

## Assignment 2

### Frequency Image Enhancement using High Pass Filters

- a) Read the given image and convert its data type to double. Display the image.  
**Store the size of image in M and N. All the loops of x, y, u and v runs till image size.**
- b) Multiply the input image by  $(-1)^{x+y}$  To Center The Transform To  $U = M/2$  And  $V = N/2$ .
- c) Compute the fourier transform of the image in part (b) using “fft2”. Display the image.
- d) Multiply the image in part (c) with **ideal high pass filter** using  $D_0=50$ .
- e) Compute inverse fourier transform of image in part (d) using “ifft2”
- f) Obtain the real part of the image in part (e) and multiply again by  $(-1)^{x+y}$  to Cancel The Multiplication Of The Input Image. Display the sharpened image.
- g) Multiply the image in part (c) with **butterworth high pass filter** using  $D_0=50$  and  $n=2$ .
- h) Compute inverse fourier transform of image in part (g) using “ifft2”
- i) Obtain the real part of the image in part (h) and multiply again by  $(-1)^{x+y}$  to Cancel The Multiplication Of The Input Image. Display the sharpened image.
- j) Multiply the image in part (c) with **Gaussian high pass filter** using  $D_0=50$  and  $n=2$ .
- k) Compute inverse fourier transform of image in part (j) using “ifft2”
- l) Obtain the real part of the image in part (k) and multiply again by  $(-1)^{x+y}$  to Cancel The Multiplication Of The Input Image. Display the sharpened image.

### Image Restoration

- a) Read the image and convert it to grayscale and display it.
- b) Add the Gaussian noise in the image using “imnoise”. Display the image.

- c) Use arithmetic mean filter image to remove the Gaussian noise using "filter2". Display the image.
- d) Use wiener filter to remove the Gaussian noise using "wiener2". Display the image.
- e) Add the salt and pepper noise in the grayscale image using "imnoise". Display the image.
- f) Use median filter to remove salt and pepper noise using "medfilt2". Display the image.
- g) Now add salt and pepper noise using imnoise with varying noise density (0.05, 0.1, 0.3) and apply medfilt2 with different window sizes
- h) Construct a code for adaptive median filtering and compare its results with the ones you got in (g)