Introduction to Wavelets II

Lecture #2: Thursday, 2 October 1997
Lecturer: Denis Zorin
Scribe: Scott Cohen
Reviewer: John Owens

1 Filter Banks as Transformations

A filter bank H transforms an input x into an output $y = H(x)$. Figure 1 shows the familiar example of a analysis filter bank that separates the low and high frequencies of a discrete input signal. If the filter bank is linear, then the corresponding transformation

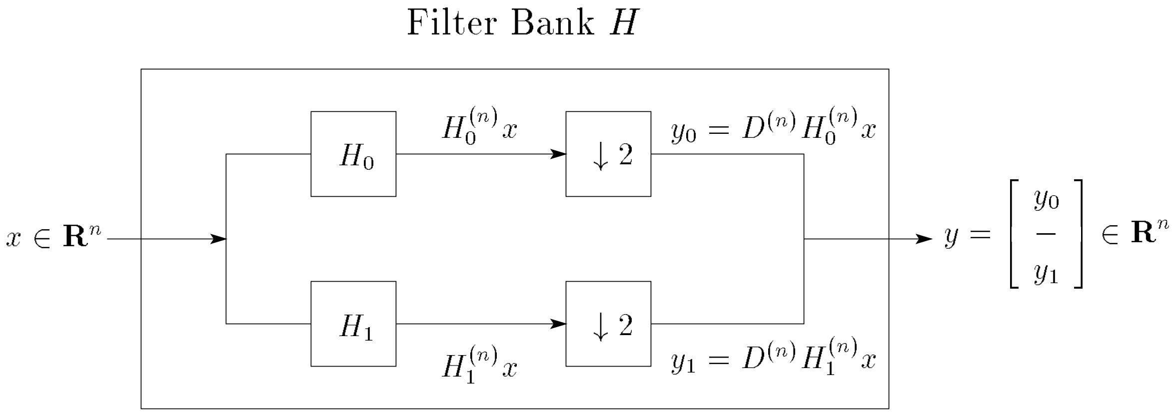


Figure 1: A analysis filter bank.

can be represented as a matrix H, and applying H to x is achieved by computing the matrix product $y = Hx$.

As an example, we shall now compute the matrix H for the Haar analysis filter bank, assuming an input x consisting of n samples ($x \in \mathbf{R}^n$). Recall that the Haar low pass filter H_0 simply averages adjacent entries of its input. This operation can be represented by the matrix[1]

$$H_0^{(n)} = \frac{1}{2}\begin{bmatrix} 1 & 1 & & & & & \\ & 1 & 1 & & & & \\ & & 1 & 1 & & & \\ & & & & \ddots & & \\ & & & & & 1 & 1 \\ & & & & & & 1 \end{bmatrix} \in \mathbf{R}^{n\times n}.$$

The Haar high pass filter H_1 computes half the difference between successive input samples, and can be represented[2] as

$$H_1^{(n)} = \frac{1}{2}\begin{bmatrix} 1 & -1 & & & & & \\ & 1 & -1 & & & & \\ & & 1 & -1 & & & \\ & & & & \ddots & & \\ & & & & & 1 & -1 \\ & & & & & & 1 \end{bmatrix} \in \mathbf{R}^{n\times n}.$$

The downsampling operation is represented as the matrix[3]

$$D^{(n)} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & \cdots & 1 & 0 \end{bmatrix} \in \mathbf{R}^{\frac{n}{2}\times n}$$

which picks out the first, third, etc. entries of x. The combined action of low pass filtering and then downsampling is represented by the matrix

$$L^{(n)} = D^{(n)}H_0^{(n)} = \frac{1}{2}\begin{bmatrix} 1 & 1 & & & & & & & \\ & & 1 & 1 & & & & & \\ & & & & \cdots & & & & \\ & & & & & 1 & 1 & & \\ & & & & & & & 1 & 1 \end{bmatrix} \in \mathbf{R}^{\frac{n}{2}\times n}.$$

