# Assignment A3: Frequency Domain Filtering

***CS 4640***
***Spring 2018***

**Assigned:** 5 February 2018

**Due:** 22 February 2018

For this problem, handin Matlab .m files for the functions described by the headers below. Note that one of these is a driver which creates inputs for each function and runs the function on those inputs to obtain the output.

Also, you are to turn in a document (A3.pdf) which includes a number of figures to demonstrate your Matlab functions. You are to use two images: im and trees found in the code/A3 subdir on my class webpage. The document should be organized as follows:

- *Section 1*: Power Spectrum: Show 2 rows each with 3 images: the original image, imshow(mat2gray()) and surf() of the test images.
- *Section 2*: Ideal Low Pass Filter: give 2 rows each for *im* and *trees* where the first row uses *imshow* to demonstrate the spatial filtering result (7x7 block), frequency filtering result, and combo of those, and row 2 uses *surf* to show the same 3 things.
- *Section 3*: Gaussian Low Pass Filter: give 2 rows each for *im* and *trees* where the first row uses *imshow* to demonstrate the spatial filtering result (7x7 block), frequency filtering result, and combo of those, and row 2 uses *surf* to show the same 3 things.
- *Section 4*: High Pass LoG Filter: give 2 rows each for *im* and *trees* where the first row uses *imshow* to demonstrate the spatial filtering result (7x7 block), frequency filtering result, and combo of those, and row 2 uses *surf* to show the same 3 things.
- *Section 5*: Sharpening: use the Gaussian low pass image and add the LoG filtered image to sharpen the image. Give 2 rows each for *im* and *trees* where the first row uses *imshow* to demonstrate the blurred image, the LoG image, and the sharpened result, and row 2 uses *surf* to show the same 3 things.

Some notes:

- Indent headers correctly (5 spaces indented lines)
- Do not exceed 72 characters per source line
- CS4640_A3_driver: should show that each function works

None of the functions should write to the interpreter, draw, etc.

```
function imc = CS4640_center(im)
% CS4640_center - center image for Fourier transform
% On input:
%     im (MxN array): input image
% On output:
%     imc (MxN array): centered image
% Call:
%     imc = CS4640_center(im);
% Author:
%     <Your name>
%     UU
%     Spring 2018
%
```

```
function imP = CS4640_power_spectrum(im,centered)
% CS4640_power_spectrum - power spectrum of FT of image
% On input:
%     im (MxN double array): input image
%     centered (Boolean): if 1 centered, else not
% On output:
%     imP (MxN double array): power spectrum
% Call:
%     imP = CS4640_power_spectrum(im,1);
% Author:
%     <Your name>
%     UU
%     Spring 2018
%
```

```
function im_pad = CS4640_pad(im,P,Q)
% CS4640_im_pad - pad an image for frequency domain filtering
% On input:
%     im_in (MxN double array): input image
% On output:
%     im_pad (PxQ double array): upper MxN is im; rest is 0
% Call:
%     imp = CS4640_pad(im,2*M,2*N);
% Author:
%     <Your name>
%     UU
%     Spring 2018
%


function im_depad = CS4640_depad(im,M,N)
% CS4640_im_depad - depad an image for frequency domain filtering
% On input:
%     im (PxQ double array): padded input image
% On output:
%     im_depad (MxN double array): upper MxN of im
% Call:
%     imdp = CS4640_depad(im,M,N);
% Author:
%     <Your name>
%     UU
%     Spring 2018
%


function g = CS4640_spatial_filter(im_in,H)
% CS4640_spatial_filter - spatial domain filtering
% On input:
%     im_in (MxN double array): input image
%     H (kxk double array, k odd): filter
% On output:
%     g (MxN double array): filtered image; i.e., im_in<conv>H
% Call:
%     im_lp = CS4640_spatial_filter(im,ones(7,7)/49);
% Author:
%     <Your name>
```

```
%     UU
%     Spring 2018
%


function g = CS4640_freq_filter(im_in,H)
% CS4640_freq_filter - frequency domain filtering
% On input:
%     im_in (MxN double array): input image
%     H (kxk double array, k odd): filter
% On output:
%     g (MxN double array): filtered image; i.e.,
%     F^{-1}[F(im_in).*F(H)]
% Call:
%     im_lp = CS4640_freq_filter(im,ones(7,7)/49);
% Author:
%     <Your name>
%     UU
%     Spring 2018
%




function CS4640_A3_driver
% CS4640_A3_driver - driver for A3 functions
% On input:
%     N/A
% On output:
%     N/A
% Call:
%     CS4640_A3_driver
% Author:
%     <Your name>
%     UU
%     Spring 2018
%
```