THE ROYAL SOCIETY

Image processing with complex wavelets

By Nick Kingsbury

Signal Processing Group, Department of Engineering, University of Cambridge, Cambridge CB2 1PZ, UK (ngk@eng.cam.ac.uk)

We first review how wavelets may be used for multi-resolution image processing, describing the filter-bank implementation of the discrete wavelet transform (DWT) and how it may be extended via separable filtering for processing images and other multi-dimensional signals. We then show that the condition for inversion of the DWT (perfect reconstruction) forces many commonly used wavelets to be similar in shape, and that this shape produces severe shift dependence (variation of DWT coefficient energy at any given scale with shift of the input signal). It is also shown that separable filtering with the DWT prevents the transform from providing directionally selective filters for diagonal image features.

Complex wavelets can provide both shift invariance and good directional selectivity, with only modest increases in signal redundancy and computation load. However, development of a complex wavelet transform (CWT) with perfect reconstruction and good filter characteristics has proved difficult until recently. We now propose the dual-tree CWT as a solution to this problem, yielding a transform with attractive properties for a range of signal and image processing applications, including motion estimation, denoising, texture analysis and synthesis, and object segmentation.

Keywords: image processing; wavelets; shift invariance; directional filters; perfect reconstruction; complex filters

1. Introduction

In this paper we consider how wavelets may be used for image processing. To date, there has been considerable interest in wavelets for image compression, and they are now commonly used by researchers for this purpose, even though the main international standards still use the discrete cosine transform (DCT). However, for image processing tasks, other than compression, the take-up of wavelets has been less enthusiastic. Here we analyse possible reasons for this and present some new ways to use wavelets that offer significant advantages.

A good review of wavelets and their application to compression may be found in Rioul & Vetterli (1991) and in-depth coverage is given in the book by Vetterli & Kovacevic (1995). An issue of the Proceedings of the IEEE (Kovacevic & Daubechies 1996) has been devoted to wavelets and includes many very readable articles by leading experts.

In § 2 of this paper we introduce the basic discrete wavelet filter tree and show how it may be used to decompose multi-dimensional signals. In § 3 we show some typical wavelets and illustrate the similar shapes of those that all satisfy the perfect reconstruction constraints. Unfortunately, as explained in § 4, discrete wavelet

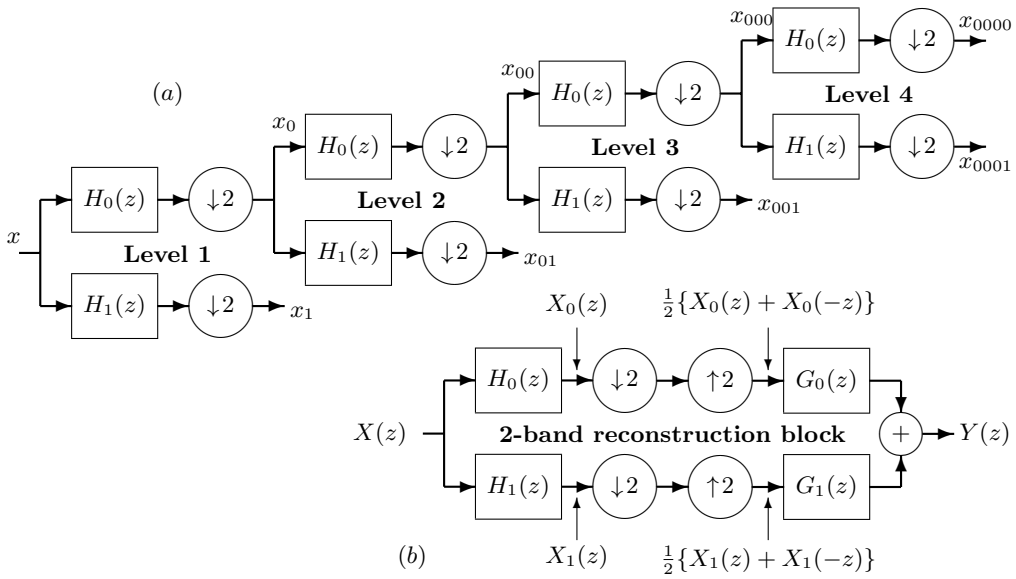

Figure 1. (*a*) Four-level binary wavelet tree of lowpass filters, H_0, and highpass filters, H_1; and (*b*) the two-band reconstruction block, used to achieve perfect reconstruction from an inverse tree employing filters, G_0 and G_1.

decompositions based on these typical wavelets suffer from two main problems that hamper their use for many image analysis and reconstruction tasks as follows.

(i) Lack of *shift invariance*, which means that small shifts in the input signal can cause major variations in the distribution of energy between wavelet transform coefficients at different scales.

(ii) Poor *directional selectivity* for diagonal features, because the wavelet filters are separable and real.

Complex wavelets are shown, in § 5, to overcome these two key problems by introducing limited redundancy into the transform. However, a further problem arises here because perfect reconstruction becomes difficult to achieve for complex wavelet decompositions beyond level 1, when the input to each level becomes complex. To overcome this, we have recently developed the dual-tree complex wavelet transform (DT CWT), which allows perfect reconstruction while still providing the other advantages of complex wavelets. This is described in § 6, and then § 7 discusses constraints on the DT CWT filter characteristics, required in order to achieve close approximations to shift invariance. Finally, in § 8, we suggest possible application areas for the DT CWT, including an example of image denoising.

2. The wavelet tree for multi-dimensional signals

For one-dimensional signals, the conventional discrete wavelet transform (DWT) may be regarded as equivalent to filtering the input signal with a bank of bandpass filters, whose impulse responses are all approximately given by scaled versions of a *mother wavelet*. The scaling factor between adjacent filters is usually 2:1, leading to octave

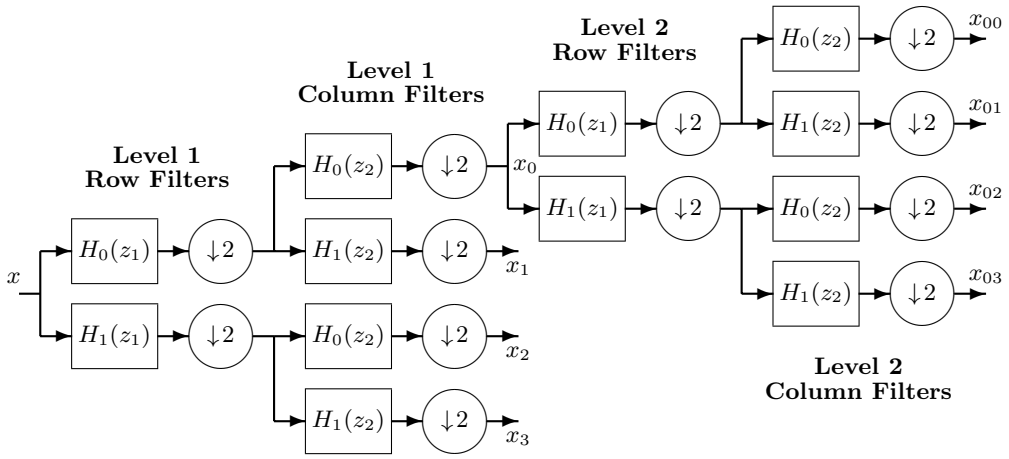

Figure 2. Two levels of the quaternary separable wavelet tree, normally used for two-dimensional signals. Row filtering is denoted by the z transform parameter z_1, and column filtering by z_2.

bandwidths and centre frequencies that are one octave apart. At the coarsest scale, a lowpass filter is also required to represent the lowest frequencies of the signal. The outputs of the filters are usually maximally decimated so that the number of DWT output samples equals the number of input samples and the transform is invertible. The octave-band DWT is most efficiently implemented by the *dyadic wavelet decomposition tree* of Mallat (1989), a cascade of two-band perfect-reconstruction filter banks, shown in figure 1*a*. Because of the decimation by two at each stage, the total output sample rate equals the input sample rate and there is *no redundancy* in the transform.

