE1 =

692.5050

MSE2 =

744.4679

FMSE =

102.3465

FMSE1 =

422.1168

FMSE2 =

539.7324

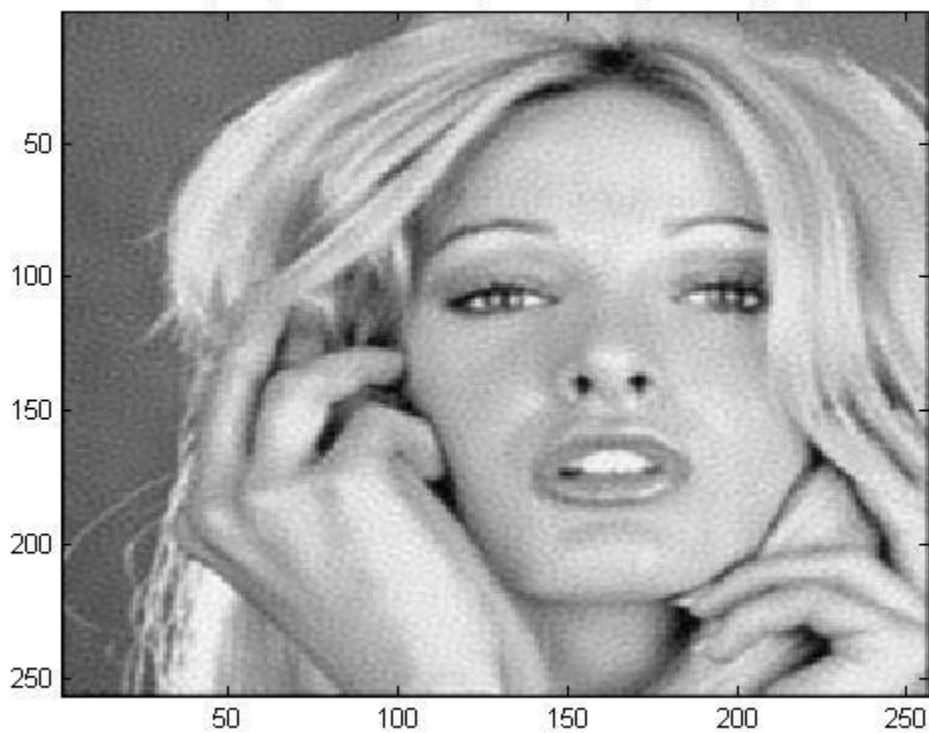
ISNR1=
4.9503

ISNR2=
3.2160

Original girl2 image after filtering and cropping



Original girl2Noise32Hi image after filtering and cropping





This filter performs better in cases of noise than both the filters used before but distorts original image.

Full scale contrast stretch program:

```
function fin=contraststretch(orgn)
[m,n]=size(orgn);
A=min(min(abs(orgn)));
B=max(max(abs(orgn)));
if A==B
    A=min(min(orgn));
    B=max(max(orgn));
end
P=(m-1)/(B-A);
L=-1*A*((m-1)/(B-A));
fin=zeros(m,n);
for i=1:m
    for j=1:n
        fin(i,j)=P*orgn(i,j)+L;
    end
end
```