CREATING AND COMPARING WAVELETS

GILBERT STRANG
Department of Mathematics, Massachusetts Institute of Technology
Cambridge, Massachusetts 02139, USA
E-mail: gs@math.mit.edu

ABSTRACT

This paper emphasizes two points about the design and application of filters and filter banks and wavelets:

- The algebra behind wavelet design is now quite simple.
- The comparison of two competing wavelets is still experimental and empirical.

As example we consider two particular 9/7 constructions (nine coefficients in analysis and seven in synthesis). Both are symmetric, so neither is orthogonal. Both have important advantages. We are unable to say which construction is better. The paper begins with the conditions on the filter coefficients for perfect reconstruction. That part is a brief summary of the exposition in Strang and Nguyen [1].

1. Introduction

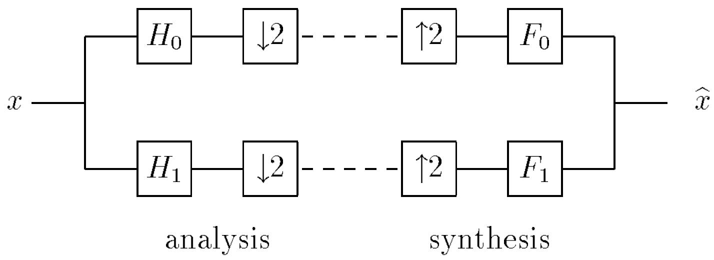
This paper discusses two-channel filter banks, leading to wavelets. The structure involves four filters (which are just convolutions, see below). The analysis bank has a lowpass filter H_0 and a highpass filter H_1. The outputs y_0 and y_1 from those filters are "downsampled" by keeping only the even-numbered components:

$$\begin{aligned} y_0 &= H_0 x \quad \text{and} \quad v_0(n) = (\downarrow 2) y_0(n) = y_0(2n) \\ y_1 &= H_1 x \quad \text{and} \quad v_1(n) = (\downarrow 2) y_1(n) = y_1(2n). \end{aligned}$$

The full-length input vector x yields two half-length vectors v_0 and v_1. *That analysis step is inverted by the synthesis step.* The v's are "upsampled" to put zeros into their odd-numbered components. The results are filtered by F_0 and F_1, and their sum is the output $\hat{x}$:

$$\hat{x} = F_0(\uparrow 2) v_0 + F_1(\uparrow 2) v_1.$$

The intention is that $\hat{x} = x$.



In that case the filter bank gives *perfect reconstruction.* The analysis wavelets and synthesis wavelets will be *biorthogonal.* So are the scaling functions (biorthogonality means $\int \tilde{\phi}(t-k)\phi(t-\ell)dt = \delta_{k\ell}$). All these functions come from solving two-scale equations, which involve t and $2t$. This second scale $2t$ corresponds to the $2n$ in downsampling.

We emphasize particularly how the properties of these functions in continuous time follow from the properties of the filters in discrete time. The heart of the construction is the choice of H_0 and F_0. These are given by the impulse responses h_0 and f_0, which are the vectors of filter coefficients: $h_0 = (h_0(0), \ldots, h_0(8))$ and $f_0 = (f_0(0), \ldots, f_0(6))$. Those choices determine H_1 and F_1 by a pattern of "alternating signs", $h_1(n) = (-1)^n f_0(n)$ and $f_1(n) = (-1)^{n+1} h_0(n)$.

We first determine the conditions on H_0 and F_0 (thus on h_0 and f_0) for perfect reconstruction. Then we explain the special importance of "zeros at $z = -1$" in the transfer functions, which are simply polynomials built from the filter coefficients:

$$H_0(z) = \sum_0^8 h_0(n) z^{-n} \qquad \text{and} \qquad F_0(z) = \sum_0^6 f_0(n) z^{-n}.$$

The algebra is all straightforward. We have a change of basis, produced by a *wavelet transform.* The components of v_0 and v_1 express the input vector x in the new basis. This transform can be applied again to the lowpass output v_0 that is normally most important. Scaling functions and wavelets appear in the limit of an infinite iteration. Four or five levels give a typical tree, in practice.