In order to reconstruct the signal, a pair of reconstruction filters G_0 and G_1 are used in the arrangement of figure 1*b*, and usually the filters are designed such that the z transform of the output signal $Y(z)$ is identical to that of the input signal $X(z)$. This is known as the condition for *perfect reconstruction*. Hence, in figure 1*a*, x_{000} may be reconstructed from x_{0000} and x_{0001}; and then x_{00} from x_{000} and x_{001}; and so on back to x, using an inverse tree of filters, $G_{\ldots}(z)$.

If the input signal is two dimensional, the binary tree may be extended into a quad-tree structure, as shown for two levels in figure 2. In a separable implementation, each level of the quad-tree comprises two stages of filtering: the first stage typically filters the rows of the image to generate a pair of horizontal lowpass and highpass subimages; and then the second stage filters the columns of each of these to produce four subimages, $x_0, \ldots, x_3$, each one-quarter of the area of x. The lowpass subimage, x_0, is similar to the original but smaller, and is typically decomposed by further levels of the two-dimensional transform.

The filtering in figure 2 is *separable* since it is performed separately in the row and column directions. This is usually the most efficient way to perform two-dimensional filtering. The technique may be extended, straightforwardly, to more than two dimensions, by applying filters to each dimension in turn. For m dimensions, the number of sub-bands (subimages) at each level increases to 2^m (including the low-frequency sub-band, passed on to the next level).

3. Common wavelets with perfect reconstruction

The art of finding good wavelets lies in the design of the set of filters, $\{H_0, H_1, G_0, G_1\}$ from figure 1*b*, to achieve various trade-offs between (signal-dependent) spatial and frequency domain characteristics while satisfying the perfect-reconstruction (PR) condition. We now briefly consider this design process.

In figure 1*b*, multi-rate filter analysis shows that

$$\begin{aligned} Y(z) &= \tfrac{1}{2}\{X_0(z) + X_0(-z)\}G_0(z) + \tfrac{1}{2}\{X_1(z) + X_1(-z)\}G_1(z) \\ &= \tfrac{1}{2}X(z)\{H_0(z)G_0(z) + H_1(z)G_1(z)\} \\ &\quad + \tfrac{1}{2}X(-z)\{H_0(-z)G_0(z) + H_1(-z)G_1(z)\}. \end{aligned} \tag{3.1}$$

The first PR condition requires aliasing cancellation and forces the above term in $X(-z)$ to be zero. Hence, $H_0(-z)G_0(z) + H_1(-z)G_1(z) = 0$, which can be achieved if

$$H_1(z) = z^{-k}G_0(-z) \quad \text{and} \quad G_1(z) = z^k H_0(-z), \tag{3.2}$$

where k must be odd (usually $k = \pm 1$).

The second PR condition is that the transfer function from $X(z)$ to $Y(z)$ should be unity; i.e. $H_0(z)G_0(z) + H_1(z)G_1(z) = 2$. If we define a product filter $P(z) = H_0(z)G_0(z)$ and substitute the results from (3.2), then this condition becomes:

$$H_0(z)G_0(z) + H_1(z)G_1(z) = P(z) + P(-z) = 2. \tag{3.3}$$

Since the odd powers of z in $P(z)$ cancel with those in $P(-z)$, this requires that $p_0 = 1$ and that $p_n = 0$ for all n even and non-zero.

$P(z)$ is the transfer function of the lowpass branch in figure 1*b* (excluding the effects of the decimator and interpolator), and $P(-z)$ is that of the highpass branch. For image processing applications, $P(z)$ should be zero phase in order to minimize distortions when the wavelet coefficients are modified in any way; so to obtain PR it must be of the form:

$$P(z) = \cdots + p_5 z^5 + p_3 z^3 + p_1 z + 1 + p_1 z^{-1} + p_3 z^{-3} + p_5 z^{-5} + \cdots . \tag{3.4}$$

To simplify the tasks of choosing $P(z)$, based on the zero-phase symmetry we usually transform $P(z)$ into $P_t(Z)$ such that:

$$P(z) = P_t(Z) = 1 + p_{t,1}Z + p_{t,3}Z^3 + p_{t,5}Z^5 + \cdots , \quad \text{where } Z = \tfrac{1}{2}(z + z^{-1}). \tag{3.5}$$

If T_s is the sampling period, the frequency response is given by $z = \mathrm{e}^{j\omega T_\mathrm{s}}$, and therefore by $Z = \cos(\omega T_\mathrm{s})$. To obtain smooth wavelets (after many levels of decomposition), Daubechies (1990) has shown that $H_0(z)$ and $G_0(z)$ should have a number of zeros at $z = -1$ ($\omega T_\mathrm{s} = \pi$), so $P_t(Z)$ needs zeros at $Z = -1$. In general, more zeros at $Z = -1$ (and, hence, at $z = -1$) produce smoother wavelets.

Figure 3 shows five different wavelets satisfying the above equations. They are shown after four levels of decomposition. On the left are the wavelets formed from the analysis filters, H, and on the right are the wavelets from the reconstruction filters, G. These filters may be swapped, so we have chosen the smoother wavelets for reconstruction, since, for many applications (particularly compression), this gives the least visible distortions.

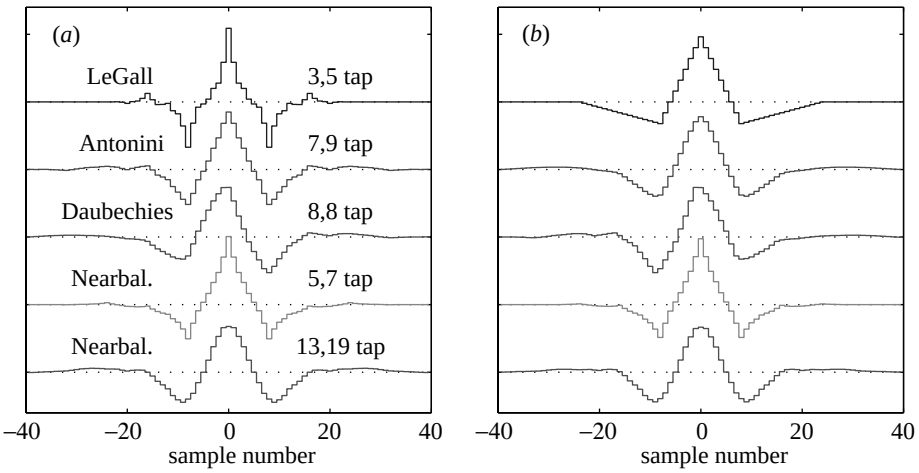

Figure 3. Comparison of some common wavelets. (*a*) Level 4 analysis wavelets. (*b*) Level 4 reconstruction wavelets.

The LeGall (3,5)-tap wavelet is designed by choosing $P_t(Z)$ to be third order, and constraining the maximum number (two) of zeros to be at $Z = -1$, consistent with the even-order term(s) in equation (3.5) being zero. This gives filters, H_0 and G_0, which are five and three taps long, respectively. The Antonini (7,9)-tap wavelet is obtained in the same way, when $P_t(Z)$ is seventh order. In this case there are four zeros at $Z = -1$. This wavelet is one of the most favoured by image-compression researchers and is used in the FBI fingerprint-compression system.

Both of these wavelets are constrained such that all the H and G filters are linear-phase finite impulse-response filters. Unfortunately, this means that the reconstruction filters cannot have the same frequency responses as the analysis filters. Wavelets of this type are known as *biorthogonal*.

