

TESI DI DOTTORATO

UNIVERSITÀ DEGLI STUDI DI NAPOLI “FEDERICO II”

DIPARTIMENTO DI INGEGNERIA ELETTRONICA  
E DELLE TELECOMUNICAZIONI

DOTTORATO DI RICERCA IN  
INGEGNERIA ELETTRONICA E DELLE TELECOMUNICAZIONI

---

**THE PROBLEM  
OF CONTOURS  
IN TRANSFORM IMAGE CODING**

---

**SARA PARRILLI**

Il Coordinatore del Corso di Dottorato  
Ch.mo Prof. Giovanni POGGI

Tutori  
Ch.mo Prof. Luisa VERDOLIVA  
Ch.mo Prof. Giovanni POGGI

A. A. 2007–2008



# Contents

<b>Contents</b>	<b>iv</b>
<b>List of Figures</b>	<b>vii</b>
<b>Introduction</b>	<b>ix</b>
<b>1 Contour coding problem</b>	<b>1</b>
1.1 Transform coding . . . . .	1
1.2 From KLT to JPEG . . . . .	4
1.3 Wavelet transform . . . . .	5
1.4 Wavelet and nonlinear approximation . . . . .	10
1.5 Wavelet limits and new approaches . . . . .	14
<b>2 Object-based coding</b>	<b>19</b>
2.1 Object-based coding . . . . .	19
2.2 The coding scheme . . . . .	21
2.3 Measurement of losses . . . . .	23
2.3.1 Methodology . . . . .	23
2.3.2 Experimental results . . . . .	26
2.4 Measurement of gains . . . . .	33
2.4.1 Methodology . . . . .	33
2.4.2 Experimental results . . . . .	35
2.5 Performance with real-world images . . . . .	37
2.6 Multispectral images . . . . .	43
2.6.1 Class and region-based approaches . . . . .	43
2.6.2 Class-based scheme . . . . .	45
2.6.3 Region-based scheme . . . . .	52

---

<b>3</b>	<b>Coding with contourlets</b>	<b>57</b>
3.1	Contourlets . . . . .	57
3.2	Hybrid contourlet-wavelet decomposition . . . . .	62
3.3	A sphit-like algorithm for Contourlets . . . . .	66
3.3.1	Experimental Results . . . . .	68
<b>4</b>	<b>Distortion evaluation in ALS</b>	<b>75</b>
4.1	Adaptive lifting scheme . . . . .	75
4.1.1	Adaptive update lifting scheme . . . . .	77
4.1.2	Adaptive prediction lifting scheme . . . . .	79
4.2	Distortion evaluation problem for adaptive lifting . . . . .	82
4.3	Computation of polyphase matrices . . . . .	83
4.3.1	Adaptive update lifting scheme . . . . .	83
4.3.2	Adaptive prediction lifting scheme . . . . .	85
4.4	Weight computation . . . . .	86
4.4.1	One level of decomposition . . . . .	86
4.4.2	Multiple levels of decomposition . . . . .	88
4.5	Extension to multi-dimensional case . . . . .	90
4.6	Experimental results . . . . .	95
4.6.1	Test Lifting Schemes . . . . .	95
4.6.2	Distortion evaluation in transformed domain . . . . .	98
4.6.3	Bit-rate allocation . . . . .	99
	<b>Conclusions</b>	<b>105</b>

# List of Figures

1.1	Transform coding scheme. . . . .	2
1.2	Example of mother wavelet. . . . .	6
1.3	Example of rescaled mother wavelet. . . . .	7
1.4	Filter bank. . . . .	8
1.5	Three levels wavelet decomposition of image Lena. . . . .	9
1.6	EZW and SPIHT trees of coefficients. . . . .	10
1.7	Approximation using Fourier basis. . . . .	12
1.8	Approximation using wavelet basis. . . . .	13
1.9	2-D wavelet transform of the image Barbara. . . . .	15
1.10	A simple model of image. . . . .	16
1.11	Example of anisotropy and directionality importance. . . . .	16
2.1	The object-based coding scheme under investigation. . . . .	22
2.2	RD curves for flat (red) and object-based (blue) coders. . . . .	25
2.3	Test images from the USC database. . . . .	26
2.4	Some maps used in the experiments. . . . .	27
2.5	RD performance with square-tile segmentation. . . . .	28
2.6	RD performance with rectangular-tile segmentation. . . . .	29
2.7	RD performance with wavy-tile segmentation. . . . .	30
2.8	RD performance with out-of-context segmentation maps. . . . .	31
2.9	Object overlapping in the transform domain. . . . .	34
2.10	RD performance with fitting SA-WT. . . . .	35
2.11	Some mosaics used in the experiments. . . . .	36
2.12	PSNR gain for wavy-tile mosaics. . . . .	37
2.13	PSNR gain for square-tile mosaics. . . . .	38
2.14	Segmentation maps for image Peppers. . . . .	39
2.15	RD performance of object-based coding for image Peppers. . . . .	40
2.16	Image Peppers compressed at 0.05. . . . .	41
2.17	RD perf. of BISK and SA-SPIHT for Peppers. . . . .	42

---

2.18	RD perf. of BISK and SA-SPIHT with A.C. for Peppers. . . . .	42
2.19	An example image and its segmentation maps . . . . .	44
2.20	The original class-based coding scheme. . . . .	45
2.21	The Landsat TM test image. . . . .	46
2.22	RD performance of the original DCT-based coder. . . . .	47
2.23	RD performance of the new wavelet-based coder. . . . .	49
2.24	Comparison wavelet-based and DCT-based CBC. . . . .	50
2.25	Comparison wavelet-based CBC and various reference coders. . . . .	51
2.26	RD curves for the AVIRIS image. . . . .	52
2.27	AVIRIS image, false colors . . . . .	54
2.28	Performance of test techniques on the TM (Lisbon). . . . .	55
2.29	Performance of test techniques on the AVIRIS (Moffett-field). . . . .	56
3.1	Laplacian pyramid. . . . .	58
3.2	Laplacian decomposition of image Lena. . . . .	58
3.3	Simple reconstruction scheme for the Laplacian pyramid. . . . .	59
3.4	Do-Vetterli reconstruction scheme for the Laplacian pyramid. . . . .	59
3.5	DFB frequency partitioning with $l = 3$ . . . . .	60
3.6	QFB with fan filter . . . . .	61
3.7	Contourlet filter bank. . . . .	62
3.8	Test images. . . . .	63
3.9	NLA with wavelets and contourlets for the image Barbara. . . . .	64
3.10	NLA with wavelets and contourlets for the image Lena. . . . .	64
3.11	NLA with wavelets and contourlets for the image Goldhill. . . . .	65
3.12	NLA with wavelets and contourlets for the image Bench. . . . .	65
3.13	Wavelet-contourlet case with 8 directions. . . . .	67
3.14	Contourlet-contourlet with 4-4 directions. . . . .	68
3.15	Contourlet-contourlet case with 4-8 directions. . . . .	69
3.16	The HWD-half transform. . . . .	70
3.17	Rate-distortion performance for the image Barbara. . . . .	71
3.18	Rate-distortion performance for the image Lena. . . . .	71
3.19	Rate-distortion performance for the image Goldhill. . . . .	72
3.20	Rate-distortion performance for the image Bench. . . . .	72
3.21	Details of reconstructed images. . . . .	73
3.22	HWD results for barbara. . . . .	74
4.1	Classical lifting scheme. Analysis. . . . .	76
4.2	Classical lifting scheme. Synthesis. . . . .	77
4.3	Lifting scheme with adaptive update first. . . . .	78

---

4.4	Lifting scheme with update first and adaptive prediction. . . .	80
4.5	Predictor selection at an ideal step edge. . . . .	80
4.6	Lifting scheme with update first and adaptive prediction. . . .	81
4.7	Equivalent filter bank for synthesis ALS. . . . .	83
4.8	Structure of the matrix $\mathbf{G}_{00}$ . AULS. . . . .	84
4.9	Structure of $\mathbf{G}_{01}$ . AULS. . . . .	85
4.10	Structure of the matrix $\mathbf{G}_{00}$ . APLS. . . . .	86
4.11	Structure of the matrix $\mathbf{G}_{01}$ . APLS . . . . .	86
4.12	The bi-dimensional signal $x$ represented via four channels. . .	90
4.13	2D AULS. . . . .	92
4.14	Theoretical 2D APLS. . . . .	93
4.15	Claypoole inspired 2D APLS. . . . .	95
4.16	RD curves for Lena with and without weights. AULS A. . . .	100
4.17	RD curves for Lena with and without weights. AULS B. . . .	101
4.18	RD curves for Lena with and without weights. AULS C. . . .	102
4.19	RD curves for Lena with and without weights. AULS CS. . . .	103
4.20	RD curves for Lena with and without weights. APLS. . . . .	104



# Introduction

**W**e live in the image society! This short sentence does not introduce a sociological analysis on the importance of look in modern times, but it wants to underline the change in our way to communicate. We live in an age in which text is gradually losing ground to image, video and sound. The new generations slogan could be “Say it with a multimedia content”.

The ‘image society’ causes a growing traffic of multimedia contents which travel on the data networks and often on wireless networks. This calls for new tools to provide a faithful and efficient representation of images and video, especially at low bit rates. As a consequence, in recent years there has been intense research on image compression, new concepts and tools have been generated, and new standards have been output, such as the wavelet-based JPEG2000 which improves upon JPEG especially at low bit rates. However, despite its significant improvements, JPEG2000 performance is still quite far from theoretical compression bounds. If we consider that the human eye can receive only 20 bits per second, as psychology studies show [1], and it typically takes only a few seconds to capture a visual scene, the bits needed to represent an image should not exceed a few hundreds. By contrast, JPEG2000, which represents the state of art, requires at least some tens of kilobytes to represent a typical  $512 \times 512$ -pixel image. This suggests that there is still a long way to go before technology reaches Nature.

The classical scheme of image compression is based on three steps: transform, quantization and lossless coding. Recently, research efforts have focused on the choice of the transform that best represents a natural image. As a matter of fact, in spite of its great success, wavelet transform is not the optimal basis for an image. Indeed, it is very effective in representing smooth signals with pointwise discontinuities (like an image row), but fails in representing discontinuities along curves, like the contours between neighboring visual objects, which typically characterize images [2].

The problem of contours in image compression is the leading thread of

this thesis work, as well as wavelet inefficiency in describing this piece of information. This is a very important issue in image compression because contours are the most resource-consuming elements of an image, especially at low bit rates, and their good reproduction is essential to perceive a satisfactory image quality. This very same subject is treated in this PhD thesis work under three different scenarios, that is, three different attempts to overcome wavelet limits on images contours: object-based coding; new directional transforms; and adaptive lifting scheme.