If successful, many components of the v's will be small. The signal is *compressed* by setting small coefficients to zero (not invertible!). The reconstructed output from the synthesis bank will no longer agree exactly with x. But if x and $\hat{x}$ are close, the input signal is now represented by a small number of components. In the new basis, the signal can be efficiently transmitted and stored. One important and unresolved difficulty is the meaning of the word "close".

2. The Conditions for Perfect Reconstruction

A filter is a convolution: $y(n) = \sum h(k)x(n-k)$. This linear transformation is represented by a Toeplitz matrix (meaning constant diagonals). The coefficient $h(k)$ appears along the kth subdiagonal. The input vector x is very long in practice and infinitely long in theory— thus the filter matrix is doubly infinite:

$$Hx = \begin{bmatrix} \cdot & \cdot & \cdot & \cdot & & & & \\ & h(3) & h(2) & h(1) & h(0) & & & \\ & & h(3) & h(2) & h(1) & h(0) & & \\ & & & h(3) & h(2) & h(1) & h(0) & \\ & & & & \cdot & \cdot & \cdot & \cdot \end{bmatrix} \begin{bmatrix} \cdot \\ x(-1) \\ x(0) \\ x(1) \\ \cdot \end{bmatrix} = \begin{bmatrix} \cdot \\ y(-1) \\ y(0) \\ y(1) \\ \cdot \end{bmatrix}.$$

Downsampling removes $y(-1)$ and $y(1)$. In the product $(\downarrow 2)H$, all the odd-numbered rows of H are removed:

$$(\downarrow 2)H = \begin{bmatrix} \cdot & \cdot & \cdot & \cdot & & & & & & \\ & & h(3) & h(2) & h(1) & h(0) & & & & \\ & & & & h(3) & h(2) & h(1) & h(0) & & \\ & & & & & & \cdot & \cdot & \cdot & \cdot \end{bmatrix}.$$

Notice the *double shift* between rows. $(\downarrow 2)H$ is the fundamental operator in wavelet analysis (1×2 block Toeplitz matrix). When the two analysis filters $(\downarrow 2)H_0$ and $(\downarrow 2)H_1$ are combined, by interleaving rows of the two matrices, we get the block Toeplitz matrix (with 2×2 blocks) that represents the analysis bank:

$$H_b = \begin{bmatrix} h_0(3) & h_0(2) & h_0(1) & h_0(0) & & \\ h_1(3) & h_1(2) & h_1(1) & h_1(0) & & \\ & & h_0(3) & h_0(2) & h_0(1) & h_0(0) \\ & & h_1(3) & h_1(2) & h_1(1) & h_1(0) \\ & & & & \cdot & \cdot \end{bmatrix}$$

The inverse of H_b is the synthesis matrix F_b. *The key feature of these matrices is that both are banded.* In the language of signal processing, all filters are FIR (finite impulse response). Banded Toeplitz matrices with banded inverses are possible only because these are *block* matrices. The inverse of a polynomial $1 + z^{-1}$ is not a polynomial. But a matrix polynomial can have a polynomial inverse: for example

$$H_p(z) = \frac{1}{2} \begin{bmatrix} 1 + z^{-1} & 1 - z^{-1} \\ 1 - z^{-1} & 1 + z^{-1} \end{bmatrix}$$

has determinant z^{-1} and inverse

$$\frac{1}{2} \begin{bmatrix} z + 1 & -z + 1 \\ -z + 1 & z + 1 \end{bmatrix}.$$

This example illustrates two further points. First, the determinant is a monomial. Since we divide by the determinant, this monomial is the key to a polynomial inverse. Second, the inverse is anticausal (powers of z) when the original filters are causal (powers of z^{-1}). To keep all filters causal and all matrices lower triangular, the convention is to separate out the monomial determinant:

$$\begin{aligned} F_p(z) &= z^{-1} H_p^{-1}(z) && \text{is causal,} \\ F_p(z) H_p(z) &= z^{-1} I && \text{is a one–step delay.} \end{aligned}$$

