Chapter 4

Beyond wavelets

> When you have only one way of expressing yourself, you have limits that you don't appreciate. When you get a new way to express yourself, it teaches you that there could be a third or a fourth way. It opens up your eyes to a much broader universe.
>
> *David Donoho*[1]

In this chapter we shall explore some additional topics that extend the basic ideas of wavelet analysis introduced previously. We first describe the theory of *wavelet packet transforms,* which sometimes provide superior performance beyond that provided by wavelet transforms. Then we discuss *continuous wavelet transforms* which are particularly useful for tackling problems in signal recognition, and for performing finely detailed examinations of the structure of signals.

4.1 Wavelet packet transforms

A *wavelet packet transform* is a simple generalization of a wavelet transform. In this section we briefly discuss the definition of wavelet packet transforms, and in the next section examine some examples illustrating their applications.

All wavelet packet transforms are calculated in a similar way. Therefore we shall concentrate initially on the Haar wavelet packet transform, which is the easiest to describe. The Haar wavelet packet transform is usually referred to as the *Walsh transform.* A Walsh transform is calculated by

[1] Donoho's quote is from [BUR].

performing a 1-level Haar transform *on all subsignals,* both trends *and* fluctuations.

For example, consider the signal $\mathbf{f}$ defined by

$$\mathbf{f} = (2, 4, 6, 8, 10, 12, 14, 16). \tag{4.1}$$

A 1-level Haar transform and a 1-level Walsh transform of $\mathbf{f}$ are identical, producing the following signal:

$$(3\sqrt{2}, 7\sqrt{2}, 11\sqrt{2}, 15\sqrt{2} \,|\, -\sqrt{2}, -\sqrt{2}, -\sqrt{2}, -\sqrt{2}). \tag{4.2}$$

A 2-level Walsh transform is calculated by performing 1-level Haar transforms on both the trend and the fluctuation subsignals, as follows:

$$\begin{aligned} (3\sqrt{2}, 7\sqrt{2}, 11\sqrt{2}, 15\sqrt{2}) &\overset{\mathbf{H}_1}{\longmapsto} (10, 26 \,|\, -4, -4) \\ (-\sqrt{2}, -\sqrt{2}, -\sqrt{2}, -\sqrt{2}) &\overset{\mathbf{H}_1}{\longmapsto} (-2, -2 \,|\, 0, 0). \end{aligned}$$

Hence the 2-level Walsh transform of the signal $\mathbf{f}$ is the following signal:

$$(10, 26 \,|\, -4, -4 \,|\, -2, -2 \,|\, 0, 0). \qquad \text{[2-level Walsh]} \tag{4.3}$$

It is interesting to compare this 2-level Walsh transform with the 2-level Haar transform of the signal $\mathbf{f}$. The 2-level Haar transform of $\mathbf{f}$ is the following signal:

$$(10, 26 \,|\, -4, -4 \,|\, -\sqrt{2}, -\sqrt{2}, -\sqrt{2}, -\sqrt{2}). \qquad \text{[2-level Haar]} \tag{4.4}$$

Comparing this Haar transform with the Walsh transform in (4.3), we see that the Walsh transform is slightly more compressed in terms of energy, since the last two values of the Walsh transform are zeros. We could, for example, achieve 25% compression of signal $\mathbf{f}$ by discarding the two zeros from its 2-level Walsh transform, but we could not discard any zeros from its 2-level Haar transform. Another advantage of the 2-level Walsh transform is that it is more likely that *all* of its non-zero values would stand out from a random noise background, because these values have larger magnitudes than the values of the 2-level Haar transform.

A 3-level Walsh transform is performed by calculating 1-level Haar transforms on each of the four subsignals that make up the 2-level Walsh transform. For example, applying 1-level Haar transforms to each of the four subsignals of the 2-level Walsh transform in (4.3), we obtain

$$\begin{aligned} (10, 26) &\overset{\mathbf{H}_1}{\longmapsto} (18\sqrt{2} \,|\, -8\sqrt{2}), \\ (-4, -4) &\overset{\mathbf{H}_1}{\longmapsto} (-4\sqrt{2} \,|\, 0), \\ (-2, -2) &\overset{\mathbf{H}_1}{\longmapsto} (-2\sqrt{2} \,|\, 0), \\ (0, 0) &\overset{\mathbf{H}_1}{\longmapsto} (0 \,|\, 0). \end{aligned}$$

Hence the 3-level Walsh transform of the signal $\mathbf{f}$ in (4.1) is

$$(18\sqrt{2}\,|\,-8\sqrt{2}\,|\,-4\sqrt{2}\,|\,0\,|\,-2\sqrt{2}\,|\,0\,|\,0\,|\,0). \qquad \text{[3-level Walsh]} \quad (4.5)$$

Here, at the third level, the contrast between the Haar and Walsh transforms is even sharper than at the second level. The 3-level Haar transform of this signal is

$$(18\sqrt{2}\,|\,-8\sqrt{2}\,|\,-4,-4\,|\,-\sqrt{2},-\sqrt{2},-\sqrt{2},-\sqrt{2}). \qquad \text{[3-level Haar]} \quad (4.6)$$

Comparing the transforms in (4.5) and (4.6) we can see, at least for this particular signal $\mathbf{f}$, that the 3-level Walsh transform achieves a more compact redistribution of the energy of the signal than the Haar transform.