For the frequency responses to be the same, the wavelets must be *orthogonal*. The Daubechies (8,8)-tap wavelet is an orthogonal wavelet, obtained from the same seventh-order $P_t(Z)$ as the Antonini wavelet, but where the factors of z in $P(z)$ are grouped into reciprocal pairs that are divided equally between H_0 and G_0 so as to provide the best approximation to linear-phase responses.

The final two wavelets in figure 3 are linear-phase wavelets that have frequency responses that are *nearly balanced* between analysis and reconstruction, and are, therefore, nearly orthogonal. To achieve this condition, we increase the order of $P_t(Z)$ by two, without adding further zeros at $Z = -1$, and hence provide an additional design parameter that can be used to obtain approximate balance.

We find (Tay & Kingsbury 1993) that the following factorization of a fifth-order $P_t(Z)$ produces good balance:

$$P_t(Z) = \tfrac{1}{50}(50 + 41Z - 15Z^2 - 6Z^3) \cdot \tfrac{1}{7}(7 + 5Z - 2Z^2) = H_{0t}(Z) \cdot G_{0t}(Z). \quad (3.6)$$

Using the transformation $Z = \frac{1}{2}(z + z^{-1})$ gives the simplest near-balanced wavelets with five and seven tap filters. These wavelets are well balanced but have quite sharp cusps, as can be seen in the fourth row of figure 3.

Smoother near-balanced wavelets may be obtained by employing a higher-order

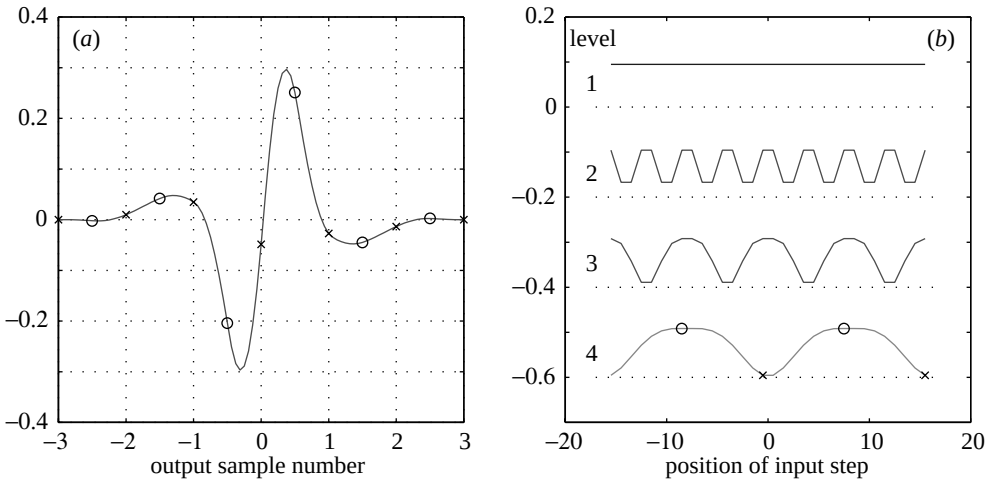

Figure 4. (*a*) Step response at level 4 of the Antonini (7,9)-tap wavelet; and (*b*) its shift dependence, showing the variation of energy in each level of wavelet coefficients, for a unit step input as the position of the input step is shifted. The zero lines (shown dotted) for each level have been offset for clarity.

transformation from Z to z, such as

$$Z = pz^3 + (\tfrac{1}{2} - p)(z + z^{-1}) + pz^{-3}.$$

Four zeros on the unit circle near $z = -1$ are achieved for each zero, $Z = -1$, if $p = -\frac{3}{32}$. When substituted into $P_t(Z)$, this gives relatively high-order filters with 13 and 19 taps (although two taps of each filter are zero and they do have an efficient ladder implementation). The final row of figure 3 shows the improved smoothness of these wavelets.

4. Some problems with common wavelets

(*a*) *Shift dependence*

Figure 3 shows a strong similarity between the shapes of various wavelets. This is because PR constrains each filter in figure 1 to be approximately a half-band filter. This causes aliasing and results in severe *shift dependence* of the wavelet transform.

When we analyse the Fourier spectrum of a signal, we expect the energy in each frequency bin to be invariant to any shifts of the input in time or space. It would be desirable if wavelet transforms behaved similarly, but, unfortunately, real wavelet transforms, even though they have perfect reconstruction properties, do not provide energy shift invariance separately at each level.

Consider a step function input signal, analysed with the DWT using Antonini (7,9)-tap filters. The step response at wavelet level 4 is shown in figure 4*a*, assuming that wavelet coefficients are computed at the full input sampling rate. In practice, they are computed at $\frac{1}{16}$ of this rate, yielding samples at points such as those of the crosses in figure 4*a*. If the input step is shifted relative to the output sampling grid, then this is equivalent to sampling the step response with a different horizontal offset; e.g. for an offset of eight input samples, we obtain samples at the circles in figure 4*a*.

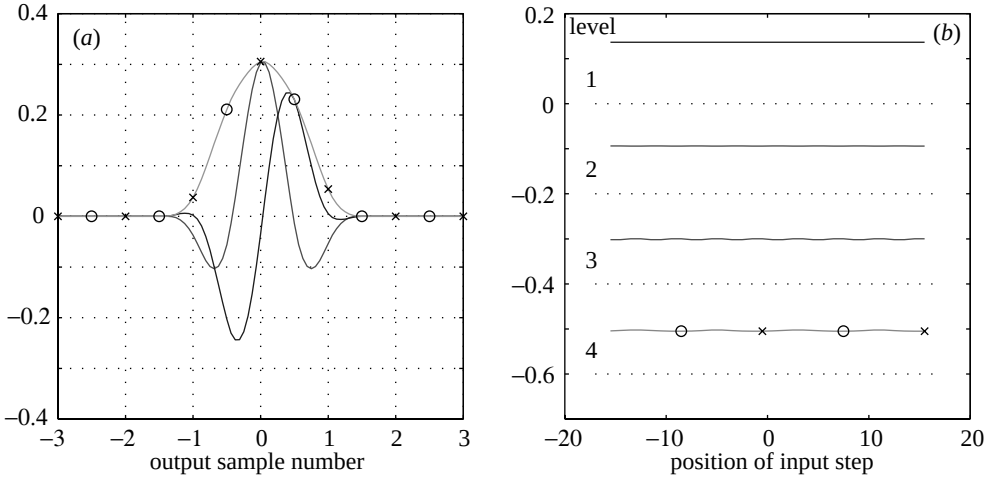

Figure 5. (*a*) Step response at level 4 of complex wavelet (*a*). (*b*) Shift dependence of output energy at levels 1 to 4, showing the variation of energy in each level of wavelet coefficient for a unit step input as the position of the input step is shifted.

Now, comparing the total energy of the samples at the crosses (which are all quite small in amplitude) with the energy of the samples at the circles (two of which are rather large), we find a large energy fluctuation (over 24:1). This illustrates a significant drawback to using the standard DWT as a tool for analysing signals; the energy distribution between the various wavelet levels depends critically on the *position* of key features of the signal relative to the wavelet subsampling grid. Ideally, we would like it to depend on just the features themselves.

This problem is illustrated more generally in figure 4*b*, which shows how the total energy at each wavelet level varies as the input step is shifted. The period of each variation equals the subsampling period at that level, and the crosses and circles in figure 4*b* show the energies at level 4 corresponding to the shift positions shown by the equivalent symbols in figure 4*a*.

Hence we conclude that, due to this shift dependence, real DWTs are unlikely to give consistent results when used to detect key features in images. This problem is caused by aliasing due to subsampling at each wavelet level. It can, therefore, be avoided by not decimating the DWT outputs. However, this produces considerable data redundancy because each subimage at each wavelet level is the same size as the input image. This is often known as the undecimated wavelet transform, but, in addition to being computationally inefficient (even when using the 'à trous' algorithm), this does not solve the other main wavelet problem, discussed below.

