Xiaolin Wu "Compression of Wavelet Transform Coefficients"
The Transform and Data Compression Handbook
Ed. K. R. Rao et al.
Boca Raton, CRC Press LLC, 2001

Chapter 8

Compression of Wavelet Transform Coefficients

Xiaolin Wu
University of Western Ontario

8.1 Introduction

Mathematical transforms are widely used in signal compression, particularly in compression of sensory data such as audio, image, and video. Although sensory signals are typically sampled and presented to users in the spatial/time domain, a direct signal representation in the spatial/time domain creates a huge volume of data with excessive redundancy. Clearly, signals in original sample form are unsuitable for transmission or storage. Transform coding is a proven paradigm for signal compression. In transform coding, signal samples are mapped from spatial/time domain into another space, typically a frequency or joint time-frequency domain in which statistical and subjective redundancies in the samples can be better understood, exploited, and removed. Transformed samples are thus more amenable to compression. This paradigm of transform-based signal compression is exemplified by the current and commercially successful industrial standards for image compression (JPEG standard [19]) and video compression (MPEG standards [4, 5]) [21]. A schematic description of a typical transform coding system is given in Fig. 8.1. The compression (encoding) process is completed in three major steps: transform of signal samples, quantization of transform coefficients, and entropy coding of quantized coefficients. The decompression (decoding) process is a reverse of the compression process.

The JPEG and MPEG standards use discrete cosine transform (DCT) in the transform step of the compression system. The acronyms JPEG and MPEG stand for the Joint Photographic Experts Group and the Moving Picture Experts Group. The two groups consist of members from both the International Standards Organization (ISO) and the International Telecommunications Union (ITU). They are charged respectively with the missions of developing international standards for the coded representation of compressed still images, and of compressed moving pictures and associated

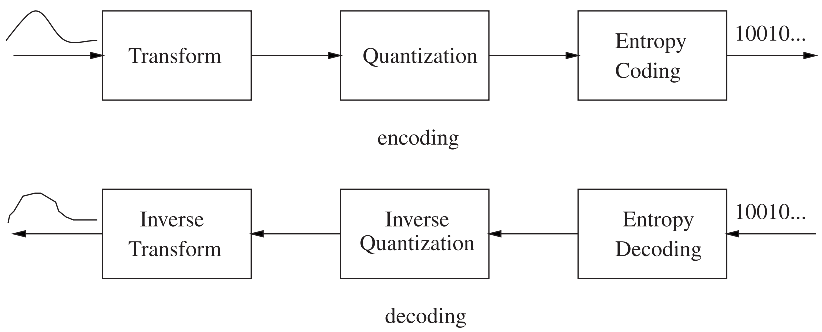

FIGURE 8.1
Schematic description of a typical transform coding system.

audio. Their efforts are instrumental for the prevalence of digital visual communications in multimedia and Internet applications. Due to the popularity of JPEG and MPEG compression standards and products, the DCT-based coding system is now considered a matured and effective technology for image and video compression. In 1988 when the JPEG members evaluated various image compression schemes and decided on the JPEG standard, the DCT-based image codecs offered the best compromise between compression performance, computational complexity (hardware complexity in particular), and coder flexibility, among other competing image compression technologies at that time, specifically vector quantization (VQ) [14] and DPCM (differential pulse coding modulation) coding.

Since the standardization of DCT-based compression technology, the past few years have seen rapidly increasing sophistication and maturity of wavelet-based image compression methods. Wavelet-based image codecs have so far delivered the best lossy compression performance in both peak signal to noise ratio (PSNR) and visual quality, over bit rates from 0.05 bits/pixel (summary quality for browsing) to 2.00 bits/pixel (visually indistinguishable from the original). During the same period, research on VQ compression and fractal compression has also advanced. But in image compression, neither VQ nor fractal compression methodology has matched the rate-distortion performance of wavelet-based image codecs at the time of this writing. Indeed, the recent success of wavelet transform in image compression has reinforced the dominance of widely practiced transform coding paradigm for signal compression. Only in the realm of lossless image compression, adaptive predictive coding has slightly (about 3%) higher compression ratio than lossless transform coding such as reversible integer wavelet codecs [34]. But this small advantage of predictive lossless coding becomes even more marginal in the presence of other unique features of wavelet lossless codecs, on which we elaborate later.

Within the transform coding family, discrete wavelet transform is threatening to unseat DCT as the transform of choice, at least for image compression applications. The current state-of-the-art wavelet image codecs significantly outperform the existing DCT-based JPEG standard in PSNR measure and subjective image quality, particularly for low bit rates at which the block effects of DCT are noticeable [25, 28, 30, 29, 36, 37]. Being encouraged by the improvements brought on by wavelet-based image compression techniques over DCT, and prompted by increasing acceptance of wavelet compression technology by industry, the JPEG committee has developed a new wavelet-based still image compression standard called JPEG 2000 [2]. Also, in 1993 the FBI chose a wavelet-based image codec to be the standard for fingerprint image compression [1].

The superior compression performance of wavelet-based image coding systems over their DCT-based counterpart might suggest that the improved performance was primarily made by replacing DCT with wavelet transform, and hence the choice of transform would matter the most to coding efficiency. However, in strict technical terms, all existing transforms used in signal compression by themselves do not lead to any data reduction. Both DCT and dyadic wavelet transforms, the two most widely used types of transforms in image and video compression, generate as many coefficients as the number of samples. Furthermore, while the original sample values of digital signals are integers, the transform coefficients are nonintegers. Therefore, without efficient coding of transform coefficients, a transform not only cannot compress but can even expand the data. The main benefit of transform to data compression is from its property of energy packing. A suitable transform can transfer the majority of signal energy into a few transform coefficients, resulting in a large number of zero and near-zero coefficients. In other words, the probability distribution of transform coefficients is much more biased than that of original samples. The more biased the distribution, the easier it is to compress signals by entropy coding. Despite the well-accepted folklore that lossy signal compression is better done via transform coding, it is the process of entropy coding that actually achieves data reduction. Informally, entropy coding refers to a family of coding techniques that uses shorter codewords for more probable symbols (smaller transform coefficients), and longer codewords for less probable symbols (larger transform coefficients). An optimal variable length code can achieve an average code length that approaches the information theoretic lower bound called entropy, hence the term entropy coding. Entropy coding is also referred to as noiseless or lossless coding since the coding process is perfectly reversible. It is a key machinery of information theory, a field fathered by Shannon [27] half a century ago and that has guided data compression engineering ever since. For background and rigorous treatment of entropy coding, we refer readers to textbooks such as Cover and Thomas [10].

In terms of energy packing capability, the principal component transform (also known as Karhunen-Loève transform [15]) is optimal in the sense that it distributes the largest amount of signal energy into the direction of the eigenvector of the largest eigenvalue (the direction of largest sample variance), and the second largest amount of signal energy into the second largest eigenvector direction, and so on. Therefore, if one is to choose only k coefficients to best approximate the original signal in L_2 metric, then the optimal choice will be the k coefficients corresponding to the eigenvectors of the k largest eigenvalues. DCT has been shown to be very close to the principal component transform when applied to the first order stationary Markov process [22]. This justifies the wide use of DCT in data compression. The energy packing capability of wavelet transform was studied by DeVore, Jawerth, and Lucier [11] who showed that wavelet bases are optimal among all possible basis functions in minmax nonlinear approximation obtained by retaining the k largest coefficients and discarding the remaining. Both DCT and wavelet transforms possess some good properties in terms of energy packing.

Wavelet transforms have two additional advantages over DCT that are important for coefficient compression. The first is the multiresolution representation of the signal by wavelet decomposition that greatly facilitates subband coding, a notion that existed long before the popularity of wavelets [32]. Fig. 8.2 shows an image pyramid associated with wavelet decomposition. It can be seen from the figure that wavelet transform preserves to some extent spatial signal features in subbands of different scales and creates self-similarities between the subbands of the same spatial orientation. This fractal-like structure reveals sample dependencies across scales to the benefit of statistical context modeling and coding of wavelet coefficients. In fact, the well-known zerotree techniques precisely exploit the self-similarity of regions of zero and near-zero coefficients. The second advantage of wavelet transform is that it reaches a good compromise between frequency and time resolutions of the signal. From the perspective of energy packing, statistically short-term signal constructs such as image edges, or transients in signal processing terminology, have much higher energy concentration in time domain; hence they can be modeled and coded far more efficiently in the time domain than in the frequency domain. However, the exact opposite is true for long-term signal constructs such as smooth shades and regular textures in images. Wavelet transforms are superior to DCT in that their basis functions offer good frequency resolution in the lower frequency range, and at the same time they yield good time resolution at a higher frequency range (see the well-preserved edge information in the three highest subbands in Fig. 8.2).

However, neither the multiresolution property nor the frequency-time characteristics of wavelets suffices for signal compression. Whether and how much signal compression can benefit from the good properties of wavelets largely depends on statistical context modeling (implicit or explicit) and entropy coding of wavelet coefficients. The difference in rate-distortion performance between the DCT-based JPEG codec and wavelet-based image codecs is mostly caused by the differences in entropy coding of transform coefficients between the two methods. Indeed, before Shapiro's zerotree technique (EZW) in 1993 [28], a landmark work on wavelet coefficient coding, wavelet transforms had not won over DCT in rate-distortion performance. More recently, particularly during the ongoing JPEG 2000 standardization process, further advances have been made in statistical context modeling and adaptive entropy coding of wavelet coefficients. The modern wavelet coefficient coding techniques [25, 36, 29, 41, 37] significantly outperform the pioneer EZW coder for any given wavelet transform. The new techniques have better rate-distortion performance over EZW because they overcome a weakness of zerotree. That is, while being an

