REU Site Program in CVMA

Lab 4 – Exercise on Filter Techniques in Frequency Domain

Problem 1: Exercises on Low-pass and High-pass Filters in the Frequency Domain

a) Design a butterworth low-pass filter of order 2 with a cutoff frequency of 80 in the frequency domain. Obtain the filtered image by filtering the original image *Sample* with the designed butterworth filter. Display the original image, the butterworth low-pass filter **(treat it as an image)**, and the filtered image in figure 1 with the appropriate titles.
b) Repeat the step a) by using the corresponding high-pass filter. Display the corresponding images (original image, high-pass filter, and filtered image) in figure 2 with the appropriate titles.
c) Close all figures and all variables in the workspace.

Problem 2: Remove Additive Cosine Noise

The noisy image *boy_noisy.gif* has been generated by adding some noise in the form of a cosine function. Your goal is to remove the cosine interference. This can be done as follows:

a) Compute the centered DFT of the noisy image.
b) Compute the magnitude and find the frequencies corresponding to the ***four largest distinct magnitudes*** (do not consider the magnitude at the center – a very large value)
c) Replace the value of each of the four frequencies, which correspond to the four largest distinct magnitudes, by the average of its 8 neighbors. (Make sure you will perform this change on their symmetric points to ensure the proper reconstruction).
d) Take the inverse DFT transform and display the original image and the resulted image side-by-side in figure 1 with the appropriate titles.
e) Explain why the four largest distinct values of the magnitude were chosen to do the processing via Matlab display command.
f) Close all figures and variables in the workspace.

Problem 3: Preliminary Wavelet Transform

a) Apply a maximum-level "db2" wavelet decomposition on *Lena* by using appropriate Matlab function(s). **You need to compute the maximum decomposition level instead of hard coding it**. Apply the inverse wavelet transform to restore the image. Use "if-else" statement to compare your restored image with the original image so the appropriate message indicating the equality or inequality between these two images should be displayed. (Note: The original image and the restored image should be the same)
b) Apply a 4-level "db2" wavelet decomposition on *Lena* by using appropriate Matlab function(s). **Independently** perform the inverse wavelet transform after each of the following operations:
   b.1) Set all the approximation coefficients as 0's.
   b.2) Set the first level detail coefficients as 0's.
   b.3) Set the second level detail coefficients as 0's.
   Note: Please solve the above three questions without using any Matlab built-in function.
   Display the reconstructed image for each of the above three operations. Summarize the reasons for current appearance of the reconstructed image using the Matlab display function.
c) Close all figures and all variables in the workspace.