

**THE UNIVERSITY OF TEXAS AT ARLINGTON, TEXAS  
DEPARTMENT OF ELECTRICAL ENGINEERING**

**EE 5356**

**DIGITAL IMAGE PROCESSING**

**PROJECT # 8**

**by**

**SOUTRIK MAITI**

**1001569883**

**Presented to**

**Dr. K.R.RAO**

**Mar 29, 2019**

**Inverse Gaussian Filter**

***MATLAB Code:***

%% Reading the images for the project

img\_lena = double(imread('lena512.bmp'));

img\_lena = img\_lena(:,:,1);

img\_goldhill = double(imread('goldhill256.BMP'));

img\_girl = double(imread('girl512.bmp'));

img\_boat = double(imread('boat512.gif'));

%% Setting the variance for the images

lena\_var = 15707;

goldhill\_var = 28198;

boat\_var = 41694;

girl\_var = 34657;

%% Applying 2D FFT for the images

dft\_lena = fft2(img\_lena);

dft\_goldhill = fft2(img\_goldhill);

dft\_boat = fft2(img\_boat);

dft\_girl = fft2(img\_girl);

%% Initializing the IGF matrix for the images

igf\_goldhill = zeros(1,128);

igf\_lena = zeros(1,256);

igf\_boat = zeros(1,256);

igf\_girl = zeros(1,256);

%% IGF for goldhill256

for i = 1:128

for j = 1:128

igf\_goldhill(i,j) = exp(((i^2)+(j^2))/(2\*goldhill\_var));

end

end

for i = 1:256

for j = 1:256

igf\_lena(i,j) = exp(((i^2)+(j^2))/(2\*lena\_var));

igf\_girl(i,j) = exp(((i^2)+(j^2))/(2\*girl\_var));

igf\_boat(i,j) = exp(((i^2)+(j^2))/(2\*boat\_var));

%% conditional for goldhill

if i<=128 && j>128

igf\_goldhill(i,j) = igf\_goldhill(i,257-j);

elseif i>128 && j<=128

igf\_goldhill(i,j) = igf\_goldhill(257-i,j);

elseif i>128 && j>128

igf\_goldhill(i,j) = igf\_goldhill(257-i,257-j);

end

end

end

%% IGF for the remaining images

for i = 1:512

for j = 1:512

%% conditional for remaining images

if i<=256 && j>256

igf\_lena(i,j) = igf\_lena(i,513-j);

igf\_girl(i,j) = igf\_girl(i,513-j);

igf\_boat(i,j) = igf\_boat(i,513-j);

elseif i>256 && j<=256

igf\_lena(i,j) = igf\_lena(513-i,j);

igf\_girl(i,j) = igf\_girl(513-i,j);

igf\_boat(i,j) = igf\_boat(513-i,j);

elseif i>256 && j>256

igf\_lena(i,j)=igf\_lena(513-i,513-j);

igf\_girl(i,j)=igf\_girl(513-i,513-j);

igf\_boat(i,j)=igf\_boat(513-i,513-j);

end

end

end

%% Displaying the 3D IGF for the images

%lena

surf(igf\_lena,'EdgeColor','none');

colormap(hsv);

shading interp;

alpha(0.7);

axis tight;

title('IGF Mask Lena');

saveas(gca,'3d\_igf\_lena.jpg');

%boat

figure; surf(igf\_boat,'EdgeColor','none');

colormap(hsv);

shading interp;

alpha(0.7);

axis tight;

title('IGF Mask Boat');

saveas(gca,'3d\_igf\_boat.jpg');

%girl

figure; surf(igf\_girl,'EdgeColor','none');

colormap(hsv);

shading interp;

alpha(0.7);

axis tight;

title('IGF Mask Girl');

saveas(gca,'3d\_igf\_girl.jpg');

%goldhill

figure;

surf(igf\_goldhill,'EdgeColor','none');

colormap(hsv);

shading interp;

alpha(0.7);

axis tight;

title('IGF Mask Goldhill');

saveas(gca,'3d\_igf\_goldhill.jpg');

%% Filtered images

f\_l = dft\_lena.\*igf\_lena;

f\_g = dft\_goldhill.\*igf\_goldhill;

f\_gr = dft\_girl.\*igf\_girl;

f\_b = dft\_boat.\*igf\_boat;