The object-based paradigm is an highly flexible tool to describe images. To begin with, considering an image as composed by objects, and not by pixels, is more intuitive and natural. Object-based coding offers a large number of high level functionalities, for example, the user can choose to decode only objects of interest, or to assign them different coding resources and different error-protection levels. Furthermore an object-based description can be used for subsequent classification tasks. In this thesis, however, we focus on the rate distortion performance of object-based schemes in the compression of the whole image, and analyze costs and advantages of a specific coding scheme. The main assumption is that, with an object-based approach, the wavelet works only on the interior of the objects, that is, almost stationary signals, and can therefore provide near-optimal performance. We consider a specific object-based compression scheme where the main coding tools are the shape-adaptive wavelet transform (SA-WT) proposed by S. Li and W. Li [3], and a shape-adaptive version of SPIHT (SA-SPIHT) [4] which extends the well-known image coder proposed by Said and Pearlman [5] to objects of arbitrary shape. Besides assessing the rate-distortion performance of such an object-based coder in typical situations of interest, our analysis will provide insight about the individual phenomena that contribute to the overall losses and gains in this approach. Then, we extend the object-based paradigm to the class of multispectral images. In this context the object-based scheme can be declined in two cases: class-based and region-based paradigms. The analysis of the rate-distortion performance for both schemes, with reference to remote-sensing images, will prove the potential of object-based paradigms for multispectral images.

New directional transforms represent a more direct solution to wavelet inefficiency on image contours. While in object-based coding the transform remains the wavelet and the intelligence is put on the scheme, here the wits is in the transform. Recent studies have shown that wavelet's inability to adequately describe image contours is due to its separability which (while allowing for a

simple implementation) cuts it away from two fundamental properties: directionality and anisotropy [6]. The new directional transforms try to overcome these limits by adding these characteristics to that of wavelet transform, such as multiresolution, localization and critical sampling. Many transforms have been proposed in the last few years (contourlets [6], directionlets [7], curvelets [2], bandelets [8], etc.), but results are mostly theoretical, while there are only a few actual coding algorithms based on these new tools. In this thesis we propose a SPIHT-like coding algorithm based on the contourlet transform. We choose the contourlet transform [6] for several reasons: it has many desirable characteristics, such as directionality, anisotropy, near-optimal theoretical performance and, unlike other directional transforms, it is easily implemented by a filter bank. Our coding algorithm attempts to exploit the peculiar properties of contourlet transform coefficients, which exhibit various types of dependency, across scales, space and direction. Our strategy for SPIHT trees construction tries to exploit all such links, without introducing a significant complexity overload. Experiments on typical images used in the scientific community show some interesting improvements for images rich in directional details.

The last part of the thesis is dedicated to adaptive lifting schemes. The lifting scheme [9] is an efficient and flexible implementation of the wavelet transform. One of the main advantages of the lifting structure is to provide a totally time domain interpretation of the wavelet transform and this feature makes simpler to design new wavelets and content-adaptive wavelets. Adaptive lifting schemes can be used to deal with the problem of contour representation, for example, by constructing directional wavelets, with the filtering direction chosen according to the local orientation of image edges [10], [11], [12], or changing the filters according to the regularity of input signal [13], [14], [15] in order to utilize different and more fit filters when contours or singularities are encountered. A major problem of adaptive lifting schemes is that they are strongly non-isometric transforms, which bars from computing the distortion in the transform domain. On the other hand, this would be highly desirable in order to perform efficient resource allocation. In this thesis we address this problem proposing a method that allows for a reliable estimation of the distortion in the transform domain. The strategy adopted is based on the observation that, although adaptive lifting schemes are nonlinear operators, they can be considered equivalent to suitable time-varying linear filters, which eventually allows us to generalize the traditional distortion computation methods.

The thesis is organized as follows:

**Chapter 1** presents the basics of transform coding and its relations with classical coding schemes. In particular, the key concepts to understand strengths and weaknesses of the wavelet transform are discussed here.

**Chapter 2** is dedicated to the measure of rate-distortion costs and gains for a wavelet object-based encoder. The analysis of the rate-distortion curves obtained in several experiments provides insight about what performance gains and losses can be expected in various operative conditions. Then two variants of object coding paradigm for multispectral images are presented and their performance analyzed.

**Chapter 3** introduces the main characteristics of contourlet transform and the motivation for an hybrid contourlet-wavelet decomposition. A new image coding technique based on this transform and built upon the well-known SPIHT algorithm is described and numerical and visual quality results are reported to confirm the potential of the proposed technique.

**Chapter 4** addresses the problem of distortion evaluation in the transform domain for adaptive lifting schemes. The analysis is conducted on two interesting classes of lifting schemes. Experiments show that the proposed method allows for a reliable estimation of the distortion in the transform domain which results in improved coding performance.

# Chapter 1

## Contour coding problem

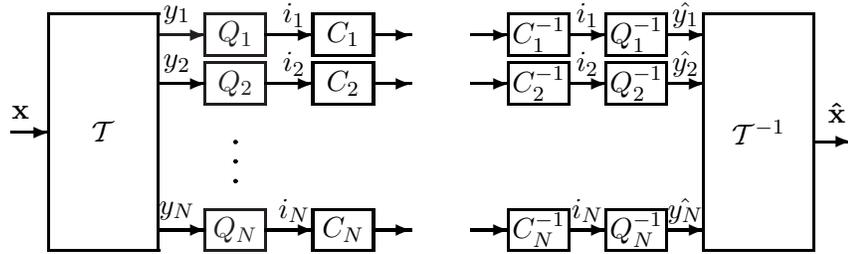
The information about discontinuities between different objects in an image is very important from a visual point of view but often it is hard to compress. In fact, many transform coding algorithms fail in describing efficiently this information. The main problem is that classical transforms do not perform well on image discontinuities. In this chapter we expose the basic principles of transform coding, and its use in image compression algorithms. We focus our attention on wavelet transform underling the reasons of its success and the motivation of its suboptimality.

### 1.1 Transform coding

Transform coding is one of the most successful and widespread technique for lossy compression. Both in audio compression, with MP3, and in image compression, where the discrete cosine transform and wavelets are part of the JPEG standards, transforms are at the heart of the compression engine.

The conventional framework of transform coding was introduced by Huang and Schultheiss [16]. A discrete-time, continuous-valued, vector source with correlated components is given; instead of vector quantization, one uses a linear transform followed by a scalar quantization. An entropy coder, even if it is not present in Huang and Schultheiss paper, is often used to improve the coding efficiency (Fig. 1.1).

Transform coding is an inherently suboptimal source coding technique but it has much lower complexity than vector quantization. In fact, computing the transform  $\mathcal{T}$  requires at most  $N^2$  multiplications and  $N(N - 1)$  additions furthermore, icing on the cake, specially structured transforms like discrete



**Figure 1.1:** Transform coding scheme.

Fourier, cosine, and wavelet transforms are often used to reduce the complexity of this step. The idea behind transform coding is that scalar coding may be more effective in the transform domain than in the original signal space.

There are other two reasons for transform coding: one intuitive and another subjective. The first one is based on the observation that the transform tends to mix the information contained in the different samples so that each transform coefficient contains part of information of all original pixels. In this way it is possible to reconstruct a reliable version of the image even without all coefficients. The subjective reason, instead, starts from studies that state that some biological systems, as human eyes, work in transformed domains.

There are a lot of theorems that define the characteristics of the optimal transform in the ideal framework, but these theorems are based on too restrictive hypotheses [17]. Nevertheless, they have been very important in the transform coding development.

The first and maybe most famous theorem is the one that affirms that Karhunen Loève transform is the optimal transform for gaussian sources. A Karhunen Loève transform (KLT) is a particular type of orthogonal transform that depends on the covariance of the source. Let us introduce formally this transform. Call  $\mathbf{x}$  the input vector, where  $\mathbf{x}$  is assumed to have mean zero, and let  $R_{\mathbf{x}}$  denote the covariance matrix  $R_{\mathbf{x}} = E[\mathbf{x}\mathbf{x}^T]$ , where  $T$  denotes the transpose. Covariance matrices are symmetric, thus orthogonally diagonalizable, so we can write  $R_{\mathbf{x}} = U\Lambda U^T$ , where the diagonal elements of  $\Lambda$  are the eigenvalues of  $R_{\mathbf{x}}$ . The KLT matrix will be  $\mathcal{T} = U^T$ . For gaussian sources it is possible to show that KLT is the optimal orthogonal transform, in the sense that it minimizes the distortion both at high rates [17] and at low rates [18].

The problem is that the hypothesis of gaussian source does not fit well

the images, and without this assumption KLT is not optimal. However it is worth noting that the KLT has the property, not related only to gaussian case, of returning uncorrelated transform coefficients. Since in gaussian hypothesis uncorrelation imply the independence it is natural to wonder if this is the property that we are looking for. Heuristically, independence of transform coefficients seems desirable for a scalar processing like the coding scheme showed in Fig. 1.1. With regard to latter observation we report an interesting theorem [17]:

**Theorem 1** *consider a high-rate transform coding system employing entropy-constrained <sup>1</sup> uniform quantization. A transform with orthogonal rows that produces independent transform coefficients is optimal, when such a transform exists.*

Unfortunately for non gaussian sources there is not always an orthogonal transform that produces independent transform coefficients (without the orthogonality hypothesis the optimality is not assured). Furthermore, even if such transform exists, it is not necessarily optimal when the rate is low. This is a great drawback because low bit-rate coding represents the real challenge for new image compression schemes since at high bit-rates the performance of actual coder are already fulfilling. We will see in Section 1.4 that a totally different theory can be used in this scenario, but it is important to underline that, even in that case, the theoretical results must be considered only as a guideline for real compression. In fact, when we consider transform coding as applied to empirical data, we typically find that a number of simple variations can lead to significant improvements w.r.t. the performance obtained with a strict application of these theorems (see for example sec. 1.2).

Practical transform coders differ from the standard model in many ways, for example transform coefficients are often not entropy coded independently and this eliminates the incentive to have independent transform coefficients. This is what happens in JPEG standard [19], where, as we shall clear in the next section, the transform coefficients with zero quantized values are jointly coded. Actually, this type of joint coding does not eliminate the effectiveness of the KLT, on the contrary, can be seen as an additional motivation for its use. In fact, the property of rendering transformed coefficients totally decorrelated is equivalent to the property of concentrating optimally the coefficient energy, so there are a large fraction of coefficients with small magnitude and this benefits the joint coding. The empirical fact that wavelet transforms have a similar, but

---

<sup>1</sup>The rate is computed from the entropy.

stronger (see Sec. 1.4), property for natural images (or rather, for piecewise smooth functions) is a key to their current popularity.