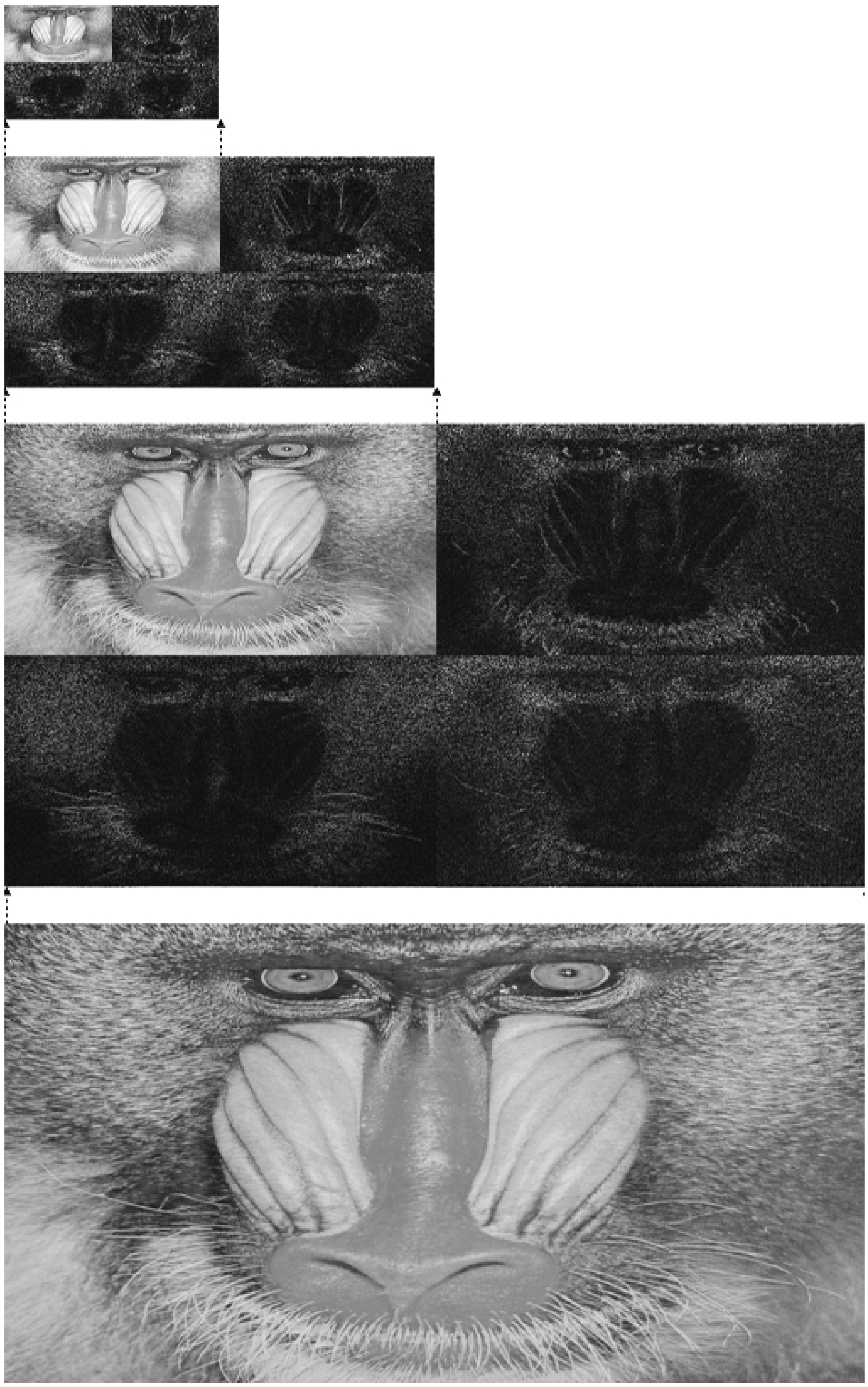

FIGURE 8.2
Dyadic wavelet decomposition of a test image.

effective technique to remove data redundancy in the form of a long-term trend, zerotree is less efficient to describe short-term signal constructs than the more advanced statistical modeling techniques discussed later in this chapter.

In summary, it is the increasing sophistication of coefficient coding, not the transforms alone, that contributes the most to the success enjoyed by wavelet image compression technology. Compression of coefficients is perhaps the most critical issue for any transform-based signal compression system. This chapter is dedicated to the problem of compression of transform coefficients. In order to make our discussions concrete and lucid, we focus on compression of wavelet coefficients in the setting of image coding. The general principles and techniques of this chapter, however, are applicable to compression of other transform coefficients and also effective with other types of signals, such as video and audio.

The structure of this chapter is as follows. Section 8.2 discusses the problem of compressing transform coefficients in wavelet-based image compression systems. Specifically, we introduce the popular approach of embedded bit-plane coding of quantized wavelet coefficients. Section 8.3 formulates the problem of statistical context modeling of wavelet coefficient and explains why this is the single most important issue that determines the compression performance. Since wavelet transforms cannot achieve total decorrelation between the signal samples, particularly when sample correlation is nonlinear, high-order statistical dependencies exist between wavelet coefficients. Therefore, optimum compression performance can be made possible only by high-order statistical context modeling of wavelet coefficients. However, if not treated with care, the number of Markov conditioning states can grow exponentially in the order of the model. This leads to a so-called problem of context dilution, addressed in Section 8.4. The challenge is how to maintain a modest number of conditioning states while still making high-order statistics available to aid entropy coding. Section 8.5 discusses how to discriminatingly choose modeling contexts in wavelet subbands as a means to control model cost. Section 8.6 introduces the process of context quantization to reduce drastically the number of conditioning states. The essence of context quantization is to merge different conditioning states that have similar symbol probability distributions. The subject is further pursued in Section 8.7, which investigates how to optimize context quantization for minimum code length. We borrow a common strategy of nonparametric multivariate statistical analysis to overcome high model cost: data projection in the direction of statistical dominance. Specifically, Fisher's linear discriminant [12] is used to guide context quantization. Section 8.8 presents a context quantizer design algorithm via dynamic programming that can minimize conditional entropy for a given number of conditioning states. Section 8.9 turns to the computational aspect of context modeling. Efficient algorithm techniques are developed to compute modeling contexts. We demonstrate that the time complexity of forming a modeling context is $O(1)$, independent of the order of the model, and thus high-order statistical context modeling is made computationally feasible. The chapter concludes with experimental results that provide convincing empirical evidence for the importance and effectiveness of context modeling and conditional entropy coding of wavelet coefficients in practical compression systems.

MSB	1	0	0	0	1		
	0	1	0	1	0	⟶	1 0 0 0 1, 0 1 0 1 0, 0 1 0 0 1, 1 0 0
	0	1	0	0	1		
LSB	1	0	0	1	1		

FIGURE 8.3
Embedded bit stream of coefficients.

8.2 Embedded Coefficient Coding

Like most transform coding systems a typical wavelet-based signal compression system consists of three cascaded modules, as depicted by Fig. 8.1, first wavelet transform, followed by quantization of wavelet coefficients, and finally entropy coding of quantized coefficients. To improve coding efficiency, one can perform adaptive wavelet transforms for better energy packing or optimal quantization for rate-distortion considerations. But most of the coding gains are usually made by conditional entropy coding of wavelet coefficients coupled with universal statistical context modeling. This is because transforms can remove only linear correlations between samples, whereas universal statistical context modeling can discover and remove more complex types of sample dependencies. In the ongoing JPEG 2000 standardization process, entropy coding of wavelet coefficients is by far the hottest subject being studied and debated by participating parties. It has been established empirically that the best rate-distortion results can be obtained by adaptive entropy coding of coefficients even with a fixed wavelet transform and uniform scalar coefficient quantization.

In order to focus this chapter on the last system component of coefficient entropy coding, in the following discussions we assume that the standard dyadic wavelet transform due to Mallat [17] is used in the transform module, and that simple uniform scalar quantization of wavelet coefficients is used in the quantization module. The input of entropy coder is the quantization indices of the coefficient magnitudes plus the signs of the coefficients. The quantized coefficients of dyadic transform are thus signed integers arranged in a two-dimensional layout of subbands as in Fig. 8.2. Image compression is finally achieved by lossless entropy coding of quantized coefficients.

A breakthrough of wavelet-based image compression technology is a coding scheme called embedded bit plane coding that was pioneered by Shapiro [28] in 1993 and then improved very rapidly by many other authors [25, 30, 43, 36, 37]. The idea is simple. Instead of coding all coefficients in one pass, and coding each coefficient once, we scan the coefficients in multiple passes, one bit plane per pass, from the most to the least significant bit, as illustrated in Fig. 8.3. Within a bit plane, the order of traversing wavelet coefficients in a two-dimensional subband layout can be arbitrary. A common traversal order is from the lowest frequency or the most coarse subband to the highest frequency or the most detailed subband, as depicted by Fig. 8.4. The binary sequence generated by such a traversal is called the embedded bit stream. An important property of the embedded bit stream is its scalability in

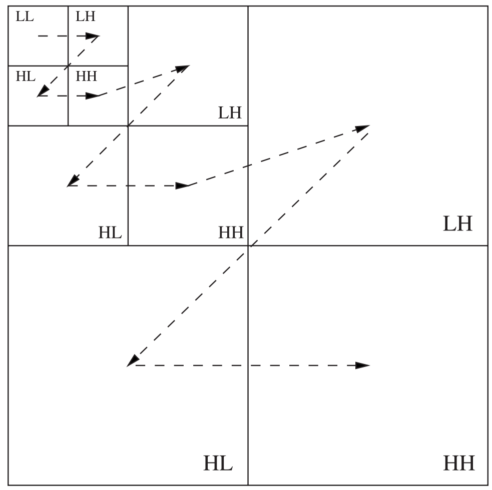

FIGURE 8.4
A traversal of subbands within a bit plane.

both spatial resolution and sample fidelity. Truncating an embedded bit stream at any point means approximating all wavelet coefficients at a certain precision; hence, the truncated bit stream can reconstruct the image at a corresponding fidelity — the longer the bit stream being used in the reconstruction, the higher the fidelity. The effect of successive refinement of a coded image via progressive transmission of an embedded bit stream is illustrated in Fig. 8.5. Scalable image and video compression allows transmission and distribution of the same source material at different quality levels to meet different bandwidth and storage capacity requirements, and the ability to do so with a single unified code stream. This feature is highly desirable in many applications, such as Internet, multimedia, medical imaging, prepress imaging, and image databases. With scalable coding one needs only to archive one master copy of the material in the database to support applications at different quality levels and under different constraints — from fast browsing to professional high quality reproduction — instead of maintaining multiple copies of the same materials at different bit rates for different bandwidths and quality trade-offs.

Scalable wavelet coding can also unify lossy and lossless compressions. If reversible integer wavelet transforms [8] are used, all coefficients are integers in the first place. No coefficient quantization is necessary; hence, there will be no quantization errors. In this case an embedded bit stream can eventually achieve perfect lossless decompression if every bit is received, while any truncation of the bit stream corresponds to a lossy decompression. The use of reversible integer wavelets for lossy to lossless scalable image compression was proposed by Zandi et al. [43] and Said and Pearlman [26]. This approach has very recently been extended to lossy and lossless compression of image sequences such as three-dimensional medical data, multi/hyperspectral remote sensing data, and video [18, 42].