(*b*) *Poor directional selectivity*

In figure 2, separable filtering of the rows and columns of an image produces four subimages at each level. The Lo–Hi and Hi–Lo bandpass subimages (e.g. x_1 and x_2) can select mainly horizontal or vertical edges, respectively, but the Hi–Hi subimage (x_3) contains components from diagonal features of either orientation. This means that the separable real DWT has *poor directional selectivity*.

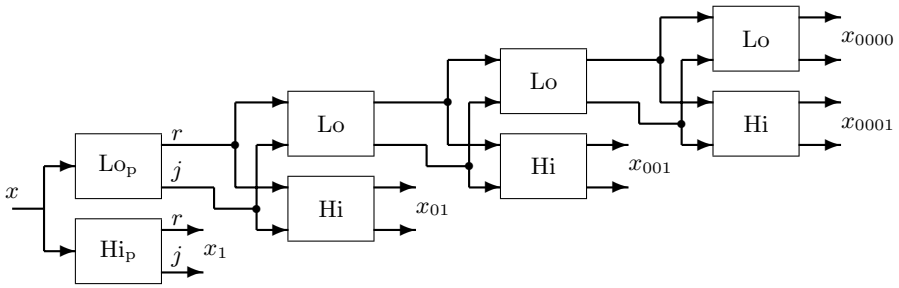

Figure 6. Four levels of the complex wavelet tree for a real one-dimensional input signal x. The real and imaginary parts (r and j) of the inputs and outputs are shown separately. Where there is only one input to a block, it is a real signal.

One way of explaining this is that real highpass row filters select both positive and negative horizontal high frequencies, while real highpass column filters select both positive and negative vertical high frequencies. Hence, the combined Hi–Hi filter must have passbands in all four quadrants of the two-dimensional frequency plane. On the other hand, a directionally selective filter for diagonal features with positive gradient must have passbands only in quadrants 2 and 4 of the frequency plane, while a filter for diagonals with negative gradient must have passbands only in quadrants 1 and 3. The poor directional properties of real separable filters make it difficult to generate steerable or directionally selective algorithms, based on the separable real DWT.

5. Properties of complex wavelets

It is found that both of the above problems can be solved effectively by the *complex* wavelet transform (CWT). The structure of the CWT is the same as in figure 1*a*, except that the CWT filters have complex coefficients and generate complex output samples. This is shown in figure 6, in which each block is a complex filter and includes down-sampling by 2 (not shown) at its outputs. Since the output sampling rates are unchanged from the DWT, but each sample contains a real and imaginary part, a redundancy of 2:1 is introduced (we shall show later that this becomes 4:1 in two dimensions).

The complex filters may be designed such that the *magnitudes* of their step responses vary slowly with input shift; only the *phases* vary rapidly. This is shown in figure 5*a*, in which the real and imaginary parts of a typical complex wavelet step response are superimposed and the uppermost curve represents the magnitude of the response. Note that the real part is an odd function while the imaginary part is even. This wavelet was derived from the following simple (4,4)-tap Lo and Hi filters:

$$\left.\begin{aligned} h_0 &= \tfrac{1}{10}[1-j, 4-j, 4+j, 1+j], \\ h_1 &= \tfrac{1}{48}[-3-8j, 15+8j, -15+8j, 3-8j]. \end{aligned}\right\} \tag{5.1}$$

The level 1 filters, $\mathrm{Lo_p}$ and $\mathrm{Hi_p}$ in figure 6, include an additional 2-tap prefilter, which has a zero at $z=-j$, in order to simulate the effect of a filter tree extending further levels to the left of level 1, as discussed by Magarey & Kingsbury (1998).

Figure 5*b* plots the energy at each level versus input step position for this CWT, and, in contrast to figure 4*b*, shows that it is approximately constant at all levels. Hence, the energy of each CWT band may be made approximately *shift invariant*.

Simoncelli *et al*. (1992) have shown that shift invariance of energy distribution is equivalent to the property of interpolability.

The other key property of these complex wavelets is that their phases vary approximately linearly with input shift (as with Fourier coefficients). Thus, based on measurement of phase shifts, efficient displacement estimation is possible and interpolation between consecutive complex samples can be relatively simple and accurate. Further details of the design of these wavelets and of their application to motion estimation are given in Magarey & Kingsbury (1998).

(*a*) *Extension to multiple dimensions*

Extension of complex wavelets to two dimensions is achieved by separable filtering along rows and then columns. However, if row and column filters both suppress negative frequencies, then only the first quadrant of the two-dimensional signal spectrum is retained. Two adjacent quadrants of the spectrum are required to represent fully a real two-dimensional signal, so an extra 2:1 factor of redundancy is required, producing 4:1 redundancy overall in the transformed two-dimensional signal. This is achieved by additional filtering with complex conjugates of either the row or column filters. If the signal exists in m-dimensional space ($m > 2$), then further conjugate pairs of filters are needed for each dimension leading to redundancy of 2^m:1.

The most computationally efficient way to achieve the pairs of conjugate filters is to maintain separate imaginary operators, j_1 and j_2, for the row and column processing, as shown in figure 7. This produces four-element 'complex' vectors: $\{a, b, c, d\} = a + bj_1 + cj_2 + dj_1j_2$. Note that these are not quaternions as they have different properties. Each 4-vector can be converted into a pair of conventional complex 2-vectors, by letting $j_1 = j_2 = j$ in one case and $j_1 = -j_2 = -j$ in the other case. This corresponds to sum and difference operations on the $\{a, d\}$ and $\{b, c\}$ pairs and produces two complex outputs, $(a-d)+(b+c)j$ and $(a+d)+(-b+c)j$, corresponding to first and second quadrant directional filters, respectively. The $\sum/\Delta$ blocks in figure 7 do this.

Complex filters in multiple dimensions provide true directional selectivity, despite being implemented separably, because they are still able to separate all parts of the m-dimensional frequency space. For example, a two-dimensional CWT produces six bandpass subimages of complex coefficients at each level, which are strongly orientated at angles of $\pm 15°$, $\pm 45°$ and $\pm 75°$, depicted by the double-headed arrows in figure 7 and by the two-dimensional impulse responses in figure 12.

6. The dual-tree complex wavelet transform

For many applications it is important that the transform be perfectly invertible. A few authors, such as Lawton (1993) and Belzer *et al*. (1995), have experimented with complex factorizations of the standard Daubechies polynomials and obtained PR complex filters, but these do not give filters with good frequency-selectivity properties. To provide shift invariance and directional selectivity, all of the complex filters should emphasize positive frequencies and reject negative frequencies, or vice versa (see §§ 5 and 7).