%% Inverse FFT of the filtered images

ifft\_l = ifft2(f\_l);

ifft\_g = ifft2(f\_g);

ifft\_gr = ifft2(f\_gr);

ifft\_b = ifft2(f\_b);

%% Displaying Original and filtered images

%lena

figure;

subplot(1,2,1);

imshow(uint8(img\_lena));

title('Original Image');

subplot(1,2,2);

imshow(uint8(ifft\_l));

title('Filtered Image');

saveas(gca,'lena\_results.jpg');

%girl

figure;

subplot(1,2,1);

imshow(uint8(img\_girl));

title('Orig. Image');

subplot(1,2,2);

imshow(uint8(ifft\_gr));

title('Filtered Image');

saveas(gca,'girl\_results.jpg');

%goldhill

figure;

subplot(1,2,1);

imshow(uint8(img\_goldhill));

title('Orig. Image');

subplot(1,2,2);imshow(uint8(ifft\_g));

title('Filtered Image');

figure;

saveas(gca,'goldhill\_results.jpg');

%boat

subplot(1,2,1);

imshow(uint8(img\_boat));

title('Orig. Image');

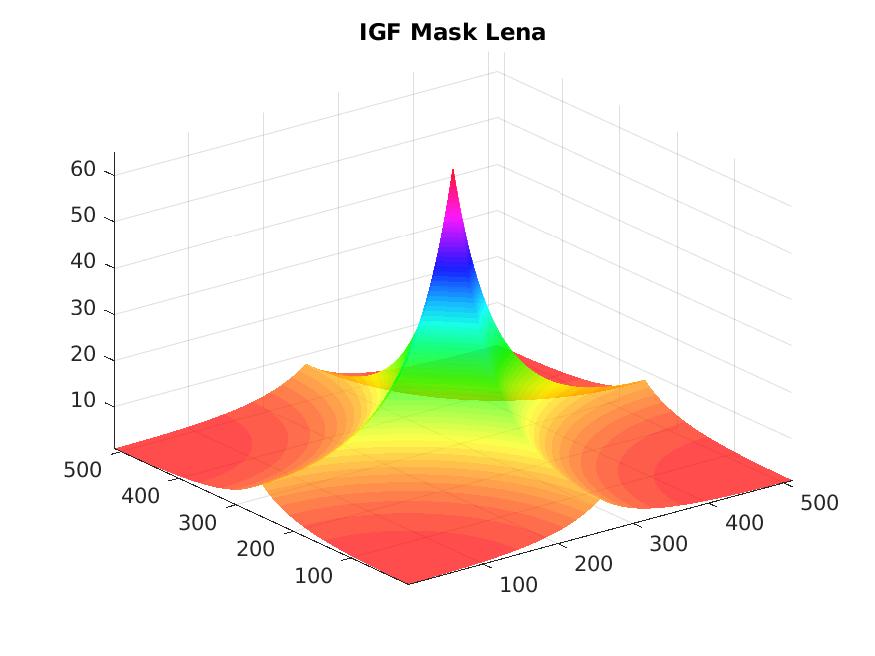
subplot(1,2,2);

imshow(uint8(ifft\_b));

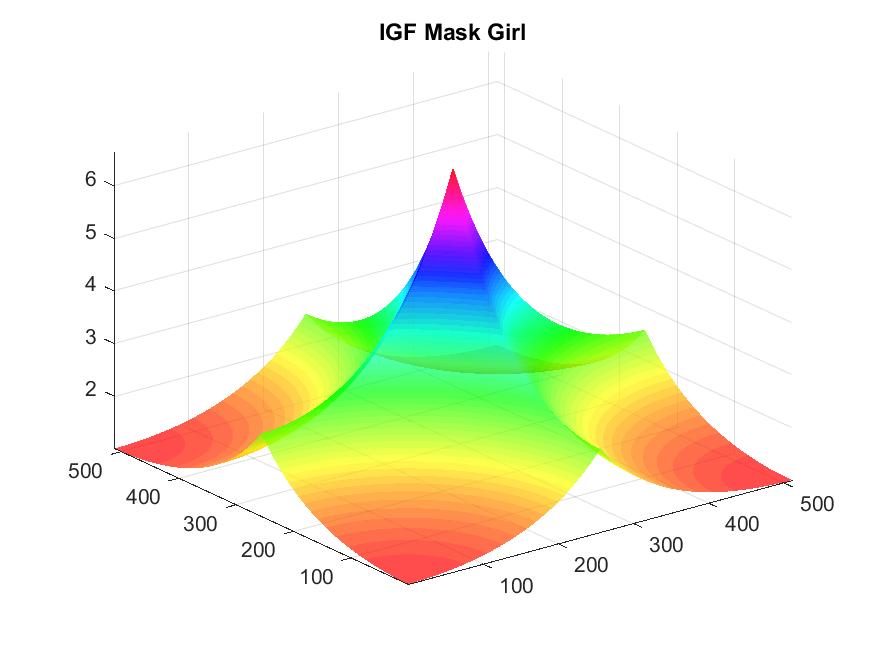
title('Filtered Image');