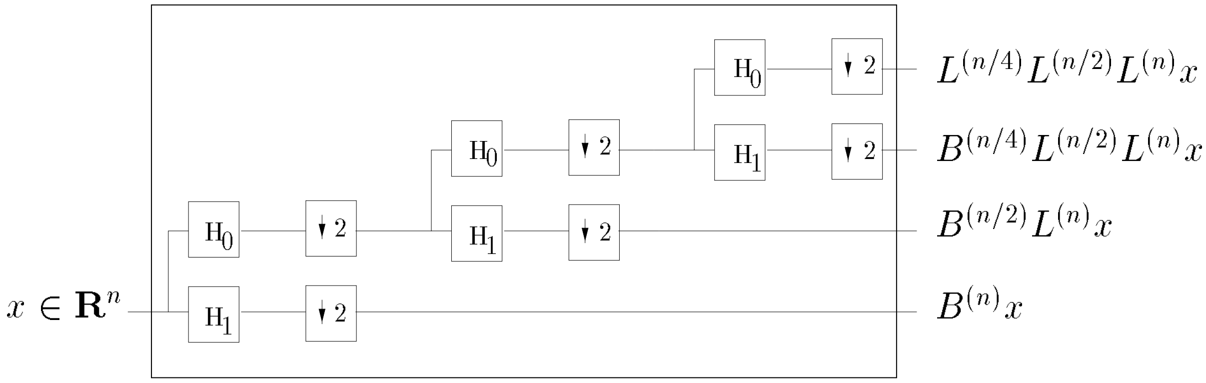
Similarly, the combined action of high pass filtering and then downsampling is represented

[1]Note that $H_0^{(n)}$ averages the final entry of x with zero since there is no next element. This is a moot point since the downsampling step that follows throws away the last entry of $H_0^{(n)}x$.

[2]The filter matrix $H_1^{(n)}$ differences the final entry of x with zero since there is no next element. This boundary anomaly is, once again, a moot point since the downsampling step that follows throws away the last entry of $H_1^{(n)}x$.

[3]Here we have tacitly assumed that n is even. Since the downsampling operation picks out the first, third, etc. components of its input, the last component will be discarded.

Filter Bank H_{sys}



$$\text{output } y = \begin{bmatrix} L^{(n/4)}L^{(n/2)}L^{(n)} \\ B^{(n/4)}L^{(n/2)}L^{(n)} \\ B^{(n/2)}L^{(n)} \\ B^{(n)} \end{bmatrix} x = H_{\text{sys}}x \in \mathbf{R}^n$$

Figure 2: Hierarchical decomposition of a signal.

as

$$B^{(n)} = D^{(n)}H_1^{(n)} = \frac{1}{2}\begin{bmatrix} 1 & -1 & & & & & & & \\ & & 1 & -1 & & & & & \\ & & & & \cdots & & & & \\ & & & & & 1 & -1 & & \\ & & & & & & & 1 & -1 \end{bmatrix} \in \mathbf{R}^{\frac{n}{2}\times n}.$$

The top and bottom branches of the filter bank produce (see Figure 1)

$$y_0 = L^{(n)}x \in \mathbf{R}^{\frac{n}{2}} \qquad \text{and} \qquad y_1 = B^{(n)}x \in \mathbf{R}^{\frac{n}{2}},$$

respectively. The output of the filter bank is

$$y = \begin{bmatrix} y_0 \\ - \\ y_1 \end{bmatrix} = \begin{bmatrix} L^{(n)}x \\ - \\ B^{(n)}x \end{bmatrix} = H^{(n)}x \in \mathbf{R}^n, \qquad \text{where } H^{(n)} = \begin{bmatrix} L^{(n)} \\ - \\ B^{(n)} \end{bmatrix} \in \mathbf{R}^{n\times n}.$$

Thus, we have expressed the operation of the Haar filter bank on an input of length n as the matrix $H^{(n)}$ shown above.

The output of the top branch of the filter bank is a coarse version of the input signal. We can build a hierarchical representation of a signal by recursively filtering the low pass output of the filter bank. This process is illustrated in Figure 2. In each step of the recursion, the signal rate decreases by a factor of two. If the signal is discrete and finite (and a power of two in length), then we eventually reach a signal with one sample. In the case of the Haar analysis filter bank, this sample will be the average of the elements of the original signal. The sum of the sizes of all the outputs from the high pass steps and the output of the final low pass step is equal to the size of the original input. The final, hierarchical representation of an input signal is a collection of signal details at various resolution levels (scales) and a coarse version of the original signal (which is the output of the final low pass filter). The entire process can be represented as a single transformation matrix H_{sys} which one can think of as rewriting the input x in terms of another basis to produce the output $y = H_{\text{sys}}x$. Figure 2 shows the H_{sys} matrix that results when the Haar filter bank is applied three times.