## 1.2 From KLT to JPEG

To make the KLT approach to block coding operational, two problems need to be addressed: the computational complexity ( $N^2$ ), due to the calculation of eigenvectors of the correlation matrix, and the estimation of correlation matrix from the data. Indeed, to have a good estimation it is necessary that the data are stationary but this is not true for images. A possible strategy is to partitioning the image in blocks in order to have a more stationarity signal, but this mean doing the estimation on a smaller quantity of data, thus a worse estimation. To overcome KLT problems, fast fixed transforms leading to approximate diagonalization of correlation matrices are used. The most popular among these transforms is the discrete cosine transform (DCT).

The discrete cosine transform of a sequence  $x(n)$  of length  $N$  is defined by:

$$y(k) = \sqrt{\frac{2}{N}} \beta(k) \sum_{n=0}^{N-1} x(n) \cos\left(\frac{k(2n+1)\pi}{2N}\right) \quad (1.1)$$

where

$$\beta(k) = \begin{cases} \sqrt{1/2} & \text{for } k = 0 \\ 1 & \text{for } 0 < k \leq N-1 \end{cases}, \quad (1.2)$$

and its inverse is:

$$x(n) = \sqrt{\frac{2}{N}} \sum_{k=0}^{N-1} \beta(k) y(k) \cos\left(\frac{k(2n+1)\pi}{2N}\right) \quad (1.3)$$

The transform for the two-dimensional signals is derived as a product basis from the one-dimensional DCT. While not necessarily best, this is an efficient way to generate a two-dimensional basis.

Since the DCT is equivalent to the discrete Fourier transform (DFT) applied to a symmetric extension of original signal it can be computed with a fast Fourier transform (with about  $N \log N$  operations). This is a key issue: the DCT achieves a good compromise between energy concentration and computational complexity. Therefore, for a given computational budget, it can actually outperform the KLT.

For these reasons DCT transform is used in the most widespread standard for image compression: JPEG (Joint Photographic Experts Group) [19]. The

way of using DCT in JPEG is quite different from what theory suggests, because the source (images) is very different from the theoretical model (gaussian sources). The image is first subdivided into blocks of size  $8 \times 8$  and these blocks are processed independently. Note that blocking the image into independent pieces allows to have a more stationary signal in each block and to adapt the compression to each block individually. The block size choice represents a compromise between the need of compacting energy and the need of limiting the signal variability.

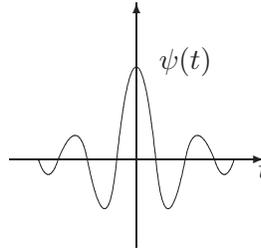
Now, quantization and entropy coding is done in a manner that is quite at variance with the classical setup. First, based on perceptual criteria, the transform coefficients are quantized with a uniform quantizers whose step-sizes, typically, small for low frequencies, and large for high ones, are stored in a quantization matrix. Technically, one could pick different quantization matrices for different blocks, but usually, only a single scale factor is used, that can be adapted depending on the block statistics. Since there is no natural ordering of the two-dimensional DCT plane a prescribed 2D to 1D scanning is used. This so-called zig-zag scan traverses the DCT frequency plane diagonally from low to high frequencies in order to have a sequence with a lot of zeros at the end. For this resulting one-dimensional sequence, nonzero coefficients are entropy-coded, and stretches of zero coefficients are encoded using run lengths coding.

### 1.3 Wavelet transform

In spite of the considerable advantages of compression scheme based on DCT, this technique is not very performing at low bit-rates. As underlined in the previous section, in the JPEG standard it is necessary to divide the image in blocks before to perform the DCT transform, in this way the correlation present on blocks edges is eliminated causing an effect called "blocking artifact", that is very visible at low bit-rates. The evolution of JPEG standard, known as JPEG-2000, is based on the Discrete Wavelet Transform (DWT)[20].

The literature about wavelet is huge and we refer to it ([21, 22, 23, 24] for a detailed exposition, in this context we only want to highlight wavelet characteristics useful for image compression, namely:

- time frequency localization;
- critical sampling;
- easy implementation;



**Figure 1.2:** Example of mother wavelet.

- multiresolution.

To explain the first property we start from continuous wavelet. Unlike Fourier analysis, that describes a function in terms of components relative to a basis made by sine and cosine, wavelet analysis has, as basis functions, the dilations and translations of a single function  $\psi$ , called mother wavelet (Fig. 1.2), having limited energy and zero mean. So the wavelet basis functions are short and oscillating from which the name of *wavelet* (“small waves”).

The mother wavelet  $\psi(t)$  can be dilated (stretched) and translated adjusting two parameters  $a$  (scale parameter) and  $b$  (translation parameter) that characterize it:

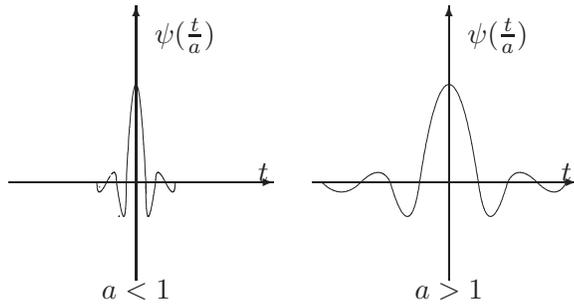
$$\psi_{ab}(t) = \frac{1}{\sqrt{|a|}} \psi\left(\frac{t-b}{a}\right) \quad (1.4)$$

with  $a \in \mathbb{R}^+$  and  $b \in \mathbb{R}$ . A value  $a > 1$  corresponds to a more stretched basis function while a value of  $a < 1$  to a more compressed one (Fig. 1.3).

The continuous Wavelet transform (CWT) of a given signal  $x(t)$  with respect to the mother wavelet  $\psi_{ab}(t)$ , is given by the following definition:

$$CWT_x(a, b) = \int_{-\infty}^{\infty} x(t) \psi_{ab}^*(t) dt = \langle \psi_{ab}(t), x(t) \rangle \quad (1.5)$$

For an assigned couple of parameter  $a$  and  $b$ , the coefficient obtained by relation 1.5 represents how well the signal  $x(t)$  and scaled and translated mother wavelet match. So, if we look at Fig. 1.3, we can notice that when wavelet basis functions are longer ( $a > 1$ ) we are taking into account lower frequencies, while when basis functions are shorter ( $a < 1$ ) we are taking into account higher frequencies. Basically, the wavelets can variate the resolution



**Figure 1.3:** Example of rescaled mother wavelet.

with frequency, in particular it is possible to obtain a time resolution arbitrarily good at high frequencies and a frequency resolution arbitrarily good at low frequencies. This property is called time-frequency localization and it is particularly suitable for natural images, that typically are composed by a mixture of low frequency components of long duration, called *trend* (for example background), and high frequency components of short duration, called *anomalies* (for example contours).

The continuous wavelet transform is highly redundant and is not adapted to compression. To obtain a non redundant representation (the critical sampling that is our second desirable characteristic) is necessary to discretize the parameters. To this regard, the most common discretization is the following:

$$\begin{cases} a = a_0^m & m \in \mathcal{Z} & a_0 > 1 \\ b = nab_0 & n \in \mathcal{Z} & b_0 > 1 \end{cases} \quad (1.6)$$

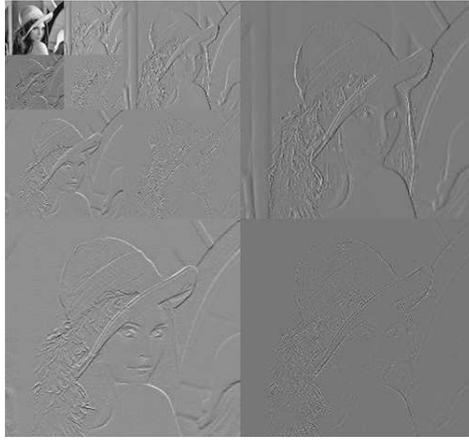
The most widespread wavelet decomposition is the dyadic one (the only considered in this thesis work) achieved setting  $a_0 = 2$  and  $b_0 = 1$ . So the wavelet transform of a signal  $x(t)$  using discrete scale and time parameters is:

$$\langle x(t), \psi_{mn}(t) \rangle = \int_{-\infty}^{\infty} x(t) 2^{-\frac{m}{2}} \psi(2^{-m}t - n) dt \quad (1.7)$$

It is possible to demonstrate that, choosing conveniently  $\psi(t)$ , the reconstruction formula is:

$$x(t) = \sum_n \sum_m \langle x(t), \psi_{mn}(t) \rangle \psi_{mn}(t) \quad (1.8)$$



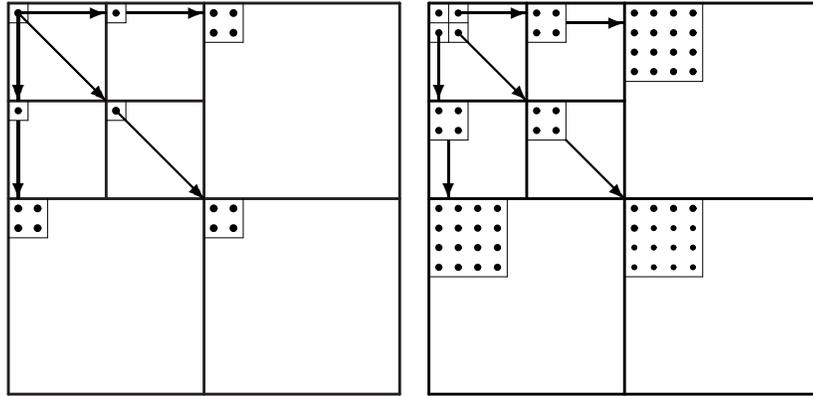


**Figure 1.5:** Three levels wavelet decomposition of image Lena.

standard JPEG2000 [27].

EZW and SPIHT are named *zero-tree* coders and are based on the same principles. They are progressive algorithms in which the most important information, which yields the largest distortion reduction, is transmitted first. This means, supposing the transform orthogonal, that the largest wavelet coefficients are sent before. To realize this approach it is possible to classify the information in bit plane; so the progressive transmission send before the most significant bits of each coefficient. At first step of the algorithm a threshold is fixed and all coefficients are compared with this threshold: the coefficients superior to threshold are labelled as *significant*, the other *insignificant*. Then the information relative to these decisions (the significance map) is encoded, and a refinement pass gives further bits to the coefficients that were already significant at previous step. Finally the threshold is halved and the procedure repeated.