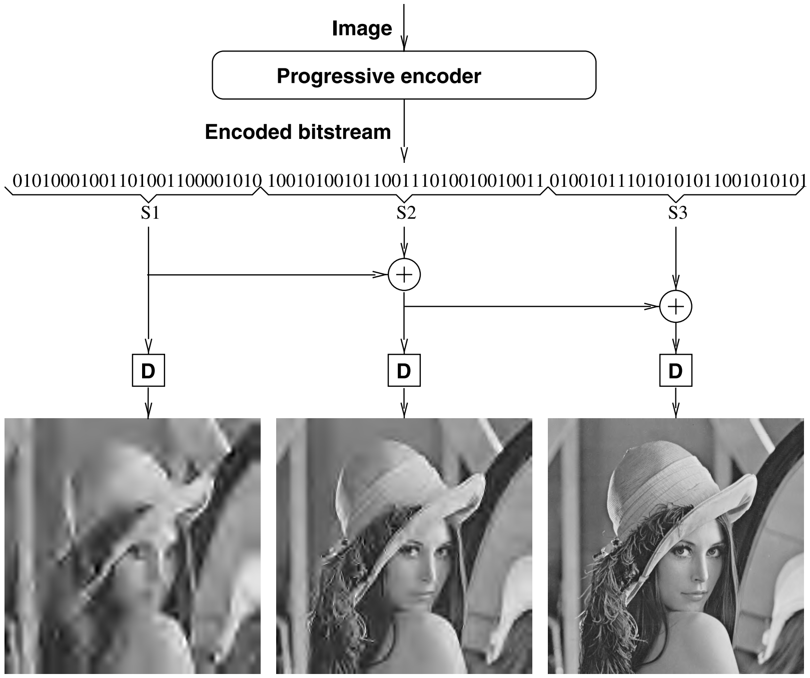

FIGURE 8.5
Progressive image reconstruction via scalable embedded code stream. Reproduced by Special Permission of *Playboy* magazine. Copyright ©1972, 2000 by Playboy.

The first published work on embedded bit plane coding of wavelet coefficients was Shapiro's zerotree algorithm [28]. Shapiro developed his embedded zerotree of wavelets (EZW) by observing that large blocks of zero coefficients exist in high frequency subbands and at bit planes of high significance. Furthermore, a block of zero coefficients statistically tends to reside in the same spatial location across different scales. If we consider a coefficient at a coarser subband as parent, and the four coefficients corresponding to the spatial location of the parent coefficient at the next finer subband as children, then coefficients of a dyadic wavelet transform can be naturally organized into quadtree data structures, as shown in Fig. 8.6. The conditional probability for all four children to be 0 given that the parent is 0 is much higher than given that the parent is 1. This statistical inheritance of 0 across different scales tends to form quadtrees of all 0 nodes, with their roots at upper levels of the multiresolution hierarchy and their leaves at the bottom level. Therefore, one can code a large number of 0 coefficients very compactly with a special code symbol for such zerotrees. This technique is very much like the zigzag technique of the existing DCT-based JPEG standard for coding long runs of 0 coefficients. In essence, the EZW technique compresses wavelet coefficients using a prior statistical model, i.e., assuming that

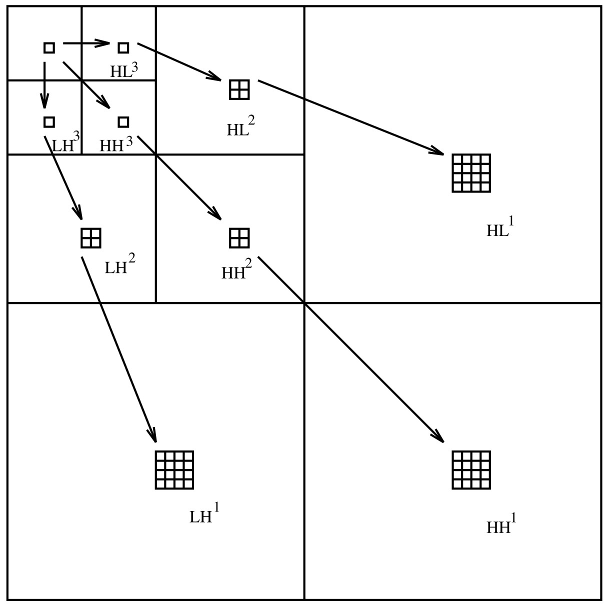

FIGURE 8.6
Coefficient quadtrees across different scales. Zerotrees are those quadtrees whose nodes are all 0.

zero and near-zero coefficients are clustered in both spatial and frequency domains, and that the regions of low sample energy are self-similar across different scales. The rate-distortion performance of the EZW technique was improved by a variant of the zerotree coder called SPIHT, proposed by Said and Pearlman [25]. Unlike the EZW algorithm that forms and codes zerotrees in a fixed spatial scanning order, SPIHT codes the zerotrees in an order that is beneficial to rate-distortion performance; those trees that are likely to generate higher reduction in distortion are coded first. The better performance of SPIHT over EZW is also due to the use of a finer tree-based classification of source symbols and the use of joint entropy (specifically, coding four binary symbols in a block).

But the best image compression results reported so far in the literature were not generated by zerotree-based methods, but rather by a sample-by-sample bit plane coding technique called ECECOW (embedded conditional entropy coding of wavelet) coefficients [36, 41]. A drawback of the zerotree, or similar quadtree type of data structures used by EZW [28] and SPIHT [25] algorithms, is that the tree imposes an artificial structure on the wavelet coefficients. Only contexts of square shape in the spatial domain can be used, whereas statistically related wavelet coefficients may form regions of arbitrary shapes. Moreover, like any run-length type codes, quadtree code cannot efficiently describe statistically short-term signal constructs, such as edges, because the implicit statistical model used by zerotree breaks down on transient sample behavior. Relative to sample-by-sample coding, zerotree can be considered as a block-based entropy code. It largely ignores the sample dependency between neighboring quadtree nodes. This limitation is particularly regrettable considering that wavelet transform represents a fundamental departure from block-based DCT. The first technique of embedded bit plane coding of wavelet coefficients without any tree constraints seems to be Taubman and Zakhor's layered zero coding (LZC) algorithm [30]. Another early wavelet image coder, called CREW (Compression via Reversible Embedded Wavelets) [43], also did not confine the formation of modeling contexts to quadtree nodes. Compared with its predecessors, the main strength of the ECECOW algorithm is its using higher-order context modeling of embedded wavelet coefficient symbol streams.

8.3 Statistical Context Modeling of Embedded Bit Stream

This section formulates the problem of entropy coding of embedded wavelet bit streams, namely, coding uniformly quantized wavelet coefficients bit plane by bit plane, scanning from the most to the least significant bits. Within each bit plane there are many possibilities of traversing different subbands, and different ways of traversing a subband other than raster scan. Flexible bit traversal can support many desirable functionalities such as region of interests, error resilience, and rate-distortion optimization [29]. The context modeling and entropy coding techniques developed in this chapter all support any traversals within a bit plane. For simplicity, we assume a raster scan in the following descriptions.

The bit plane coding deals with only two source symbols: 0 or 1. However, accompanying the most significant bit of a coefficient, its sign should also be coded. Since the sign is a binary event, we again have only two possible source symbols in this situation. Therefore, in bit plane coding, all wavelet coefficients of an image can be conveniently converted into a sequence of binary symbols: $x_1, x_2, \ldots, x_n$, $x_i \in \{0, 1\}$. The minimum code length of the binary sequence in bits is given by

$$-\log_2 \prod_{i=1}^{n} P\left(x_i | x^{i-1}\right) , \tag{8.1}$$

where x^{i-1} denotes the sequence $x_{i-1}, x_{i-2}, \ldots, x_1$. If the conditional probability $P(x_i|x^{i-1})$ is known, then arithmetic coding can approach this minimum rate. Arithmetic coding is a powerful entropy coding technique with an arbitrarily high coding efficiency (limited only by the precision of arithmetic operations). It was pioneered by Rissanen and Langdon [23] and popularized by Witten, Neal, and Cleary [31]. Since embedded wavelet symbol sequence is binary, it can be compressed by adaptive binary arithmetic coding, the simplest and fastest version of adaptive arithmetic coding. Efficient, good approximation algorithms, such as QM coder [20], for adaptive binary arithmetic coding have been well studied and can be easily implemented by both software and hardware. Indeed, QM coder and other variants of binary arithmetic coding are used in many image compression standards, such as the new lossless JPEG standard JPEG-LS (JPEG-LS high-performance extension, LS mean lossless) [3], the JBIG (Joint Binary Image Group) lossless binary image compression standard [7], the JPEG 2000 standard [2], and others [21]. In addition to facilitating binary arithmetic coding, the binarization of the wavelet coefficients also offers great operational advantages for high-order context modeling, as appreciated in subsequent sections.

With arithmetic coding, we can separate the entropy coding completely from statistical context modeling, i.e., the problem of estimating $P(x_i|x^{i-1})$. Given a probability estimate $\hat{P}(x_i|x^{i-1})$, arithmetic coding can achieve the code length $-\log_2 \prod_{i=1}^{n} \hat{P}(x_i|x^{i-1})$. The remaining problem, also a far more difficult one, is how to reach a good estimate $\hat{P}(x_i|\underline{x}^{i-1})$ of $P(x_i|x^{i-1})$, where $\underline{x}^{i-1}$ denotes a subsequence of x^{i-1} that consists of past samples of statistical significance to x_i. Note that the most relevant past subsequence $\underline{x}^{i-1}$ is not necessarily a prefix of x^{i-1}. In image coding, $\underline{x}^{i-1}$ or a causal template for x_i consists of adjacent symbols in both time and frequency. The estimated conditional probability mass function $\hat{P}(x_i|\underline{x}^{i-1})$ serves as a statistical model of the source. The modeling context is the set of past observations $\underline{x}^{i-1}$ on which the probability of the current symbol is conditioned.

In fact, statistical context modeling in the form of probability estimation lies at the heart of any compression system. Ultimately it is the model quality, or the precision of probability estimate, that determines the rate-distortion performance. The true magic of wavelet transforms to compression is in their support of context modeling of sample dependencies via the localization of signal energy in both frequency and time/spatial domains. Specifically, wavelet coefficients of similar magnitudes statistically cluster in frequency subbands and in time/spatial locations. Large wavelet coefficients in different frequency subbands tend to register at the same spatial locations. This localization property makes statistical context modeling of the image signals much easier in wavelet domain than in time/spatial domain or other transform spaces. Specifically, the choice of relevant modeling context $\underline{x}^{i-1}$ becomes easier in the wavelet domain, as explained below.

We take a universal source coding approach to compression of the binary sequence x^n, assuming no prior knowledge about $P(x_i|x^{i-1})$. The central task is to estimate the conditional probability $P(x_i|\underline{x}^{i-1})$ "on the fly" based on the past coded bits and to use the estimate $\hat{P}(x_i|\underline{x}^{i-1})$ to drive an adaptive binary arithmetic coder. For easy reference to individual samples x_i in the binary sequence x^n, we denote the b-th bit of a coefficient $\textsc{c}$ by $\textsc{c}_b$, the i-th through j-th bits of $\textsc{c}$, $j > i$, by $\textsc{c}_{j..i}$, and all the bits of $\textsc{c}$ that are above the b-th bit by $\textsc{c}_{..b+1}$. In the sequel, the notation $\textsc{c}_{j..i}$ always refers to the bits in the binary encoding of coefficient magnitude $|\textsc{c}|$. The sign of $\textsc{c}$ is denoted by $\tilde{\textsc{c}}$. Note that the bits of $\textsc{c}_{j..i}$ are not consecutive in an embedded wavelet bit stream but are scattered around. If the most significant bit of $\textsc{c}$ is lower than b, then $\textsc{c}_{..b}$ is considered to be 0. We use directional notations N, W, S, E, NW, NE, NN, WW, and so on, to denote the coefficients to the north, west, south, east, northwest, northeast, northnorth, and westwest of the current coefficient C. Similarly, we denote the parent coefficient by P, and those coefficients in the parent subband to the north, west, south, and east of P by PN, PW, PS, and PE.