When the input to the system is an image (a 2D signal), the first filter bank application produces a blurred version of the image (the low pass output) and the details of the original image (sharp edges) which are not contained in the blurred version (the output of the high pass filter). The second low pass filter application takes the blurred version from step one and blurs it even further. The differences between the second blurred image and the first blurred image are captured in the output of the second high pass filter. This recursive process transforms an image into a collection of images that capture image details at different scales and one final coarse image.

2 The Haar Wavelet Basis

If a hierarchical decomposition via filter banks writes a discrete signal in terms of new basis. Can we find a similar decomposition for continuous signals? Answering this question is where the dilation equation

$$\phi(t) = 2\sum_k h_0(k)\phi(2t-k) \tag{1}$$

and the wavelet equation

$$w(t) = 2\sum_k h_1(k)\phi(2t-k) \tag{2}$$

come into the picture. The function $\phi(t)$ is called the *scaling function*, and the function $w(t)$ is called the *wavelet function*. The dilation equation and wavelet equation must hold for all t. Replacing t by $2^{j-1}t$ gives

$$\begin{aligned} \phi(2^{j-1}t) &= 2\sum_k h_0(k)\phi(2^j t-k) && (3)\\ w(2^{j-1}t) &= 2\sum_k h_1(k)\phi(2^j t-k) && (4) \end{aligned}$$

This last set of equations is more convenient for describing the construction of the wavelet basis corresponding to a filter bank.

We now illustrate the wavelet basis construction from the dilation equations using the Haar filter bank. Recall that the low pass Haar filter H_0 is defined by $h_0(0) = h_0(1) = 1/2$

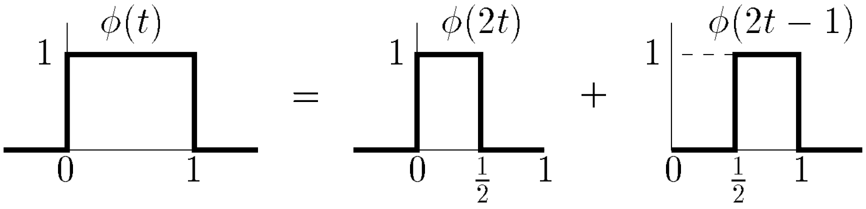


Figure 3: The Haar scaling functions $\phi(t)$, $\phi(2t)$, and $\phi(2t-1)$.

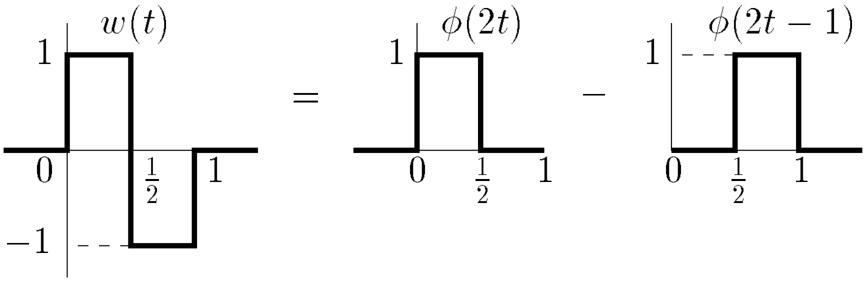


Figure 4: The Haar wavelet function $w(t)$.