In general, a wavelet packet transform is performed by calculating a particular 1-level wavelet transform for each of the subsignals of the preceding level. For instance, a 3-level Daub4 wavelet packet transform would be calculated in the same way as a 3-level Walsh transform, but with 1-level Daub4 wavelet transforms being used instead of 1-level Haar transforms. Because all of the 1-level wavelet transforms that we have discussed enjoy the Conservation of Energy property and have inverses, it follows that all of their wavelet packet transforms also enjoy the Conservation of Energy property and have inverses. What this implies is that our discussions of compression and denoising of signals in Chapters 1 and 2 apply, essentially without change, to wavelet packet transforms. In particular, the threshold method is still the basic method for compression and noise removal with wavelet packet transforms.

In two dimensions, a wavelet packet transform is performed by adopting the same approach that we used in one dimension. First, a 1-level wavelet transform is performed on the 2D image. Then, to compute a 2-level wavelet packet transform, this 1-level wavelet transform is applied to each of the four subimages—$\mathbf{a}^1$, $\mathbf{d}^1$, $\mathbf{h}^1$, and $\mathbf{v}^1$—from the first level. This produces 16 subimages that constitute the 2-level wavelet packet transform of the image. To compute a 3-level wavelet packet transform, the 1-level wavelet transform is applied to each of these 16 subimages, producing 64 subimages. This process continues in an obvious manner for higher level wavelet packet transforms.

Because of the great similarity between wavelet transforms and wavelet packet transforms, we shall now end our discussion of the mathematics of these transforms, and turn to a discussion of a few examples of how they can be applied.

4.2 Applications of wavelet packet transforms

In this section we shall discuss a few examples of applying wavelet packet transforms to audio and image compression. While wavelet packet transforms can be used for other purposes, such as noise removal, because of space limitations we shall limit our discussion to the arena of compression.

For our first example, we shall use a Coif30 wavelet packet transform to compress the audio signal *greasy,* considered previously in Section 2.5. In that section we found that a 4-level Coif30 wavelet transform—with trend values quantized at 8 bpp and fluctuations quantized at 6 bpp, and with separate entropies computed for all subsignals—achieved a compression of *greasy* requiring an estimated $11,305$ bits. That is, this compression required an estimated 0.69 bpp (instead of 8 bpp in the original). However, if we use a 4-level Coif18 wavelet packet transform and quantize in the same way, then the estimated number of bits is $10,158$, i.e., 0.62 bpp. This represents a slight improvement over the wavelet transform.

In several respects—in bpp, in RMS Error, and in total number of significant values—the wavelet packet compression of *greasy* is nearly as good as or slightly better than the wavelet transform compression. See Table 4.1. Whether these slight improvements are worth the extra computations needed to calculate with the wavelet packet transform is certainly open to question. Our next example, from the field of image compression, is more definitive.

Table 4.1 Wavelet and wavelet packet compressions of *greasy*

Transform	*Sig. values*	*Bpp*	*RMS Error*
wavelet	3685	0.69	0.839
w. packet	3072	0.62	0.868

Our second example deals with an image compression. In Figure 4.1(a) we show an image of a woman, which we shall refer to as *Barb.* If a 4-level Coif12 wavelet transform is applied to this image—with the trend quantized at 8 bpp, the fluctuations quantized at 6 bpp, and separate entropies computed for each subimage—then an estimated 0.67 bpp are needed to encode the compressed image. The compressed image is virtually indistinguishable from the original image; so we will not display it. There are some noticeable differences in details at sufficiently high magnification, as can be seen by comparing Figures 4.1(b) and 4.1(c).

If we compute a 4-level Coif12 wavelet packet transform, using the same quantizations as for the wavelet transform, then the compressed image re-

