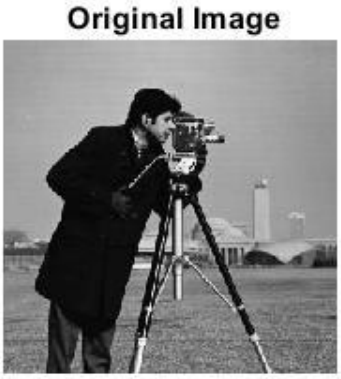
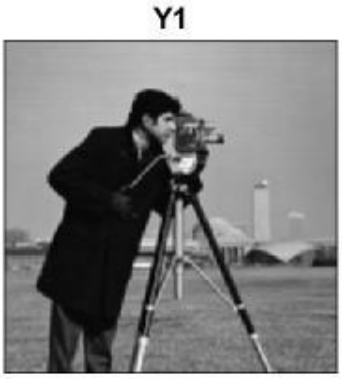
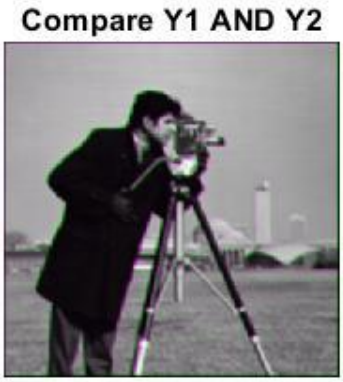
```
%We perform the Matlab code for Image Processing and perform 1D and 2D
convolution
close all
a=imread('cameraman.jpg');          %reads the image cameraman.jpg
ax1=subplot(2,2,1);
imshow(a)                           %Display the Original image
title(' Original Image')

h= [1/9 1/9 1/9; 1/9 1/9 1/9; 1/9 1/9 1/9] %Creates the 3X3 2D impluse response
matrix

y1= conv2(a,h);                     % Performs filtering using conv2() function
ax2= subplot(2,2,2);
imshow(uint8(y1))
title('Y1')                         % Filtered image Y1

y2= filter2(h,a);                   % Performs filtering using filter2() function
ax3= subplot(2,2,3);
imshow(uint8(y2))
title('Y2')                         %Filtered image Y2

ax4=subplot(2,2,4);
compare_1 = imshowpair(y1,y2);
title('Compare Y1 AND Y2')          % Compared image output

S= isfilterseparable(h) % if the return is logic 1 the matrix h is separable

%Our impluse response h(n1,n2) is separable.
%Separability can also be found by finding the rank of the matrix and if the rank
is 1 then it is separable.
%Separable matrix means the given matrix can be derived from convolution
%of two 1D vectors either row and column.

[U S V]= svd(h)
%we perform svd to find out row and column whoes multiplication can result in
matrix h

h1 = U*sqrt(S)                          % we take squareroot of S
h2 = sqrt(S)*V'
h1 = U(:,1)*sqrt(S(1,1))                %column
h2 = sqrt(S(1,1))*V(:,1)'               %row

% we multiply h1*h2 to check whether we get the h matrix again, so (separable_a =
h)

separable_a= h1*h2

for i= 1:256                              %Dconv(i,:) are the rows of the matrix
    Dconv(i,:)= conv(h2,a(i,:));          %conv the rows with the image a
end
for i = 1:256                             %DDconv(:,i) are the columns of the
matrix
    DDconv2(:,i) = conv(h1, Dconv(:,i));  %conv the column with the each column
of Dconv(i,:)
end

y3= uint8(DDconv2);                       % Image after 1D convolution
```

```
figure(2)
imshow(y3)
title('Y3');

figure(3)
compare_2 = imshowpair(y1,y3);
title('Compare Y1 AND Y3')                % Compared image output

%Our filter is a low pass filter. We will convert it to high pass filter

n=[1,0,0;0,0,0;0,0,0];                   %Defining delta function
h2=n-h;                                  % This forms the highpass filter

n_highpass=filter2(h2,a);
figure(4),
imshow(uint8(n_highpass))                %Displaying the highpass filtered image
title('Image after highpass filtering')

A=mean(mean(a));                          %taking the mean of the original image

B=mean(mean(n_highpass));                 %taking the mean of the high pass
filtered image

scalingfactor= A/B;

h2new=scalingfactor*h2                    %multiplying the the scaling factor to
the new h2

y4=conv2(a,h2);

figure(5)

imshow(uint8(y4))

title('Y4')                               %Filtered image output Y4

figure(6)

k=imadd(y2,n_highpass,'uint8');

imshow(k,[])

title('Reconstructed Image')              %Reconstructed image output
```

Outputs:-







Step 4 : Generated the matrix and it is 3X3 matrix with all the values 1/9 at each elements of matrix.