(all other coefficients are zero). Substituting the Haar low pass filter into equation (1), we get

$$\phi(t) = \phi(2t) + \phi(2t-1).$$

The solution to this recurrence is the Haar scaling function

$$\phi(t) = \begin{cases} 1 & \text{if } t \in [0,1) \\ 0 & \text{otherwise} \end{cases} . \qquad (5)$$

The functions $\phi(t)$, $\phi(2t)$, and $\phi(2t-1)$ are shown in Figure 3. The high pass Haar filter H_1 is defined by $h_1(0) = 1/2$ and $h_1(1) = -1/2$. Substituting into (2) yields

$$w(t) = \phi(2t) - \phi(2t-1).$$

It follows easily from (5) that the Haar wavelet function is

$$w(t) = \begin{cases} 1 & \text{if } t \in [0,1/2) \\ -1 & \text{if } t \in [1/2,1) \\ 0 & \text{otherwise} \end{cases} .$$

The wavelet function $w(t)$ is shown in Figure 4. The scaling function $\phi(t)$ is the continuous analog of the discrete low pass filter H_0. Applying $\phi(t)$ to $f(t)$ yields

$$<\phi, f> = \int_{-\infty}^{\infty} \phi(t) f(t)\, dt = \int_0^1 f(t)\, dt,$$

the average value of f over the interval $[0,1)$. The wavelet function $w(t)$ is the continuous analog of the discrete high pass filter H_1. Applying $w(t)$ to $f(t)$ yields

$$< w, f > = \int_{-\infty}^{\infty} w(t) f(t)\, dt = \int_0^{\frac{1}{2}} f(t)\, dt - \int_{\frac{1}{2}}^{1} f(t)\, dt.$$

The filter ϕ is an averaging operator, and the filter w is a differencing operator.

Now consider functions defined on the interval $[0,1)$. Let V^j denote the set of functions that are constant on the 2^j subintervals $[l/2^j, (l+1)/2^j)$, $l = 0, 1, \ldots, 2^j - 1$. Any function in V^j can be represented exactly by a linear combination of the 2^j functions

$$\phi_{jk}(t) = \phi(2^j t - k), \quad k = 0, \ldots, 2^j - 1.$$

This should be clear (at least for the case $j = 3$) from the upper left hand corner in Figure 5 which shows the $2^j = 8$ functions $\phi_{3k} = \phi(2^3 t - k)$, $k = 0, \ldots, 7$. Similarly, the wavelet functions are denoted by

$$w_{jk}(t) = w(2^j t - k), \quad k = 0, \ldots, 2^j - 1.$$

We collect the scaling and wavelet functions at fixed resolution level j in the sets

$$\Phi_j = \{\ \phi_{jk}(t)\ :\ k = 0, \ldots, 2^j - 1\ \} \quad \text{and} \quad \Omega_j = \{\ w_{jk}(t)\ :\ k = 0, \ldots, 2^j - 1\ \},$$

respectively.

As previously mentioned, Φ_j is a basis for V^j. Applying equations (3) and (4) with $j = 3$ to the functions $\phi_{3k}(t) = \phi(2^3 t - k)$, $k = 0, \ldots, 7$ yields the functions $\phi_{2k}(t) = \phi(2^2 t - k)$, $k = 0, \ldots, 3$ and $w_{2k}(t) = w(2^2 t - k)$, $k = 0, \ldots, 3$. In detail,

$$\begin{bmatrix} \phi_{20} \\ \phi_{21} \\ \phi_{22} \\ \phi_{23} \\ w_{20} \\ w_{21} \\ w_{22} \\ w_{23} \end{bmatrix} = 2H^{(8)} \begin{bmatrix} \phi_{30} \\ \phi_{31} \\ \phi_{32} \\ \phi_{33} \\ \phi_{34} \\ \phi_{35} \\ \phi_{36} \\ \phi_{37} \end{bmatrix}.$$

The results of this step are shown in the middle and last columns of the first row. The final column in the first row shows the separation of the low pass output functions Φ_2 from the high pass output functions Ω_2. Note that $\Phi_2 \cup \Omega_2$ is also a basis for V^3. The low pass output functions are further refined by once again applying equations (3) and (4), this time with $j = 2$. In matrix notation,

$$\begin{bmatrix} \phi_{10} \\ \phi_{11} \\ w_{10} \\ w_{11} \end{bmatrix} = 2H^{(4)} \begin{bmatrix} \phi_{20} \\ \phi_{21} \\ \phi_{22} \\ \phi_{23} \end{bmatrix}.$$