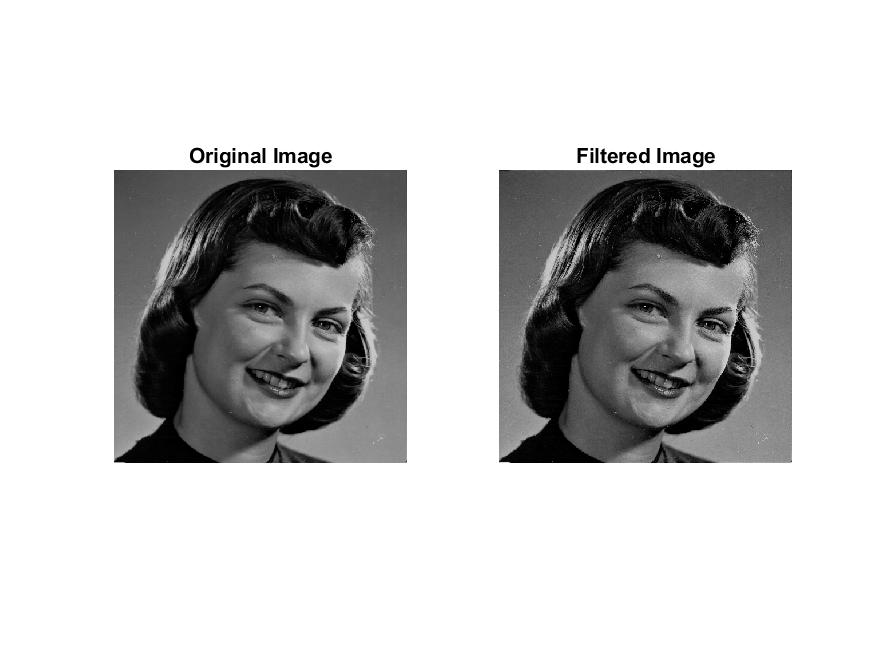
saveas(gca,'boat\_results.jpg');