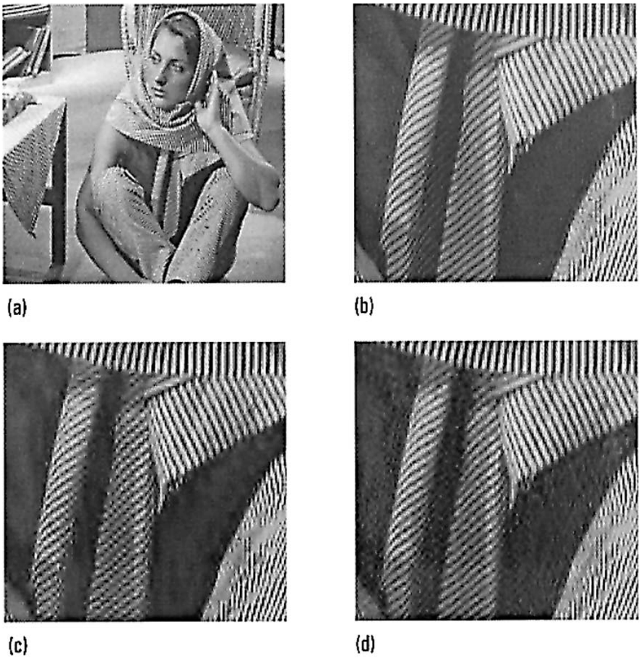

FIGURE 4.1
(a) *Barb* image. (b) Detail of original. (c) Detail of wavelet compressed image. (d) Detail of wavelet packet compressed image.

quires an estimated 0.51 bpp for encoding. This represents a 24% improvement over the wavelet transform compression. The wavelet packet transform performs significantly better in several respects, as summarized in Table 4.2.

Table 4.2 Wavelet and wavelet packet compressions of *Barb*

Transform	*Sig. values*	*Bpp*	*Rel. 2-norm error*
wavelet	28370	0.67	.0486
w. packet	21755	0.51	.0462

There is also improved accuracy of detail in the wavelet packet compression, as shown in Figure 4.1(d). In particular, the two sets of diagonal stripes aligned along the two vertical folds of *Barb*'s scarf are better preserved in Figure 4.1(d) than in Figure 4.1(c).

There is insufficient space to pursue a thorough explanation of why the wavelet packet transform performs better in this example. Nevertheless,

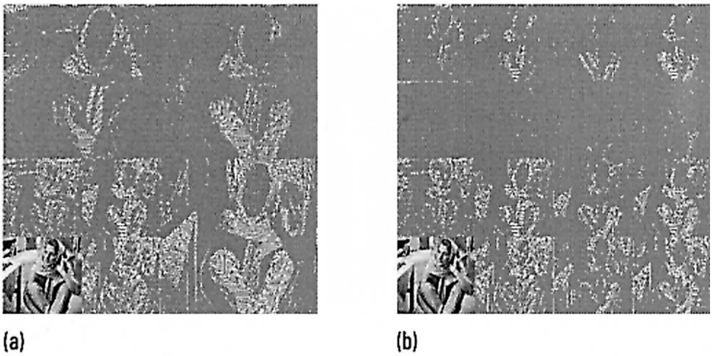

FIGURE 4.2
(a) 2-level Coif12 wavelet transform of *Barb*. (b) 2-level Coif12 wavelet packet transform of *Barb*.

we can gain some understanding of the situation by comparing a Coif12 wavelet transform of *Barb* with a Coif12 wavelet packet transform. In Figure 4.2(a) we show a 2-level Coif12 wavelet transform of *Barb*. Notice that the 1-level fluctuations—$\mathbf{h}^1$, $\mathbf{d}^1$, and $\mathbf{v}^1$—all reveal considerable structure. It is easy to discern ghostly versions of the original *Barb* image within each of these fluctuations. The presence of these ghostly subimages, along with the trend subimage, suggests the possibility of performing wavelet transform compression on all of these subimages. In other words, we compute another 1-level wavelet transform on each of the four subimages, which produces the 2-level wavelet packet transform shown in Figure 4.2(b). Notice that, in the regions corresponding to the horizontal fluctuation $\mathbf{h}^1$ and the diagonal fluctuation $\mathbf{d}^1$ in the wavelet transform, there is a considerable reduction in the number of significant values in the wavelet packet transform. This reduction enables a greater compression of the *Barb* image. For similar reasons, the 4-level wavelet packet transform exhibits a more compact distribution of significant coefficients, hence a greater compression, than the 4-level wavelet transform.

For our last example, we consider a compression of a fingerprint image. In the previous example, we saw that a 4-level wavelet packet transform performed better on the *Barb* image than a 4-level wavelet transform. Consequently, we are led to try a similar compression of Fingerprint 1 [see Figure 2.18(a)]. Instead of the 4-level Coif18 wavelet transform used on Fingerprint 1 in Section 2.9, here we shall try a 4-level Coif18 wavelet packet transform. Using the same quantizations as before—9 bpp for the trend and 6 bpp for the fluctuations—we obtain an estimated 0.49 bpp. That represents a 36% improvement over the 0.77 bpp estimated for the wavelet compression discussed in Section 2.9. In Table 4.3 we show a comparison of these two compressions of Fingerprint 1. Although the wavelet packet transform compression does not produce as small a relative 2-norm error as the wavelet transform compression, nevertheless, a value of 0.043 is still better than the 0.05 rule of thumb value for an acceptable approximation. In fact, the compressed version of Fingerprint 1 produced by the wavelet packet transform is virtually indistinguishable from the original (hence we do not include a figure of it).[2] Taking into account the other data from Table 4.3—the number of significant transform values and the number of bpps—it is clear that the wavelet packet compression of Fingerprint 1 is superior to the wavelet compression.