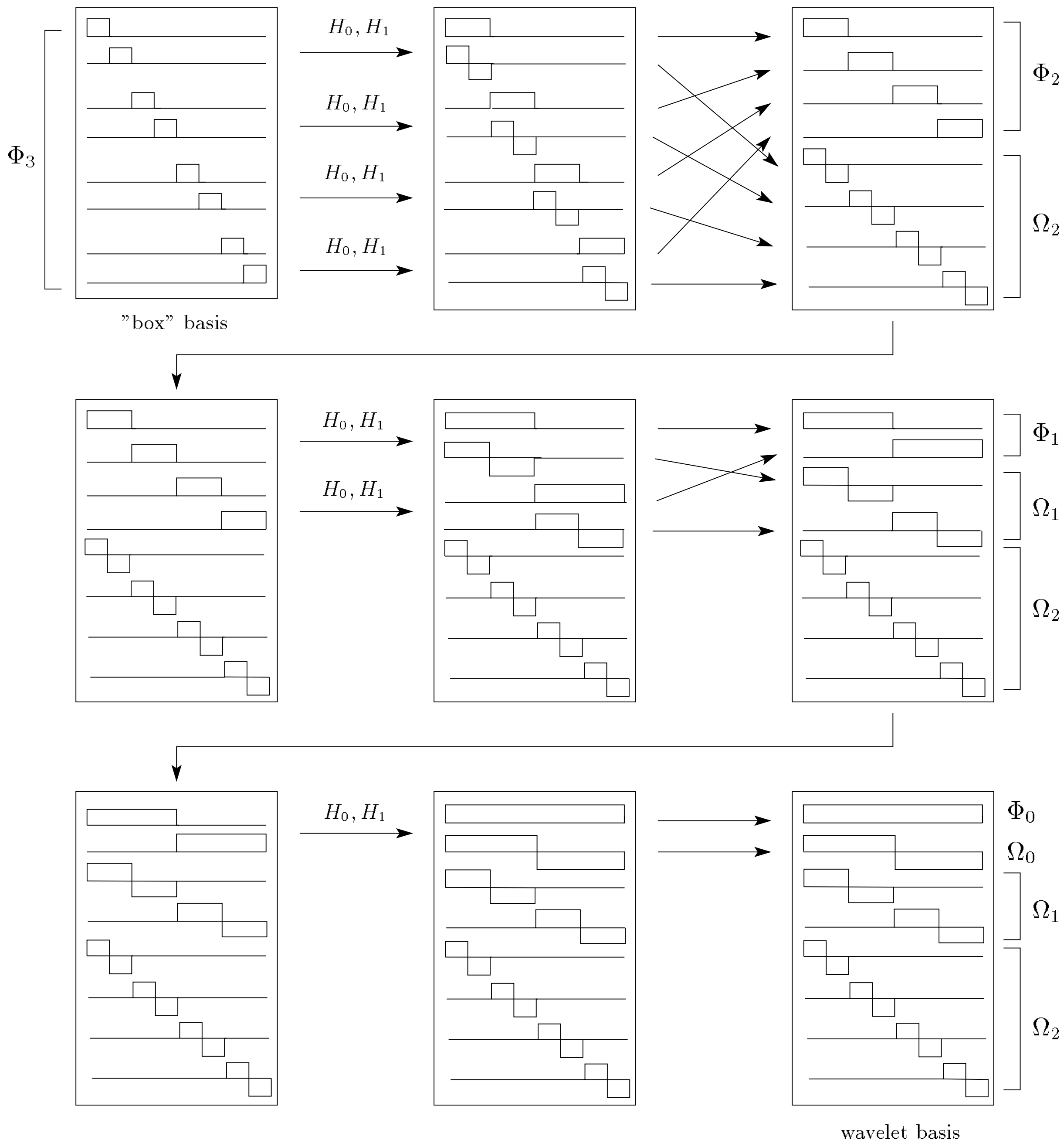


Figure 5: The Haar wavelet basis.