The strategy adopted by EZW and SPIHT for map significance encoding is based on two observations, the first is that images are substantially low pass, and the second that among wavelet subbands there is a sort of *self-similarity* (see Fig. 1.5), due to the fact that the coefficients present in different subbands represent the same spatial locations in the image. So, if there is an insignificant coefficient in the subband at lower frequency, probably, the coefficients relative to the same spatial location, that lie in the subbands at higher frequency, are zero and it is possible to jointly describe all these zero coefficients by a tree structure (Fig. 1.6).



**Figure 1.6:** EZW and SPIHT trees of coefficients.

The coding scheme used in JPEG2000 standard is EBCOT (Embedded Block Coding with Optimized Truncation) and it was originally proposed by Taubman [20]. It is a block coding scheme that generates an embedded bitstream. The block coding is independently performed on non overlapping blocks within individual subbands of same dimension (except for the blocks on the lower or on the right boundaries). EBCOT organizes the bitstream in layers, within each layer each block is coded with a variable number of bits. The partitioning of bits between blocks is obtained using a Lagrangian optimization that dictates the truncation points. The quality of the reproduction is proportional to the number of layers received. The embedded coding scheme is similar in philosophy to the EZW and SPIHT algorithms but the data structure used is different. In fact, EBCOT codes independently blocks that reside entirely within a subband, which precludes the use of trees. Instead EBCOT use a quadtree data structure. As in EZW and SPIHT there is a comparison with a threshold to state the significance of the coefficient, and the algorithm makes multiple passes: significance map encoding and refinement. The bits generated during these procedures are encoded using arithmetic coding.

## 1.4 Wavelet and nonlinear approximation

We already noted that the importance of wavelet in compression is related to a low-rate scenario, in fact DCT and JPEG have good performance for medium

and high rates. In the first section of this chapter we presented some theorems that could be seen as a guideline to choose the transform, but all these theorems are valid only for high-rate compression. An analysis of the low-rate behavior of transform coders has been done by Mallat and Falzon [28] and by Cohen, Daubechies, Guleryuz, and Orchard [29]. In these works, the authors link the low bit-rate performance of a coder to the nonlinear approximation (NLA) behavior of its transform.

Before explaining why NLA and low bit-rate compression are linked, we need to introduce the concepts of linear and nonlinear approximation.

Given a signal  $x$  of a function space  $S$ , it can be represented by a linear combination of the elementary signals  $\{\phi_n\}_{n \in \mathcal{N}}$  that form an orthogonal basis for  $S$ :

$$x = \sum_{n=0}^{\infty} c_n \phi_n, \text{ where } c_n = \langle x, \phi_n \rangle. \quad (1.9)$$

We call linear approximation (LA) the representation of signal that we obtain keeping the first  $M$  components of its basis expansion:

$$\hat{x}_M = \sum_{n=0}^{M-1} c_n \phi_n. \quad (1.10)$$

This is referred to as linear approximation since it is equivalent to projecting the input object onto the subspace spanned by the first  $M$  basis elements.

Instead, we talk of nonlinear approximation when we keep the best  $M$  components<sup>3</sup>:

$$\hat{x}_M = \sum_{n \in I_M} c_n \phi_n, \quad (1.11)$$

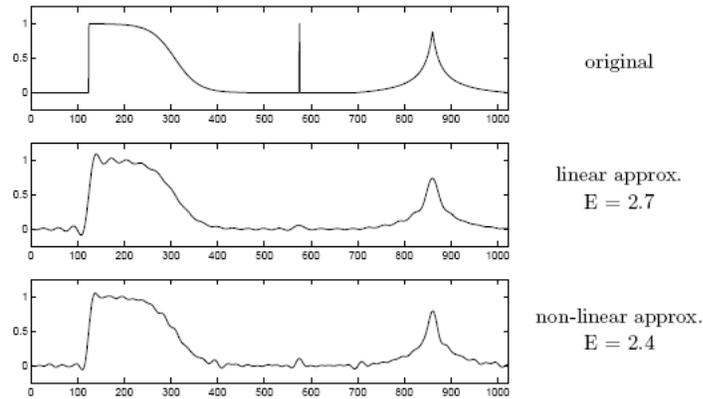
where  $I_M$  is the set of  $M$  indices corresponding to the  $M$ -best  $|c_n|$ . This is referred to as nonlinear approximation since the sum of two arbitrary elements expressed by Eq. 1.11 generally uses more than basis functions with the index included in  $I_M$ .

The results of both schemes depend on the signal and on the basis choice but the nonlinear approximation scheme certainly works better than linear one. To clear the different behavior of these two types of approximation we report an example taken from [30]. We want approximate a one-dimensional piecewise smooth signal with  $N = 1024$  samples (showed at top of Fig. 1.7) using only  $M = 64$  coefficients. In figure 1.7 we show the results for both types of

---

<sup>3</sup>That, for orthonormal transform, are the largest ones.

approximation using the Fourier basis. In this case we see that the nonlinear approximation is not necessarily better than linear approximation, but both provide poor results. Figure 1.8, instead, shows the results of the same exper-



**Figure 1.7:** Approximation using Fourier basis. On top, the original signal. In the middle, linear approximation. At the bottom, nonlinear approximation.

iment but using a wavelet basis. In this case, with the linear approximation scheme the resulting signal is very poor, but using nonlinear approximation the signal is almost perfectly recovered.

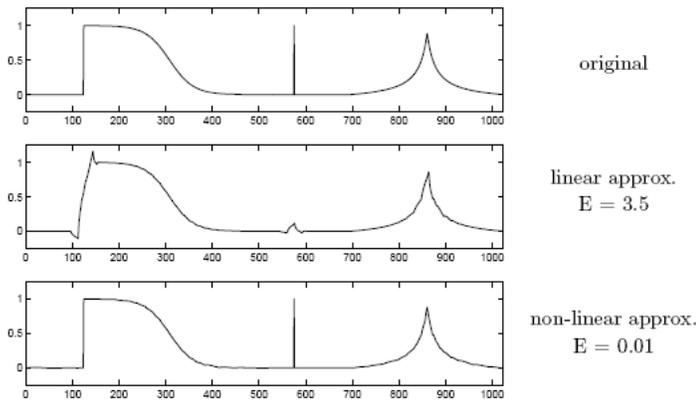
In the first section, talking about KTL, we said that this transform decorrelates optimally namely maximizes the energy concentration. This characteristic can be expressed in another way saying that KLT is the basis that minimizes the linear approximation error:

$$E\left[\|x - \sum_{n=0}^{M-1} \langle x, \phi_n \rangle \phi_n\|^2\right]. \quad (1.12)$$

So, in a low bit-rate regime, the key feature is not the energy concentration or the decorrelation but the capacity of transform to minimize the error of nonlinear approximation:

$$E\left[\|x - \sum_{n \in I_M} \langle x, \phi_n \rangle \phi_n\|^2\right]. \quad (1.13)$$

When a basis satisfies this property, is named *sparse*.



**Figure 1.8:** Approximation using wavelet basis. On top, the original signal. In the middle, linear approximation. At the bottom, nonlinear approximation.

Even if we refer the reader to [28] and [29] for a deeper analysis, here we want just to mention the reason for nonlinear approximation importance in low bit-rate compression. Many image coder schemes, as for example EZW, SPIHT and EBCOT, tend to put to zero small coefficients, typically, the lower bit-rates the smaller the number of coefficients different from zero. So at these rates, there are two different contributions to distortion, one due to quantization and another due to the fact that we are taking only a part of total coefficients. It is possible to show that the latter contribution is predominant. So if the coding is made in an appropriate way the rate behavior follows that of nonlinear approximation.

The success of wavelet basis in nonlinear approximation was interpreted by mathematicians to be the true reason of the usefulness of wavelets in signal compression, rather than their potential for decorrelation [29].

At this point we can declare with sufficient confidence that what we want from a transform in image coding is that it gives a sparse representation of the data. But, while the better basis for linear approximation is note for all kind of signals and is the KTL, the best basis for nonlinear approximation depends on the particular class of signals considered.

Recently, to find the best basis for images, a lot of researchers used a completely different approach from that of classical information theory, in fact they utilized a deterministic approach [31]. In this setting the images are not seen

as realization of a random process (for example gaussian) but as function of a particular class. While in the probabilistic approach all results are average results, the results in deterministic setting are relative to the worst case, namely instead of minimizing

$$E[\|x - \sum_{n \in I_M} \langle x, \phi_n \rangle \phi_n\|^2], \quad (1.14)$$

we minimize:

$$\max_{x \in S} [\|x - \sum_{n \in I_M} \langle x, \phi_n \rangle \phi_n\|^2], \quad (1.15)$$

where  $S$  is the function class.

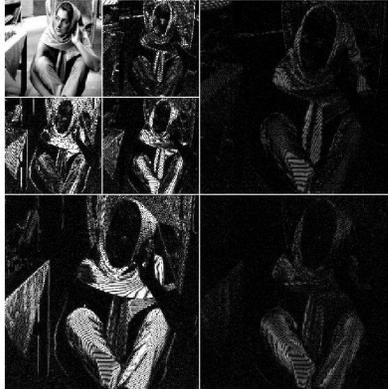
Particular, of great interest for compression community are some results of harmonic analysis. The harmonic analysis involves different mathematical problems, even very heterogeneous, but can be summarized in the tentative of representing some functions as superposition of fundamental waves, called harmonic. Basically, the harmonic analysis try to identify a class of mathematically defined objects (functions, operators, etc.), to develop tools to characterize the class of objects and to find, if possible, an optimal representation [31]. Part of the results we are going to talk about rise in this context, notably in the sphere of computational harmonic analysis (CHA), an harmonic analysis branch that, as well as being interested in finding an optimal representation, try to identify a fast algorithm to compute this representation.

A very interesting CHA result, that justify the success of wavelet transform in image coding, is that wavelets are an optimal basis for Besov spaces, where with Besov spaces is intended a class of smooth functions with point discontinuities. Therefore wavelets are optimal for a class of functions that well represents image rows and columns.