Step5: Use the function conv2() to filter the image using the filter with impulse response h(n1; n2). Call the resulting image y1. Display and generate a printout of y1.

Ans5 : Output Y1 in the above figure.

Step6: Repeat the above step with the filter2() command. Call the result y2. Compare y1 and y2. Using the type() command, list and study the function filter2().

Ans6: Output Y2 in the above figure. Compared output Compare Y1 AND Y2 in the above figure.

The relationship can be stated by saying that filter2() implements correlation, whilst conv2() implements convolution. In addition, filter2 implements an optimisation for the case of separable masks. This shouldn't affect significantly the numerical values of the results, just the speed. So in the image Y1 which is output after conv2() we see a sharp black border while in the image Y2 we don't see that. However, image Y2 is more blur than Y1. The pixel size for y1 is 258x258 and for y2 is 256x256.

Step7: Show that h(n1,n2) is separable; using MATLAB, generate the resulting filtered image using the 1-D conv() function. Call the image y3. Display and print y3. Compare y3 and y1.

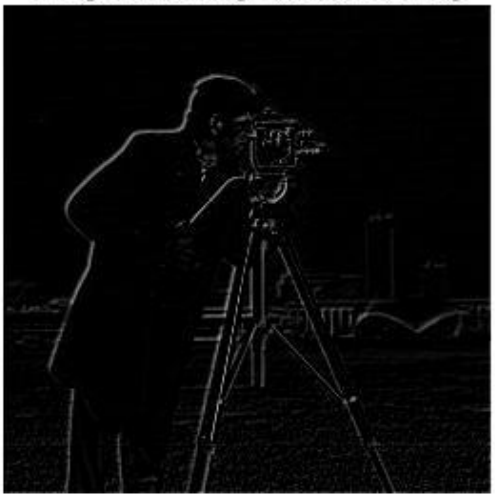
Ans 7: Matrix h is separable because it can be represented in the form of two vectors. The rank of the matrix is also 1 so if the rank of matrix is one it is separable. In the program I have used a function which takes the input as matrix and if the return is 1 then that matrix is separable. Performed the 1D convolution using “for loop” with first doing convolution of rows and then taking that matrix and performing the convolution with columns.

Y3

Step8: Is the filter of Step 4 h(n1; n2) lowpass or highpass? (JUSTIFY your answer). If it is highpass, show how you can transform it into a lowpass filter with impulse response h2(n1; n2); otherwise, show how you can transform it into a highpass filter h2(n1; n2). Scale h2(n1; n2) properly so that the image mean is unchanged by filtering (Hint: mean of image is multiplied by H2(0; 0), the Discrete-Time Fourier Transform of h2(n1; n2) evaluated at w1 = w2 = 0).

Ans8: Our impulse response h is low pass filter as when performed convolution of impulse response with image we got blur image. Lowpass filter smoothens the image and hence blurs the image. It averages the value of pixel with the neighbor pixels. Now, we have to convert the low pass filter to high pass filter and for that we use delta function and subtract the low pass filter from it.

Image after highpass filtering



We perform the scaling of the h2(n1,n2) so that the mean of the image is unchanged. I found the mean of the image and then the mean of the image after 1D convolution and then found the scaling factor which can be multiplied to the h2 and we get h2new.

Step9: Store the resulting filter impulse response h2(n1; n2) as a Matlab matrix h2. Repeat Step 5 using h2 instead of h. Call the resulting filtered image y4. Display and print y4. What is the effect of filtering with h2(n1; n2)?

Ans9: The image becomes very dark and sharp due to the effect of highpass filtering.

Y4

Reconstructed image :-