

|  |
| --- |
| **Title:** To implement low pass and high pass filtering in spatial and frequency domain |

**Objective:** To learn and understand the effects of filtering in spatial and frequency domain on images using Matlab.

**Expected Outcome of Experiment:**

|  |  |
| --- | --- |
| **CO** | **Outcome** |
| **CO4** | Design & implement algorithms for digital image enhancement, segmentation & restoration. |

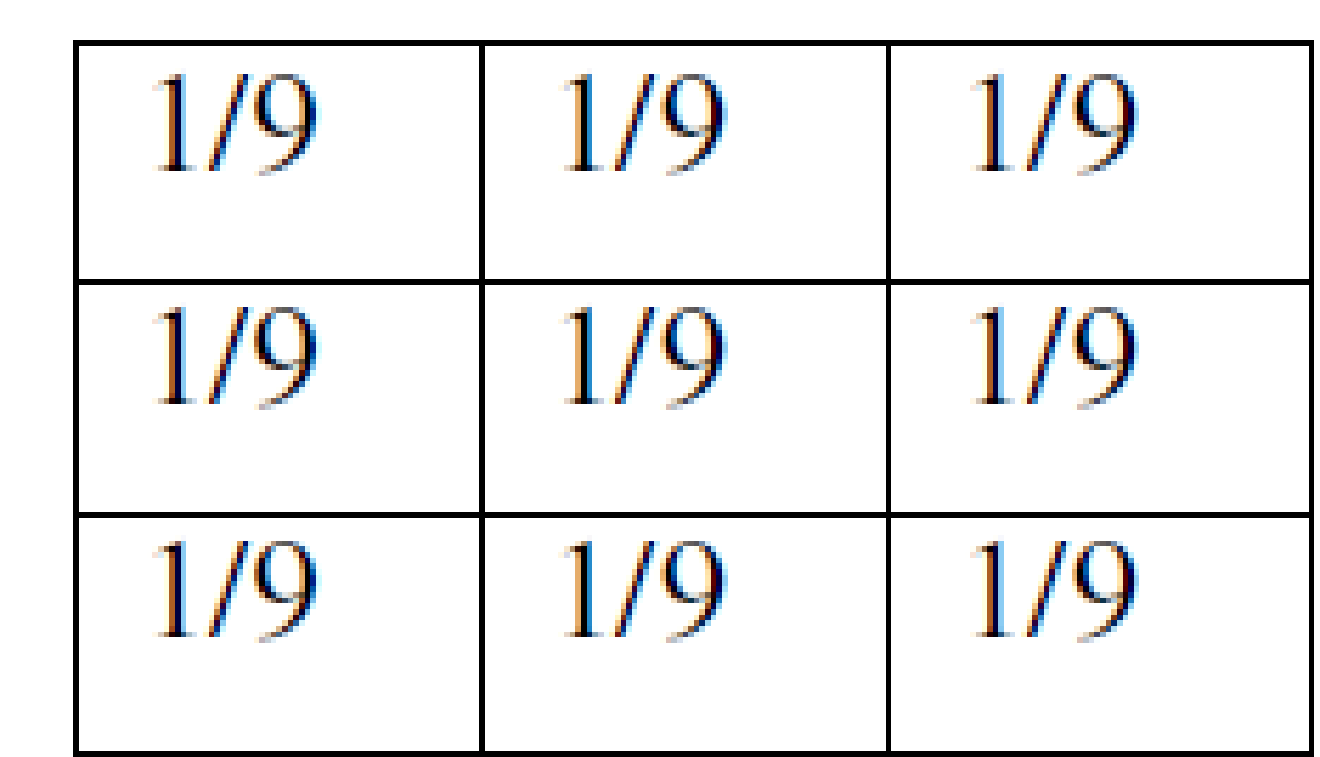
**Books/ Journals/ Websites referred:**

1. http://www.mathworks.com/support/
2. www.math.mtu.edu/~msgocken/intro/intro.html.
3. R. C.Gonsales R.E.Woods, “Digital Image Processing”, Second edition, Pearson Education
4. S.Jayaraman, S Esakkirajan, T Veerakumar “Digital Image Processing “Mc Graw Hill.
5. S.Sridhar,”Digital Image processing”, oxford university press, 1st edition."

