**EE610 Image Processing**

**Assignment 3 – Wavelets and Image Restoration**

**Instructions:**

1. Form a group of 2 people. Only one person needs to submit the assignment, and both with get equal points.
2. Use python and libraries such as numpy, matplotlib, scipy.signal, PIL, and OpenCV.
3. Submit your assignment, preferably a single .ipynb file, by Oct 16, 2022, 11:59pm on Moodle at: https://moodle.iitb.ac.in/mod/assign/view.php?id=95510
4. Include copious comments about your thoughts in choosing the operations for a particular problem, and your observations about what worked and what did not work, including possible reasons.
5. Comment every line of code to demonstrate understanding of how the code works. Write as much code as you can on your own, but you can use functions from the aforementioned libraries.
6. Cite sources referred at the place where they were used, and include a reference section at the end.

**Problems:**

1. Write the following basic functions using numpy and scipy signal libraries only in an efficient manner:
    a. Add additive white Gaussian noise (AWGN) with variable variance to any given image, whether gray scale or color. The noise should be independent in RGB channels for color images. [1]
    b. Measure PSNR (with 255 as the default peak, but it should also be variable) between a restored image and a ground-truth image. Do not import a PSNR function directly, even if available. [1]
    c. Measure SSIM (with power 1 for all three elements -- mean, variance, correlation -- which can be varied too) between a restored and a ground-truth image. Do not import an SSIM function directly, even if available. [1]
    d. Isotropic Gaussian blurring (Gaussian low pass filtering) of an image for a given variance. [1]
2. Write the following denoising functions for wavelet transforms (you can use a library for DWT and IDWT):
    a. Hard thresholding: set to zero any detail (LH, HL, HH) coefficient that is below a threshold (threshold should be variable). [1]
    b. Soft threhsolding: set to zero any detail (LH, HL, HH) coefficient that is below a threshold (threshold should be variable), and subtract the threshold from those that are above the threshold. [1]
    c. The method proposed in the paper "Image Denoising using Neighbouring Wavelet Coefficients" by Chen et al. [3]
3. Compare PSNR and SSIM (with plots) when you vary the noisy added to clean color images by varying:
    a. Noise variance [1]
    b. Gaussian low pass filtering with various variances [1]
    c. The two thresholding methods, where threshold is given by $\lambda = \sigma\sqrt{2\log n^2}$ (as suggested by Donoho et al.), where $\sigma$ is noise variance, and $n$ is window size. [2]
    d. The method of Chen et al. [2]
    e. When the method is independently applied to RGB channels, versus when the method is applied to I channel of HSI transform. [2]
4. Compare a few of the methods from part 3 on real noisy images, where noise variance is not known, by assuming various values of the noise variance as trial and error. [2]
5. Optional, for fun only: Repeat Wiener filtering of assignment 2 on ImgBlur.png with a filter that is defined by np.ones((2,11)), and check if you can now read the number.

Links: https://www.egr.msu.edu/~aviyente/neighshrink.pdf https://docs.scipy.org/doc/scipy/reference/signal.html
https://iitbacin.sharepoint.com/sites/EE610-2022/Shared%20Documents/Forms/AllItems.aspx?id=%2Fsites%2FEE610%2D2022%2FShared%20Documents%2FClass%2FAssignments