Unfortunately, it is very difficult to design an inverse transform, based on complex filters of the type defined at the start of § 5, which has good frequency selectivity *and*

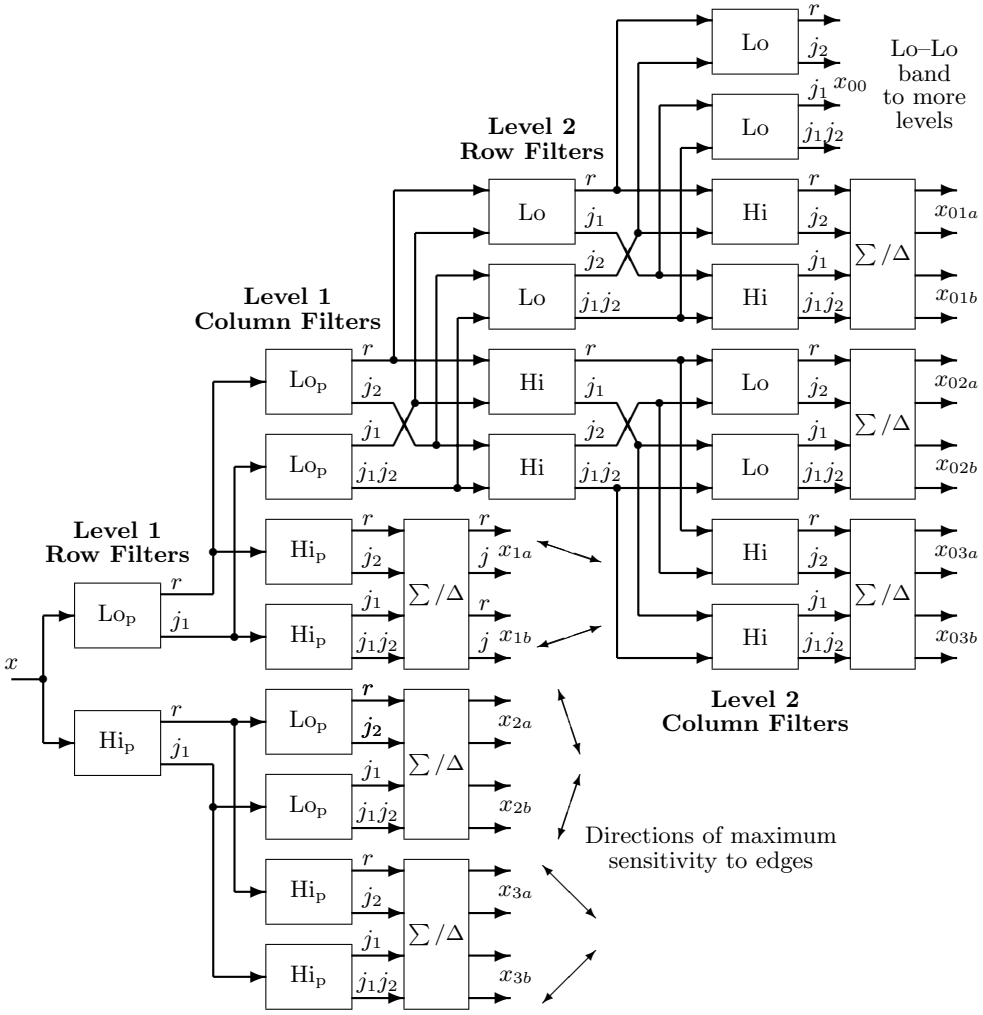

Figure 7. Two levels of the complex wavelet tree for a real two-dimensional input image x, giving six directional bands at each level (the directions are shown for level 1). Components of four-element 'complex' vectors are labelled r, j_1, j_2, j_1j_2.

PR at all levels of the transform. Although such filters can be designed to give PR quite easily at level 1 of the tree by applying the constraint that the reconstructed output signal must be real, a similar constraint cannot be applied at further levels where inputs *and* outputs are complex. For PR below level 1, the set of four filters in figure 1*b* must have a flat overall frequency response. However, this is not possible if all of the filters tend to reject negative frequencies. Hence, a different approach to generating a complex filter tree is needed.

In Kingsbury (1998*a*, *b*), we introduced the DT CWT, which added perfect reconstruction to the other attractive properties of complex wavelets: shift invariance; good directional selectivity; limited redundancy; and efficient order-N computation.

The dual-tree transform was developed by noting that approximate shift invariance can be achieved with a *real* DWT by doubling the sampling rate at each level of the

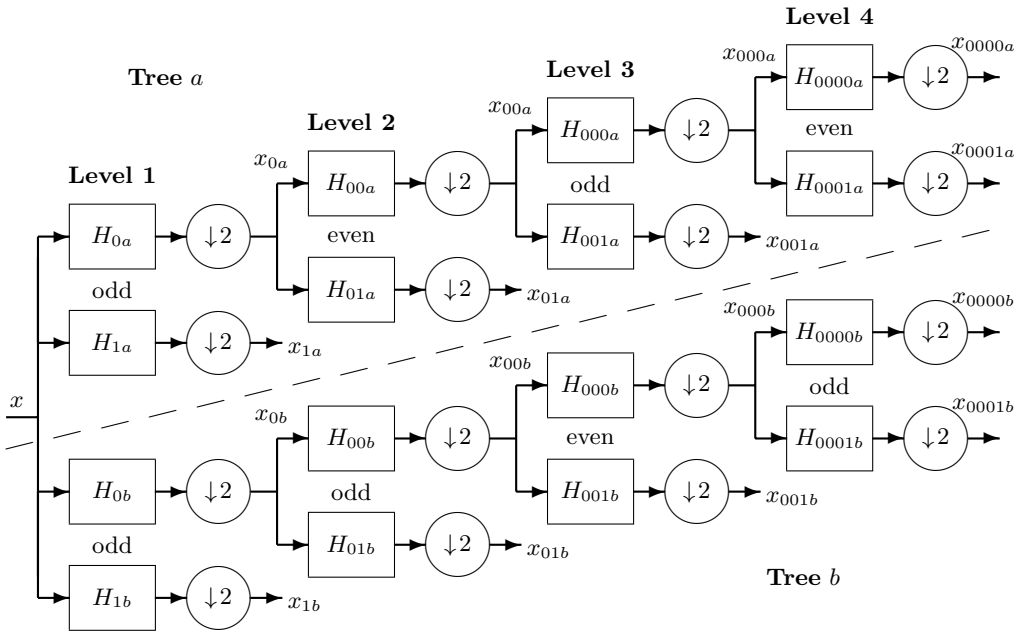

Figure 8. Dual tree of real filters for the DT CWT, showing whether the filters have odd or even length. The two trees give the real and imaginary parts of the complex coefficients.

Input samples	← Block of 16 input samples →															
x:	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•
Level 1 samples																
odd Lo x_{0a}:	a		a		a		a		a		a		a		a	
odd Lo x_{0b}:		b		b		b		b		b		b		b		b
odd Hi x_{1a}:		a		a		a		a		a		a		a		a
odd Hi x_{1b}:	b		b		b		b		b		b		b		b	
Level 2 samples																
even Lo x_{00a}:		a				a				a				a		
odd Lo x_{00b}:				b				b				b				b
Hi x_{01a}, x_{01b}:		$*$				$*$				$*$				$*$		
Level 3 samples																
odd Lo x_{000a}:		a								a						
even Lo x_{000b}:						b								b		
Hi x_{001a}, x_{001b}:						$*$								$*$		
Level 4 samples																
even Lo x_{0000a}:						a										
odd Lo x_{0000b}:														b		
Hi x_{0001a}, x_{0001b}:						$*$										

Figure 9. Effective sampling instants of odd and even filters in figure 8 assuming zero phase responses. (a = tree a, b = tree b, $*$ = combined samples.)

tree. For this to work, the samples must be evenly spaced. We can double all the sampling rates in figure 1*a* by eliminating the down-sampling by 2 after the level 1 filters. This is equivalent to having two parallel fully decimated trees, a and b in figure 8, provided that the delays of H_{0b} and H_{1b} are one sample offset from H_{0a} and H_{1a}. We then find that, to get uniform intervals between samples from the two trees *below* level 1, the filters in one tree must provide delays that are half a sample different

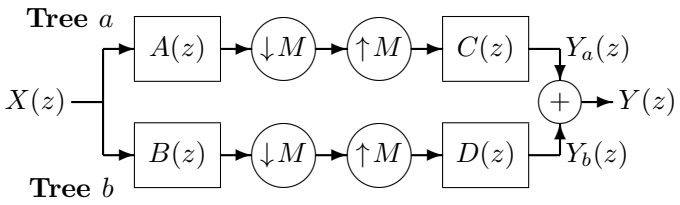

Figure 10. Basic configuration of the dual tree if either wavelet or scaling-function coefficients from just level m are retained ($M = 2^m$).