## 1.5 Wavelet limits and new approaches

In the previous section, we motivated wavelet efficiency with its good NLA behavior on smooth signals with point discontinuity. Unfortunately, contours in the image are not pointwise but they lie on regular curves. So commonly used separable wavelet transforms is suboptimal in higher dimensions. To clear this concept in Fig. 1.9 we show an example of the 2-D wavelet transform of the image Barbara. We can notice that wavelet representation is not sufficiently sparse, in fact there are a lot of significant wavelet coefficients (shown in white). The significant pixels are around the image contours, or in general,

the points with sharp contrasts in the intensity. The key point to note here is that the locations of these significant coefficients exhibit visible geometrical correlation as they form simple curves; geometrical correlation that wavelet, being a separable transform, can not exploit. This suboptimality can be for-



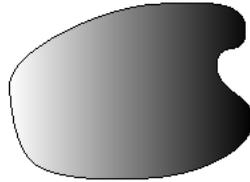
**Figure 1.9:** 2-D wavelet transform of the image Barbara. Significant coefficients are shown in white.

malized mathematically if we consider a model for natural images of type  $C^2/C^2$ , where with the expression  $C^2/C^2$  we mean an image composed of piecewise smooth regions with a smoothness of type  $C^{2.4}$  separated by smooth boundaries having the same regularity (see Fig. 1.10). In fact if we consider this model it is possible to show that optimal asymptotic NLA behavior is of type  $M^{-2}$  while wavelet NLA behavior is only of type  $M^{-1}$  [2]. The wavelet inefficiency on contours make necessary find new approaches for image compression. The arguments treated in this thesis work are placed in three different scenarios that can be considered three different answers to this request: object-based coding (Chapter 2), new directional transforms (Chapter 3) and adaptive lifting scheme (Chapter 4).

The first approach that we want to investigate is object-based coding. The scheme that we considered is the following: the image is segmented, this process produces a segmentation map that is coded without loss of information; then this information is used to lossy compress the interior of each object; finally, the overall bit-budget is allocated among the various objects. In this way we have two advantages: first of all we preserve information about contours

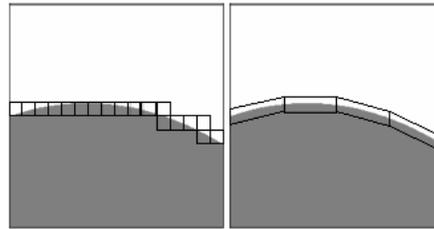
---

<sup>4</sup>A function is of class  $C^2$  if has two derivatives continuous and bounded.



**Figure 1.10:** A simple model of image.

and this improves the visual quality. Second, lossy compression algorithms (read wavelets) work on stationary signals and so they perform better.



**Figure 1.11:** Example of anisotropy and directionality importance.

An alternative solution for contours problem is the use of new, and more effective, transforms. We already noted that the cause of wavelets suboptimality reside in its inability to see the smoothness along contours. This is due to its being a separable transform so it lacks of two fundamental characteristics: directionality and anisotropy [6]. To visualize the importance of these features in representing images with smooth contours, we consider the scenario showed in Fig. 1.11 where we have a simple image with a contour. The little squares represent the supports of the wavelet basis functions: they are square because we use the same wavelet on the rows and on the columns. We can consider to have a large coefficient whenever our basis functions intersect the contour, so to describe this information we need a lot of coefficients. But if we had basis functions with a support rather elongated and directional, we could represent the same information with less coefficients, as we can see on the right of Fig. 1.11. The new transforms look for these new goals, directionality and anisotropy, while at same time, keeping the desirable wavelet characteristics

---

such as multiresolution, localization and critical sampling. The importance of these features is also supported by the physiological studies [32], [33] from which we know that the receptive fields in the visual cortex are characterized as being localized, oriented and bandpass.

Lifting structure is an efficient and popular implementation of wavelet transforms [9]. Using lifting structure, several adaptive wavelet transforms have been proposed. The ways in which adaptivity could be useful for contour problem are numerous. A very spread strategy concerns the filtering directions, that are modified in order to follow the orientations of edges and textures [10], [11], [12]. Others types of adaptive lifting scheme, instead, operating on prediction or update steps, change the filters according to the regularity of the input signal [13], [14], [15]. So when important features like contours or singularities are highlighted, different filters, from that used in regular part of images, are used.



## Chapter 2

# Object-based coding

Object-based image coding is drawing great attention for the many opportunities it offers to high level applications. At same time this scheme could be seen as a mean to cope with wavelet inefficiency on contours. However, its value, in terms of rate-distortion performance, is still uncertain, because the gains provided by an accurate image segmentation are balanced by the inefficiency of coding objects of arbitrary shape, with losses that depend on both the coding scheme and the object geometry. In this chapter we measure rate-distortion costs and gains for a wavelet-based shape-adaptive encoder similar to the shape-adaptive texture coder adopted in MPEG-4. The analysis of the rate-distortion curves obtained in several experiments provides insight about what performance gains and losses can be expected in various operative conditions and shows the potential of such an approach for image coding. Then we introduce two variants of object coding paradigm for multispectral images and we analyze their performance showing the validity of the approach for this kind of images.

### 2.1 Object-based coding

Object-based image coding is an increasingly active area of research, dating back to early works on second generation coding techniques [34] and gaining momentum more recently thanks to the driving force of the MPEG-4 video coding standard [35]. The major conceptual reason for object-based coding is that images are *naturally* composed by objects, and the usual pixel-level description is only due to the lack of a suitable language to efficiently represent them. Once objects have been identified and described, they can be

treated individually for the most diverse needs. For example they can be assigned different coding resources and different error-protection levels based on their relative importance for the user [36, 37], can be edited in various way by high-level applications, or can be used for subsequent classification tasks (e.g., biometric applications).

In some instances, object-based coding is obviously the most reasonable solution. In the context of MPEG-4 video coding, for example, when a number of arbitrarily shaped foreground objects move in front of a fixed background, which is a full-frame sprite, conventional coding is clearly inefficient. Additionally, there exist applications (e.g., [38]) in which data are available only for part of the image frame, and one has no choice but to either code an arbitrarily-shaped object or artificial pad the object out to a full-frame.

In the context of this thesis work, we are interested in object-based from a coding efficiency point of view, in fact, as noticed in Cap. 1, the object-based description of an image could be a mean to overcome problems of classical coding scheme in describing contours. Indeed, component regions turn out to be more homogeneous, and their individual encoding can lead to actual rate-distortion gains. Moreover, an accurate segmentation carries with it information on the image edges, and hence contributes to the coding efficiency and perceived quality. Of course, there are also costs; first of all, since objects are separate entities, their shape and position must be described by means of some segmentation map, sent in advance as side information. In addition, most coding techniques become less efficient when dealing with regions of arbitrary size and shape. These observations justify the need of analysis of potential cost and advantages of object-based approach [39].

In this analysis, we focus on a wavelet-based shape-adaptive coding algorithm. The main coding tools are the shape-adaptive wavelet transform (SA-WT) proposed by Li and Li [3], and a shape-adaptive version of SPIHT (SA-SPIHT) [4] (similar to that formerly proposed in [40] and further refined in [41]) which extends to objects of arbitrary shape the well-known image coder proposed by Said and Pearlman . The choice of the specific coding scheme is justified by the fact that Li and Li's SA-WT is by now a *de facto* standard for object based-coding, and SPIHT guarantees a very good performance, and is widespread and well-known in the compression community. In addition, the algorithm analyzed here is very similar to the standard texture coder of MPEG-4 [35]. Of course, this is not the only reasonable choice, and other coding algorithms based on shape-adaptive wavelet have been proposed in recent years [42, 43, 44, 45], sometimes with very interesting results, but a comparison with

some of these algorithms, deferred to the Sec.2.5, is of marginal interest here. Our target is to analyze the quite general mechanisms that influence the efficiency of wavelet-based shape-adaptive coding and to assess the difference in performance with respect to conventional wavelet-based coding.

In more detail, we can identify three causes for the additional costs of object-based coding: the reduced performances of the WT and the reduced coding efficiency of SPIHT that arise in the presence of regions with arbitrary shape and size, and the cost of side information (segmentation map, object coding parameters). Note that this classification is somewhat arbitrary, since the first two contributions are intimately connected, nonetheless it will help us in our analysis. As for the possible gains, they mirror the losses, since they arise for the increased sparsity of the WT representation, when dominant edges are removed, and for the increased coding efficiency of SPIHT when homogeneous regions have to be coded.

A theoretical analysis of such phenomena is out of the question, and in the literature attempts have been made only for very simple cases, like 1-d piecewise-constant signals [46]. Therefore, we measure losses and gains by means of numerical experiments carried out in controlled conditions. This allows us to isolate with good reliability the individual contributions to the overall performance, point out weaknesses and strengths of this approach, and hence give insight about the behavior of the proposed coding scheme in situations of practical interest.

In order to assess losses and gains related to the SA-WT only, we remove the cost of side information, and use an “oracle” coder which mimics the progressive bit-plane coding of SPIHT but knows in advance the location of significant coefficients within each bit-plane, thereby removing all sorting-pass costs<sup>1</sup>. Within this framework, we use several classes of images and of segmentation maps, both synthetic and natural, so as to study all the relevant phenomena. Subsequently, for the same set of images and maps, we add the actual coding phase: the additional gains and losses can be therefore attributed to SA-SPIHT or to its interactions with the SA-WT.

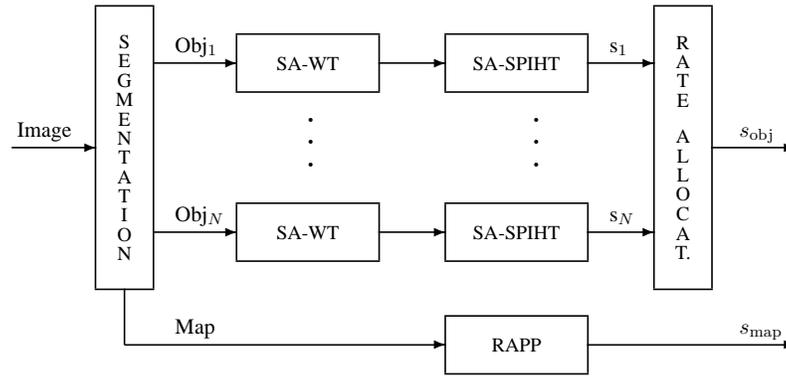
## 2.2 The coding scheme

We implemented an object-based coding scheme with the following elementary steps (see Fig. 2.1)

---

<sup>1</sup>Note that the very same oracle coder works for all bit-plane oriented coders that use Li and Li’s SA-WT, like for example [43] and [47].