**Pre Lab/ Prior Concepts:**

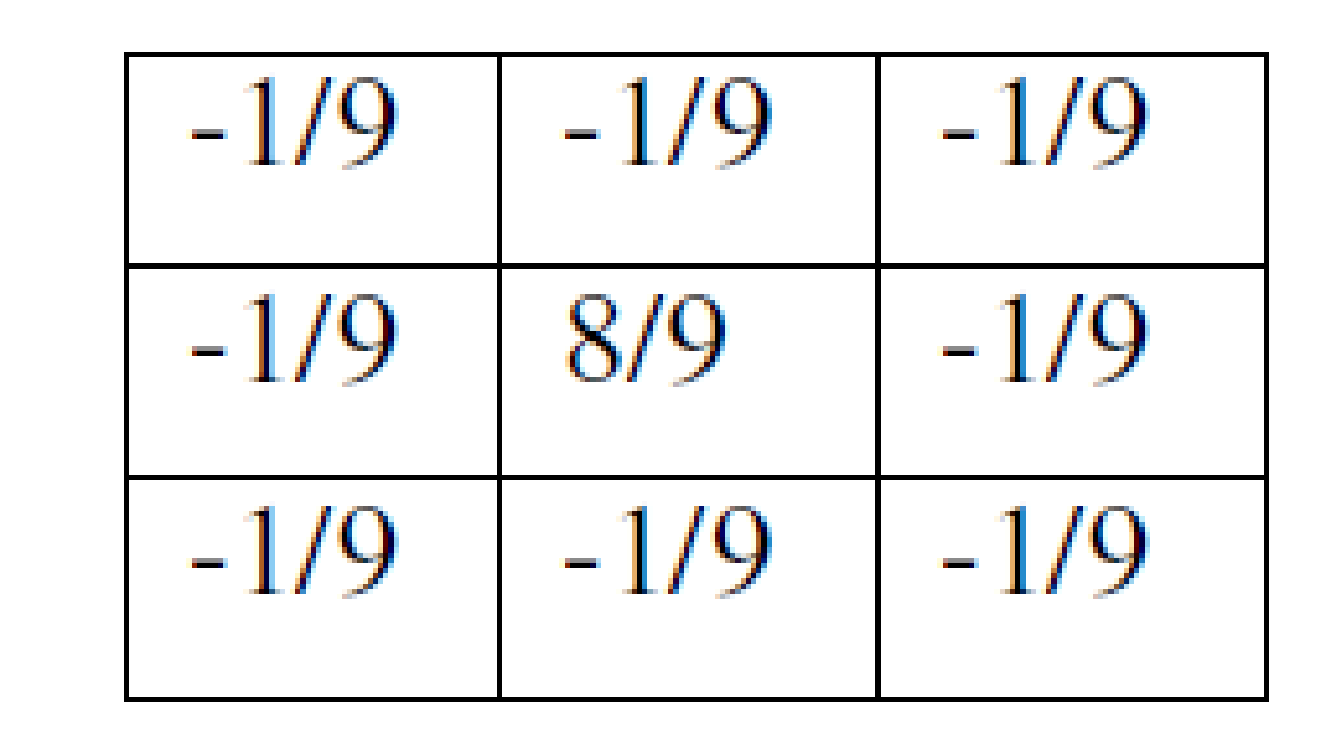
**Filtering in Spatial Domain:**

**Low pass filtering** as the name suggests removes the high frequency content from the image. It is used to remove noise present in the image. Mask for the low pass filter is:



One important thing to note from the spatial response is that all the coefficients are positive. We could also use 5 x 5 or 7 x 7 mask as per our requirement. We place a 3 x 3 mask on the image. We start from the left hand top corner. We cannot work with the borders and hence are normally left as they are. We then multiply each component of the image with the corresponding value of the mask. Add these values to get the response. Replace the centre pixel of the o/p image with these responses. We now shift the mask towards the right till we reach the end of the line and then move it downwards.

