BME 790.01: Engineering Programming and Signal Processing
Fall 2013

Homework Assignment #2
Assigned Monday, September 23, 2013
Due: Friday, October 4, 2013, in class

First, read Chapter 2, sections 2.1–2.14 of Hayken, 2nd ed. and then complete the following problems.

1. Problem 2.34 (a), (b), and (c)
2. Problem 2.38 (a) and (b) and answer the following questions:
   (a) Is this $h(t)$ causal, memoryless, linear, time invariant? How can you tell?
   (b) How could you change it's causality? I.e. if your $h(t)$ is causal, how could you make it non causal, and vice versa? Sketch your new $h(t)$.
   (c) What kind of filter is $h(t)$? How do you know?
3. Problem 2.40 (a), (c), and (k)
4. Redo your work for (3) for the part of your choice using MATLAB's conv function and verify that you get the same answer. You will need to convert your continuous function into a discrete one. Plot your result as a function of time and answer the following questions:
   (a) Using the hold on command, plot the mathematical result of your convolution as determined by Problem (3) on top of the MATLAB computed convolution. To avoid color printing experiment with MATLAB's plot() command, using '*' for black and white display purposes. Does MATLAB give you the same answer? Why or why not? Does math work?
   (b) How does changing your sampling rate (dt) of the continuous function affect your result? Try changing your sampling rate and plot the result on top of your original solution using the 'hold on' command. Hint: if the results are not approximately the same something is wrong.
5. A common BME application of discrete convolutions is image processing. To that end, we're going to investigate using convolutions to alter images using our friend MATLAB.
   (a) Create your 'image', I[n], in MATLAB which in our case is going to be a 1-D array of length 128 pixels, with pixels 40 to 82 equal to 1, and all other pixel values are 0. This is equivalent to a rect function, and is the simplest form of image we can use. Plot I[n] as a function of n.
   (b) Design an impulse response function, $h_1[n]$, that is equivalent to one of the functions in Problem 2.34. Is your $h_1[n]$ causal? Is it memoryless? What do you hypothesize your $h_1[n]$ will do to an image? Could you classify it as a low pass or high pass filter?

(c) Convolve $h_1[n]$ with I[n] and plot the result. How was I[n] affected? Was your hypothesis correct? Why or why not?

(d) Create a second impulse response function, $h_2[n]$ that is another function from 2.34 (your choice). Is this $h_2[n]$ causal? Memoryless? What do you hypothesize $h_2[n]$ will do to your image? Is your impulse response function a low pass or high pass filter?

(e) Convolve $h_2[n]$ with I[n] and plot the result. Was your hypothesis correct?

(f) Download the file head.mat (which corresponds to an MRI brain image) and import it into MATLAB

(g) Use imagesc or imshow to visualize the image the brain. Note: for some reason visualization of images in MATLAB is particularly annoying as frequently the image does not display as you expect it, usually because your image is out of the range of the image viewer. If you get an entirely white image that means your values of the image are too large (try normalizing your matrix). If you get an entirely black image your values are too small (again try normalizing your matrix).

(h) Convolve the brain image with $h_1[n]$ and view the result (search conv or conv2 in MATLAB). Did it do what you expected? Note: You may have to modify your $h_1[n]$ for 2D convolution in MATLAB.

(i) Convolve the brain image with $h_2[n]$ and view the result. Did it do what you expected?

(j) Does the filter you've created affect both x and y directions? Why or why not?

(k) Convolve the brain image with $h_1[n]'$, that is the transpose of $h_1[n]$. How did the convolution affect the brain image this time? How was the result different than from part h?

(l) Convolve the brain image with $h_2[n]'$, again the transpose of $h_2[n]$. How did the convolution affect the brain image and how was it different than part i?

(m) Modify either your $h_1[n]$ or $h_2[n]$ to affect your image in both x and y directions. Mesh plot (mesh3) your new h[n] and perform a 2D convolution with your brain image. Did it work as expected? Be sure to display all brain images with clearly labeled titles in a subplot format. You should have a total of 6 brain images (original, $h_1[n]$, $h_2[n]$, $h_1[n]'$, $h_2[n]'$, and 2D $h[n]$) in one figure.

(n) Extra Credit! Pretend either your $h_1[n]$ or your $h_2[n]$ are continuous functions (see Problem 2.40 as many of the functions used are continuous versions of the discrete functions seen in 2.34). Change your sampling rate of the $h[n]$ function, that is, sample your continuous function more often or less often to generate a result that is wider or narrower than your original $h[n]$. To see a result, you may need to increase/decrease your sampling by a factor of 2 or more. How will this new kernel, $h[n]$, affect your image? Convolve your brain image with this new filter and plot the result. Why does sampling rate change the resultant image?