Table 4.3 Two compressions of Fingerprint 1

Transform	*Sig. values*	*Bpp*	*Rel. 2-norm error*
wavelet	33330	0.77	0.035
w. packet	20796	0.49	0.043

This last example gives us some further insight into the standard method adopted by the FBI for fingerprint compression, the WSQ method. The WSQ method achieves *at least* 15:1 compression, without noticeable loss of detail, on all fingerprint images. It achieves such a remarkable result by applying a hybrid wavelet transform compression that combines features of both wavelet and wavelet packet transforms. In this hybrid transform, not every subimage is subjected to a further 1-level wavelet transform, but a large percentage of subimages are further transformed. For an example illustrating why this might be advantageous, consider the two transforms of the *Barb* image in Figure 4.2. Notice that the vertical fluctuation $\mathbf{v}^1$ in the lower right quadrant of Figure 4.2(a) does not seem to be significantly compressed by applying another 1-level wavelet transform. Therefore, some advantage in compression might be obtained by *not* applying a 1-level wavelet transform to this subimage, while applying it to the other subimages. An exact description of the hybrid approach used by the WSQ method can be found in the articles listed in the Notes and references section for this chapter.

4.3 Continuous wavelet transforms

In this section and the next we shall describe the concept of a *continuous wavelet transform* (CWT), and how it can be approximated in a discrete

[2] You can, however, find the compressed image at the FAWAV website.

form using a computer. We begin our discussion by describing one type of CWT, known as the *Mexican hat* CWT, which has been used extensively in seismic analysis. In the next section we turn to a second type of CWT, the Gabor CWT, which has many applications to analyzing audio signals. Although we do not have space for a thorough treatment of CWTs, we can nevertheless introduce some of the essential ideas.

The notion of a CWT is founded upon many of the concepts that we introduced in our discussion of discrete wavelet analysis in Chapters 1 through 3, especially the ideas connected with discrete correlations and frequency analysis. A CWT provides a very redundant, but also very finely detailed, description of a signal in terms of both time and frequency. CWTs are particularly helpful in tackling problems involving signal identification and detection of hidden transients (hard to detect, short-lived elements of a signal).

To define a CWT we begin with a given function $\Psi(x)$, which is called the *analyzing wavelet.* For instance, if we define $\Psi(x)$ by

$$\Psi(x) = 2\pi w^{-1/2} \left[1 - 2\pi (x/w)^2 \right] e^{-\pi (x/w)^2} , \quad w = 1/16, \tag{4.7}$$

then this analyzing wavelet is called a *Mexican hat wavelet,* with *width parameter* $w = 1/16$. See Figure 4.3(a).

It is possible to choose other values for w besides $1/16$, but this one example should suffice. By graphing the Mexican hat wavelet using different values of w, it is easy to see why w is called a width parameter. The larger the value of w, the more the energy of $\Psi(x)$ is spread out over a larger region of the x-axis.

The Mexican hat wavelet is not the only kind of analyzing wavelet. In the next section, we shall consider the Gabor wavelet, which is very advantageous for analyzing recordings of speech or music. We begin in this section with the Mexican hat wavelet because it is somewhat easier to explain the concept of a CWT using this wavelet.

Given an analyzing wavelet $\Psi(x)$, then a CWT of a discrete signal $\mathbf{f}$ is defined by computing several correlations of this signal with discrete samplings of the functions $\Psi_s(x)$ defined by

$$\Psi_s(x) = \frac{1}{\sqrt{s}} \Psi \left(\frac{x}{s} \right) , \quad s > 0. \tag{4.8}$$

The parameter s is called a *scale parameter.* If we sample each signal $\Psi_s(x)$ at discrete time values $t_1, t_2, \dots, t_N$, where N is the length of $\mathbf{f}$, then we generate the discrete signals $\mathbf{g}_s$ defined by

$$\mathbf{g}_s = (\, \Psi_s(t_1), \Psi_s(t_2), \dots, \Psi_s(t_N)\,) .$$

A CWT of $\mathbf{f}$ then consists of a collection of discrete correlations $(\mathbf{f} : \mathbf{g}_s)$ over a finite collection of values of s. A common choice for these values is