In coding of the b-th bit plane, we may condition c_b on

$$\mathrm{C}_{..b+1}, \mathrm{N}_{..b}, \mathrm{W}_{..b}, \mathrm{S}_{..b+1}, \mathrm{E}_{..b+1}, \mathrm{NW}_{..b}, \mathrm{NE}_{..b}, \\ \mathrm{NN}_{..b}, \mathrm{WW}_{..b}, \mathrm{P}_{..b}, \mathrm{PN}_{..b}, \mathrm{PS}_{..b}, \mathrm{PW}_{..b}, \mathrm{PE}_{..b}, \ldots \tag{8.2}$$

We treat all the known bits, up to the moment of coding c_b, of the neighboring coefficients in current and parent subbands as potential feature events in modeling context $\underline{x}^{i-1}$ of $x_i = \mathrm{c}_b$. Unlike in the EZW and SPIHT algorithms, our modeling context of c_b contains some future information if one considers that the octave-raster scanning of coefficients produces a time series. Specifically, this refers to the use of $\mathrm{S}_{..b+1}$, $\mathrm{E}_{..b+1}$, $\mathrm{PS}_{..b}$, $\mathrm{PE}_{..b}$, and the like in context modeling of c_b. The ability of looking into the future in a time series significantly reduces the uncertainty of c_b.

8.4 Context Dilution Problem

The modeling context of Eq. (8.2) leads to a statistical model

$$P\left(\mathrm{c}_b | \mathrm{N}_{..b}, \mathrm{W}_{..b}, \mathrm{S}_{..b+1}, \mathrm{E}_{..b+1}, \mathrm{NW}_{..b}, \mathrm{NE}_{..b}, \mathrm{P}_{..b} \cdots\right) \tag{8.3}$$

of very high order or long memory. High-order context modeling is necessary for optimal compression performance because image features such as edges can involve pixels that are spatially far apart. Given a modeling context $(\mathrm{N}_{..b}, \mathrm{W}_{..b}, \mathrm{S}_{..b+1}, \mathrm{E}_{..b+1}, \mathrm{NW}_{..b}, \mathrm{NE}_{..b}, \mathrm{P}_{..b} \cdots)$, the average code length of c_b is bounded from below by the conditional entropy

$$\begin{aligned} &H\left(\mathrm{c}_b | \mathrm{N}_{..b}, \mathrm{W}_{..b}, \mathrm{S}_{..b+1}, \mathrm{E}_{..b+1}, \ldots\right) \\ &\quad = -E\left\{\log P\left(\mathrm{c}_b | \mathrm{N}_{..b}, \mathrm{W}_{..b}, \mathrm{S}_{..b+1}, \mathrm{E}_{..b+1}, \ldots\right)\right\} . \end{aligned} \tag{8.4}$$

The fact that conditional entropy is monotonically nonincreasing [10] seems to suggest that the higher the order of the context model, the shorter the code length. But this is not necessarily true.

In universal source coding we do not have prior knowledge of the source. The model itself must be either explicitly sent to the decoder or learned on the fly from the samples. In the former case, we need to add side information to the total description length of the source. In the latter case, the learning requires a large number of samples to fit a statistical model to the source. The number of possible conditioning states grows exponentially with the order of the context, an image of finite resolution may not provide sufficient samples to reach a robust estimate of the underlying conditional probability

$$\hat{P}\left(\mathrm{c}_b | \mathrm{N}_{..b}, \mathrm{W}_{..b}, \mathrm{S}_{..b+1}, \mathrm{E}_{..b+1}, \mathrm{NW}_{..b}, \mathrm{NE}_{..b}, \mathrm{P}_{..b} \cdots\right) . \tag{8.5}$$

In order words, too high an order of modeling context spreads sample statistics too thin among all possible modeling states to yield statistical significance. The code length will actually increase when the order of modeling context gets too high. Thus, from an implementation point of view, high order of context modeling is more than a problem of high time and space complexity. It can reduce coding efficiency as well. This problem is commonly known as *context dilution* and formulated by Rissanen analytically as *model cost* [24]. Intuitively, the higher the model complexity (i.e., the more model parameters), the longer the time the model takes to *learn* from the samples to set the parameters right. Before the model converges to the underlying statistics via online learning, entropy coding cannot achieve the minimum code length of Eq. (8.1). Therefore, the context model has an inherent cost to the total description length, either in the form of side information to describe the model, as in two-pass coding, or in the form of extra code length due to model mismatch in the beginning of the learning process, as in one-pass coding.

By now, one may appreciate an advantage of turning the wavelet coefficients into a binary sequence. Since a conditional binary probability has only two parameters, we do not need nearly as many samples to obtain a good probability estimate as for a large symbol alphabet. But even with c_b being binary, we still have to reduce Eq. (8.2) to a modest number of conditioning states; otherwise the benefits of context modeling will be negated by high model cost. Indeed, in his original EZW paper [28], Shapiro remarked that Markov conditioning did not offer significant coding gains over "single histogram strategy" (entropy coding based on symbol probability without context modeling). In their original SPIHT paper [25], Said and Pearlman also implied that in their experiments high-order context modeling made only marginal coding gains over the simple Huffman coding. But their observations did not mean the lack of high-order statistical dependencies between samples in the wavelet domain. Their experimental results with context modeling were somewhat disappointing only because the problem of context dilution was not considered. The challenge is to reduce the model cost and still capture statistically significant structures of high orders between the samples.

8.5 Context Formation

One way to reduce the number of model parameters, and thus to reduce the model cost, is to include into the modeling context only those past samples that are statistically related to the current sample being coded. For one-dimensional sources, such as text, speech, and audio, the modeling context selection criterion can be some prefix of the current sample because the amount of sample dependency is proportional to the distance between the samples. Similarly, for image and video sources the general practice is to choose a spatial and temporal neighborhood to form the modeling context. However, as we pointed out in the previous section, the resulting context can be of a very high order. A more selective rule than k nearest neighbors, where k is the size of context template, should be used if we have any prior knowledge about sample structures.

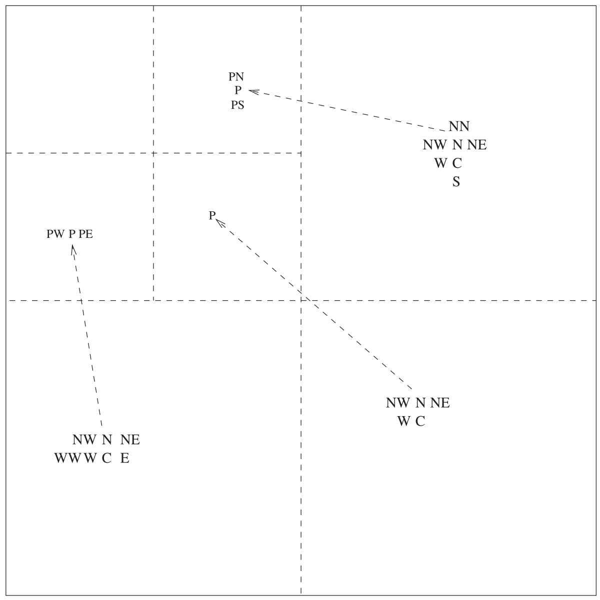

FIGURE 8.7
Modeling contexts in different subbands.

The feature orientations of different wavelet subbands are the kind of prior knowledge that is useful for reducing the model cost. For instance, the LH subband exhibits predominantly vertical sample structures, while the HL subband exhibits predominantly horizontal sample structures. Therefore, we choose a subset of Eq. (8.2):

$$S_{LH} = \{\text{N}_{..b}, \text{W}_{..b}, \text{NW}_{..b}, \text{NE}_{..b}, \text{NN}_{..b}, \text{S}_{..b+1}, \text{P}_{..b}, \text{PN}_{..b}, \text{PS}_{..b}\} \tag{8.6}$$

to be the modeling context of c_b in LH subbands. This choice of conditioning events forms a vertically prolonged modeling context. Similarly, we use a horizontally prolonged modeling context

$$S_{HL} = \{\text{N}_{..b}, \text{W}_{..b}, \text{NW}_{..b}, \text{NE}_{..b}, \text{WW}_{..b}, \text{E}_{..b+1}, \text{P}_{..b}, \text{PW}_{..b}, \text{PE}_{..b}\} \tag{8.7}$$

in modeling of c_b in HL subbands.

Note that we include corner samples $\text{NW}_{..b}$ and $\text{NE}_{..b}$ in the northwest and northeast directions, but not $\text{SW}_{..b+1}$ and $\text{SE}_{..b+1}$ in the southwest and southeast directions. The reason is that the former two samples have one bit more precision and are therefore statistically more significant than the latter two samples in the raster scan of bit planes. In our experiments, including two more samples at the southwest and southeast corner did not bring any compression gains, and in some cases it could even increase the code length due to the effect of context dilution. In practice, when choosing a modeling context one can monitor the resulting code length as the order of modeling context increases. This will empirically detect the point where the increased model cost just starts to have negative impact on compression. Thus one can choose an appropriate order of the model by not adding to the modeling context samples of less statistical significance to c.

In HH subbands, sample structures tend to be much weaker than in LH and HL subbands. A smaller modeling context can be used without reducing compression performance. In our experiments, we found that maximum coding gains can be made by conditioning a c_b in an HH subband on the following set of samples:

$$S_{HH} = \{\mathrm{N}_{..b}, \mathrm{W}_{..b}, \mathrm{NW}_{..b}, \mathrm{NE}_{..b}, \mathrm{S}_{..b+1}, \mathrm{E}_{..b+1}, \mathrm{P}_{..b}, \mathrm{C}_{HL}, \mathrm{C}_{LH}\} \tag{8.8}$$

where c_{HL} and c_{LH} are two sister coefficients of c that are at the same spatial location in the HL and LH subbands of the same scale. The different shapes and orientations of modeling contexts used in different subbands are illustrated in Fig. 8.7.