(at each filter input rate) from those in the other tree. For linear phase, this requires *odd-length* filters in one tree and *even-length* filters in the other. Greater symmetry between the two trees occurs if each tree uses odd and even filters alternately from level to level, but this is not essential. In figure 9 we show the effective sampling instants of the filter output samples when the filters are odd and even, as in figure 8. For example, at level 2, the tree a filters are even length, so the a output samples (x_{00a}) occur midway between pairs of samples from the a lowpass filter at level 1 (x_{0a}), whereas the tree b filters are odd length, so the b output samples (x_{00b}) are aligned with samples from the b lowpass filter at level 1 (x_{0b}).

To invert the transform, the PR filters G are applied in the usual way to invert each tree separately, and, finally, the two results are averaged. Compared with the undecimated (à trous) wavelet tree, which eliminates down-sampling after *every* level of filtering, the dual tree effectively eliminates down-sampling *only* after the level 1 filters, and, hence, its redundancy is much less.

Thus far, the dual tree does not appear to be a complex transform at all. However, when the outputs from the two trees in figure 8 are interpreted as the real and imaginary parts of complex wavelet coefficients, the transform effectively becomes complex. If the filters are from linear-phase PR biorthogonal sets, the odd-length highpass filters have even symmetry about their midpoint, while the even-length highpass filters have odd symmetry. The impulse responses of these then look very like the real and imaginary parts of the complex wavelets of the previous section. In fact, the block diagrams of figures 6 and 7 still apply to the dual-tree transform, although the operations within each filter block do change: the real and imaginary parts of each block's output are no longer calculated based on the usual rules of complex algebra, but, instead, each is based on filtering just one part of the block's input with either an odd- or even-length filter.

7. Shift-invariant filter design

In order to show the shift-invariant properties of the dual tree, we consider what happens when the signal is reconstructed using coefficients of just one type (wavelet or scaling function) from just one level of the dual tree. This models (in an extreme way) the virtual elimination of certain sub-bands that commonly occur in many algorithms. For example, we might choose to retain only the level-3 wavelet coefficients, x_{001a} and x_{001b}, from figure 8, and set all others to zero. If the signal y, reconstructed from just these coefficients, is free of aliasing, then the transform is defined to be shift invariant at that level.

Figure 10 shows the simplified analysis and reconstruction parts of the dual tree when coefficients of just one type and level are retained. All down(up)-sampling operations are moved to the output (input) of the analysis (reconstruction) filter banks and the cascaded filter transfer functions are combined. $M = 2^m$ is the total down-sampling factor. For example, if we retain only x_{001a} and x_{001b}, then $M = 8$, $A(z) = H_{0a}(z)H_{00a}(z^2)H_{001a}(z^4)$ and $B(z)$, $C(z)$, $D(z)$ are obtained similarly.

Letting $W = \mathrm{e}^{j2\pi/M}$, multirate analysis of figure 10 gives:

$$Y(z) = \frac{1}{M}\sum_{k=0}^{M-1} X(W^k z)[A(W^k z)C(z) + B(W^k z)D(z)]. \tag{7.1}$$

For shift invariance, the aliasing terms (for which $k \neq 0$) must be negligible. So we design $B(W^k z)D(z)$ to cancel out $A(W^k z)C(z)$ for all non-zero k that give overlap of the pass or transition bands of the filters $C(z)$ or $D(z)$ with those of the shifted filters $A(W^k z)$ or $B(W^k z)$. Separate strategies are needed depending on whether the filters are lowpass (for scaling functions) or bandpass (for wavelets).

For level m in the dual tree, the lowpass filters have passbands from $(-f_\mathrm{s}/2M)$ to $(f_\mathrm{s}/2M)$, where f_s is the input sampling frequency. The W^k terms in (7.1) shift the passbands in multiples, k, of f_s/M. If $A(z)$ and $C(z)$ have similar frequency responses (as required for near-orthogonal filter sets) and significant transition bands, it is not possible to make $A(Wz)C(z)$ small at all frequencies $z = \mathrm{e}^{j\theta}$, because the frequency shift, f_s/M, of $A(z)$ due to W is too small (the A and C responses tend to overlap at their -3 dB points). However, it is quite easy to design $A(W^2 z)C(z)$ to be small since the frequency shift of A is twice as great and the responses no longer overlap significantly. Hence, for the lowpass case, we design $B(W^k z)D(z)$ to cancel $A(W^k z)C(z)$ when k is odd by letting

$$B(z) = z^{\pm M/2}A(z) \quad \text{and} \quad D(z) = z^{\mp M/2}C(z), \tag{7.2}$$

so that $B(W^k z)D(z) = (-1)^k A(W^k z)C(z)$. In this way, the unwanted aliasing terms, mainly at $k = \pm 1$, are approximately cancelled out. This is equivalent to a single tree with a decimation by $\frac{1}{2}M$ rather than M.

Now consider the bandpass case. Here we find that the edges of the positive frequency passband of C or D, $(f_\mathrm{s}/2M) \rightarrow (f_\mathrm{s}/M)$, will tend to overlap with the edges of the negative frequency passband of A or B, which gets shifted either to $0 \rightarrow (f_\mathrm{s}/2M)$ or to $(f_\mathrm{s}/M) \rightarrow (3f_\mathrm{s}/2M)$ when $k = 1$ or 2, respectively. Similarly for the opposite passbands when $k = -1$ or -2. We find that the main aliasing terms are always caused by the overlap of opposing frequency passbands (i.e. passbands that have opposite polarity of centre frequency in the unshifted filters). It happens that the solution here is to give B and D positive and negative passbands of opposite polarity while A and C have passbands of the same polarity (or vice versa).

Suppose we have prototype *complex* filters $P(z)$ and $Q(z)$, each with just a single passband $(f_\mathrm{s}/2M) \rightarrow (f_\mathrm{s}/M)$ and zero gain at all negative frequencies, then we let

$$\left.\begin{aligned} A(z) &= \mathrm{Re}[2P(z)] = P(z) + P^*(z), \\ B(z) &= \mathrm{Im}[2P(z)] = -j[P(z) - P^*(z)], \\ C(z) &= \mathrm{Re}[2Q(z)] = Q(z) + Q^*(z), \\ D(z) &= \mathrm{Im}[-2Q(z)] = j[Q(z) - Q^*(z)], \end{aligned}\right\} \tag{7.3}$$

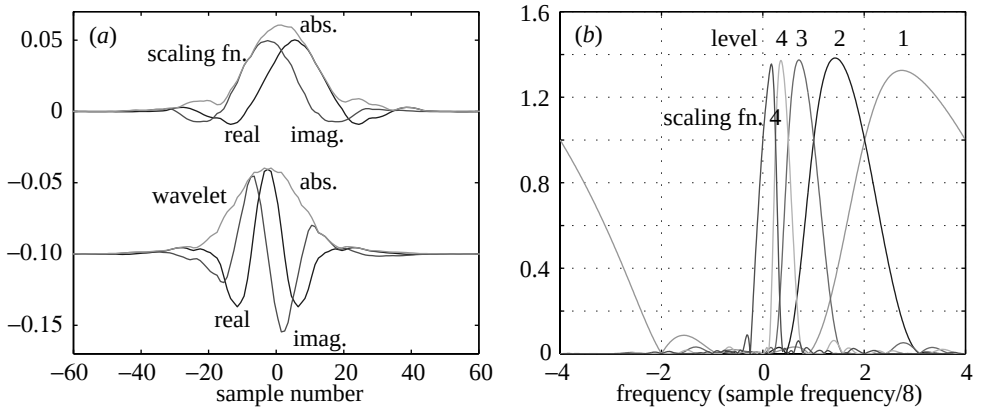

Figure 11. (*a*) Impulse responses at level 4 of the DT CWT scaling function, and wavelet; (*b*) frequency responses of the wavelets at levels 1–4 and of the level 4 scaling function.