$$s = 2^{-k/M} , \quad k = 0, 1, 2, \dots, I \cdot M$$

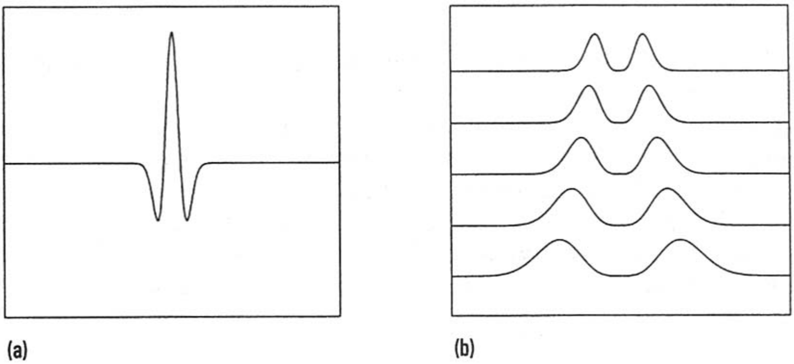

FIGURE 4.3
(a) The Mexican hat wavelet, $w = 1/16$. (b) DFTs of discrete samplings of this wavelet for scales $s = 2^{-k/6}$, from $k = 0$ at the top, then $k = 2$, then $k = 4$, then $k = 6$, down to $k = 8$ at the bottom.

where the positive integer I is called the number of *octaves* and the positive integer M is called the number of *voices* per octave. For example, 8 octaves and 16 voices per octave is the default choice in FAWAV. Another popular choice is 6 octaves and 12 voices per octave. This latter choice of scales corresponds—based on the relationship between scales and frequencies that we describe below—to the scale of notes on a piano (also known as the *well-tempered scale*).

At this point the reader may well be wondering what the point of all this is. One purpose of computing all these correlations that make up a CWT is that *a very finely detailed frequency analysis* of a signal can be carried out by making a judicious choice of the width parameter w and the number of octaves and voices. To see this, we observe that Formula (3.40) tells us that the DFTs of the correlations $(\mathbf{f}\colon \mathbf{g}$

$$(\mathbf{f}\colon \mathbf{g}_s) \overset{\mathcal{F}}{\longmapsto} \mathcal{F}\mathbf{f}\,\overline{\mathcal{F}\mathbf{g}_s}. \tag{4.9}$$

For a Mexican hat wavelet, $\mathcal{F}\mathbf{g}_s$ is real-valued; hence $\overline{\mathcal{F}\mathbf{g}_s} = \mathcal{F}\mathbf{g}_s$. Therefore Equation (4.9) becomes

$$(\mathbf{f}\colon \mathbf{g}_s) \overset{\mathcal{F}}{\longmapsto} \mathcal{F}\mathbf{f}\,\mathcal{F}\mathbf{g}_s. \tag{4.10}$$

Formula (4.10) is the basis for a very finely detailed frequency decomposition of a discrete signal $\mathbf{f}$. For example, in Figure 4.3(b) we show graphs of the DFTs $\mathcal{F}\mathbf{g}_s$ for the scale values $s = 2^{-k/6}$, with $k = 0$, 2, 4, 6, and 8. These graphs show that when these DFTs are multiplied with the DFT of $\mathbf{f}$, they provide a decomposition of $\mathcal{F}\mathbf{f}$ into a succession of finely resolved frequency bands. It should be noted that these successive bands overlap each other, and thus provide a very redundant decomposition of the DFT of $\mathbf{f}$. Notice also that the bands containing higher frequencies correspond to smaller scale values; *there is a reciprocal relationship between scale values and frequency values.*

A couple of examples should help to clarify these points. The first example we shall consider is a test case designed to illustrate the connection between a CWT and the frequencies in a signal. The second example is an illustration of how a CWT can be used for analyzing an ECG signal.

For our first example, we shall analyze a discrete signal $\mathbf{f}$, obtained from 2048 equally spaced samples of the following analog signal:

$$\begin{aligned} &\sin(40\pi x)e^{-100\pi(x-.2)^2} \\ &+ \left[\sin(40\pi x) + 2\cos(160\pi x)\right] e^{-50\pi(x-.5)^2} \\ &+ 2\sin(160\pi x)e^{-100\pi(x-.8)^2} \end{aligned} \tag{4.11}$$

over the interval $0 \leq x \leq 2$. See the top of Figure 4.4(a).

The signal in (4.11) consists of three terms. The first term contains a sine factor, $\sin(40\pi x)$, of frequency 20. Its other factor, $e^{-100\pi(x-.2)^2}$, serves as a damping factor which limits the energy of this term to a small interval centered on $x = 0.2$. This first term appears most prominently on the left-third of the graph at the top of Figure 4.4(a). Likewise, the third term contains a sine factor, $2\sin(160\pi x)$, of frequency 80, and this term appears most prominently on the right-third of the signal's graph. Notice that this frequency of 80 is four times as large as the first frequency of 20. Finally, the middle term

$$\left[\sin(40\pi x) + 2\cos(160\pi x)\right] e^{-50\pi(x-.5)^2}$$

has a factor containing both of these two frequencies, and can be observed most prominently within the middle of the signal's graph.