Due to the use of ubiquitous L_2 metric in wavelet approximation, samples in all subbands except the one in the lowest frequency are drawn from zero mean processes. The coefficient sign $\tilde{\mathrm{c}}$ has equal probability to be positive and negative. Consequently, we have $H(\tilde{\mathrm{c}}) = 1$; i.e., the self entropy of coefficient sign is at the maximum. But this does not necessarily mean that the signs are uncompressible. In fact, the conditional entropy of the signs can be significantly lower than 1. The waveform structures of the input image are often exhibited by sign patterns of wavelet coefficients. In Fig. 8.8 we plot the spatial distributions of signs for parts of two popular test images, "Barb" and "Lena" that have high textures. The clearly visible structures of signs suggest that the sign bits of wavelet coefficients can be modeled as a Markov process and compressed by conditional entropy coding.

During embedded bit plane coding, the sign $\tilde{\mathrm{c}}_{..b}$ of a wavelet coefficient c has three states: $+$, $-$, and 0. At the b-th bit plane, $\tilde{\mathrm{c}}_{..b}$ is still unknown to the decoder if $\mathrm{c}_{..b} = 0$; i.e., the most significant bit of c is below b. In this case the coder assigns state 0 to $\tilde{\mathrm{c}}_{..b}$; otherwise it assigns $+$ or $-$ to $\tilde{\mathrm{c}}_{..b}$ by the conventional meanings of sign. Here the state 0 is a dynamic concept; it may change to $+$ or $-$ as the coding process advances to deeper bit planes. We distinguish 0 from $+$ and $-$ because such a distinction yields a more revealing modeling context for the signs. The use of three states of signs in context modeling exploits the correlation between the signs and the magnitudes of neighboring wavelet coefficients because $\tilde{\mathrm{c}}_{..b} = 0$ indicates a relatively small $|\mathrm{c}|$. The modeling context for $\tilde{\mathrm{c}}$ commonly consists of sign status of c's four immediate neighboring samples, namely it is the set

$$\tilde{S} = \left\{\tilde{\mathrm{N}}_{..b}, \tilde{\mathrm{W}}_{..b}, \tilde{\mathrm{S}}_{..b}, \tilde{\mathrm{E}}_{..b}\right\} . \tag{8.9}$$

8.6 Context Quantization

Careful selection of past samples to be used in modeling context based on subband orientations is only a screening process. The number of possible conditioning states of

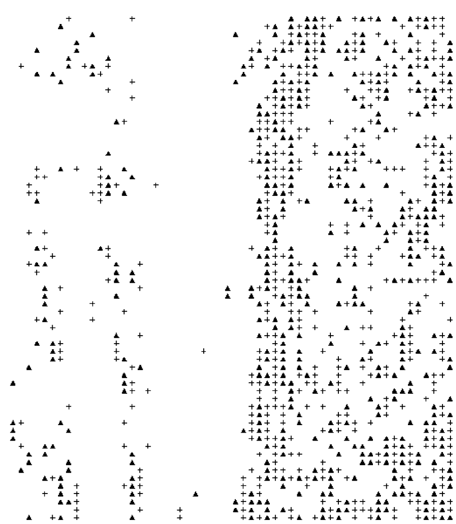

FIGURE 8.8
Sign patterns in parts of "Barb" (top) and "Lena" (bottom). The triangles are for negative signs, + for positive signs, and spaces for insignificant coefficients up to the current bit plane.

the chosen context is still far too large. Context dilution remains a serious problem. A rule of thumb for the right number of conditioning states in embedded wavelet image coding is about 64. The use of more than 100 conditioning states hardly makes any compression gain, and in many cases it can even increase the bit rate. A common technique of reducing the number of conditioning states for entropy coding is context quantization. The idea is to merge conditioning states in which the sample probability distributions are close in terms of Kullback-Leibler distance or relative entropy [10].

A simple scheme is scalar quantization of samples in the modeling context. In a modeling context that consists of eight or so samples, such as those in Eqs. (8.6) and (8.7), scalar quantization has to be very coarse in order to bring the number of conditioning states under 100. Indeed, many of the wavelet image coders reported in the literature use binary quantization of feature samples [43, 29]. In other words, feature samples $\text{N}_{..b}$, $\text{W}_{..b}$, $\text{S}_{..b}$, $\text{E}_{..b}$, etc. are entered into the context as either 1 (already significant at the current bit plane) or 0 (not yet significant at the current bit plane). Such a coarse quantization can obscure some subtleties in correlations between C and the energy level of the neighboring coefficients.

In order to capture the correlation between C and its neighbors in the wavelet domain, we use a linear estimator Δ of the magnitude of C, one for each of three orientations (LH, HL, and HH) of subbands:

$$\Delta_\theta = \sum_{z_i \in S_\theta} \alpha_{\theta,i} z_i , \quad \theta \in \{LH, HL, HH\} , \tag{8.10}$$

where the terms z_i are conditioning events in the context chosen for the given subband of C as described above. The parameters $\alpha_{\theta,i}$ are determined by linear regression so that Δ_θ is the least-squares estimate of C in the given subband orientation. The linear regression can be done offline for a general set of training images, a given class of images, and even for a given image. Of course in the last case, the optimized parameters have to be sent as side information.

For each of Δ_θ we can design an optimal quantizer Q_θ to minimize the conditional entropy

$$H\left(\text{C}|Q_\theta\left(\Delta_\theta\right)\right) = E\left\{\log P\left(\text{C}|Q_\theta\left(\Delta_\theta\right)\right)\right\} . \tag{8.11}$$

Since Δ_θ is a scalar random variable, the optimal quantizer Q_θ to achieve minimum conditional entropy can be computed via a standard dynamic programming process [33]. The optimization is carried out offline using a training set, and the quantizer parameters are stored and available at both the encoder and decoder. (In order not to interrupt the flow of our presentation we defer the details of dynamic programming process for designing minimum conditional entropy quantizers to Section 8.8.)

Besides the correlation between C and the local energy level Δ, the wavelet coefficient C also has dependence on spatial patterns of its neighboring coefficients, particularly at locations of strong edges or high textures. This dependence is due to the fact that a wavelet transform offers certain time resolution of the signal at the expense of frequency resolution. Therefore it is necessary to model the spatial sample patterns in the wavelet domain to maximize coding gains. Again the required statistical modeling has to be done without drastically increasing the number of conditioning states. To achieve this, we quantize the sample spatial pattern around C into a binary vector (bit pattern) $T_b = t_4 t_3 t_2 t_1 t_0$ by

$$
\begin{aligned}
t_0 &= \text{N}_{..b} > \text{C}_{..b+1} ?0 : 1 \; ; \\
t_1 &= \text{W}_{..b} > \text{C}_{..b+1} ?0 : 1 \; ; \\
t_2 &= \text{S}_{..b+1} > \text{C}_{..b+1} ?0 : 1 \; ; \\
t_3 &= \text{E}_{..b+1} > \text{C}_{..b+1} ?0 : 1 \; ; \\
t_4 &= \begin{cases} \text{P}_{..b} + \text{PN}_{..b} + \text{PS}_{..b} > 6\text{C}_{..b+1} ?0 : 1 & \text{in LH subbands ;} \\ \text{P}_{..b} + \text{PW}_{..b} + \text{PE}_{..b} > 6\text{C}_{..b+1} ?0 : 1 & \text{in HL subbands .} \end{cases}
\end{aligned}
\tag{8.12}
$$

The type of binary context quantization as in Eq. (8.12), as we mentioned earlier, is directly used to form conditioning states in many embedded wavelet image/video coders [29, 30, 43]. But significantly higher compression gains can be made by combining quantized energy level $Q_\theta(\Delta_\theta)$ and the spatial pattern T_b of C's neighboring coefficients to form conditioning states in entropy coding of C. Specifically, C is coded by an adaptive binary arithmetic coder driven by probability estimate

$$\hat{P}\left(\text{C}_b | Q_\theta\left(\Delta_\theta\right), T_b\right) \; . \tag{8.13}$$

8.7 Optimization of Context Quantization

The previous section introduced context quantization as a necessary component for statistical modeling and entropy coding of wavelet coefficients and presented some context quantization techniques. However, these techniques are largely based on heuristics, albeit being proven to be useful in practice. This section investigates the problem of context quantization in a multivariate analysis approach of statistics and develops algorithms for designing optimum context quantizer for minimum conditional entropy.

Context quantization is a special form of vector quantization whose criterion should ideally be minimum conditional entropy. It is well-known that optimal vector quantization is NP-complete — a problem is said to be NP-complete if its exact solution requires an amount of computations that increases exponentially in the input size [13]. In other words, for a large training set which is required if the derived VQ solution is to have any statistical significance, designing the globally optimal vector is computationally intractable. Thus, we necessarily resort to alternative techniques that are computationally feasible. Since high dimensionality is the main cause for the complexity of the problem, we would naturally like to reduce the dimensionality of the problem. A classical approach in multivariate analysis is to project sample vectors of high dimensions onto a lower dimensional space that contains most of the statistical variations.

A high-order modeling context such as the one in Eq. (8.2) can be viewed as a modeling event vector $\mathbf{v} = (v_1, v_2, \ldots, v_d)$, where v_i is a modeling event. Let $V = \{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_k\}$ be a training set of event vectors. V can be the set of all event vectors observed so far in an online learning process, or an offline training set. The former is necessary if the context quantizer is designed on the fly in one-pass coding, whereas the latter is for offline context quantizer design. We partition V into V_0 and V_1, where subset V_0 (V_1) contains all the modeling event vectors associated with $c_b = 0$ ($c_b = 1$). If there exists a hyperplane or some other surface in the d-dimensional event space that can completely separate V_0 and V_1, then the binary symbol c_b to be coded is uniquely determined by its modeling context. In this ideal case the conditional entropy of c_b is 0. In reality, however, the two point subsets V_0 and V_1 are mingled in the event space in a complicated way. To simplify the problem, we can project all training event vectors onto a line and hope that V_0 and V_1 form distinct clusters along the line. This approach is due to Fisher [12]. Let the projection be

$$u_i = \mathbf{a}^T \mathbf{v}_i, \quad i = 1, 2, \ldots, |V| . \tag{8.14}$$

Given a training set V, we want to determine the projection direction $\mathbf{a}$ such that

$$G(\mathbf{a}) = \frac{(\mu_0 - \mu_1)^2}{\sigma_0^2 + \sigma_1^2} \tag{8.15}$$

is maximized, where

$$\mu_j = E\left\{u_i | \mathbf{v}_i \in V_j\right\} = E\left\{\mathbf{a}^T \mathbf{v}_i | \mathbf{v}_i \in V_j\right\}, \quad j = 0, 1 \tag{8.16}$$

and

$$\sigma_j^2 = E\left\{\left(\mathbf{a}^T \mathbf{v}_i - \mu_j\right)^2 | \mathbf{v}_i \in V_j\right\}, \quad j = 0, 1 . \tag{8.17}$$

The criterion of maximum $G(\mathbf{a})$ can be intuitively understood as maximum separation of V_0 and V_1. The numerator demands maximum distance between the projected means of V_0 and V_1 in direction $\mathbf{a}$, whereas the denominator requires minimum overlap of V_0 and V_1 along the projection line.