1. image segmentation;
2. lossless coding of the segmentation map (object shapes);
3. shape-adaptive wavelet transform of each object;
4. shape-adaptive SPIHT coding of each object;
5. optimal post-coding rate allocation among objects.



**Figure 2.1:** The object-based coding scheme under investigation.

The accurate segmentation of the image is of central importance for the success of object-based coding, and is by itself a very challenging task and a “hot” topic. However, faithful image segmentation is not of interest here and is not investigated. Moreover, to study the effects of different object geometries on the coding performance, we need to change rather freely the geometrical/statistical parameters of objects, and therefore resort, in most of the analysis, to artificial regular segmentation maps, independent of the actual image content. Only in our final experiments we do consider meaningful segmentation maps.

The segmentation maps are encoded without loss of information, because of their importance, by means of a modified version of the RAPP algorithm [48], originally proposed for palette images, which proves very efficient for this task. The cost for coding the map, as well as all other side information costs, can become significant and even dominant in some instances, and hence must be always taken into account in the overall performance.

As for the SA-WT, we resort to Li and Li's algorithm, as already said, which is almost universally used in the literature and also adopted in the MPEG-4 standard. For a detailed description we refer to the original paper [3], but it is worth recalling here its most relevant features. First of all, the number of coefficients equals the number of pixels in the original object, so there is no new redundancy introduced. Second, spatial relationships among pixels are retained, so there are no new spurious "frequencies" in the transform. Finally, the SA-WT falls back to ordinary WT for rectangular objects. All these reasons, together with its simple implementation and experimentally good performance, justify the success of this algorithm. In the implementation, we use five levels of decomposition, Daubechies 9/7 biorthogonal filters, and the global subsampling option which secures experimentally the best performance.

After SA-WT, we use the well-known SPIHT algorithm, in the shape-adaptive extension proposed in [4]. It is worth recalling that it is a bit-plane coder of the wavelet coefficients. For each bit-plane there are essentially two tasks, locating the significant bits, and specifying their value (also the coefficient signs must be encoded of course). Other algorithms of interest here share the same general approach, and differ only in the way significant bits are located. Our shape-adaptive version of SPIHT is very similar to the basic algorithm with the differences that only active nodes, that is nodes belonging to the support of the SA-WT transform, are considered, and that the tree of coefficients has a single ancestor in the lowest frequency band.

After coding, the rate-distortion (RD) curves of all objects are analyzed so as to optimally allocate bits among them for any desired encoding rate, like in the post-compression rate allocation algorithm of JPEG-2000. This process is intrinsically performed in conventional coding, while is a necessary step in object-based coding, and also an extra degree of freedom as bits could be also allocated according to criteria different from RD optimization.

## 2.3 Measurement of losses

### 2.3.1 Methodology

As clarified in the Chapter 1, the performance of a transform-based compression algorithm depends essentially on the efficiency of the transform, which is therefore the first item we must quantify.

It's important to notice that the shape-adaptive WT is non-isometric in

an unpredictable way. This depends on the need to transform signal segments composed by a single pixel: in Li and Li's algorithm, this generates a single coefficient which is put in the low-pass transform band and, in order not to introduce discontinuities in otherwise flat areas, is multiplied by a constant. This multiplication (which can occur many times in the SA-WT of an object) modifies the transform energy and makes impossible to compare SA-WT and WT coefficients directly.

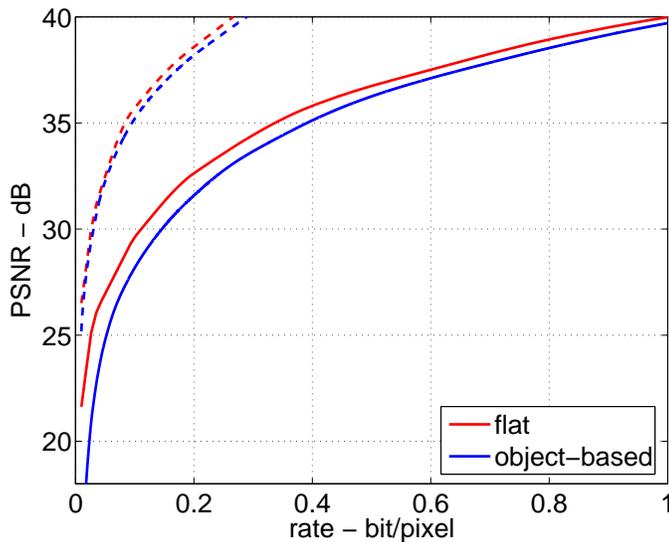
For this reason, we propose here an experimental methodology to compare the efficiency of SA-WT and its conventional (or "flat") version. The basic idea is to apply both the shape-adaptive and the flat transforms to the same image, quantize the resulting coefficients in the same way, and compare the resulting RD curves. In order for the comparison to be meaningful, the transforms must operate on exactly the same source, and hence *all* objects of the image must undergo the SA-WT and be processed together. The total number of coefficients produced by the SA-WT is equal to the number of image pixels and hence to the number of WT coefficients. These two sets of coefficients are sent to an oracle encoder which implements a bit-plane quantization scheme like that of SPIHT and most other engines used in object-based coders. All these algorithms spend some coding bits to locate the significant coefficients in each plane (sorting pass, in SPIHT terminology), and some others to encode their sign and to progressively quantize them (refinement pass). Our oracle coder knows in advance all significance maps and spends its bits only for the sign and the progressive quantization of coefficients. As a consequence, the rate-distortion performance of this virtual coder depends only on how well the transform captured pixel dependencies, what we call transform efficiency<sup>2</sup>.

As an example, consider the RD curves of Fig. 2.2. Although the object based coder (solid red) performs clearly worse than the flat coder (solid blue), at least at low rates, their oracle counterparts (dashed red and dashed blue) perform nearly equally well. This means that, as far as the transforms are concerned, the shape-adaptive WT is almost as efficient as the conventional WT, and therefore the losses must be ascribed to coding inefficiencies or to the side information. Actually, since the cost of side information is known, we can also easily compute the losses caused by SA-SPIHT inefficiencies, the second major item we are interested to measure.

There are two reasons why shape-adaptive SPIHT could be less efficient than flat SPIHT

---

<sup>2</sup> This measure is similar to NLA, but it takes into account also the effects of quantization on the most significant coefficients



**Figure 2.2:** RD curves for flat (red) and object-based (blue) coders. Solid and dashed lines are, respectively, for actual and oracle coders.

- the presence of incomplete trees of coefficients;
- the interactions with the SA-WT.

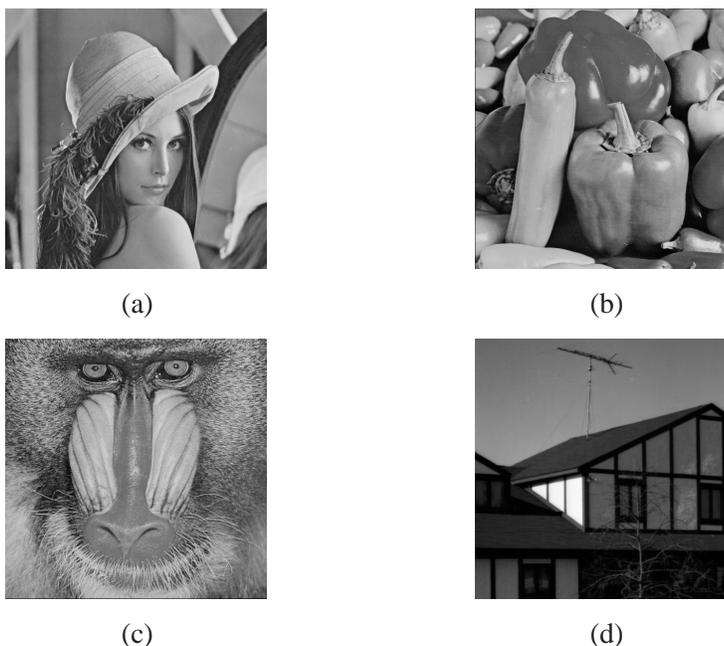
Much of the efficiency of SPIHT, especially at low rates, is due to the use of zerotrees, that is, trees of coefficients that are all insignificant w.r.t. a given threshold and can be temporarily discarded from further analysis. A single information bit can therefore describe a whole zerotree, comprising a large number of coefficients. With an arbitrarily shaped object, the support of the transform can be quite irregular, and incomplete zerotrees can appear, which lack some branches and comprise less coefficients than before. As a consequence, the zerotree coding process becomes less efficient, at least at the lowest rates.

The second item concerns a more subtle phenomenon, the fact that the reduced WT efficiency affects indeed *both* quantization and sorting. In fact, when the WT does not give a sufficiently sparse representation, the energy is more scattered throughout the trees and more bits are spent sorting in order to isolate the significant coefficients at each iteration. Hence, computing these losses as due to SA-SPIHT is somewhat arbitrary, but it is also true that a

different coder could be less affected by this phenomenon.

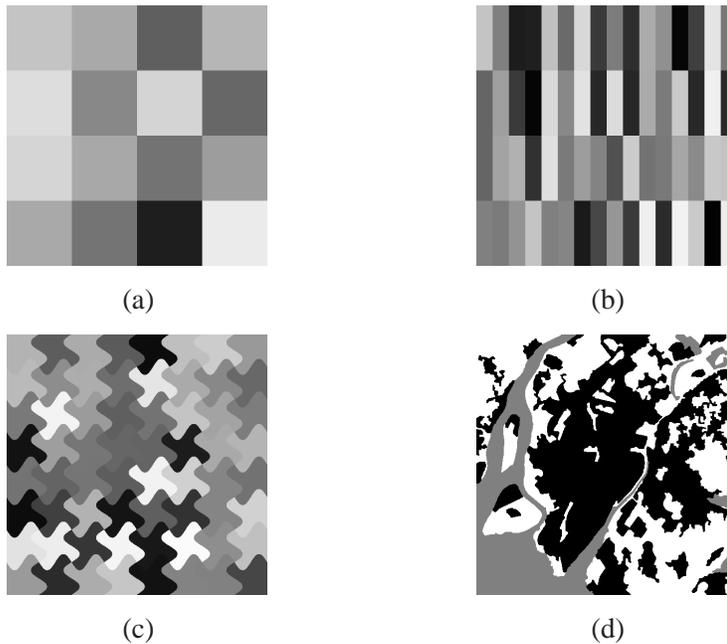
### 2.3.2 Experimental results

To measure losses, we encode some natural images of the USC database [49] with both the oracle and the actual object-based coders using synthetic segmentation maps of various types formed by square tiles, rectangular tiles, wavy tiles, irregular tiles. Test images ( $512 \times 512$  pixels, 8 bit/pixel) are shown in Fig. 2.3, while Fig. 2.4 shows some examples of segmentation maps. By using such synthetic maps, which are not related to the actual image to be coded, we introduce and measure only the *losses* due to object shape and size, while no gain can be expected because object boundaries do not coincide with actual region boundaries.