where conjugation is given by

$$P^*(z) = \sum_r p_r^* z^{-r},$$

and produces negative frequency passbands. The overlap terms are of the form $Q(z)P^*(W^k z)$ for $k = 1$, 2, and $Q^*(z)P(W^k z)$ for $k = -1$, -2, which all cancel when $B(W^k z)D(z)$ is added to $A(W^k z)C(z)$ in (7.1) to give

$$A(W^k z)C(z) + B(W^k z)D(z) = 2P(W^k z)Q(z) + 2P^*(W^k z)Q^*(z). \tag{7.4}$$

Hence, we now need only design the filters such that the positive frequency complex filter $Q(z)$ does not overlap with shifted versions of the similar filter $P(z)$. This is quite easy since the complex filter bandwidths are only $f_s/2M$, while the shifts are in multiples of f_s/M. The formulations in equation (7.3) show that the highpass filter outputs from trees a and b should be regarded as the *real and imaginary parts of complex processes*.

For the lowpass filters, equation (7.2) implies that the tree b samples should interpolate midway between the tree a samples, effectively doubling the sampling rate, as shown by the interleaved a and b samples at each level in figure 9. This may be achieved by two identical lowpass filters (either odd or even) at level 1, offset by 1 sample delay, and then by pairs of odd and even length filters at further levels to achieve the extra delay difference of $\frac{1}{4}M$ samples, to make the total difference $\frac{1}{2}M$ at each level.

The responses of $A(z)$ and $B(z)$ also need to match, which can be achieved exactly at level 1, but only approximately beyond this. We do this by designing the even-length $H_{00a}(z)$ to give minimum mean squared error in the approximation

$$z^{\pm 2} H_{0a}(z) H_{00a}(z^2) \approx H_{0b}(z) H_{00b}(z^2).$$

Note that $H_{00b}(z) = H_{0b}(z) = zH_{0a}(z)$, so this is just a matrix pseudo-inverse problem. Then the H_{01a} can be designed to form a perfect reconstruction set with H_{00a}, such that the reconstruction filters, G_{00a} and G_{00b}, also match each other closely.

Table 1. *Table of coefficients for the DT CWT analysis filters*
(The reconstruction filters are obtained by negating alternate coefficients and swapping bands.)

odd $H_{\ldots 0}$ 13-tap	odd $H_{\ldots 1}$ 19-tap	even $H_{\ldots 0}$ 12-tap	even $H_{\ldots 1}$ 16-tap
	−0.000 070 6		
	0		−0.000 464 5
−0.001 758 1	0.001 341 9		0.001 334 9
0	−0.001 883 4	−0.005 810 9	0.002 200 6
0.022 265 6	−0.007 156 8	0.016 697 7	−0.013 012 7
−0.046 875 0	0.023 856 0	−0.000 064 1	0.001 536 0
−0.048 242 2	0.055 643 1	−0.083 491 4	0.086 900 8
0.296 875 0	−0.051 688 1	0.091 953 7	0.083 355 2
0.555 468 8	−0.299 757 6	0.480 715 1	−0.488 595 7
0.296 875 0	0.559 430 8	0.480 715 1	0.488 595 7
−0.048 242 2	−0.299 757 6	0.091 953 7	−0.083 355 2
⋮	⋮	⋮	⋮

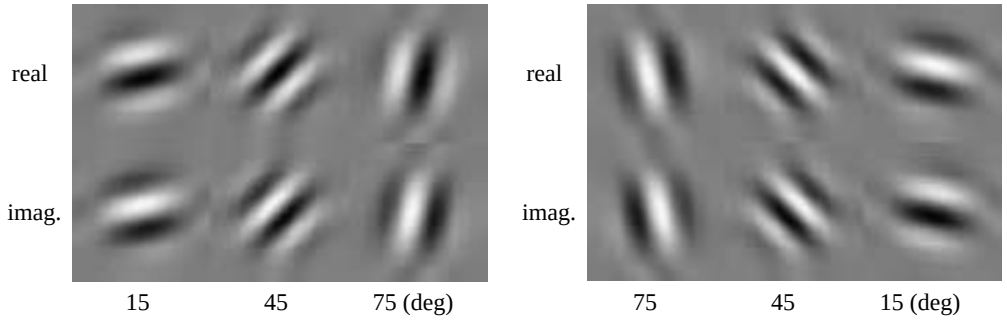

Figure 12. Real and imaginary parts of two-dimensional impulse responses for the six bandpass bands at level 4. (See table 1 for the coefficients for the DT CWT analysis filters.)

Finally, the symmetry of the odd-length highpass filters and the anti-symmetry of the even-length highpass filters produce the required phase relationships between the positive and negative frequency passbands, and equation (7.3) is approximately satisfied too.

These filters can then be used for all subsequent levels of the transform. Good shift invariance (and wavelet smoothness) requires that frequency response sidelobes of the cascaded multi-rate filters should be small. This is achieved if each lowpass filter has a stopband covering $\frac{1}{3}$ to $\frac{2}{3}$ of its sample rate, so as to reject the image frequencies due to subsampling in the next lowpass stage. If the highpass filters then mirror this characteristic, the conditions for no overlap of the shifted bandpass responses in (7.4) are automatically satisfied.

As an example, we selected two linear-phase PR biorthogonal filter sets that meet the above conditions quite well and are also nearly orthogonal. For the odd-length set, we took the (13,19)-tap filters, designed using the transformation of variables method given at the end of § 3, and then designed a (12,16)-tap even-length set to

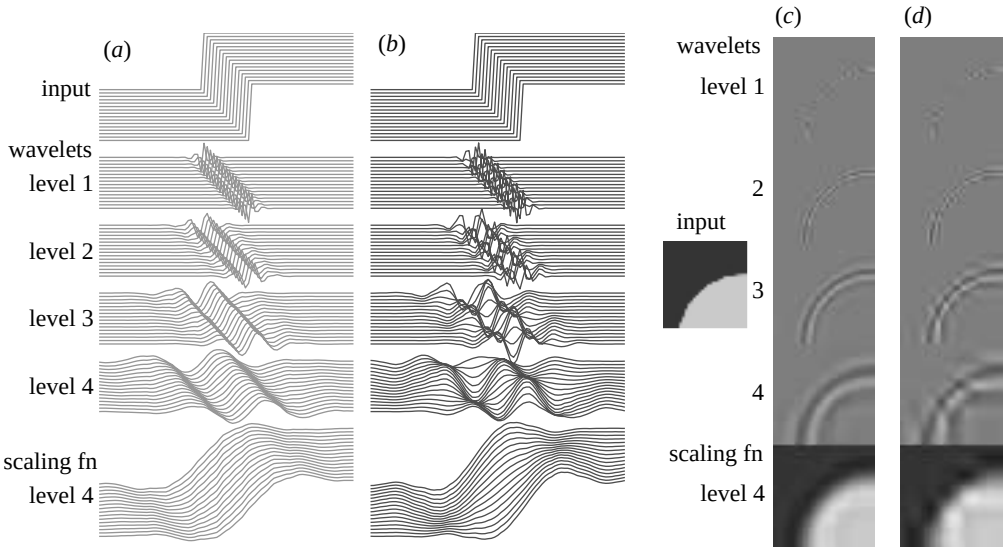

Figure 13. Wavelet and scaling function components at levels 1–4 of 16 shifted step responses using the DT CWT (a) and real DWT (b); and of one quadrant of a 'disc' image using the DT CWT (c) and real DWT (d).

match (as above). Figure 11 shows impulse responses and frequency responses of the reconstruction filter bank; and the analysis filters are very similar. The coefficients are listed in table 1 and the two-dimensional versions of the level 4 impulse responses are shown in figure 12 (note their strong directionality).