This means that the output $\hat{x}(n)$ from the filter bank agrees with $x(n-1)$, not with $x(n)$. We still call this perfect reconstruction! In general the filters produce ℓ delays and the output is $\hat{x}(n) = x(n - \ell)$. The product of the lower triangular block

Toeplitz matrices F_b and H_b is the shift matrix that has identity blocks on the ℓth subdiagonal. Our real task is to find the conditions on the coefficients $f_0(n)$ and $h_0(n)$ for this to happen.

The algebra of convolution is summed up in the convolution rule. This transforms the matrix equation $y = Hx$ into a simpler equation for the associated polynomials. These polynomials are just multiplied: $Y(z) = H(z)X(z)$ is

$$\left(\sum y(n)z^{-n}\right) = \left(\sum h(k)z^{-k}\right)\left(\sum x(\ell)z^{-\ell}\right).$$

The terms z^{-k} and $z^{-(n-k)}$ in the factors give z^{-n} in the product. Thus the coefficient in $y(n)z^{-n}$ is the convolution $\sum h(k)x(n-k)$ that comes from matrix multiplication.

The algebra of $(\downarrow 2)$ and $(\uparrow 2)$ is almost as neat. For vectors,

$$(\uparrow 2)(\downarrow 2)\begin{bmatrix} y(-2) \\ y(-1) \\ y(0) \\ y(1) \\ y(2) \end{bmatrix} = (\uparrow 2)\begin{bmatrix} y(-2) \\ y(0) \\ y(2) \end{bmatrix} = \begin{bmatrix} y(-2) \\ 0 \\ y(0) \\ 0 \\ y(2) \end{bmatrix}.$$

For the z-transform $Y(z) = \sum y(n)z^{-n}$, only even powers remain. The result of downsampling and upsampling is the even part

$$\frac{1}{2}\Big(Y(z) + Y(-z)\Big) = \frac{1}{2}\Big(H(z)X(z) + H(-z)X(-z)\Big).$$

That term with $-z$ reflects aliasing. Two inputs can give the same output. The constant vector $y(n) \equiv 1$ and the alternating vector $y(n) = (-1)^n$ have the same even components, and therefore they look the same after downsampling.

The conditions for perfect reconstruction $\hat{x}(n) = x(n-\ell)$ come by following the signal through the filter bank. We do it in the z-domain, starting with $X(z)$. The lowpass channel yields $Y_0(z) = H_0(z)X(z)$. Then it takes the even part. Then it multiplies by $F_0(z)$. The highpass channel has 1 in place of 0, and we add:

$$\hat{X}(z) = \begin{array}{c} \frac{1}{2}F_0(z)\Big(H_0(z)X(z) + H_0(-z)X(-z)\Big) \\ + \\ \frac{1}{2}F_1(z)\Big(H_1(z)X(z) + H_1(-z)X(-z)\Big) \end{array} = z^{-\ell}X(z).$$

The coefficient of $X(z)$ is $z^{-\ell}$ (no distortion, only a delay). The coefficient of $X(-z)$ must be zero (no aliasing in the final output). These are the PR conditions:

$$F_0(z)H_0(z) + F_1(z)H_1(z) = 2z^{-\ell} \tag{1}$$
$$F_0(z)H_0(-z) + F_1(z)H_1(-z) = 0. \tag{2}$$

It is the anti-aliasing equation (2) that leads to the "alternating sign" constructions, $h_1(n) = (-1)^n f_0(n)$ and $f_1(n) = (-1)^{n+1} h_0(n)$. In terms of polynomials, this is $H_1(z) = F_0(-z)$ and $F_1(z) = -H_0(-z)$. Then (2) is automatically satisfied, and (1) reduces to an equation for the *product filter* $P_0(z) = F_0(z)H_0(z)$:

$$P_0(z) - P_0(-z) = 2z^{-l}. \tag{3}$$

This is the key equation for perfect reconstruction.