**Figure 2.3:** Test images from the USC database: (a) Lena, (b) Peppers, (c) Baboon, (d) House.

In the first experiment we segment the natural images in square tiles of size going from  $512 \times 512$  (whole image) down to  $32 \times 32$  (256 objects), and encode

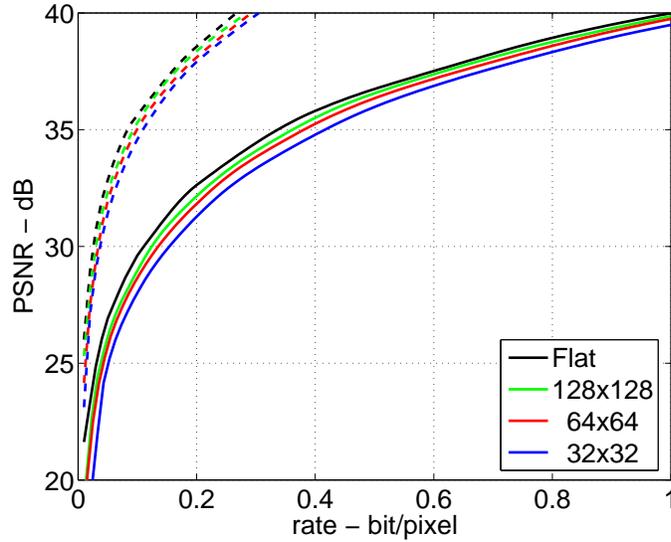


**Figure 2.4:** Some maps used in the experiments: (a) square  $128 \times 128$  tiles, (b) rectangular  $128 \times 32$  tiles, (c) wavy tiles with  $C=1$ ,  $A=16$ , (d) out-of-context map.

them as described before. In Fig. 2.5 we report the rate-distortion curves obtained by the object-based coders for each tile size: solid lines refer to the actual coder, and dashed lines to the oracle coder. Note that the flat case corresponds to the  $512 \times 512$  coder, that is, conventional WT and SPIHT. Curves refer to the image Lena of Fig. 2.3 (a), as will always be in the following unless otherwise stated, but similar results have been obtained with all other images. A first important observation is that the quantization rate is always a small fraction, about one fourth, of the total rate, at least in the range considered here<sup>3</sup>. As a consequence, the same relative loss of efficiency is much more critical for SPIHT than for the WT. In this experiment, however, losses are always quite limited. Performances worsen as the tile size decreases, but the rate increment is always less than 20% (except a very low rates) and the PSNR gap is less

<sup>3</sup>At higher rates, the RD slope is the same in all cases because we are only coding noise-like residuals, and hence the analysis loses interest

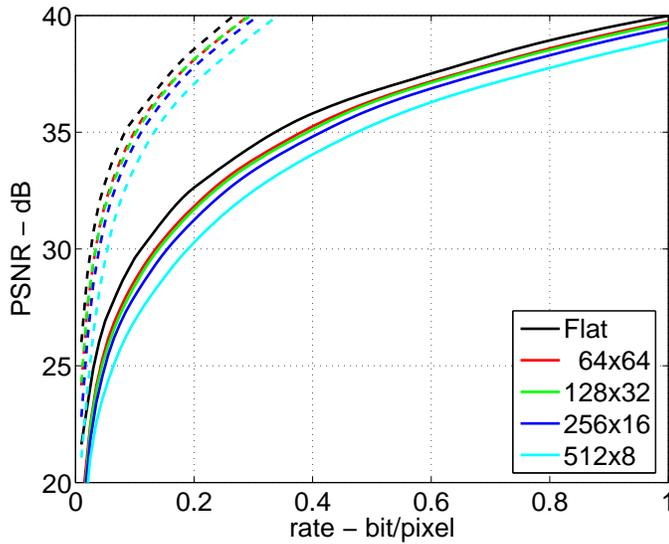
than half dB at high rates, and about 1 dB at lower rates. Most of these losses are due, directly or indirectly, to the reduced sparsity of the SA-WT, since the zerotrees are always complete, and the fixed cost of side information, 0.013 bit/pixel in the worst case, is quite small. Note, however, that this last cost cannot be neglected if one looks at very low rates.



**Figure 2.5:** RD performance with square-tile segmentation. Solid and dashed lines are, respectively, for actual and oracle coders. Black lines are for flat (conventional) coding of the whole image, colored lines are for object-based coding.

To begin investigating the influence of region shapes, in the second experiment we consider rectangular tiles of fixed size (4096 pixels) but different aspect ratios, from  $64 \times 64$  to  $512 \times 8$ . The RD curves are reported in Fig. 2.6, together with those for the flat case, and show that the aspect ratio does matter, but only when very short segments are considered. Indeed, the performance is very close for  $64 \times 64$ ,  $128 \times 32$ , and even  $256 \times 16$  tiles, while it becomes significantly worse for  $512 \times 8$  tiles, because the WT cannot perform well with segments as short as 8 pixels. For example, the PSNR loss at high rate is 1.15 dB for the  $512 \times 8$  case and less than 0.6 dB for all the other cases. One might suspect that the sharp decline in performance in the  $512 \times 8$  case is also related

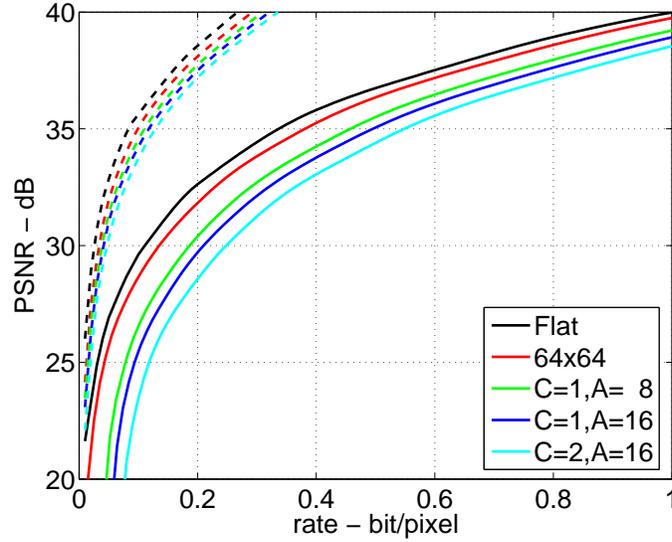
with our use of 5 levels of decomposition when 3 or 4 would have been more appropriate for such short segments. In fact, this mismatch produces several single coefficients, after some levels of WT, which are further filtered and lead to an artificial increase in energy. However, all our experiments show that adapting the number of decomposition levels to the object size has no measurable effects on the performance, and that a fixed 5-level SA-WT is the optimal choice, at least for our  $512 \times 512$  images.



**Figure 2.6:** RD performance with rectangular-tile segmentation.

Let us now consider more complex tiles, obtained by remodeling the boundaries of a  $64 \times 64$  square as sine-waves with amplitude  $A$  pixels, and frequency  $C$  cycles/tile. One such segmentation map, obtained for  $A=16$  and  $C=1$ , is shown in Fig. 2.4 (c). In Fig. 2.7 we report the RD curves for some significant values of  $A$  and  $C$ , together with the reference curves for square  $64 \times 64$  tiles and for flat coding. As expected, the performance worsens as the tiles become less regular. At high rates the impairment is not dramatic, with a PSNR loss that lies between 1 and 2 dB, while the situation is much worse at low rates, with losses of 4-5 dB or, for a given PSNR, a coding rate that doubles w.r.t. flat coding. Apparently, such losses are mainly due to the side information and SA-SPIHT inefficiencies, and only in minimal part to the SA-WT, since the RD curves for the oracle coder are all very close, but we should

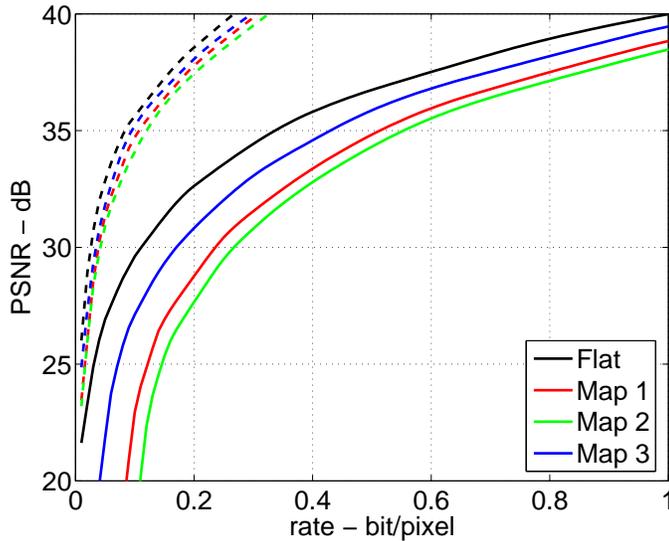
not forget the WT-SPIHT interactions, and will soon come back to this topic.



**Figure 2.7:** RD performance with wavy-tile segmentation.

In our fourth experiment, we use segmentation maps obtained for unrelated (remote-sensing) images of the same size as ours. These maps, one of which is shown in Fig. 2.4 (d), present many elementary tiles, with quite different size and shape, some with regular boundaries and some not. Fig. 2.8 shows RD curves for this case, which resemble closely those of Fig. 2.7, and for which the same comments apply, suggesting that the wavy-tiles segmentation can be a good tool to mimic actual segmentation maps.

To take a closer look at these results, let us consider Tab. 2.1 where we have collected the individual contributions of side information, quantization, and sorting pass to the overall coding cost, at a PSNR of 30 dB, corresponding to the low-rate range. We see that the increase of the quantization cost w.r.t. the flat case is quite steep, from 15% up to 100%, due to the reduced efficiency of the transform. As for the sorting cost, it also increases w.r.t. the flat case. The increase is obviously larger in the last six cases, when the tile geometry is more challenging, but also non negligible in the first six cases, with square and rectangular tiles. This is quite telling, because with straight boundaries there are no incomplete trees to impair performance, and hence all losses must be charged to the reduced sparsity of transform coefficients. Therefore, one can



**Figure 2.8:** RD performance with out-of-context segmentation maps.

even hypothesize that transform inefficiencies are the ultimate cause of most of the overall losses, even though the effects are more evident in the sorting pass, a conjecture that we will further analyze shortly. As a synthetic measure of performance, we reported in the last column the overall rate increase w.r.t. flat coding, including all contributions, which is quite large in all realistic cases, confirming that object-based coding can be very penalizing at low rates.

