Frequency Domain Filters

Examples
Example 1

- Apply to the moon.tif image:
  - Ideal low pass filter
  - Gaussian low pass filter
  - Butterworth low pass filter

```matlab
clear all
close all

f=double(imread('moon.tif'));
h = size(f,1);
w = size(f,2);
figure, improfile(f, [10 80], [h/2 h/2]);
```
Example 1 (continued)

```matlab
\% Create ideal low pass filter – a circle in middle of image
R = 30;  \% cutoff frequency
H = zeros(h,w);
for v=1:h
    for u=1:w
        if (v-h/2)^2 + (u-w/2)^2 < R^2
            H(v,u) = 1;
        end
    end
end
figure, imshow(H, []);

H = ifftshift(H);  \% put zero freq in upper left corner
figure, imshow(H, []);

F = fft2(f);
G = H .* F;
g = real(ifft2(G));
figure, imshow(g, []);
figure, improfile(g, [10 80], [h/2 h/2]);
```
Example 1 (continued)

• (repeat for Gaussian and Butterworth low pass filters)
Example 2

• Take an image (if it is a color image, convert it to grayscale using “rgb2gray”). Find the Fourier transform of the image using “fft2”. What is the value of the transform at the location (1,1)?

• Verify that it is the same as the total sum of all the pixels in the original image.
Example 3

• Perform Sobel filtering by multiplying in the frequency domain:
  – Create the 3x3 Sobel filter (shown below) that approximates a derivative in the horizontal (x) direction.

\[
\begin{pmatrix}
-1 & 0 & 1 \\
-2 & 0 & 2 \\
-1 & 0 & 1
\end{pmatrix}
\]

  – Take the Fourier transforms of the image and the Sobel filter. Multiply them, point by point, in the frequency domain. Then invert the result (using “ifft2”) to get the filtered image back in the spatial domain.
Example 3 (continued)

- Show that the resulting filtering image is identical to that obtained by spatial filtering with the Sobel filter (i.e., doing a correlation using imfilter), except for a small shift in position. (You can inspect the pixel values of each resulting image using “imtool”, and choosing the menu item “Tools->Pixel Region”.)

- Why is there a shift in position between the two images?
Example 4

• Apply notch filter to remove periodic noise in
  – Fig0464(a).tif
  – Fig0465(a).tif
Example 4 (continued)

```matlab
f = imread('Fig0464(a).tif');
figure, imshow(f, []);

F = fftshift(fft2(double(f)));
S = log(abs(F));
imwrite( S/max(S(:)), 'mask.tif');

% Edit image 'mask.tif' with another application such as "Paint".
% Draw black squares or circles at noise locations. Save it back
% to 'mask.tif'.
pause;

M = imread('mask.tif');
M = M(:,:,1); % Use only first band of color image
M = double((M>0)); % Threshold, so 0's are at noise locations

G = M .* F;
g = real( ifft2( ifftshift(G) ) );
```
Example 5

- Apply notch filter to enhance periodic structure in
  - ripples.jpg

- Use the same methods as before, but replace lines with

```matlab
M = imread('mask.tif');
M = M(:,:,1); % Use only first band of color image
M = double((M==0)); % Threshold, so 1's are at desired frequencies
% Enhance mask values at desired frequencies.
alpha = 5.0;
M = alpha*M + 1;
```