TEL411 – Digital Image Processing

Doutsi Effrosyni, PhD

Assignment 8

Due date: Wednesday, December 9, 2020

1. Read the 'cameraman.tif' image I (512x512 pixels).
2. Rescale image I and generate a new Image I_{new} of the size 30x30 pixels (you can use the *imresize()* function with the default parameters).
3. Compute the Fast Fourier Transform of I_{new} (you can use the *fft2()* function).
4. Illustrate the rescaled image I_{new} and its Fast Fourier Transform (you need to use functions *fftshift()*).
5. Build a 2D Gaussian function of the size 9x9 and standard deviation $\sigma = 0.8$ according to the following definition
$$G = \frac{1}{2\pi\sigma^2} e^{-\frac{(y^2+x^2)}{2\sigma^2}}.$$
You are allowed to use the *meshgrip()* function.
6. Compute the Fast Fourier Transform of the filter.
7. Illustrate the Gaussian filter in spatial and in frequency domain (you can use the *mesh()* function). Comment these results.
8. Compute the convolution of the image by the Gaussian filter (use the *conv2()* function).
9. Compute the multiplication between the FFT of the rescaled image and the FFT of the filter (you need to be careful of the size).
10. Compute the Inverse Fast Fourier Transform of step 9 (you are allowed to use the *ifft2()* function).

11. Compute the convolution using the Toeplitz matrix (you are allowed to use the *toeplitz()* function).
12. Illustrate the outcome of steps 8, 10 and 11.
13. Compute the Mean Square Error between steps 8,10 and 11 (You are allowed to use the *immse()* function).

What to turn in

You should turn in your code and a short report. You need to include all the requested images, report the MSE values and comment your results.

Toeplitz Matrix

The Toeplitz matrix have constant entries along their diagonals. A special form of Toeplitz matrix called "circulant matrix" is used in applications involving circular convolution and Discrete Fourier Transform (DFT). Matrix h below describes the Toeplitz matrix.

$$\begin{bmatrix} y[0] \\ y[1] \\ y[2] \\ y[3] \\ y[4] \\ y[5] \end{bmatrix} = \begin{bmatrix} h[0] & 0 & 0 \\ h[1] & h[0] & 0 \\ h[2] & h[1] & h[0] \\ h[3] & h[2] & h[1] \\ 0 & h[3] & h[2] \\ 0 & 0 & h[3] \end{bmatrix} \begin{bmatrix} x[0] \\ x[1] \\ x[2] \end{bmatrix}$$