**High pass filtering** as the name suggests removes the low frequency content from the image. It is used to highlight fine detail in an image or to enhance detail that has been blurred. Mask for the high pass filter is:



One important thing to note from the spatial response is that sum of all the coefficients is zero. We could also use 5 x 5 or 7 x 7 mask as per our requirement. We place a 3 x 3 mask on the image. We start from the left hand top corner. We cannot work with the borders and hence are normally left as they are. We then multiply each component of the image with the corresponding value of the mask. Add these values to get the response. Replace the centre pixel of the o/p image with these responses. We now shift the mask towards the right till we reach the end of the line and then move it downwards.

**Median filtering** is a signal processing technique developed by tukey that is useful for noise suppression in images. Here the input pixel is replaced by the median of the pixels contained in the window around the pixel. The median filter disregards extreme values and does not allow them to influence the selection of a pixel value which is truly representative of the neighbourhood.

**Implementation Details:**

**Low Pass Filter:**

A = imread('taj.bmp');

B = zeros(130,130,'uint8');

LP = ones(3,3);

x = 2;

y = 2;

for i=1:128

for j=1:128

B(x,y) = A(i,j);

y = y + 1;

end

y = 2;

x = x + 1;

end

for i=2:129

for j=2:129

C(i,j) = filter(i-1,j-1,B,LP)/9;

end

end

subplot(1,2,1);

imshow(B);

title('Original Image');

subplot(1,2,2);

imshow(C);

title('Low Pass Filtered');

function sum = filter(x,y,B,F)

p=x;

q=y;

T = zeros(3,3);

for i=1:3

for j=1:3

T(i,j) = B(p,q);

q = q + 1;

end

q = y;

p = p + 1;

end

T = F .\* T;

sum = 0;

for i=1:3

for j=1:3

sum = sum + T(i,j);

end

end

end

Output:

****

**High Pass Filter:**

A = imread('taj.bmp');

B = zeros(130,130,'uint8');

HP = zeros(3,3);

for i=1:3

for j=1:3

if i == 2 && j == 2

HP(i,j) = 8;

else

HP(i,j) = -1;

end

end

end

x = 2;

y = 2;

for i=1:128

for j=1:128

B(x,y) = A(i,j);

y = y + 1;

end

y = 2;

x = x + 1;

end

for i=2:129

for j=2:129

C(i,j) = filter(i-1,j-1,B,HP)/9;

end

end

subplot(1,2,1);

imshow(B);

title('Original Image');

subplot(1,2,2);

imshow(C);

title('High Pass Filtered');

function sum = filter(x,y,B,F)

p=x;

q=y;

T = zeros(3,3);

for i=1:3

for j=1:3

T(i,j) = B(p,q);

q = q + 1;

end

q = y;

p = p + 1;

end

T = F .\* T;

sum = 0;