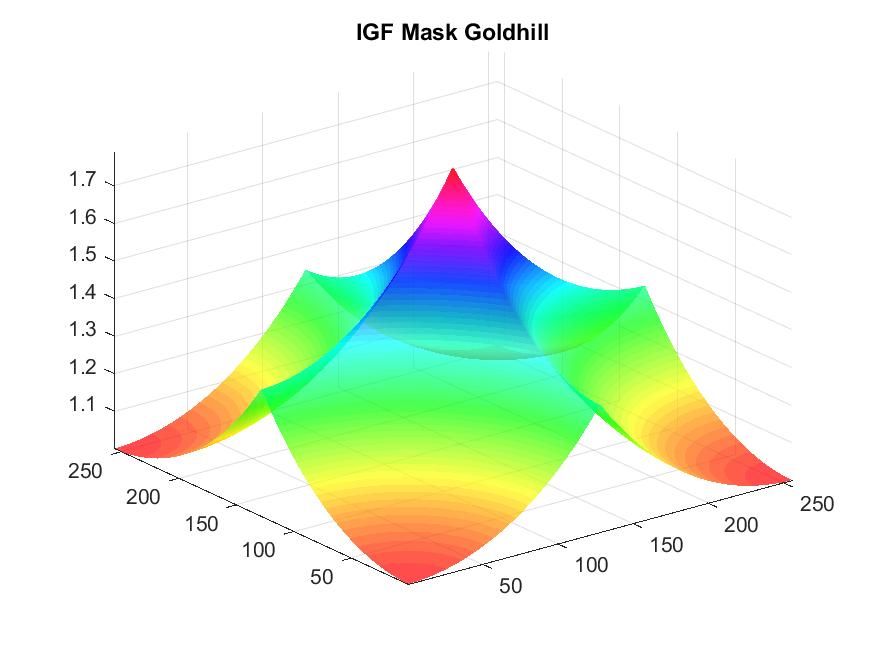
***Results:***

******

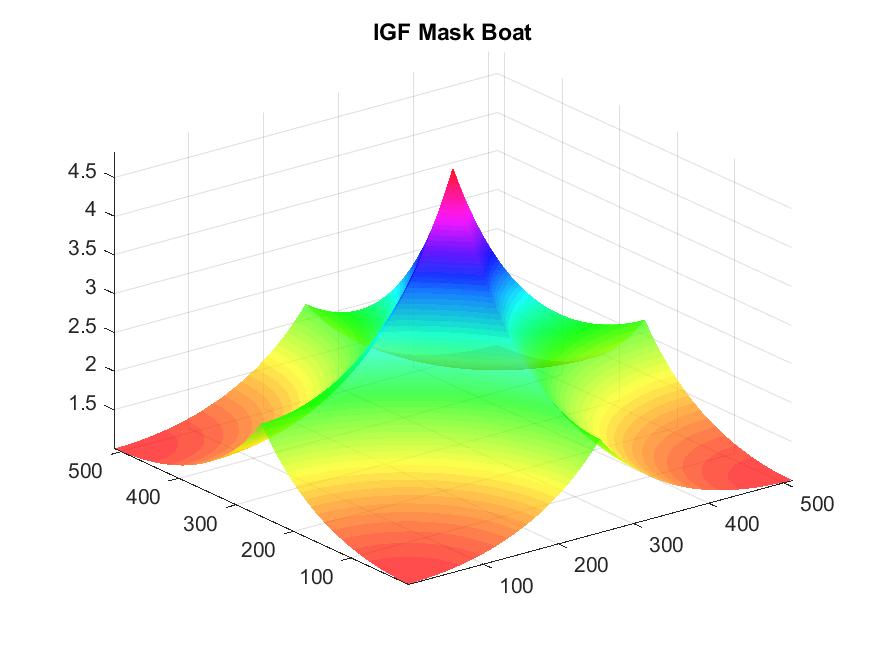


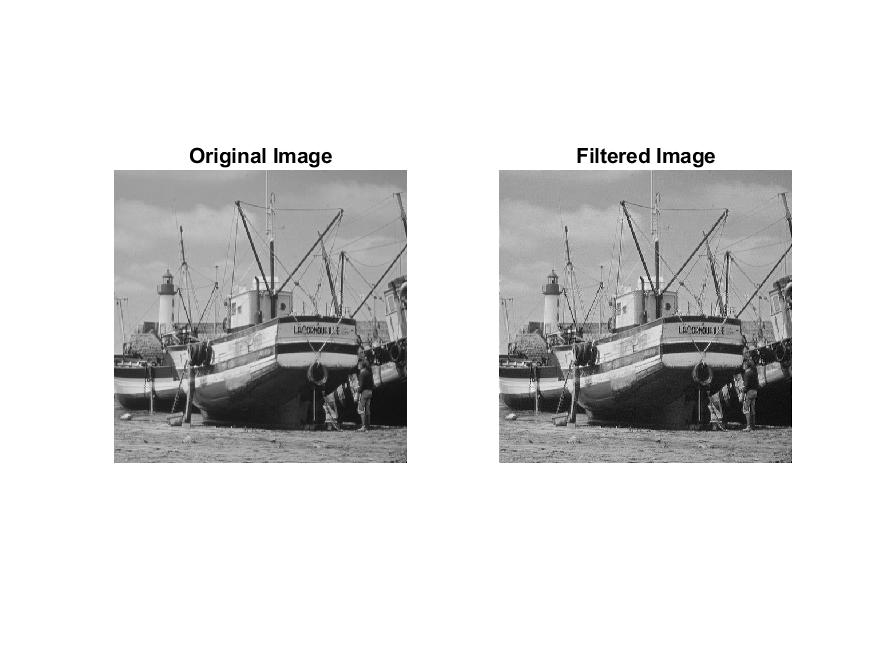




******

******

******

******

***Conclusion:***

* *A 2D DFT is done for each image.*
* *Inverse Gaussian filter is then applied*
* *Finally, inverse 2D DFT is done for each image*
* *The 3D IGF is also displayed for each image*