This defines the four new functions $\phi_{1k}(t) = \phi(2t - k)$, $k = 0, 1$, and $w_{1k}(t) = w(2t - k)$, $k = 0, 1$ shown in the second row of Figure 5. The last column of this row shows the separation of the low pass outputs Φ_1 and high pass outputs Ω_1. The set $\Phi_1 \cup \Omega_1 \cup \Omega_2$ is, again, a basis for V^3. Finally, the third row shows the result of applying the dilation and wavelet equations with $j = 1$ to $\phi_{1k}(t) = \phi(2t - k)$, $k = 0, 1$ to produce the scaling function $\phi_{0k}(t) = \phi(t - k)$, $k = 0$ and the wavelet function $w_{0k}(t) = w(t - k)$, $k = 0$. More precisely,

$$\begin{bmatrix} \phi_{00} \\ w_{00} \end{bmatrix} = 2H^{(2)} \begin{bmatrix} \phi_{10} \\ \phi_{11} \end{bmatrix}.$$

The eight functions ϕ_{00}, w_{00}, w_{10}, w_{11}, w_{20}, w_{21}, w_{22}, w_{23} in $\Phi_0 \cup \Omega_0 \cup \Omega_1 \cup \Omega_2$ form the the Haar wavelet basis for V^3. In general, the Haar wavelet basis for V^j contains the 2^j functions in $\Phi_0 \cup \Omega_0 \cup \Omega_1 \cup \cdots \cup \Omega_{j-1}$.

3 Some Examples

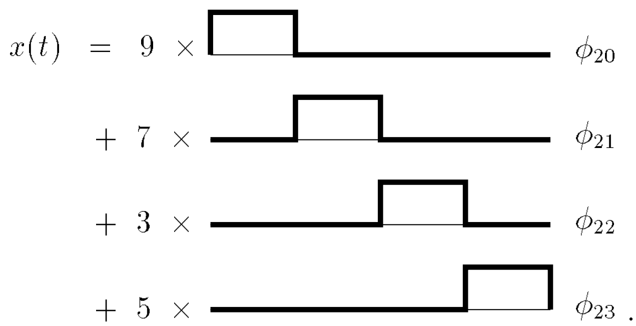
Suppose we have a function[4] $x(t)$ defined on $[0, 1)$ by

$$x(t) = \begin{cases} 9 & \text{if } t \in [0, 1/4) \\ 7 & \text{if } t \in [1/4, 1/2) \\ 3 & \text{if } t \in [1/2, 3/4) \\ 5 & \text{if } t \in [3/4, 1) \end{cases}.$$

The function $x(t)$ is in the space V^2. In terms of the basis Φ_2, $x(t)$ has representation

$$x = \begin{bmatrix} 9 \\ 7 \\ 3 \\ 5 \end{bmatrix}.$$

A graphical representation of $x(t)$ in terms of the basis Φ_2 is



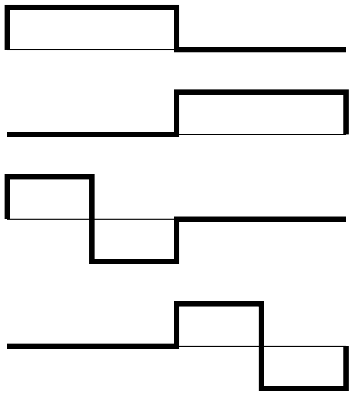
Applying the Haar sythesis filter bank $H^{(4)}$ to x gives

$$z = H^{(4)}x = \begin{bmatrix} 8 \\ 4 \\ 1 \\ -1 \end{bmatrix} = \begin{bmatrix} (9+7)/2 \\ (3+5)/2 \\ (9-7)/2 \\ (3-5)/2 \end{bmatrix}.$$

[4] This example is taken directly from the paper "Wavelets for Computer Graphics: A Primer (Part 1)" by Eric J. Stollnitz, Tony D. DeRose, and David Salesin in *IEEE Computer Graphics and Applications*, 15(3):76-84, May 1995. It is also available online at `http://www.cs.washington.edu/research/graphics/projects/wavelets/article/`.