for i=1:3

for j=1:3

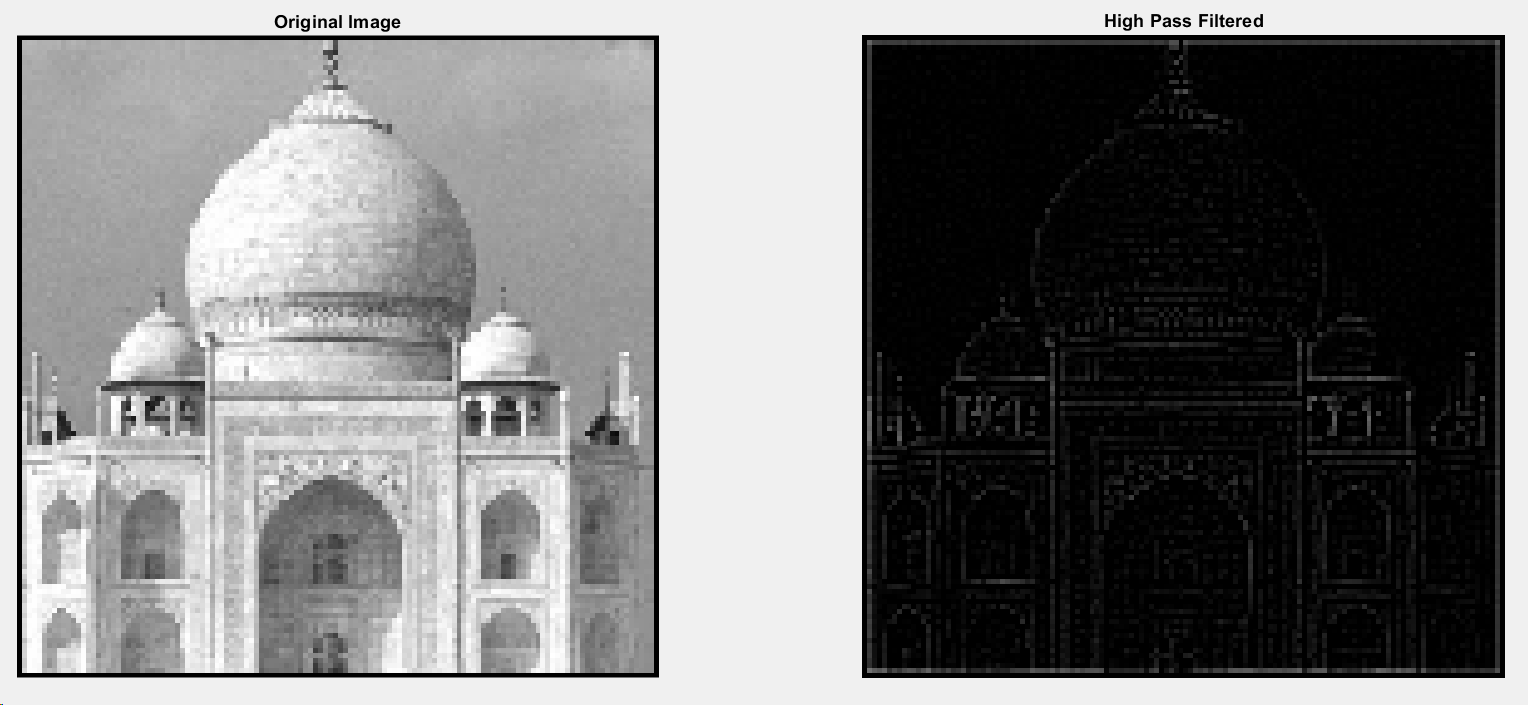
sum = sum + T(i,j);

end

end

end

Output:



**High Boost Filter:**

A = imread('taj.bmp');

B = zeros(130,130,'uint8');

HB = zeros(3,3);

for i=1:3

for j=1:3

if i == 2 && j == 2

HB(i,j) = 9;

else

HB(i,j) = -1;

end

end

end

x = 2;

y = 2;

for i=1:128

for j=1:128

B(x,y) = A(i,j);

y = y + 1;

end

y = 2;

x = x + 1;

end

for i=2:129

for j=2:129

C(i,j) = filter(i-1,j-1,B,HP)/9;

end

end

C=C+B;

subplot(1,2,1);

imshow(B);

title('Original Image');

subplot(1,2,2);

imshow(C);

title('High Boost Filtered');

function sum = filter(x,y,B,F)

p=x;

q=y;

T = zeros(3,3);

for i=1:3

for j=1:3

T(i,j) = B(p,q);

q = q + 1;

end

q = y;

p = p + 1;

end

T = F .\* T;

sum = 0;

for i=1:3

for j=1:3

sum = sum + T(i,j);

end

end

end

Output:



**Image with Noise:**

A = imread('taj.bmp');

B = zeros(130,130,'uint8');

x = 2;

y = 2;

for i=1:128

for j=1:128

B(x,y) = A(i,j);

y = y + 1;

end

y = 2;

x = x + 1;

end

nimg = imnoise(B,'salt & pepper',0.02);

subplot(1,2,1);

imshow(B);

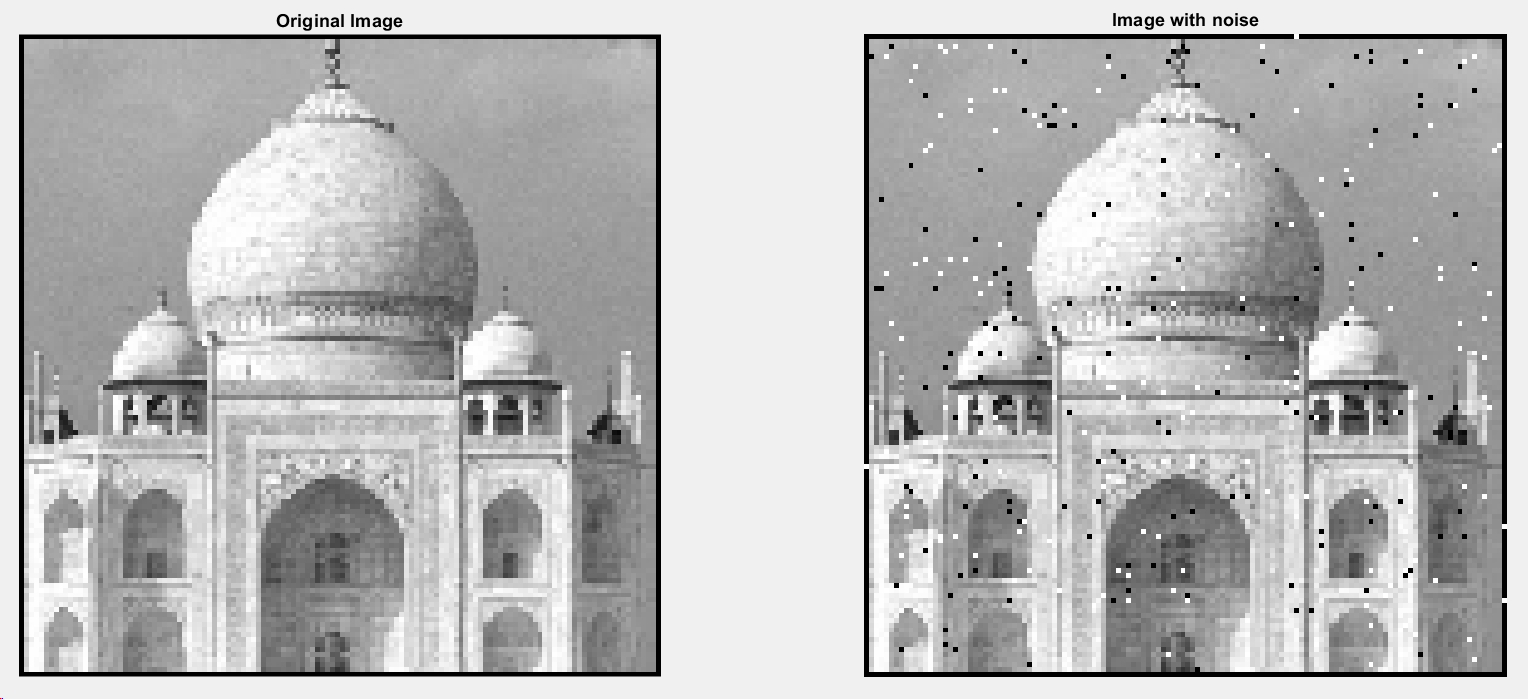
title('Original Image');

subplot(1,2,2);,

imshow(nimg);,

title('Image with noise');