The CWT, also known as a *scalogram,* for this signal is shown at the bottom of Figure 4.4(a). The analyzing wavelet used to produce this CWT was a Mexican hat wavelet of width 1/16, with scales ranging over 8 octaves and 16 voices. The labels on the right side of the figure indicate *reciprocals* of the scales used. Because of the reciprocal relationship between scale and frequency noted above, this reciprocal-scale axis can also be viewed as a frequency axis. Notice that the four most prominent portions of this scalogram are aligned directly below the three most prominent parts of the signal. Of equal importance is the fact that these four portions of the scalogram are centered on two reciprocal-scales, $1/s \approx 2^{2.2}$ and $1/s \approx 2^{4.2}$. The second reciprocal scale is four times larger than the first reciprocal scale, just as the frequency 80 is four times larger than the frequency 20. Bearing this fact in mind, and recalling the alignment of the prominent regions of the scalogram with the three parts of the signal, we can see that the CWT provides us with a *time-frequency portrait* of the signal.

Although we have shown that it is possible to correctly interpret the meaning of this scalogram; nevertheless, we can produce a much simpler

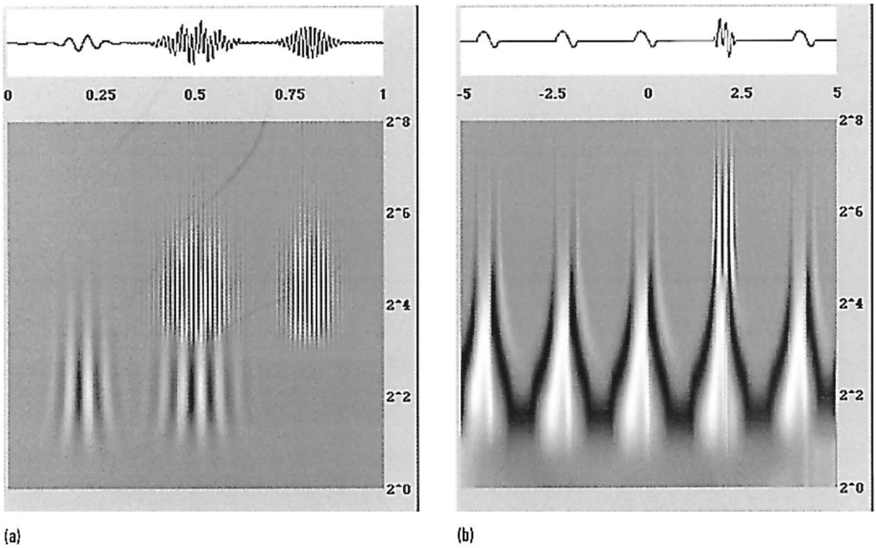

FIGURE 4.4
(a) Mexican hat CWT (scalogram) of a test signal with two main frequencies. (b) Mexican hat scalogram of simulated ECG signal. Whiter colors represent positive values, blacker values represent negative values, and the grey background represents zero values.

and more easily interpretable scalogram for this test signal using a Gabor analyzing wavelet. See Figure 4.5(a). We shall discuss this Gabor scalogram in the next section.

Our second example makes use of a Mexican hat CWT for analyzing a signal containing several transient bursts, a simulated ECG signal that we first considered in Section 3.4. See the top of Figure 4.4(b). The bottom of Figure 4.4(b) is a scalogram of this signal using a Mexican hat wavelet of width 2, over a range of 8 octaves and 16 voices. This scalogram shows how a Mexican hat wavelet can be used for detecting the onset and demise of each heartbeat. In particular, the aberrant, fourth heartbeat is singled out from the others by the longer vertical ridges extending upwards to the highest frequencies (at the eighth octave). Although this example is only a simulation, it does show the ease with which the Mexican hat CWT detects the presence of short-lived parts of a signal. Similar identifications of transient bursts are needed in seismology for the detection of earthquake tremors. Consequently, Mexican hat wavelets are widely used in seismology.

4.4 Gabor wavelets and speech analysis

In this section we describe Gabor wavelets, which are similar to the Mexican hat wavelets examined in the previous section, but provide a more powerful tool for analyzing speech and music. We shall first go over their definition, and then illustrate their use by examining a couple of examples.

A *Gabor wavelet,* with width parameter w and frequency parameter ν, is the following analyzing wavelet:

$$\Psi(x) = w^{-1/2} e^{-\pi(x/w)^2} \, e^{i2\pi\nu x/w}. \tag{4.12}$$

This wavelet is complex valued. Its real part $\Psi_{\rm R}(x)$ and imaginary part $\Psi_{\rm I}(x)$ are

$$\Psi_{\rm R}(x) = w^{-1/2} e^{-\pi(x/w)^2} \, \cos(2\pi\nu x/w), \tag{4.13a}$$

$$\Psi_{\rm I}(x) = w^{-1/2} e^{-\pi(x/w)^2} \, \sin(2\pi\nu x/w). \tag{4.13b}$$

The width parameter w plays the same role as for the Mexican hat wavelet; it controls the width of the region over which most of the energy of $\Psi(x)$ is concentrated. The frequency parameter ν provides the Gabor wavelet with an extra parameter for analysis.