Figure 13a demonstrates the shift invariance in one dimension of the DT CWT with these filters. The input of the transform (at the top) is a unit step, shifted to 16 adjacent sampling instants in turn (each shift is displaced down a little to give a waterfall style of display). Below this are shown the components of the output of the *inverse* transform, reconstructed in turn from the wavelet coefficients at each of levels 1–4, and from the scaling function coefficients at level 4. This follows our definition of shift invariance given at the start of this section. Note that summing these components reconstructs the input steps perfectly. Good shift invariance is seen from the fact that each of the reconstructed output components in figure 13a has a shape and amplitude that hardly varies as the input is shifted. This shows that the DT CWT has decomposed the input step into five separate components, which are virtually independent of the location of the step relative to the sub-sampling points of the transform.

For comparison, figure 13b shows the equivalent components if the real DWT is used. The DWT components are much less consistent with shift. The energies of the DT CWT coefficients at any given level vary over the 16 shifts by no more than 1.025:1, whereas the DWT coefficient energies vary by up to 5.45:1: a big difference! Although the five components of the DWT still sum to give perfectly reconstructed steps at each shift position, the decomposition varies considerably with shift. If we are trying to detect features of a signal from its wavelet decomposition, then it is highly confusing to have the sort of unwanted variation with shift shown in figure 13b.

Figure 13c, d shows the equivalent comparison in two dimensions using an image of a circular disc as input. The gradual shift and rotation of the edge of the disc

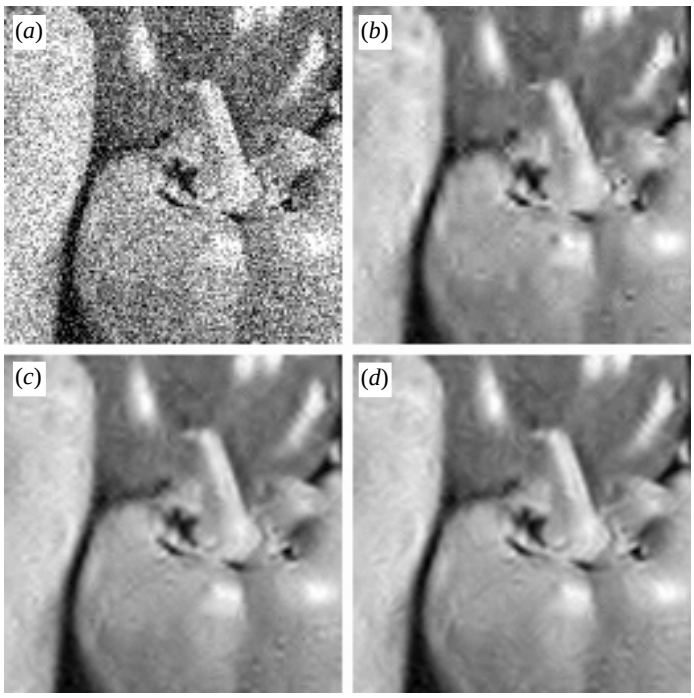

Figure 14. 128 × 128 pel portions of the pepper's image: (*a*) with white Gaussian noise added to give SNR = 3.0 dB; (*b*) denoised with real DWT, SNR = 12.24 dB; (*c*) denoised with undecimated WT, SNR = 13.45 dB; (*d*) denoised with dual-tree CWT, SNR = 13.51 dB.

with respect to the rectangular sub-sampling grids of the transform (not shown) form a good test of shift and rotational dependencies. In these images, all bandpass coefficients at a given wavelet level are retained. In figure 13*c* we see the near-perfect circular arcs, generated by the components at each level for the DT CWT, which show good shift and rotational invariance. Contrast these with the badly distorted arcs for the DWT in figure 13*d*, caused by aliasing.

8. Applications of the complex wavelet transform

The shift invariance and directionality of the CWT may be applied to advantage in many areas of image processing, for example: denoising, restoration, texture modelling, steerable filtering, registration/motion processing, object segmentation, and image classification. We have space for only one example here.

In figure 14 we show an example of denoising. Image (*d*) is the result of denoising image (*a*) using the DT CWT and a simple soft thresholding method that suppresses all complex wavelet coefficients x of low amplitude with a raised cosine gain law: $g(x) = \frac{1}{2}(1-\cos\{\pi|x|/T\})$ for $|x| < T$, and $g(x) = 1$ elsewhere. For comparison, we show images (*b*) and (*c*), which were obtained using the same soft thresholding method with the real DWT in its decimated and undecimated forms, respectively. (*b*) shows significantly worse artefacts than (*d*), while (*c*) is very similar to (*d*) but requires about five times as much computation. In all cases, the thresholds T were selected so as to get minimum mean-squared error from the original (clean) image.

In practice, more complicated thresholding methods may be used, such as in Malfait & Roose (1997), which uses Markov random fields in conjunction with an undecimated WT. It is likely that, by replacing the undecimated WT with the CWT, the effectiveness of the MRFs at coarser wavelet levels can be improved, owing to the more appropriate sampling rates of the CWT.

In conclusion, we are now investigating a number of applications of the CWT and are finding that it may have many uses as a multiresolution front end for processing images and other multidimensional signals.

References

Belzer, B., Lina, J.-M. & Villasenor, J. 1995 Complex, linear-phase filters for efficient image coding. *IEEE Trans. Signal Proc.* **43**, 2425–2427.

Daubechies, I. 1990 The wavelet transform, time-frequency localisation and signal analysis. *IEEE Trans. Informat. Theory* **36**, 961–1005.

Kingsbury, N. G. 1998*a* The dual-tree complex wavelet transform: a new technique for shift invariance and directional filters. In *Proc. 8th IEEE DSP Workshop, Bryce Canyon, UT, USA*, paper no. 86.

Kingsbury, N. G. 1998*b* The dual-tree complex wavelet transform: a new efficient tool for image restoration and enhancement. In *Proc. EUSIPCO 98, Rhodes, Greece*, pp. 319–322.

Kovacevic, J. & Daubechies, I. (eds) 1996 Special issue on wavelets. *Proc. IEEE* **84**, 507–685.

Lawton, W. 1993 Applications of complex valued wavelet transforms to sub-band decomposition. *IEEE Trans. Signal Proc.* **41**, 3566–3568.

Magarey, J. F. A. & Kingsbury, N. G. 1998 Motion estimation using a complex-valued wavelet transform. *IEEE Trans. Signal Proc.* **46**, 1069–1084.

Malfait, M. & Roose, D. 1997 Wavelet-based image denoising using a Markov random field a priori model. *IEEE Trans. Image Proc.* **6**, 549–565.

Mallat, S. G. 1989 A theory for multiresolution signal decomposition: the wavelet representation. *IEEE Trans. PAMI* **11**, 674–693.

Rioul, O. & Vetterli, M. 1991 Wavelets and signal processing. *IEEE Signal Processing Mag.* **8**, 14–38.

Simoncelli, E. P., Freeman, W. T., Adelson, E. H. & Heeger, D. J. 1992 Shiftable multiscale transforms. *IEEE Trans. Informat. Theory* **38**, 587–607.

Tay, D. B. H. & Kingsbury, N. G. 1993 Flexible design of multidimensional perfect reconstruction FIR 2-band filters using transformations of variables. *IEEE Trans. Image Proc.* **2**, 466–480.

Vetterli, M. & Kovacevic, J. 1995 *Wavelets and sub-band coding*. Englewood Cliffs, NJ: Prentice-Hall.