

DAY 3

Monday, July 24th, 2006

I. Lecture: (total 47 slides)

I.1. Multiresolution Analysis (MRA): (22/22 slides)

1. Requirements for MRA
2. Nested Spaces and Complementary Spaces
3. Scaling Functions and Wavelets

I.2. Refinement Equation: (18/25 slides)

1. Iterative and Recursive Solution Techniques
2. Infinite Product Formula
3. Filter Bank Approach for Computing Scaling Functions and Wavelets

Link:

- <http://ocw.mit.edu/NR/ronlyres/Mathematics/18-327Wavelets--Filter-Banks-and-ApplicationsSpring2003/B4C7391D-94B8-47E9-A677-9282AC95E9CF/0/Slides9.pdf>
- <http://ocw.mit.edu/NR/ronlyres/Mathematics/18-327Wavelets--Filter-Banks-and-ApplicationsSpring2003/D7C1D6B6-159D-43E6-82FC-497EA758554A/0/Slides10.pdf>

Exercise and Lab:

I.2. Exercise:

1. Problem Set 6.1:

Problem 1: Explain why the scaling requirement, that $f(t)$ is in V_j if and only if $f(2t)$ is in V_{j+1} , can be restated as $\hat{f}(\omega)$ is in \hat{V}_j if and only if $\hat{f}(2\omega)$ is in \hat{V}_{j-1} . Here \hat{V}_j is the space of Fourier transforms of functions in V_j .

Problem 2: Find 2 by 2 matrices $c(0)$ and $c(1)$ so that the box function $\phi_1(t)$ and sloping line $\phi_2(t) = 1 - 2t$ satisfy

$$\begin{bmatrix} \phi_1(t) \\ \phi_2(t) \end{bmatrix} = c(0) \begin{bmatrix} \phi_1(2t) \\ \phi_2(2t) \end{bmatrix} + c(1) \begin{bmatrix} \phi_1(2t-1) \\ \phi_2(2t-1) \end{bmatrix}.$$

2. Problem Set 6.2:

Problem 5: Suppose the filter coefficients $h(k)$ are $\frac{1}{2}, 0, 0, \frac{1}{2}$. Starting from the box function, take one step of the cascade algorithm and draw $\phi^{(1)}(t)$. Then take the second step and draw $\phi^{(2)}(t)$. Describe $\phi^{(i)}(t)$ - on what fraction of the interval $[0, 3]$ does $\phi^{(i)}(t) = 1$?

Problem 6: Suppose the only filter coefficient is $h(0) = 1$. Starting from the box function $\phi^{(0)}(t)$, draw the graphs of $\phi^{(1)}(t)$ and $\phi^{(2)}(t)$. In what sense does $\phi^{(i)}(t)$ converge to the delta function $\delta(t)$? To verify the dilation equation $\delta(t) = 2\delta(2t)$, multiply by the any smooth $f(t)$ and compare the integrals of both sides.

3. Problem Set 6.3:

Problem 3: Show that the convolution $\phi_1(t) * \phi_2(t)$ does satisfy a dilation equation with the coefficients from $h_1 * h_2$.

4. Problem Set 6.4:

Problem 4: If $H(\omega)$ has p zeros at $\omega = \pi$, show that $\hat{\phi}(\omega)$ has p zeros at $\omega = 2n\pi$ for each $n \neq 0$.

I.3. Matlab:

1. 1-D signal analysis using Wavelets:

- Use biorwavf.m to generate symmetric biorthogonal wavelet filters with Nr = 4 is number of zeros at pi in the synthesis lowpass filter and Nd =4 is number of zeros at pi in the analysis lowpass filter.
- Find and plot the zeros of the synthesis and analysis lowpass filters.
- Use biorfilt.m to find and plot the impulse response of the complete set of filters.
- Examine the frequency response of the filters.
- Load a test signal using: load noisdopp; x = noisdopp;
- Compute the lowpass and highpass coefficients using convolution and downsampling.
- Obtain the same result by directly using the function dwt.m.
- Reconstruct the signal using upsampling and convolution. Only keep the middle L coefficients of the reconstructed signal i.e. the ones that correspond to the original signal.
- Reconstruct the signal by using the function idwt.m.
- Plot the input and output signal and the lowpass and highpass coefficients.

- Draw the structure of the iterated analysis filter bank of a three level decomposition by using `wtree.m`.
- Use `wavedec.m` to find the the vector `wc` of wavelet transform coefficients of the input signal.
- Use the functions `appcoe.m` and `detcoef.m` to extract the lowpass coefficients `a3` and the various highpass coefficients `d3`, `d2`, `d1` from `wc`. Then plot them.
- Reconstruct the signal from the lowpass coefficients `a3` and the various highpass coefficients `d3`, `d2`, `d1` by using `upcoef.m`.
- Reconstruct the signal from the from the full vector `wc` of transform coefficients by using `wrcoef.m`.
- Compare the error between the reconstructed signal and the original signal by using `upcoef.m` and `wrcoef.m`.

2. Practical problem:

Description: Voice scrambling techniques were widely used in World War II to secure telephone conversation. These scramblers operated by a number of principles. One was the manipulation of the “subbands” of speech on a telephone. A voice audio signal is a very complicated and unpredictable waveform, but it can be broken down into a set or “spectrum” of different simple “component” waveforms with a range of frequencies, with the spectrum changing over time.

The highest frequencies in normal human voice conversations range to over 7,000 hertz (Hz), but most of the higher frequencies don’t make much difference, and so telephones only transmit in the “frequency band” between 300 Hz and 3,300 Hz. As anyone who uses a phone knows, this is perfectly adequate for speech communications, though it’s certainly not high-fidelity sound. It is possible to shift out different “subbands” from the voice signal using electronic filters. For example, we could build a set of five filters to tear apart the spectrum of the voice signal as follows:

- Filter 1: 300 to 900 Hz
- Filter 2: 900 to 1,500 Hz
- Filter 3: 1,500 to 2,100 Hz
- Filter 4: 2,100 to 2,700 Hz
- Filter 5: 2,700 to 3,300 Hz

These five subband signals could then be “modulated”, or shifted in frequency, in various ways and mixed back together again to generate sounds that are hopefully incomprehensible to any eavesdropper. A parallel set of subband filters in the decoder would then sort out the scrambled speech and reassemble it correctly.

One simple subband scrambling scheme simply “inverted” the subbands, converting the output of filter 1 to the 2,700 to 3,300 Hz range, converting the output of filter 5 to the 300 to 900 Hz range, and so on. Another simple approach

was to shift the output of filters 1 through 4 up a subband, and convert the output of filter 5 to the 300 to 900 Hz range.

Your task:

- Design a 2-channel filter bank to split the sound signal $x(n)$ into 2 subband signals $v_0(n)$ and $v_1(n)$, then suppose a sophisticated way to merge these subband signals (both intra and inter signal) to form an encrypted signal $y(n)$.
- Listen to your encrypted signal
- Design a receiver to reconstruct this encrypted signal to the original sound.