We use a well-known procedure in multivariate analysis literature for maximizing $G(\mathbf{a})$ based on sample event vectors. We rewrite Eq. (8.15), by scaling, in terms of sample scatter matrices S_0 and S_1 for V_0 and V_1, respectively,

$$G(\mathbf{a}) = \frac{(\mu_0 - \mu_1)^2}{\mathbf{a}^T (S_0 + S_1) \mathbf{a}} . \tag{8.18}$$

The scatter matrix is defined by

$$S_i = \sum_{\mathbf{v} \in V_i} (\mathbf{v} - \mathbf{m}_i)(\mathbf{v} - \mathbf{m}_i)^T, \quad i = 0, 1 \tag{8.19}$$

where

$$\mathbf{m}_i = \frac{1}{|V_i|} \sum_{\mathbf{v} \in V_i} \mathbf{v}, \quad i = 0, 1 \, . \tag{8.20}$$

We also express the numerator of Eq. (8.15) in terms of sample means:

$$(\mu_0 - \mu_1)^2 = \mathbf{a}^T \, (\mathbf{m}_0 - \mathbf{m}_1) \, (\mathbf{m}_0 - \mathbf{m}_1)^T \, \mathbf{a} \, . \tag{8.21}$$

Letting $M = (\mathbf{m}_0 - \mathbf{m}_1)(\mathbf{m}_0 - \mathbf{m}_1)^T$ and $S = S_0 + S_1$, we have

$$G(\mathbf{a}) = \frac{\mathbf{a}^T M \mathbf{a}}{\mathbf{a}^T S \mathbf{a}} \, . \tag{8.22}$$

Differentiating $G(\mathbf{a})$ and setting $\partial G / \partial \mathbf{a} = 0$ to determine the direction $\hat{\mathbf{a}}$ that maximizes $G(\mathbf{a})$, we arrive at

$$\frac{\hat{\mathbf{a}}^T M \hat{\mathbf{a}}}{\hat{\mathbf{a}}^T S \hat{\mathbf{a}}} S \hat{\mathbf{a}} = M \hat{\mathbf{a}} \, . \tag{8.23}$$

Now the underlying optimization problem reduces to one of an eigenvalue with the scalar term $\lambda = (\hat{\mathbf{a}}^T M \hat{\mathbf{a}}) / (\hat{\mathbf{a}}^T S \hat{\mathbf{a}})$. If S^{-1} exists, the direction of $\hat{\mathbf{a}}$ is given by

$$\hat{\mathbf{a}} = S^{-1} M \hat{\mathbf{a}} \, . \tag{8.24}$$

Since $M\hat{\mathbf{a}}$ has the direction of $\mathbf{m}_0 - \mathbf{m}_1$, it follows that

$$\hat{\mathbf{a}} = S^{-1} \, (\mathbf{m}_0 - \mathbf{m}_1) \, . \tag{8.25}$$

The simple solution above is made possible by the binarization of source symbols via embedded bit plane coding. The binarization conveniently lends Fisher's linear classifier with two classes to our context quantization problem. In Fisher's original work, the objective is to find a linear discriminant to classify between V_0 and V_1 for minimum classification error. But in reality, the projected samples of V_0 and V_1 in the direction of $\hat{\mathbf{a}}$ can be intermingled in such complicated ways that Fisher's discriminant leaves a significant degree of uncertainty. Much finer context quantization is required to further resolve the uncertainty and to approach rate-distortion optimality.

8.8 Dynamic Programming for Minimum Conditional Entropy

Once the direction of maximum separation $\hat{\mathbf{a}}$ is determined, we project all training event vectors onto a line in this direction. On this line the projection establishes an order of training event vectors by their projection values $u_i = \hat{\mathbf{a}}^T \mathbf{v}_i$, $i = 1, 2, \ldots, |V|$, namely $\mathbf{v}_i \leq \mathbf{v}_j$ if $u_i \leq u_j$. This linear ordering enables a constrained optimization approach of dynamic programming to design a K-level context quantizer. The constraint is that all quantizer cells are perpendicular to direction $\hat{\mathbf{a}}$. Under the constraint, the K-level context quantizer can be globally optimized for minimum conditional entropy, which is better than a gradient descent method that may be trapped in a local minimum. It is easy to see that in Section 8.6, the least-square estimator Δ of Eq. (8.10) also corresponds to a projection in high-dimensional feature space and establishes an order of training event vectors via the projection. Therefore, the same dynamic programming process to be developed in this section can be used to solve the optimization problem posted around Eq. (8.11) as well.

Let $\underline{u} = \min_i u_i$, $\bar{u} = \max_i u_i$, and denote by $Q(\tau, k)$ the set of all possible k-dimensional vectors $\mathbf{q} = (q_1, q_2, \ldots, q_k)$ such that

$$\underline{u} \equiv q_0 < q_1 < q_2 < \cdots < q_{k-1} < q_k = \tau < \bar{u} \, . \tag{8.26}$$

In designing the context quantizer, we associate each modeling event vector $\mathbf{v} \in V$ with the random variable $\mathrm{c}_b \in \{0, 1\}$ being modeled. Then the optimal context quantizer that minimizes conditional entropy is given by

$$\hat{\mathbf{q}} = \arg \min_{\mathbf{q} \in Q(\bar{u}, K)} \sum_{k=1}^{K} P\left(u_i \in (q_{k-1}, q_k]\right) H\left(\mathrm{c}_b | u_i \in (q_{k-1}, q_k]\right) \tag{8.27}$$

where

$$H\left(\mathrm{c}_b | u_i \in (q_{k-1}, q_k]\right) = -E\left\{\log P\left(\mathrm{c}_b | u_i \in (q_{k-1}, q_k]\right)\right\} \, . \tag{8.28}$$

In the formulation, the k-th quantizer cell corresponds to a subset $\Psi_k = \{\mathbf{v}_i | q_{k-1} < u_i \leq q_k\}$ of training event vectors. Denote by $n_0(q_{k-1}, q_k]$ the number of modeling event vectors in Ψ_k that are associated with $\mathrm{c}_b = 0$, and by $n_1(q_{k-1}, q_k] = |\Psi_k| - n_0(q_{k-1}, q_k]$ the number associated with $\mathrm{c}_b = 1$. Also let

$$\begin{aligned} L_0\big(q_{k-1}, q_k\big] &= n_0\big(q_{k-1}, q_k\big] \log n_0\big(q_{k-1}, q_k\big] \\ L_1\big(q_{k-1}, q_k\big] &= n_1\big(q_{k-1}, q_k\big] \log n_1\big(q_{k-1}, q_k\big] \\ L\big(q_{k-1}, q_k\big] &= \big|\Psi_k\big| \log \big|\Psi_k\big| \, . \end{aligned} \tag{8.29}$$

When working with the training set V and using the notations above, the minimization problem of Eq. (8.27) becomes

$$\hat{\mathbf{q}} = \arg \min_{\mathbf{q} \in Q(\bar{u}, K)} \sum_{k=1}^{K} \Big(L\big(q_{k-1}, q_k\big] - L_0\big(q_{k-1}, q_k\big] - L_1\big(q_{k-1}, q_k\big]\Big) \, . \tag{8.30}$$

The optimal K-level context quantizer $\hat{\mathbf{q}}$ as given by Eq. (8.30) can be efficiently computed by observing the following recursion:

$$
\begin{aligned}
&\min_{\mathbf{q}\in Q(r,j)} \sum_{k=1}^{j} \big(L\big(q_{k-1}, q_k\big] - L_0\big(q_{k-1}, q_k\big] - L_1\big(q_{k-1}, q_k\big]\big) \\
&= \min_{\tau<r} \Bigg\{ \min_{\mathbf{q}\in Q(\tau,j-1)} \sum_{k=1}^{j-1} \big(L\big(q_{k-1}, q_k\big] - L_0\big(q_{k-1}, q_k\big] - L_1\big(q_{k-1}, q_k\big]\big) \\
&\quad + L(\tau, r] - L_0(\tau, r] - L_1(\tau, r] \Bigg\} . \qquad (8.31)
\end{aligned}
$$

The recursion means that the solution for the problem of size j can be constructed on the solutions of subproblems of size $j - 1$. Because of this property (called the principle of optimality, in optimization literature), we can use a straightforward dynamic programming algorithm to solve Eq. (8.30). The primitive operations in the dynamic programming process are those in Eq. (8.29). We can precompute and store L_0, L_1, and L for all possible subsets in $O(|V|^2)$ time. The expensive logarithmic computations can be done via table lookup. After the preprocessing, the dynamic programming algorithm takes $O(K|V|^2)$ time.

8.9 Fast Algorithms for High-Order Context Modeling

High-order context modeling is indispensable for good rate-distortion performance of wavelet image coders. But if care is not taken in algorithm design and implementation, the formation of high-order modeling contexts can be both CPU and memory intensive, creating a computation bottleneck for wavelet coding systems. Indeed, our earlier research prototype of ECECOW, a high-order embedded conditional entropy coder of wavelet coefficients, spent 70% of its execution time on context modeling. It is unacceptable for most applications that a module of a wavelet image codec is six times more expensive than the wavelet transform itself. In this section, we focus on the operational aspects of high-order statistical context modeling and introduce some fast algorithmic techniques that can drastically reduce both time and space complexities of high-order context modeling in the wavelet domain.

Two computationally intensive parts in context formation are the linear combination Eq. (8.10) of neighboring samples and the texture pattern extraction Eq. (8.12) from neighboring samples. Once Δ is computed, its quantization is very fast via table lookup. Although Eq. (8.10) and Eq. (8.12) involve only basic arithmetic and logic operations — namely additions, comparisons, and bit manipulations — straightforward computations of Eq. (8.10) and Eq. (8.12) require a large number of operations per binary symbol. Furthermore, forming a high-order context that spans over several scan lines needs to access data (modeling events) stored in distant memory locations. This activity can cause excessive cache misses on modern hardware architecture, slowing down the computation. The high computational complexity is seemingly inherent in high-order context modeling. In order to speed up context formation we have to question if the computational complexity of statistical context modeling is necessarily proportional to the order of the model. The answer is pleasantly, if somewhat surprisingly "no," as we will see shortly.