Note that the left side of (3) is an odd function, so ℓ must be odd. The equation says that *the only odd term in $P_0(z)$ is $z^{-\ell}$ with coefficient 1.* We can separate the design of a PR filter bank into three steps:

1. Choose a polynomial $P_0(z)$ that satisfies (3).
2. Factor $P_0(z) = F_0(z)H_0(z)$.
3. Choose $H_1(z) = F_0(-z)$ and $F_1(z) = -H_0(-z)$.

This simplicity is what was meant in our first point in the abstract. It is deceptive, because it does not indicate what makes one design better than another. Part of the answer (only part!) is in the number of zeros at $z = -1$.

3. Zeros at -1: Approximation and Vanishing Moments

For a lowpass filter, the polynomial

$$H(z) = \sum h(n)z^{-n} = \sum h(n)e^{-in\omega}$$

is near zero at the highest frequency $\omega = \pi$. In the z-plane, with $z = e^{i\omega}$, this is the point $z = e^{i\pi} = -1$. Then the multiplication $Y(z) = H(z)X(z)$ ensures that high frequencies in x (at or near $\omega = \pi$) are nearly annihilated in y. The lowest frequency $\omega = 0$ (or $z = +1$) passes through the filter provided $H(1) = 1$. Hence the name "lowpass".

This condition $H(-1) = 0$ is fundamental in wavelet theory. It must hold exactly, not just approximately, to have any chance of continuous scaling functions and wavelets. The scaling function $\phi(t)$ solves the *dilation equation* or *refinement equation*:

$$\phi(t) = 2\sum h(k)\phi(2t - k). \tag{4}$$

The Fourier transform of this equation is

$$\hat{\phi}(\omega) = \sum h(n)e^{-i\omega n/2}\hat{\phi}(\omega/2) = H(e^{i\omega/2})\hat{\phi}(\omega/2). \tag{5}$$

Periodicity gives $H(e^{ik\pi}) = 0$ for all odd integers k. Then equation (5) yields $\hat{\phi}(2\pi n) = 0$ for every $n \neq 0$. This is the first of the so-called "Strang–Fix conditions", implying that the translates of $\phi(t)$ add to a constant, which we may normalize to 1:

$$\sum_{-\infty}^{\infty} \phi(t-n) \equiv 1. \tag{6}$$

This conclusion could also be reached directly from the dilation equations

$$\phi(t-n) = 2\sum h(k)\phi(2t-2n-k)$$

by summing on n. The sum $S(t)$ is 1-periodic, and we restrict to $0 \le t < 1$. Separating even from odd k leads to

$$S(t) = \left(2 \sum_{\text{even } k} h(k)\right) S(2t) + \left(2 \sum_{\text{odd } k} h(k)\right) S(2t-1). \tag{7}$$

The coefficients are $H(1) \pm H(-1)$. Thus both coefficients are 1. The sum satisfies the "Haar equation" $S(t) = S(2t) + S(2t-1)$, whose solution on the period interval $[0,1)$ is $S(t) \equiv 1$.

We often use the simplified notation $H(\omega)$ to replace $H(e^{i\omega})$. Note that the transform of $\phi(2t)$ is $\frac{1}{2}\hat{\phi}(\omega/2)$. Then (4) applies recursively to $\omega/2, \omega/4, \ldots$ and leads to an infinite product formula for $\hat{\phi}(\omega)$:

$$\hat{\phi}(\omega) = H(\frac{\omega}{2})H(\frac{\omega}{4})\hat{\phi}(\frac{\omega}{4}) = \cdots = \prod_{1}^{\infty} H(\frac{\omega}{2^j}). \tag{8}$$

This product converges for each ω. But is it the transform of a smooth $\phi(t)$?

