EEE 391

Basics of Signals and Systems
Spring 2016–2017
Computer Assignment 3
due: 3 May 2017, Wednesday, by 23:55 on Moodle

Choose one of the four 512×512 gray scale images provided on Moodle (houses, clown, couple, bridge). The images are provided as 512×512 bmp (bitmap) files. Use the command: `A=imread('nameofimage.bmp');` to read the image file 'nameofimage.bmp' and convert it to a matrix `A` whose values range from 0 to 255. Use the command: `J=mat2gray(A, [0 255]);` to take the matrix `A` and convert it to a matrix `J` whose values are between 0 and 1. The value 0 represents black and the value 1 represents white and intermediate values are gray scale values.

Let n and m be the independent variables (integer indices) indexing the pixels (matrix elements) along the vertical and horizontal dimensions of the image, respectively.

1) Implement a one-dimensional M-point averaging filter to this image for $M = 11$, $M = 31$, $M = 61$, such that each pixel value is replaced by the equal-weighted average of its $(M-1)/2$ neighbors to its left, $(M-1)/2$ neighbors to its right, and the pixel value itself. For instance, for $M = 3$, it would be like:

$$y[n, m] = \frac{1}{3}\left(x[n, m-1] + x[n, m] + x[n, m+1]\right)$$

Notice that the averaging is one-dimensional (1D), that is, along only one of the dimensions (horizontal dimension). Assume that values lying outside of the image are zero while averaging.

Display the averaged images and the magnitude of the frequency response function $H(e^{j\hat{\omega}})$ of the filter for $M = 11$, $M = 31$, $M = 61$ for $-\pi \leq \hat{\omega} \leq \pi$.

Display the averaged images as 2D gray scale images, not as 2D or 3D graphs/plots of values. To display an image provided as a matrix `Y`, use the command: `imshow(Y)`

Although a 2D system is involved, the index n has no role so you can treat the system effectively as a 1D system while finding the frequency response function; that is, as if:

$$y[m] = \frac{1}{3}\left(x[m-1] + x[m] + x[m+1]\right)$$

The frequency variable $\hat{\omega}$ is a continuous variable. While calculating $H(e^{j\hat{\omega}})$ you will have to choose suitable samples of $\hat{\omega}$. Choose these samples close enough so that the graph of $|H(e^{j\hat{\omega}})|$ versus $\hat{\omega}$ looks smooth.

Be sure to label your plots and the axes appropriately and completely so we can understand what different frequencies mean with respect to the images. To do this you will have to learn the MATLAB commands that allow labeling the axes etc.

Comment on:

(i) How can you describe the visual effect created by the filter?

(ii) What happens to the details in the image? How is this affected by changing the value of M?

(iii) Compare the visual effect in the dimension n versus the dimension m (vertical versus horizontal)?

(iv) Recall that a signal can be written as the weighted sum of cosine/sines of different frequencies. Referring to the magnitude of the frequency response, describe the effect of the filter in terms of what happens to the different frequency components in the image. How is this affected by changing the value of M?

(v) Comment on what happens close to the edges of the image? How is this affected by changing the value of M?

Now, add a random variable to each element of the matrix, which will be a model for noise. The random variable should be uniformly distributed between $[-\frac{c}{2}, \frac{c}{2}]$. To obtain this, you can use a standard random number generator that generates random numbers between $[0, 1]$, subtract 0.5 and multiply by c. Use the values of $c = 0.2$ and $c = 1$.

For these two values of c, apply filters with $M = 11$, $M = 31$, $M = 61$, and display the images, so you should produce six images. Compare these with the noise-free images you displayed earlier and comment on:

(vi) Does averaging help to reduce/eliminate the noise? How is this affected by changing the value of M?

(vii) Is there an undesirable side-effect of eliminating the noise in this manner? How is this affected by changing the value of M?

(viii) What do you think is the best choice of M here?

2) Using the same image that you chose for part 1) as an input, implement a first differencer along the horizontal dimension as follows:

$$y[n, m] = x[n, m] - x[n, m - 1]$$

Assume that values lying outside of the image are again zero while calculating the output.

Display the output image as a 2D gray scale image.

Derive the magnitude of the frequency response of the filter and plot it in a manner similar to that in part 1).

Be sure to label your plots and the axes appropriately and completely so we can understand what the different frequencies mean with respect to the images.

Comment on:

(i) How can you describe the visual effect created by the filter?

(ii) Compare this to what you would theoretically expect it to be.

(ii) Compare the visual effect in the dimension n versus the dimension m (vertical versus horizontal)?

(iv) Referring to the magnitude of the frequency response, describe the effect of the filter in terms of what happens to different frequencies in the image.

Submit the results of your own work in the form of a well-documented report on Moodle. Borrowing full or partial code from your peers or elsewhere is not allowed and will be punished. Please include all evidence (plots, screen dumps, MATLAB codes, MATLAB command window print-outs, etc.) as needed in your report. Append your MATLAB code at the end of your assignment, do not upload it separately. The axes of all plots should be scaled and labeled. Typing your report instead of handwriting some parts will be better. Please do not upload any photos/images of your report. Your complete report should be uploaded on Moodle as a single good-quality pdf file by the given deadline. Please DO NOT submit any hardcopies or files by e-mail or on memory stick/CD.