Output:



**Image without Noise, Median Filter:**

A = imread('taj.bmp');

B = zeros(130,130,'uint8');

x = 2;

y = 2;

for i=1:128

for j=1:128

B(x,y) = A(i,j);

y = y + 1;

end

y = 2;

x = x + 1;

end

nimg = imnoise(B,'salt & pepper',0.02);

for i=2:129

for j=2:129

F(i,j) = mfilter(i-1,j-1,nimg);

end

end

subplot(1,2,1);

imshow(B);

title('Original Image');

subplot(1,2,2);

imshow(F);

title({['Image without noise'],['Median Filter']});

function val = mfilter(x,y,B)

temp = zeros(1,9);

k=1;

p=x;

q=y;

for i=1:3

for j=1:3

temp(k)=B(p,q);

k=k+1;

q=q+1;

end

q=y;

p=p+1;

end

for i=1:9

for j=1:9

if(temp(j)<temp(i))

t=temp(j);

temp(j)=temp(i);

temp(i)=t;

end

end

end

val = temp(5);

end

Output:



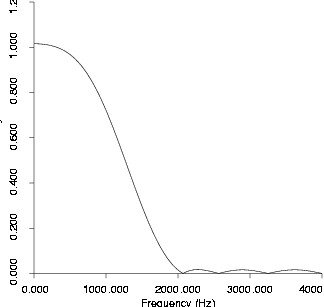
**Conclusion:-** The concept of high, low, median pass filters was implemented, the masks and their applications were understood. These filters were applied on a given image

**Post Lab Descriptive Questions**

1. Explain filtering in frequency domain

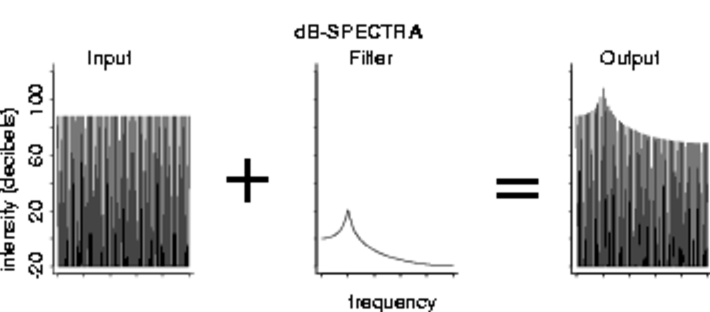
Ans.

Filtering is an operation on an acoustic signal that changes the shape of the spectrum of the signal. A filter is a device (or an computer program) which applies such a change to a signal. Examples of filters are the bass/treble controls on your stereo which attenuate the higher or lower frequency ranges in the music signal. Each filter has a characteristic spectral shape which describes which parts of the original signal will be attenuated and which will be left alone. An example for a low pass filter might be as below,



**An example of the spectral shape for a low pass filter**

The effect of applying a filter to a signal is to modify the shape of the signal's spectrum. In the frequency domain the effect of applying a filter to a signal is to *multiply* the spectrum of the signal by that of the filter. (Note that we multiply the raw spectra and not the dB spectra -- since dB spectra are logarithms they should be added to achieve the same effect since log(a\*b) = log(a)+log(b).) The result is a spectrum that combines the features of those of the input signal and the filter. An example is shown infigure below, of a simple impulse train input signal (which has a `flat' spectrum) filtered through a bandpass filter; the output signal shows the characteristics of both spectra.



**Applying a filter in the frequency domain**

Having applied a filter by combining the spectra, a new filtered signal can be derived by performing an inverse fourier transform on the new spectrum.

The question arises as to how we can find the spectrum of a filter, or indeed how a filter is represented digitally. A digital filter can be characterised by an acoustic signal called an *impulse response* -- literally this is the result of passing an impulse (a 1 followed by lots of zeros) through the filter. The spectrum of a filter can be derived by taking the spectrum of its impulse response.

Filters are an incredibly important topic in the study of speech acoustics.