Summary

The special zero $H(-1) = 0$ leads to $S(t) \equiv 1$. Constant polynomials can be produced from translates of $\phi(t)$. By standard results in approximation theory, these translates give at least first-order approximation to any smooth function $f(t)$:

$$\|f(t) - \sum a_k \phi(t-k)\| \le C\|f'(t)\|,$$

for suitable a_k. The particular choice $a_k = f(k)$ is the "quasi–interpolate". When the mesh size changes from 1 to h, by rescaling t, the familiar factor h appears on the right side and the approximation error is $O(h)$.

All this followed from a simple zero $H(-1) = 0$. *Suppose that the zero at $z = -1$ has higher multiplicity p.* Then the corresponding steps lead to the Strang–Fix condition of order p: the derivatives are $\hat{\phi}^{(j)}(2\pi n) = 0$ for $n \ne 0$ and $j < p$. This is equivalent to pth-order approximation by the translates of $\phi(t)$.

Theorem 1 *A pth-order zero of $H(z)$ at $z = -1$ implies these properties, provided $\phi(t)$ exists in L^2:*

1. *The translates $\phi(t-n)$ can reproduce all polynomials of degree less than p.*

2. *The translates give pth-order approximation of a smooth $f(t)$:*

$$\|f(t) - \sum a_k \phi(t-k)\| \leq C_p \|f^{(p)}(t)\| \tag{9}$$

   for suitable a_k. Again the rescaling of t produces the factor h^p in (9).

3. *The wavelets that are orthogonal to the scaling functions have p vanishing moments.*

Orthogonality to the scaling functions $\phi(t-k)$ means orthogonality to their combinations $1, t, \ldots, t^{p-1}$. This gives the vanishing moments:

$$\int_{-\infty}^{\infty} t^m \tilde{w}_{jk}(t)dt = \int_{-\infty}^{\infty} t^m 2^{j/2} \tilde{w}(2^j t - k)dt = 0 \qquad \text{for } m < p.$$

Note! The convention is that $\phi(t)$ without the tilde is the *synthesis* function. Thus the coefficients in (4) and (5) should be written $f_0(k)$ instead of $h(k)$. The *analysis* functions $\tilde{\phi}(t)$ and $\tilde{w}(t)$ are constructed from

$$\begin{aligned} \tilde{\phi}(t) &= 2\sum h_0(k)\tilde{\phi}(2t-k) \\ \tilde{w}(t) &= 2\sum h_1(k)\tilde{\phi}(2t-k). \end{aligned}$$

Biorthogonality between tilde and non-tilde follows from the perfect reconstruction conditions, at every step of the iteration that solves the dilation equation. This iteration is the "cascade algorithm":

$$\tilde{\phi}^{(i+1)}(t) = 2\sum h_0(k)\tilde{\phi}^{(i)}(t). \tag{10}$$

The initial $\tilde{\phi}^{(0)}(t)$ is the box function on [0,1]. *But the iteration may not converge.* There is a condition on the eigenvalues of the matrix $T = (\downarrow 2)2HH^T$:
Condition E: All eigenvalues satisfy

$$|\lambda(T)| < 1$$

except for a simple eigenvalue $\lambda = 1$.

This gives L^2 convergence of the cascade algorithm [1]. The scaling function basis and the wavelet basis are stable [2]. Furthermore the smoothness of $\tilde{\phi}(t)$ and $\tilde{w}(t)$ are determined [3] by the spectral radius $\rho = |\lambda_{\max}(T)|$, when we exclude the special eigenvalues $\lambda = 1, \frac{1}{2}, \frac{1}{4}, \ldots, (\frac{1}{2})^{2p-1}$ that are automatic from the pth order zero of $H(z)$ at $z = -1$. The number of derivatives of $\tilde{\phi}(t)$ and $\tilde{w}(t)$ in L^2 is given by

$$s_{\max} = -\log \rho / \log 4. \tag{11}$$

Each additional factor $(1+z^{-1})/2$ in $H(z)$ increases p by 1 and $s_{\max}$ by 1. The new $\tilde{\phi}(t)$ is just the convolution of the old $\tilde{\phi}(t)$ with the box function. The perfect examples are *B-splines*, which come from the "pure" filter

$$H(z) = \left(\frac{1+z^{-1}}{2}\right)^p .$$

The B–spline of degree $p-1$ has $s_{\max} = p - 1/2$. Of course the B–splines are not biorthogonal to themselves (except for the box function when $p=1$). The product $F_0(z)H_0(z)$ is allowed only one odd power $z^{-\ell}$. We now create filters with this perfect reconstruction property, and compare the functions that come out of the cascade algorithm.

