GEORGIA INSTITUTE OF TECHNOLOGY

SCHOOL of ELECTRICAL and COMPUTER ENGINEERING

ECE 6250 Fall 2018

Problem Set #5

Assigned: 26-Sep-18
Due Date: 3-Oct-18

As stated in the syllabus, unauthorized use of previous semester course materials is strictly prohibited in this course.

All assignments will be due at the begining of class on the due date.

PROBLEM 5.1:

Using your class notes, prepare a 1-2 paragraph summary of what we talked about in class in the last week. I do not want just a bulleted list of topics, I want you to use complete sentences and establish context (Why is what we have learned relevant? How does it connect with other things you have learned here or in other classes?). The more insight you give, the better.

PROBLEM 5.2:

Write two MATLAB functions, called `mydct.m` and `myidct.m`, that implement the discrete cosine transform (DCT) and its inverse for a vector of length N. You code should be short (4 lines or less per function, no loops), efficient (it should make use of the `fft` command), and match the output of MATLAB's `dct` and `idct` commands. You can verify the latter with

```
x = randn(1000000,1);
d1 = mydct(x);
d2 = dct(x);
norm(d1-d2)

y = randn(1000000,1);
w1 = myidct(y);
w2 = idct(y);
norm(w1-w2)
```

The norms of the differences should be small in both cases. (The one thing you really have to handle here is the fact that the DCT uses cosines that are shifted by half a sample.)

PROBLEM 5.3:

Let $\psi(t)$ be the "hat" function centered at $t = 0$:

$$\psi(t) = \begin{cases} t+1, & -1 \le t \le 0 \\ 1-t, & 0 \le t \le 1, \\ 0, & \text{otherwise.} \end{cases}$$

Sketch $\psi(t)$. Then sketch $\psi(at - b)$ for some $a, b > 0$. Label your plot carefully; critical points on the t axis should be labeled in terms of a, b.

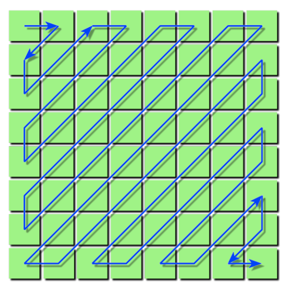
Last updated 8:49, September 27, 2018

PROBLEM 5.4:
In this problem, we will explore the main concepts behind the JPEG compression standard. Start by looking over the Wikipedia page on JPEG:
http://en.wikipedia.org/wiki/JPEG
Turn in your code for all parts, plots and reconstructed images for a few values of M for part c, and your calculations and reconstructed image for part d.

1. Write a MATLAB function `block_dct2.m` which takes an $N \times N$ pixel image, divides it into 8×8 pixel blocks (you may assume that N is divisible by 8), and returns the discrete cosine transform coefficients for each block. You will find the MATLAB command `dct2.m` helpful.

   Write another MATLAB function `iblock_dct2.m` which is the inverse of the above: that takes the blocked DCT coefficients and returns the image.

2. The provided function `jpgzzind.m` orders the indexes of a block from low frequencies to high frequencies, as shown below:

   If `xb` is an 8×8 block, then `xb(jpgzzind(8,8))` is a 64×1 vector containing the same elements, just in the "correct" order.

   Using this utility, write a MATLAB function `block_dct2_approx.m` that takes an $N \times N$ image and a number M, and returns an $N \times N$ approximation $\tilde{\boldsymbol{x}}_M$ formed by keeping the first M DCT coefficients in each block.

3. Download the file `bb.tiff`. Read it into MATLAB using `x=double(imread('bb.tiff'))`. For this image, plot[1]

   $$\log_{10}\left(\frac{\|\boldsymbol{x} - \tilde{\boldsymbol{x}}_M\|_2^2}{\|\boldsymbol{x}\|_2^2}\right)$$

   versus M (choose a range and number of values of M that make this plot meaningful). Using the `imagesc.m` command, show the approximation for $M = 1, 3, 8$.

