E9 241: DIGITAL IMAGE PROCESSING

ASSIGNMENT 5; DUE OCTOBER 17/18, 2019

TOPIC: IMAGE FILTERING

PROGRAMMING TASKS (DUE OCTOBER 17/18)

(1) Write a Matlab/Python script to accept a grayscale image (size $N \times N$) as input and generate an extended image of size $3N \times 3N$ under the following boundary conditions: (i) zero-padding; (ii) mirror-symmetry; (iii) periodic extension. These three options should appear as radio-buttons in your implementation with the default being mirror-symmetry boundary conditions. Report your results on the *Cameraman* image.

(2) Implement an $L \times L$-point moving-average filter with a complexity independent of L. Your program must have the option of selecting the boundary conditions from one of the options above and then produce the appropriate output. Your program must also have the provision for selecting the value of L through a slider, starting from a minimum of 3 to a maximum of 19 in steps of 2. Report your results on the *Baboon* image. The *Baboon* image is often used to refer to the effect of processing on the detailed image features such as the whiskers of the Baboon.

(3) Write a Matlab/Python script to accept a grayscale image as input and generate a noisy image of a desired peak signal-to-noise ratio (PSNR) as the output. The noise should be zero-mean, white, Gaussian, and additive. The PSNR value should be available for selection on a slider ranging from 10 dB to 30 dB. The display of the noisy image should get automatically updated when the slider is moved. The program should have the option to select any grayscale image from a folder of grayscale images, not all of them having the same size/dimension. The program should execute successfully and correctly regardless of the image size. Recall that the PSNR is defined as the ratio of the maximum value of the noise-free image to the noise standard deviation, measured on a logarithmic scale. Report your results on the *Cameraman* image.

(4) Implement an iterated $L \times L$ moving-average filter (no. of iterations N_{iter}) to perform approximate Gaussian filtering of a certain variance input as a numerical parameter by the user. The parameters N_{iter} and L should be computed based on the input variance. Report your results on noisy versions of the *Cameraman* image as the input. Consider PSNRs in the range 10 dB to 30 dB. Determine the PSNR of the output. The PSNR gain is the difference between the output PSNR and the input PSNR measured on a log scale. Show a plot of the PSNR gain versus the input PSNR. Since the noise is random and changes from one realization to another, the PSNR gain would change from one realization to another even if the input PSNR value remains the same. For each input PSNR value, repeat the experiment 20 times and show the average PSNR gain curve and the one-σ standard deviation (`errorbar`) about the average versus the input PSNR. Report your results for the *Baboon* and *Peppers* images. Also show the input and output images for PSNR values of 20 dB to serve as an illustration.

(5) Write a Matlab/Python script to perform median filtering over a sliding window of size $L \times L$, where L is an odd number. L takes a minimum of 3 and a maximum of 23, in steps of 2. The value of L should be selected by a slider and the output should get updated dynamically. Report your results on the *Orion* image. The image is courtesy of Adrian Pelletier (`unsplash.com`).