4. Three Choices of 9/7 Symmetric Biorthogonal Filters

The most popular filters are constructed by factoring a Daubechies polynomial of degree $4p-2$:

$$D_{4p-2}(z) = (1+z^{-1})^{2p} Q_{2p-2}(z).$$

$Q(z)$ is needed so that $D(z)$ will have only one odd power (with coefficient 1). The unique polynomial Q_{2p-2} of lowest degree $2p-2$ is the Daubechies choice. Examples are

$$\begin{aligned} D_6(z) &= (1+z^{-1})^4\ (-1+4z^{-1}-z^{-2})/32 \\ D_{14}(z) &= (1+z^{-1})^8\ (-5+40z^{-1}-131z^{-2}+208z^{-3}-131z^{-4}+40z^{-5}-5z^{-6})/2^{12}. \end{aligned}$$

The polynomial Q_{2p-2} comes directly from the binomial expansion of $(1-y)^{-p}$, truncated after p terms [1]. This polynomial of degree $(p-1)$ in $y = (2-z-z^{-1})/4$ becomes a polynomial of degree $2p-2$ in z^{-1} (after shifting by z^{1-p}).

Here are three filter banks with interesting properties. Many others are interesting too!

1. The FBI 9/7 filters were constructed by Daubechies and chosen by the FBI in digitizing fingerprints. $H_0(z)$ and $F_0(z)$ are factors of $D_{14}(z)$, each with $p=4$ zeros at $z=-1$. (Thus $(1+z^{-1})^8$ is split down the middle.) The other factors of degree 4 in H_0 and 2 in F_0 are chosen to preserve the symmetry of $Q_6(z)$. Real reciprocal roots z and $1/z$ go into F_0, and complex reciprocal roots $z, \bar{z}, 1/z, 1/\bar{z}$ go into H_0. *The coefficients are not rational*, and we give (inadequately) two decimals of the filter coefficients:

$$\begin{aligned} h_0 &= [\ 0.03\ \ -0.02\ \ -0.08\ \ 0.27\ \ 0.60\ \ 0.27\ \ -0.08\ \ -0.02\ \ 0.03] \\ f_0 &= [\ -0.05\ \ -0.03\ \ 0.30\ \ 0.56\ \ 0.30\ \ -0.03\ \ -0.05]. \end{aligned}$$

These filters are frequently used in image compression. Our normalization is

$$\sum h_0(k) = \sum f_0(k) = 1.$$

Then an extra $\sqrt{2}$ is needed in all four filters $H_0,\ H_1,\ F_0,\ F_1$, by Eq.(1).

2. The spline 9/7 filters come from a different factorization of $D_{14}(z)$, in which $F_0(z) = (1+z^{-1})^6/64$. Then $\phi(t)$ is the extremely smooth B–spline of degree 5 (with 4 continuous derivatives). But this leaves only two zeros at -1 for $H_0(z)$, which must swallow $Q_6(z)$ whole:

$$\begin{array}{lllllllllll} h_0 & = [& -5 & 30 & -56 & -14 & 154 & -14 & -56 & 30 & -5]/64 \\ f_0 & = [& 1 & 6 & 15 & 20 & 15 & 6 & 1]/64. \end{array}$$

This is a foolish choice. We will see that there is no L_2 solution to the dilation equation involving h_0. Condition E fails.

3. The "binary" 9/7 filter bank selects $F_0(z) = D_6(z)$. This choice (by the physicist Tomas Arias of M.I.T.) surprised the author. In itself it has one odd power z^{-3}—which leads to the useful interpolating property $\phi(n) = \delta(n-3)$. But it is the product F_0H_0 that must have one odd power, and what is H_0? The biorthonormal filter is needed in compression, if not in physics.