8.9.1 Context Formation via Convolution

By tracing the major causes of high computational complexity, we come to the following key observation. High-order modeling contexts for neighboring samples have large overlaps in the wavelet domain. This means that samples are accessed and operated on repetitively. We can improve computational efficiency by eliminating repetitive arithmetic, logic, and memory operations in spatially overlapped modeling contexts. This idea leads to an incremental algorithm to compute Δ in $O(1)$ time independent of the order of modeling context. Given that the wavelet coefficients are coded in raster scan order at a given bit plane and in a given subband, we denote the coefficient vector in the current row by $x_0[t]$, where $t = 0, 1, \ldots$ represents spatial locations. The coefficient vector in the next row is denoted by $x_1[t]$, and likewise in the previous two rows by $x_{-1}[t]$ and $x_{-2}[t]$, respectively. The x values are up to the current decoded precision in bit plane coding, i.e., in the notations of previous sections, $x_{-2}[\cdot]$, $x_{-1}[\cdot]$, and $x_0[\tau]$, $\tau < t$, are $\mathrm{c}_{..b}$, while $x_0[\tau]$, $\tau \geq t$, and $x_1[\cdot]$ are $\mathrm{c}_{..b+1}$. We drop the subscripts for bit ranges $..b$, $..b + 1$ because they are clearly implied in spatial locations of the wavelet coefficients x.

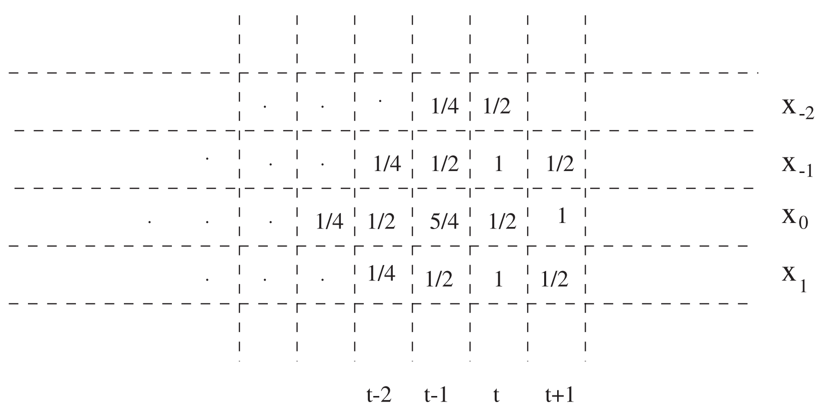

FIGURE 8.9
Convolution kernel effected by the incremental Δ computation of Eq. (8.32).

In sequential coding of $x_0[t]$ for increasing t, we compute incrementally

$$
\begin{aligned}
\alpha_t &= x_{-1}[t+1] + x_1[t+1] \\
\beta_t &= \frac{\beta_{t-1}}{2} + \alpha_{t-1} + x_0[t-1] + x_0[t+1] + \frac{x_{-2}[t]}{2} \\
\Delta_t &= \frac{\alpha_t}{2} + \beta_t \, .
\end{aligned}
\tag{8.32}
$$

Expanding the recursion above reveals

$$
\begin{aligned}
&\Delta_t = x_{-1}[t] + x_1[t] + x_0[t-1] + x_0[t+1] \\
&+ \frac{x_0[t]}{2} \frac{x_{-2}[t] + x_{-1}[t-1] + x_{-1}[t+1] + x_1[t-1] + x_1[t+1] + x_0[t-2]}{2} \\
&+ \frac{\beta_{t-2} + x_{-2}[t-1]}{4} \, .
\end{aligned}
\tag{8.33}
$$

This corresponds to a high-order linear filter whose kernel is graphically depicted in Fig. 8.9. Note that Fig. 8.9 illustrates only the part of the filter kernel with coefficients larger than $1/4$ — the 14 most important modeling events with respect to $x_0[t]$. We can see that Δ_t is a weighted sum of N, W, S, E, NW, NE, SW, SE, NN, WW, and many other past observations with the weights proportional to their distances to $x_0[t]$. Therefore, Δ_t offers a modeling context of $x_0[t]$ of order higher than 14. But Δ_t can be computed by the incremental algorithm of Eq. (8.32) in only six additions, five memory accesses, and two bit shifts — less than half the number of operations required by a direct implementation of Eq. (8.10). In fact, the computational complexity of the proposed incremental algorithm is independent of the order of modeling contexts. Indeed, we can rewrite the second line of Eq. (8.32) as

$$
\beta_t = \lambda \beta_{t-1} + \cdots \tag{8.34}
$$

where λ is a forgetting factor. Increasing λ gives higher weights to the past observations and hence increases the order of modeling context. Therefore, we derived an $O(1)$ time algorithm for computing Δ_t that can increase the order of context modeling for a fixed number of operations. The optimal value of λ is determined by the length of memory in the source. In practice, we empirically found that $\lambda = 1/2$ gave very close to optimal compression results on natural images while avoiding divisions. The incremental computations of Eq. (8.32) have a simple convolution structure, and hence particularly suitable for hardware implementation.

8.9.2 Shared Modeling Context for Signs and Textures

Next we consider efficient computations of Eq. (8.12) and Eq. (8.9) and introduce algorithmic techniques to greatly reduce the amount of computation and memory accesses to set up spatial texture patterns T_b and sign contexts. As we did for Δ, we dropped the references to parent subband in T_b, i.e., not using t_4 in Eq. (8.12). Then there are four status bits $t_3 t_2 t_1 t_0$ to be set depending on the outcomes of four comparisons between $\text{C}_{..b+1}$ and $\text{N}_{..b}$, $\text{W}_{..b}$, $\text{S}_{..b+1}$, and $\text{E}_{..b+1}$. By a careful organization of computations in Eq. (8.12) and Eq. (8.9), we can save the comparison and bit setting operations. The basic idea is to let sign modeling and texture modeling share as much context information as possible.

For each coefficient C, we introduce a *syndrome byte* $S_b = s_7 s_6 \cdots s_1 s_0$:

$$\begin{aligned} s_0 &= \text{N}_{..b} > 0?1:0\,, \quad s_4 = \tilde{\text{N}}\,; \\ s_1 &= \text{W}_{..b} > 0?1:0, \quad s_5 = \tilde{\text{W}}\,; \\ s_2 &= \text{S}_{..b+1} > 0?1:0, \quad s_6 = \tilde{\text{S}}\,; \\ s_3 &= \text{E}_{..b+1} > 0?1:0; \quad s_7 = \tilde{\text{E}}\,. \end{aligned} \tag{8.35}$$

where N, W, S, E are the four neighbors of C. Syndrome bytes for all coefficients are initialized to 0 and updated if necessary for decreasing bit planes. Syndrome bytes S_b support context modeling of signs by allowing three dynamic states of signs in embedded bit plane coding. Specifically, in S_b if bit $s_i = 0$, $i = 0, 1, 2, 3$, then the status bit s_{i+4} is not used or is only a “don’t care” bit (although the bit is physically set to 0 at initialization). In sign modeling for $x_0[t]$, the coder needs to fetch only the syndrome byte S_b of $x_0[t]$ and then uses it as the modeling context for signs.

Note that in the embedded bit plane coding, the most significant bit and the sign of a wavelet coefficient are set at the same time. It is then immediate from Eq. (8.12) and Eq. (8.35) that

$$T_b = t_3 t_2 t_1 t_0 = s_3 s_2 s_1 s_0 = S_b, \quad \text{if } \text{C}_{..b+1} = 0. \tag{8.36}$$

Therefore, syndrome byte S_b can be used not only directly for modeling signs, but also for modeling textures. The entropy coder simply extracts the last four bits of S_b and sets texture pattern $T_b = s_3 s_2 s_1 s_0$, if $\text{C}_{..b+1} = 0$. All the computations of Eq. (8.12) become unnecessary and can be saved.

Each bit in syndrome byte S_b for $x_0[t]$ is set at most once. Two bits, s_i and s_{i+4}, are set when the most significant bit and the sign of one of the four neighbors N, W, S, E of $x_0[t]$ are scanned and coded. This means that the sign of a coefficient will never be accessed and examined more than once in embedded bit plane coding. Likewise, no coefficients will be accessed and tested more than once for their significance in setting T_b for decreasing b. The proposed algorithm has therefore minimized the number of arithmetic operations and memory accesses in context formation. This optimality in time complexity is achieved by eliminating repetitive computations in spatially overlapped contexts, and it is operationally realized by the use of syndrome bytes S_b. Clearly, the number of syndrome bytes is the same as the number of wavelet coefficients in the buffer to be coded. This working memory is very modest in size and well justified by the great savings in computation.

Before leaving the subject discussed above, we would like to point out that the JPEG 2000 verification model also uses a collection of status bits for each wavelet coefficient which have similar roles as the syndrome bytes [29].

The algorithmic techniques for fast context formation presented above have led to an efficient wavelet-based image coder [38] that is 20% faster than the popular SPIHT image coder (its arithmetic coded version), and at the same time it outperforms SPIHT in rate-distortion performance, as we see in the following section.

8.10 Experimental Results

In order to demonstrate the effects of different techniques for context modeling and conditional entropy coding of wavelet coefficients on coding efficiency, we present compression results of some well-known and recently published wavelet image compression algorithms and compare them with the algorithms that are described in this chapter. We evaluate both lossy and lossless compression performance of these wavelet image coders.

8.10.1 Lossy Case

For the sake of common references, we use in our evaluation two JPEG test images, "Lena" and "Barbara," that are widely used for rate-distortion comparisons in the image compression literature. Image qualities, measured by PSNR, of various wavelet image coding algorithms at different bit rates are tabulated in Tables 8.1 and 8.2. In the tables, EZW and SPIHT algorithms are well-known and were introduced earlier in this chapter. SFQ is the space-frequency quantization method by Xiong, Ramchandran, and Orchard [40], EQ is an *estimation-quantization* method by LoPresto, Ramchandran, and Orchard [16], and C/B is a context-based entropy coding method by Chrysafis and Ortega [9]. ECECOW is a coder based on techniques presented in Sections 8.5 and 8.6, and also in Wu [36]. The best results were obtained by replacing the context quantizer of ECECOW with the context quantization scheme guided by Fisher's linear discriminant and via dynamic programming [37], as described in Sections 8.7 and 8.8. This algorithm is identified as "NEW" in the tables.

Table 8.1 Rate(bpp)/PSNR(dB) Results for "Lena"

rate	EZW	SPIHT	SFQ	EQ	C/B	ECECOW	NEW
0.25	33.17	34.13	34.33	34.57	34.31	34.81	34.89
0.50	36.28	37.24	37.36	37.68	37.52	37.92	38.02
1.00	39.55	40.45	40.52	40.88	40.80	40.85	41.01

The NEW method outperforms all others in terms of rate-distortion performance, although by smaller margins against ECECOW. We need to stress that the good performances of the NEW and ECECOW methods are solely due to high-order adaptive context modeling. In our experiments, both coders used dyadic wavelet transform of the popular bi-orthogonal 9/7 filter [17]. Neither filter kernel nor coefficient quantizer was optimized for specific images. On the other hand, some of the other methods in

Table 8.2 Rate(bpp)/PSNR(dB) Results for "Barbara"

rate	EZW	SPIHT	SFQ	C/B	ECECOW	NEW
0.25	26.77	27.57	27.20	28.48	28.85	29.21
0.50	30.53	31.39	31.33	32.63	32.69	33.06
1.00	35.14	36.41	36.96	37.61	37.65	38.05

our comparison group used much longer filter kernels. Furthermore, ECECOW is an embedded coding scheme, like EZW and SPIHT, while SFQ, EQ, and C/B are not. Our experimental results clearly demonstrate the importance of high-order context modeling in compressing wavelet coefficients.