One advantage that Gabor wavelets have when analyzing sound signals is that they contain factors of cosines and sines [see (4.13a) and (4.13b)]. These cosine and sine factors allow the Gabor wavelets to create easily interpretable scalograms of those signals which are combinations of cosines and sines—the most common instances of such signals are recorded music and speech. We shall see this in a moment, but first we need to say a little more about the CWT defined by a Gabor analyzing wavelet.

Because a Gabor wavelet is complex valued, it produces a complex-valued CWT. For many signals, it is often sufficient to just examine the magnitudes[3] of the Gabor CWT values. In particular, this is the case with the signals analyzed in the following two examples.

For our first example, we use a Gabor wavelet with width 1 and frequency 2 for analyzing the signal in (4.11). The graph of this signal is shown at the top of Figure 4.5(a). As we discussed in the previous section, this signal consists of three portions with associated frequencies of 20 and 80. The magnitudes for a Gabor scalogram of this signal, using 8 octaves and 16 voices, are graphed at the bottom of Figure 4.5(a). We see that this *magnitude-scalogram* consists of essentially just four prominent, and clearly

[3] Recall that a complex number z has a *magnitude* $|z|$ equal to its distance from the origin in the complex plane.

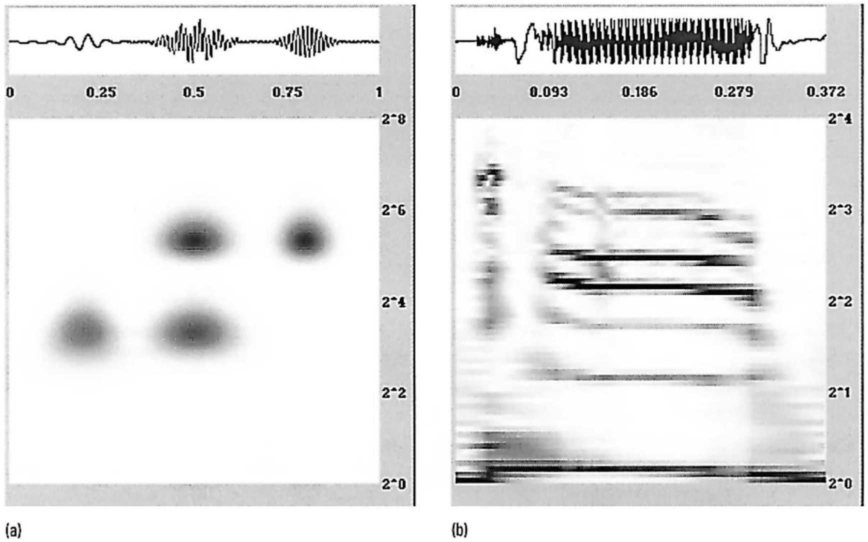

FIGURE 4.5
(a) Magnitudes of Gabor scalogram of test signal. (b) Magnitudes of Gabor scalogram of *call* sound. Darker regions denote larger magnitudes; lighter regions denote smaller magnitudes.

separated, spots aligned directly below the three most prominent portions of the signal. These four spots are centered on the two reciprocal-scale values of $2^{3.38}$ and $2^{5.38}$, which are in the same ratio as the two frequencies 20 and 80.

It is interesting to compare Figures 4.4(a) and 4.5(a). The simplicity of Figure 4.5(a) makes it much easier to interpret. The reason that the Gabor CWT is so clean and simple is because, for the proper choices of width w and frequency ν, the test signal in (4.11) consists of terms that are identical in form to one of the functions in (4.13a) or (4.13b). Therefore, when a scale value s produces a function $\Phi_{\mathrm{R}}(x/s)/\sqrt{s}$, or a function $\Phi_{\mathrm{I}}(x/s)/\sqrt{s}$, having a form similar to one of the terms in (4.11), then the correlation $(\mathbf{f} : \mathbf{g}_s)$ in the CWT will have some high-magnitude values.

This first example might appear to be rather limited in scope. After all, how many signals encountered in the real world are so nicely put together as this test signal? Our next example, however, shows that a Gabor CWT performs equally well in analyzing a real signal: a speech signal.

In Figure 4.5(b) we show a Gabor magnitude-scalogram of a recording of the author saying the word *call*. The recorded signal, which is shown at the top of the figure, consist of two main portions. These two portions correspond to the two sounds, *ca* and *ll*, that form the word *call*. The *ca* portion occupies a narrow area on the far left side of the *call* signal's graph, while the *ll* portion occupies a much larger area consisting of the middle half of the *call* signal's graph.