4. Using the quantization table in `jpeg_Qtable.mat`, quantize your transform coefficients using

   $$\tilde{\alpha}_{k,\ell} = Q_{k,\ell} \cdot \mathrm{round}\left(\frac{\alpha_{k,\ell}}{Q_{k,\ell}}\right).$$

   Calculate how many of the resulting coefficients are non-zero, and compute

   $$\log_{10}\left(\frac{\|\boldsymbol{x} - \tilde{\boldsymbol{x}}_q\|_2^2}{\|\boldsymbol{x}\|_2^2}\right)$$

[1] If x is an image, then `norm(x,'fro')` returns the standard sqrt-sum-of-squares of the entries norms. This is the Frobenius norm. If you don't specify the `'fro'`, MATLAB will return the operator norm (largest singular value) of x.

where $\tilde{\boldsymbol{x}}_q$ is the image reconstructed from the quantized coefficients. Verify Parseval by checking that

$$\|\tilde{\boldsymbol{\alpha}} - \boldsymbol{\alpha}\|_2 = \|\tilde{\boldsymbol{x}} - \boldsymbol{x}\|_2.$$

Using the `imagesc.m` command, show the reconstructed image.

PROBLEM 5.5:

Implement the Haar wavelet transform and its inverse in MATLAB. Do this by writing two MATLAB functions, `haar.m` and `ihaar.m` that are called as `w=haar(x,L)` and `x = ihaar(w,L)`. Here, `x` is the original signal, and `L` is the number of levels in the transform.

You may assume that the length of `x` is dyadic; that is, the length of the input is 2^J for some positive integer J. In this situation, the Haar transform (no matter what L is) will have exactly 2^J terms, so the length of `x` and `w` should be the same.

If we interpret the input `x` as being scaling coefficients at scale J, then the vector `w` should consist of the scaling coefficients at scale $J-L$ stacked on top of the wavelet coefficients for scale $J-L$ stacked on top of the wavelet coefficients for scale $J-L+1$, etc.

Try your transform out on the data in `blocks.mat` and `bumps.mat`. For these two inputs, take a Haar wavelet transform with $L = 3$ levels, and plot the scaling coefficients at scale $J-3$, and the wavelet coefficients at scales $J-3$ down to $J-1$. (For both of these signals, $J = 10$.)

Also, verify that your transform is energy preserving.

Turn in printouts of your code along with the plots mentioned above.

PROBLEM 5.6:

(Optional) Consider the function[2]

$$\psi_0(t) = 2\cos\left(\frac{3\pi t}{2}\right)\frac{\sin(\pi t/2)}{\pi t},$$

and let

$$\psi_{0,n}(t) = \psi_0(t-n).$$

1. Show that
$$\langle \boldsymbol{\psi}_{0,n}, \boldsymbol{\psi}_{0,k}\rangle \;=\; \int_{-\infty}^{\infty} \psi_{0,n}(t)\psi_{0,k}(t)\ \mathrm{d}t \;=\; \begin{cases} 1 & n = k \\ 0 & n \neq k \end{cases}.$$

2. What is $\operatorname{span}\left(\{\psi_{0,n}(t),\ n \in \mathbb{Z}\}\right)$? That is, for what space is $\{\boldsymbol{\psi}_{0,n}\}_{n\in\mathbb{Z}}$ an orthobasis?

3. Let
$$\psi_j(t) = 2^{j/2}\psi_0(2^j t) \quad \text{and} \quad \psi_{j,n}(t) = \psi_j(t - 2^{-j}n) = 2^{j/2}\psi_0(2^j t - n)$$
Notice that the $\{\psi_{j,n}(t)\}$ are dyadic shifts of $\psi_j(t)$ with spacing 2^{-j}. What is $\operatorname{span}\left(\{\psi_{j,n}(t),\ n \in \mathbb{Z}\}\right)$?

4. Let $\phi(t)$ be the standard sinc function
$$\phi(t) = \frac{\sin(\pi t)}{\pi t}, \quad \phi_n(t) = \phi(t-n).$$
What is the span of the union of the $\{\phi_n(t)\}$ and all of the $\{\psi_{j,k}(t)\}_k$ for $j = 0, 1, \ldots, J$?

5. For a signal $x(t)$, how can we obtain $\langle \boldsymbol{x}, \boldsymbol{\psi}_{j,n}\rangle$ for all n using a filter-then-sample architecture?

[2] General hint for this question: Do everything in the Fourier domain. Start by sketching the continuous-time Fourier transform of $\psi_0(t)$ and $\psi_j(t)$ (when you get to part c).