The elements of z are the coefficients of the representation of $x(t)$ in terms of the basis $\Phi_1 \cup \Omega_1$.

$$x(t) = 8 \times \phi_{10} + 4 \times \phi_{11} + 1 \times w_{10} + -1 \times w_{11}.$$

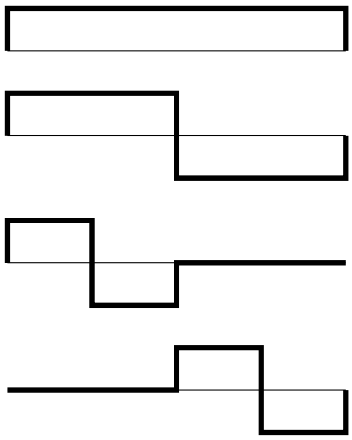


Applying the Haar analysis filter bank $H^{(2)}$ to the first two elements in z and leaving the high pass outputs in place gives

$$y = \begin{bmatrix} H^{(2)} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} \\ z_3 \\ z_4 \end{bmatrix} = \begin{bmatrix} 6 \\ 2 \\ 1 \\ -1 \end{bmatrix} = \begin{bmatrix} (8+4)/2 \\ (8-4)/2 \\ 1 \\ -1 \end{bmatrix}.$$

The elements in y are the coefficients of the representation of $x(t)$ in terms of the Haar wavelet basis $\Phi_0 \cup \Omega_0 \cup \Omega_1$.

$$x(t) = 6 \times \phi_{00} + 2 \times w_{00} + 1 \times w_{10} + -1 \times w_{11}.$$



The Haar wavelet transform of the signal $x = [\ 9\ \ 7\ \ 2\ \ 5\ ]$ is $y = [\ 6\ \ 2\ \ 1\ \ -1\ ]$.

We can also compute the wavelet transform of x by multiplying the system filter bank matrix H_{sys} by x. Figure 2 shows H_{sys} when the Haar analysis filter bank is applied three times. In the example in this section, we only need two applications and the input vector has length $n = 4$. This results in the matrix

$$H_{\text{sys}} = \begin{bmatrix} L^{(2)}L^{(4)} \\ B^{(2)}L^{(4)} \\ B^{(4)} \end{bmatrix} = \begin{bmatrix} \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} \\ \frac{1}{4} & \frac{1}{4} & -\frac{1}{4} & -\frac{1}{4} \\ \frac{1}{2} & -\frac{1}{2} & 0 & 0 \\ 0 & 0 & \frac{1}{2} & -\frac{1}{2} \end{bmatrix}.$$

The system output is

$$y = H_{\text{sys}} x = \begin{bmatrix} 6 \\ 2 \\ 1 \\ -1 \end{bmatrix},$$

just as we derived previously using the hierarchical construction.

In general, computing the representation of an n-dimensional vector in a different basis via an $n \times n$ matrix–vector multiplication requires $O(n^2)$ arithmetic operations. For computing the Haar wavelet transform, however, we can do better. The hierarchical construction given at the beginning of this section requires only $O(n)$ arithmetic operations. In fact, if applying the analysis filter bank to an n-sample signal requires at most kn operations for some constant k (as it does for the Haar filter bank), then the total number of operations to compute the wavelet transform is at most

$$kn + k\frac{n}{2} + k\frac{n}{4} + \cdots + k\frac{n}{2^{\log_2 n}} \le kn(1 + \frac{1}{2} + \frac{1}{4} + \cdots) \le 2kn = O(n).$$

Essentially, the hierarchical construction implements a fast matrix-vector multiply by taking advantage of the structure of H_{sys}.

Another informative example to consider is the Haar transform of a constant signal, say $x = [\; c \;\; c \;\; c \;\; c \;\; c \;\; c \;\; c \;\; c \;]$. Computing the Haar transform of x is equivalent to decomposing the function $x(t) \equiv c$, $t \in [0, 1)$ in terms of the basis $\Phi_0 \cup \Omega_0 \cup \Omega_1 \cup \Omega_2$ for V^3. Applying the Haar analysis filter bank three times gives

$$x = \begin{bmatrix} c \\ c \\ c \\ c \\ c \\ c \\ c \\ c \end{bmatrix} \rightarrow H^{(8)} x = \begin{bmatrix} c \\ c \\ c \\ c \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \rightarrow \begin{bmatrix} H^{(4)} \begin{bmatrix} c \\ c \\ c \\ c \end{bmatrix} \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} c \\ c \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \rightarrow \begin{bmatrix} H^{(2)} \begin{bmatrix} c \\ c \end{bmatrix} \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} c \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} = y.$$

The output y contains only one nonzero coefficient, namely the coefficient for $\phi_{00}(t) = \phi(t)$. This transform result is obvious from the functional point of view since $x(t) = c\phi(t)$. Discarding the zero coefficients in the output leaves a very compact representation of the signal x (one which is eight times smaller than the original representation). In general, the Haar basis provides a compact representation for parts of a signal with little variation.