The lowest degree is 8 for a symmetric H_0 with a zero at $z=-1$. A direct calculation gives

$$\begin{array}{lllllllllll} h_0 & = [& 1 & 0 & -8 & 16 & 46 & 16 & -8 & 0 & 1]/64 \\ f_0 & = [& -1 & 0 & 9 & 16 & 9 & 0 & -1]/32. \end{array}$$

Notice that all coefficients are integers divided by powers of 2. This means perfect arithmetic, and fast execution on a chip.

We communicated this construction to Wim Sweldens who responded that he had already found the same binary filters. His method of "lifting" is extremely useful[5]. Starting from one admissible pair, in this case $H_0 \equiv 1$ and $F_0 = D_6$, the choice $H_{\text{new}}(z) = H_0(z) + F_0(-z)S(z^2)$ also gives perfect reconstruction. The unrestricted S allows our filter (Wim's filter) to have two zeros at $z=-1$. With four zeros the lengths become 13/7, and our compression experiments were less satisfactory—we don't know why.

5. Smoothness and Behavior of the Filters

We come now to a *comparison* of the three examples: FBI, spline and binary. Various measures are easy to compute. They give partial information:

Zeros at $z=-1$: 4/4, 2/6, 2/4 (thus binary loses)
Smoothness $s_{\max}$ of $\tilde{\phi}(t)$ and $\phi(t)$: 1.4/2.1, -2.2/5.5, 0.59/2.44.

The largest non–special eigenvalues of the matrix T for the binary h_0 and f_0 were 0.4394 and 0.0339. The smoothness $s_{\max} = 0.59$ and 2.44 came directly from Eq. (11).

The largest non–special eigenvalue for the foolish choice h_0, biorthonormal to the quintic B–spline filter, was 21.314. The function $\tilde{\phi}(t)$ is a wild distribution (-2.2 derivatives in L_2). This example is *not* to show that splines are a poor choice—they are often very good. But we must keep enough zeros at $z = -1$ in both $H_0(z)$ and $F_0(z)$. When f_0 with binomial coefficients gives a spline, stability and smoothness may require a longer h_0 than would be needed for perfect reconstruction.

A third important quantity is the *coding gain*, to give the expected compression for inputs that have Markov correlation 0.95 between neighboring pixels. None of those measures is totally consistent with human visual perception. Therefore filters are generally chosen for good looks on well–known images.

We were able to compare the FBI 9/7 with the binary 9/7 on a "boats" image [1].

25:1 compression			50:1 compression	
FBI	binary		FBI	binary
40.55	40.14	MEAN SQUARE ERROR	84.55	85.92
32.05	32.10	PEAK SIGNAL/NOISE RATIO	28.86	28.79
43.71	45.07	MAXIMUM ERROR	68.81	81.99
OK	better	PERCEPTUAL QUALITY	OK	better

Note the slight advantage of the FBI, objectively. Note the equally slight advantage of the binary filters, subjectively. We believe that the binary property and the interpolation property may be significantly useful in applications.

Acknowledgement I would like to add a less conventional tribute to Ron Mitchell by thanking him for teaching my sons to play "football".

6. References

1. Gilbert Strang and Truong Nguyen, *Wavelets and Filter Banks*, Wellesley–Cambridge Press (1996).
2. Albert Cohen and Ingrid Daubechies, A stability criterion for biorthogonal wavelet bases and their related subband coding schemes, Duke Math. J., 68:313–335 (1992).
3. Lars Villemoes, Wavelet analysis of refinement equations, SIAM J. Math. Anal., 25:1433–1466 (1994).
4. Gilbert Strang, Eigenvalues of $(\downarrow 2)H$ and convergence of the cascade algorithm, IEEE Trans. on Signal Processing (1996).
5. Wim Sweldens, The lifting scheme: a custom–design construction of biorthonormal wavelets, Appl. Comput. Harm. Anal., (To appear).