8.10.2 Lossless Case

Since entropy coding of coefficients is independent of wavelet transforms and coefficient quantization, the NEW method can be readily applied to invertible wavelet transforms [8] for lossless image compression. Invertible wavelet transforms map integer pixel values to integer wavelet coefficients. Thus no coefficient quantization is required prior to coefficient coding. Because of the absence of quantization distortion, entropy decoding of wavelet coefficients followed by inverse transform leads to lossless reconstruction. Table 8.3 compares the lossless compression performance of the NEW method with other state-of-art lossless image coders on an ISO set of test images. In the comparison group JPEG-LS is the new lossless JPEG standard [6]. CALIC is a well-known lossless image coder that seems to have the best compression performance among practical lossless image coders [34, 35]. But one needs to keep in mind that both JPEG-LS and CALIC are predictive coding schemes without progressive transmission capability. The S+P algorithm, by Said and Pearlman [26], is a pioneer work in wavelet-based, embedded lossless image compression. On average, the NEW method obtains only about 2% less lossless compression than CALIC, but it outperforms all others.

8.11 Summary

For typical transform-based signal compression systems, data reduction is mostly achieved by entropy coding of transform coefficients. If context-based adaptive arithmetic coding is used to compress the transform coefficients, then the pivotal issue that determines the compression performance is statistical context (Markov) modeling of the coefficients, or, more specifically, how to estimate the underlying conditional probability of the coefficients. In this chapter, we introduced a number of modern techniques for context modeling and adaptive entropy coding of wavelet coefficients.

Table 8.3 Lossless Rates (bits/pixel) of ECECOW Compared with Other Lossless Image Coders on an ISO Set of Test Images

Image	NEW	ECECOW	S+P	CALIC	JPEG-LS
Balloon	2.85	2.86	2.97	2.78	2.90
Barb 1	4.30	4.34	4.53	4.33	4.69
Zelda	3.69	3.71	3.84	3.72	3.89
Hotel	4.36	4.38	4.53	4.22	4.38
Barb 2	4.53	4.57	4.71	4.49	4.69
Board	3.61	3.62	3.82	3.50	3.68
Girl	3.79	3.81	3.96	3.71	3.93
Gold	4.41	4.42	4.56	4.38	4.48
Boats	3.85	3.86	4.03	3.77	3.93
Lena	4.07	4.09	4.16	4.04	4.24
Average	**3.96**	**3.98**	**4.12**	**3.89**	**4.08**

These techniques are used in some state-of-the-art wavelet image codecs. Although the chapter mostly relates to wavelet-based image compression for the concreteness of the discussions, the principles and techniques described here can be used in conjunction with other transforms, such as DCT, and are also readily applicable to compression of other types of signals, such as audio and video.

References

[1] Criminal Justice Information Services, *WSQ Gray-Scale Fingerprint Image Compression Specification* (ver. 2.0), Federal Bureau of Investigation, Feb. 1993.

[2] ISO/IEC WD15444-1, JPEG 2000 — Lossless and lossy compression of continuous-tone and bi-level still images, ISO, Dec. 1999.

[3] ISO/IEC JTC 1/SC 29/WG 1 WD14495, JPEG LS image coding system, July, 1996.

[4] ISO/IEC JTC1 CD 11172, Coding of moving pictures and associated audio for digital storage media up to 1.5 Mbits/s, ISO, 1992.

[5] ISO/IEC JTC1 CD 13818, Generic coding of moving pictures and associated audio, ISO, 1994.

[6] ISO/IEC JTC 1/SC 29/WG 1, JPEG LS image coding system, *ISO Working Document ISO/IEC JTC1/SC29/WG1 N399 - WD14495*, July 1996.

[7] Arps, R.B. and Truong, T., Comparison of international standards for lossless still image compression, *Proc. of the IEEE,* 82(6), 889–899, 1994.

[8] Calderbank, A.R., Daubechies, I., Sweldens, W., and Yeo, B.L., Wavelet transforms that map integers to integers, *J. Applied and Computational Harmonic Analysis,* 5(3), 332–369, 1998.

[9] Chrysafis, C. and Ortega, A., Efficient context-based entropy coding for lossy wavelet image compression, *Proc. 1997 Data Compression Conference,* 241–250, Mar. 1997.

[10] Cover, T.M. and Thomas, J.A., *Elements of Information Theory*, John Wiley & Sons, New York, 1991.

[11] DeVore, R.A., Jawerth, B.J., and Lucier, B.J., Image compression through wavelet transform coding, *IEEE Trans. Info. Theory*, 38(2), 719–746, 1992.

[12] Fisher, R.A., The use of multiple measurements in taxonomic problems, in *Contributions to Mathematical Statistics*, John Wiley & Sons, New York, 1950.

[13] Garey, M.R. and Johnson, D.S., *Computers and Intractability, A Guide to the Theory of NP-Completeness,* Freeman, New York, 1979.

[14] Gersho, A. and Gray, R.M., *Vector Quantization and Signal Compression,* Kluwer Academic Publishers, Boston, 1992.

[15] Loeve, M., *Probability Theory,* 2nd ed., Van Nostrand Reinhold, Princeton, NJ, 478, 1960.

[16] LoPresto, S.M., Ramchandran, K., and Orchard, M.T., Image coding based on mixture modeling of wavelet coefficients and a fast estimation-quantization framework, *Proc. 1997 Data Compression Conf.,* 241–250, Mar. 1997.

[17] Mallat, S., A theory for multiresolution signal decomposition: a wavelet representation, *IEEE Trans. Patt. Anal. Machine Intelligence*, 11(7), 674–693, 1989.

[18] Pearlman, W.A., Kim, B.-J., and Xiong, Z., Embedded video subband coding with 3D SPIHT, *Wavelet Image and Video Compression,* Topiwala, P., Ed., Kluwer Academic Publishers, Boston, 1998.

[19] Pennebaker, W.B. and Mitchell, J.L., *JPEG Still Image Data Compression Standard,* Van Nostrand Reinhold, New York, 1993.

[20] Pennebaker, W.B., Mitchell, J.L., Langdon, G.G., and Arps, R.B., An overview of the basic principles of the Q-coder adaptive binary arithmetic coder, *IBM J. Res. & Devel.,* 32(6), 717–726, 1988.

[21] Rao, K.R. and Hwang, J.J., *Techniques and Standards for Image, Video and Audio Coding,* Prentice Hall, Englewood Cliffs, NJ, 1996.

[22] Rao, K.R. and Yip, P., *Discrete Cosine Transform,* Academic Press, New York, 1990.

[23] Rissanen, J. and Langdon, G.G., Universal modeling and coding, *IEEE Trans. Info. Theory,* 27, 12–23, 1981.

[24] Rissanen, J., Universal coding, information, prediction, and estimation, *IEEE Trans. Info. Theory,* 30, 629–636, 1984.

[25] Said, A. and Pearlman, W.A., New, fast, and efficient image codec based on set partitioning in hierarchical trees, *IEEE Trans. Circ. & Sys. Video Tech.,* 6(3), 243–249, June 1996.

[26] Said, A. and Pearlman, W.A., An image multiresolution representation for lossless and lossy compression, *IEEE Trans. on Image Proc.,* 5(9), 1303–1310, 1996.

[27] Shannon, C.E., A mathematical theory of communication, *Bell Syst. Tech. J.,* 27, 379–423, 623–656, 1948.

[28] Shapiro, J.M., Embedded image coding using zerotrees of wavelet coefficients, *IEEE Trans. Signal Processing,* 41(12), 3445–3462, 1993.

[29] Taubman, D., EBCOT: Embedded block coding with optimized truncation, ISO/IEC JTC 1/SC 29/WG 1, No. 1020, Oct. 1998.

[30] Taubman, D. and Zakhor, A., Multirate 3-D subband coding of video, *IEEE Trans. Image Processing,* 3(5), 572–588, 1994.

[31] Witten, I.H., Neal, R.M., and Cleary, J.G., Arithmetic coding for data compression, *Communications of the ACM,* 30, 520–540, 1987.

[32] Wood, J.W. and O'Neil, S.D., Subband coding of images, *IEEE Trans. Acoustics, Speech, and Signal Processing,* 34(5), 1278–1288, 1986.

[33] Wu, X., Optimal quantization by matrix-searching, *J. Algorithms,* 12(4), 663–673, 1991.

[34] Wu, X., Lossless compression of continuous-tone images via context selection and quantization, *IEEE Trans. on Image Proc.,* 6(5), 656–664, 1996.

[35] Wu, X. and Memon, N.D., Context-based adaptive lossless image coding, *IEEE Trans. Comm.,* 45(4), 437–444, 1997.

[36] Wu, X., High-order context modeling and embedded conditional entropy coding of wavelet coefficients for image compression, *Proc. of 31st Asilomar Conf. on Signals, Systems, and Computers,* 1378–1382, 1997.

[37] Wu, X., Context quantization with fisher discriminant for adaptive embedded wavelet image coding, *Proc. of 1999 Data Compression Conference,* 102–111, Mar. 1999.

[38] Wu, X., Low complexity high-order context modeling of embedded wavelet bit streams, *Proc. of 1999 Data Compression Conference,* 112–120, Mar. 1999.

[39] Wu, X. and Xiong, Z., An empirical study of high-order context modeling and entropy coding of wavelet coefficients, ISO/IEC JTC 1/SC 29/WG 1, No. 771, Feb. 1998.

[40] Xiong, Z., Ramchandran, K., and Orchard, M.T., Space frequency quantization for wavelet image coding, *IEEE Trans. Image Processing,* 6, 677–693, 1997.

[41] Xiong, Z. and Wu, X., Wavelet image coding using trellis coded space-frequency quantization, *IEEE Signal Processing Letters,* 6(7), 158–161, 1999.

[42] Xiong, Z., Wu, X., Yun, D.Y., and Pearlman, W.A., Progressive coding of medical volumetric data using three-dimensional integer wavelet packet transform, *Proc. of 1998 IEEE Workshop on Multimedia Signal Processing,* 553–558, Dec. 1998.

[43] Zandi, A., Allen, J.D., Schwartz, E.L., and Boliek, M., CREW: compression by reversible embedded wavelets, *Proc. of Data Compression Conf.,* 212–221, IEEE Press, Piscataway, NJ, 1995.