For the final example of this section, consider the Haar transform of the delta signal

$$x = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \longrightarrow H_{\text{sys}} x = \begin{bmatrix} \frac{1}{16} \\ -\frac{1}{16} \\ -\frac{1}{8} \\ 0 \\ 0 \\ -\frac{1}{4} \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ -\frac{1}{2} \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} = y.$$

Note that 11 of the 16 output coefficients are zero. This is not the case for the Fourier transform of x which contains energy at all frequencies. The coefficients of the Haar basis functions whose support interval (i.e. the interval over which the basis function is nonzero) does not overlap with the support of the represented function are all zero. The Haar basis allows one to make spacially localized (at some scale) changes to a function by simply adjusting the coefficient(s) for the basis function(s) of the appropriate scale (j) and location (k). This is very different from the Fourier representation in which a change to a single basis function coefficient causes spacially global changes to the represented function.

4 Orthogonal Filter Banks and Wavelet Bases

In Section 1, we showed how to view a filter bank as a transformation represented by a square matrix H (assuming the input and output are the same size). We say that a filter bank is orthogonal if its corresponding matrix is orthogonal. A matrix H is orthogonal if its inverse is equal to its transpose: $H^T H = HH^T = I$. The set of column vectors and the set of row vectors of an orthogonal matrix are both orthonormal sets of vectors (it would make more sense to call such matrices orthonormal, but the historical term for such matrices is *orthogonal*). With a suitable scaling of the input before low pass and high pass filtering, the Haar filter bank is indeed orthogonal. For example, for $n = 4$ we have

$$H^{(n)} = H^{(4)} = \begin{bmatrix} L^{(4)} \\ - \\ B^{(4)} \end{bmatrix} = \frac{1}{2}\begin{bmatrix} 1 & 1 & & \\ & & 1 & 1 \\ 1 & -1 & & \\ & & 1 & -1 \end{bmatrix},$$

and

$$(H^{(n)})^T H^{(n)} = H^{(n)}(H^{(n)})^T = \frac{1}{2}I.$$

Therefore, the matrix $\sqrt{2}H^{(n)}$ is orthogonal. A modified Haar filter bank which includes a scaling by $\sqrt{2}$ before the low pass and high pass operations is orthogonal.

A basis of functions $f_0, f_1, \ldots$ (for some class of functions on the real line) is an orthonormal basis iff

$$< f_i, f_j > = \int_{t=-\infty}^{\infty} f_i(t) f_j(t)\, dt = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{if } i \neq j \end{cases}.$$

It is very easy to write a function f in terms of an orthonormal basis. Taking the inner product with f_j with both sides of

$$f = \sum a_i f_i$$

gives

$$\begin{aligned} < f, f_j > &= < \sum_i a_i f_i, f_j > \\ &= \sum_i a_i < f_i, f_j > \\ < f, f_j > &= a_j. \end{aligned}$$

Therefore, the coefficient a_j in the decomposition of f is simply the inner product $< f, f_j >$ of f with the jth basis function f_j. In Section 2 we saw how a filter bank gives rise to a wavelet basis via the dilation and wavelet equations (at least in the case of the Haar filter bank). What conditions on the filters will guarantee an orthonormal wavelet basis?

It turns out that orthogonality of a filter bank implies orthonormality of the basis generated by the filter bank though the dilation and wavelet equations. The proof of this fact will be given in the next lecture. Since the normalized Haar filter bank is orthogonal, this fact implies that the corresponding normalized Haar wavelet basis is orthonormal. The wavelet basis given in Section 2 is not normalized, but its orthogonality can be verified directly from the equations for its scaling function $\phi(t)$ and wavelet function $w(t)$.