To analyze the *call* signal, we used a Gabor wavelet of width 1/8 and frequency 16, with scales ranging over 4 octaves and 16 voices. The resulting magnitude-scalogram is composed of two main regions lying directly underneath the two portions of the *call* signal. The largest region is a collection of several horizontal bands lying below the *ll* portion. The smaller region is a narrow, vertical segment consisting of several dark spots aligned directly underneath the *ca* portion. We shall now examine these two regions of the magnitude-scalogram, and relate their structure to the two portions of the *call* signal.

Let's begin with the larger region consisting of seven horizontal bands lying directly below the *ll* portion. These horizontal bands are centered on the following approximate reciprocal-scale values:

$$2^{0.17},\ 2^{1.17},\ 2^{1.7},\ 2^{2.17},\ 2^{2.5},\ 2^{2.97},\ 2^{3.17}. \tag{4.14}$$

If we divide each of these values by the smallest one, $2^{0.17}$, we get the following approximate ratios:

$$1,\ 2,\ 3,\ 4,\ 5,\ 7,\ 8. \tag{4.15}$$

Since reciprocal-scale values correspond to frequencies, we can see that these bands correspond to frequencies on a harmonic (musical) scale. In fact, in Figure 4.6(b) we show a graph of the spectrum[4] of a sound clip of the *ll* portion of the *call* signal. This spectrum shows that the frequencies of peak energy in the *ll* portion have the following approximate values:

$$140,\ 280,\ 420,\ 560,\ 700,\ 980,\ 1120. \tag{4.16}$$

Notice that these frequencies have the same ratios to the lowest frequency of 140 as the ratios in (4.15). There is even a missing frequency of $6 \times 140 = 840$, corresponding to a missing horizontal band in the magnitude-scalogram that appears to be centered along the reciprocal-scale $2^{2.7}$ (a small part of this missing band is visible below the right edge of the *ll* portion). In fact, the reciprocal-scale $2^{2.7}$ is about 6 times the lowest value of $2^{0.17}$.

This region illustrates an important property of many portions of speech signals, the property of *frequency banding*. These frequency bands are called

[4] The spectrum of a signal was discussed in Section 3.2.

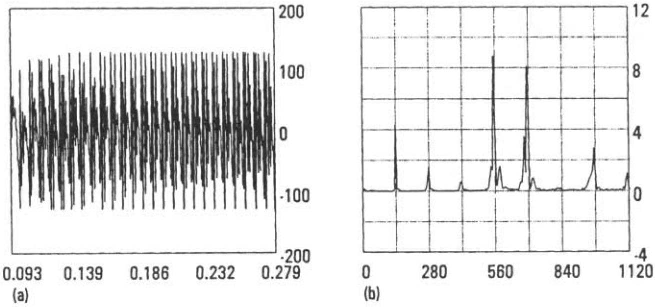

FIGURE 4.6
(a) A portion of the *ll* sound in the *call* signal; the horizontal axis is the time axis. (b) Spectrum of the signal in (a); the horizontal axis is the frequency axis.

formants in linguistics. All speakers, whether they are native speakers of English or not, produce a sequence of such frequency bands for the *ll* portion of *call.* For some speakers, the bands are horizontal, while for other speakers the bands are curved. The *ll* sound is a fundamental unit of English speech, called a *phoneme.*

The second region of the magnitude-scalogram lies below the *ca* portion. The *ca* sound is distinguished clearly from the *ll* portion by its lack of formants. From the magnitude-scalogram, we see that the *ca* portion is composed of a much more widely dispersed, almost continuous, range of frequencies without any significant banding.

This last example shows what a powerful tool the Gabor CWT provides for analyzing a speech signal. We were able to use it to clearly distinguish the two portions in the *call* sound, to understand the formant structure of the *ll* portion, and to determine that the *ca* portion lacks a formant structure.

Another application of these Gabor scalograms is that, when applied to recordings of different people saying *call*, they produce visibly different scalograms. These scalograms function as a kind of “fingerprint” for identifying different speakers. Furthermore, the ribboned structure of formants for the *ll* portion is displayed for all speakers, although they trace out different curves for different speakers. For the reader who wishes to verify these statements, we have included several recordings of different speakers saying the word *call* at the FAWAV website.

4.5 Notes and references

The best introductory material on wavelet packet transforms can be found in [WI1] and [CW1]. There is also a good discussion in [CW2]. A very thorough treatment of the subject is given in [WI2]. The relation between wavelet packet transforms and the WSQ method is described in [BBH].

Rigorous expositions of the complete theory of CWTs can be found in [DAU] and [LMR]. A more complete treatment of the discrete version described in this primer is given in [MAL]. For a discussion of the uses of the CWT for analysis of ECGs, see [STC]. Applying Gabor CWTs to the detection of engine malfunctions in Japanese automobiles is described in [KOB]. An interesting relationship between CWTs and human hearing, with applications to speech analysis, is described in [DAM]. Background on formants and phonemes in linguistics can be found in [ODA].