The picture, however, is quite different at high rates. Tab. 2.2 is similar to Tab. 2.1 except that all costs are computed at a PSNR of 38 dB, hence at the right end of our range. It is obvious that the cost of side information becomes less relevant, and even in the more challenging situations the cost of quantization and sorting presents only a limited increase. In the last column, we report a more familiar measure of performance, the PSNR loss w.r.t. flat coding at 0.8 bit/pixel, which is never more than 2 dB, and quite often under just 1 dB showing that, at high rates, object-based coding can be used without paying much attention to the rate-distortion performance. It is also worth remembering that, in most practical situations where object-based coding is used, there is only a small number of objects, and therefore these measures of loss can be assumed as upper bounds.

We conclude this section with one last insightful experiment, which sheds

tiling	absolute rates			percent increase		
	side.i.	quant.	sorting	quant.	sorting	total
whole image		.026	.085	–	–	–
128×128	.003	.030	.091	15.4	7.3	11.7
64×64	.005	.034	.096	30.9	13.1	21.6
32×32	.013	.037	.104	42.9	22.0	38.7
128×32	.005	.034	.100	31.2	17.8	25.2
256×16	.005	.040	.110	53.5	29.3	39.6
512×8	.005	.054	.131	106.9	54.0	71.1
C=1,A=8	.032	.038	.116	48.4	36.3	67.5
C=1,A=16	.044	.041	.125	58.6	46.7	89.1
C=2,A=16	.060	.047	.141	80.6	65.8	123.4
Map 1	.083	.038	.127	48.3	49.9	123.4
Map 2	.105	.042	.135	61.2	59.2	154.0
Map 3	.042	.034	.105	33.0	24.0	63.0

**Table 2.1:** Indicators of losses at low rates (PSNR = 30 dB).

some more light on the nature of SPIHT losses. Li and Li’s SA-WT, when applied to all objects of an image, like the simple example of Fig. 2.9 (a), produces transforms that do not fit together, namely, cannot be put together in a single image as the pieces of a mosaic, because some coefficients overlap, as the circled coefficients shown in Fig. 2.9 (b). This is unavoidable if all single coefficients must be put in the low-pass band after filtering. However, we can modify the algorithm and put single coefficients either in the low-pass or high-pass band depending on their coordinates. This way, we might sacrifice part of the SA-WT efficiency, but obtain object transforms that fit together as shown in Fig. 2.9 (c). After all the SA-WTs have been carried out, we can encode the coefficients by using SA-SPIHT on each object, or conventional SPIHT on all the coefficients arranged as a single image. The flat and object-based coders thus operate exactly on the same set of coefficients, and all possible impairments can be ascribed to SA-SPIHT coding inefficiencies. The RD curves obtained with flat and SA-SPIHT for various segmentation maps are reported in Fig. 2.10, and show clearly that the efficiency gap between shape-adaptive and flat SPIHT is always very limited, and at high rates never exceeds 0.3 dB<sup>4</sup>.

<sup>4</sup>As an aside, our experiments show also that the performance of this new scheme (fitting

tiling	absolute rates			percent increase		$\Delta$ PSNR @ 0.8 b/p
	side.i.	quant.	sorting	quant.	sorting	
whole image		.176	.488	–	–	–
128×128	.003	.184	.498	4.2	2.0	0.15
64×64	.005	.195	.512	10.6	4.9	0.31
32×32	.013	.204	.534	15.5	9.4	0.62
128×32	.005	.194	.519	10.2	6.3	0.37
256×16	.005	.209	.542	18.2	11.0	0.60
512×8	.005	.241	.590	36.4	20.9	1.14
C=1,A=8	.032	.211	.563	19.3	15.2	0.95
C=1,A=16	.044	.221	.589	25.2	20.6	1.35
C=2,A=16	.060	.234	.622	32.6	27.3	1.82
Map 1	.083	.209	.591	18.5	21.1	1.33
Map 2	.105	.225	.611	27.5	25.2	1.89
Map 3	.042	.197	.544	11.7	11.3	0.78

**Table 2.2:** Indicators of losses at high rates (PSNR = 38 dB).

This seems to be a conclusive proof that the losses arising in the sorting pass, although dominant w.r.t. those of the quantization pass, are mostly related to the reduced sparsity of the SA-WT.

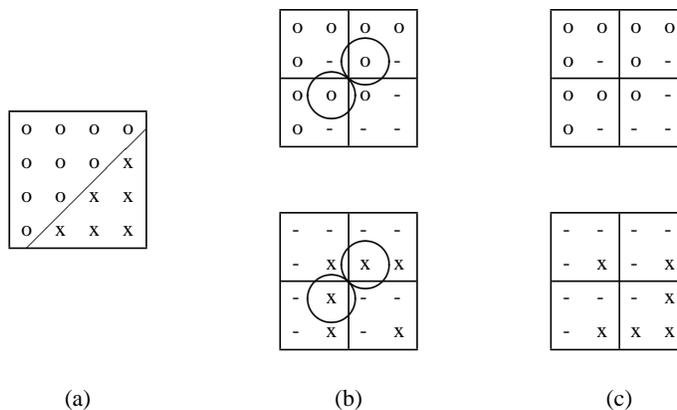
## 2.4 Measurement of gains

### 2.4.1 Methodology

The rate-distortion potential of object-based coding strongly depends on the ability of the segmenter to single out accurately the component objects. When this happens, in fact, the segmentation map describes automatically many expensive high-frequency components, related to the edges between different objects. In terms of SA-WT, this means dealing with a signal (within the object) that is much smoother than the original signal, since strong edges have been re-

---

SA-WT + flat SPIHT) is very close to that of our object-based algorithm. However, this new scheme is not object-based anymore.

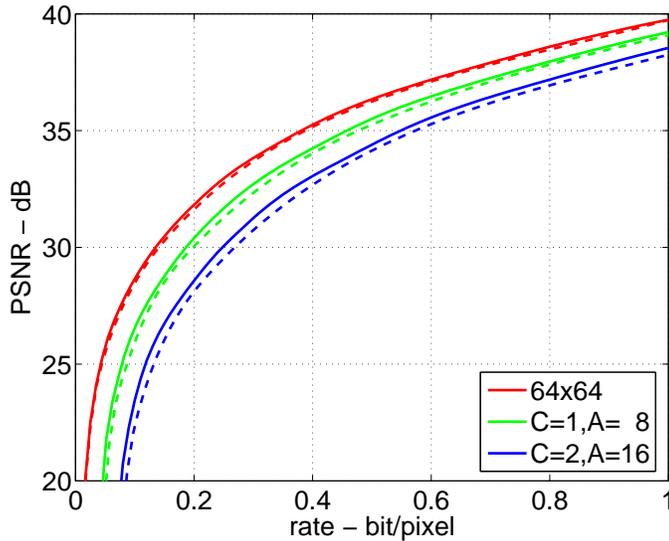


**Figure 2.9:** Object overlapping in the transform domain. The  $4 \times 4$  original image with two objects (a) is subject to 1 level of SAWT: the supports of the two objects overlap with Li and Li SA-WT (b) but not with the fitting SA-WT (c).

moved, which leads in turn to a much increased efficiency because most of the encoding resources, especially at low rates, are normally used for describing edges. Of course, the actual success of this approach depends on many factors, such as the profile of edges, the statistical properties of the signal within the objects, and the accuracy of segmentation.

In order to measure the potential performance gains, we get rid of the dependence on the segmentation algorithm, which is not the object of this analysis, by building some mosaics in which neighboring tiles are extracted from different images. Of course, one must keep in mind that this condition is very favorable for object-based coding since objects are clear-cut and we know their shape perfectly. Our mosaics vary not only for the form of the tiles, but also for the source images from which they are drawn, that can be

- synthetic images where the signal is polynomial in the spatial variables;
- natural images from the USC database;
- natural textures from the Brodatz database, also available at [49];

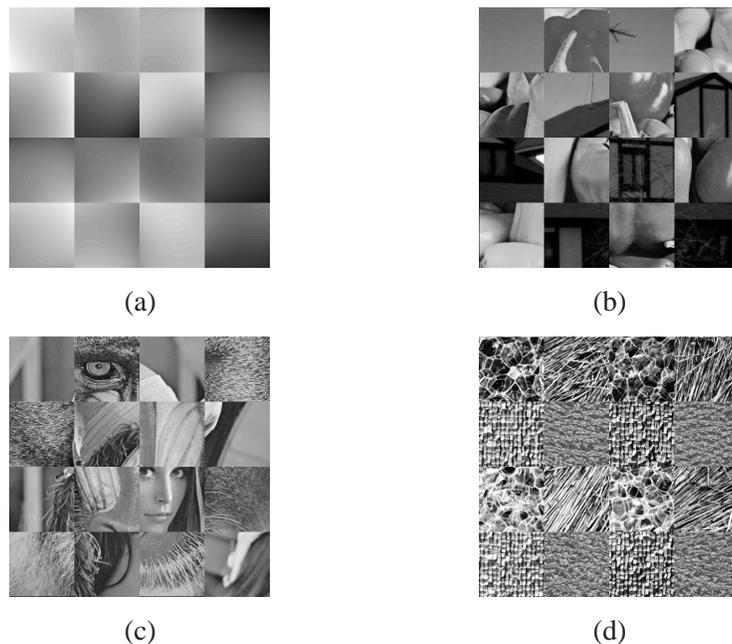


**Figure 2.10:** RD performance with fitting SA-WT. Solid lines are for flat coding of the mosaic formed by the object transform, dashed lines are for actual object-based coding.

Some examples are shown in Fig. 2.11. By changing the source images we go from the most favorable case, like that of Fig. 2.11 (a), where all tiles are from polynomial images, to the most challenging, like that of Fig. 2.11 (d), where even within the tiles there are strong signal components at the medium and high frequencies due to the original textures. In between these extremes, there are more realistic cases where the objects are drawn from natural images predominantly smooth, like Fig. 2.11 (b), or with significant texture components, like Fig. 2.11 (c).

### 2.4.2 Experimental results

Fig. 2.12 shows the PSNR differences between the object-based and the flat coders when mosaics are composed by wavy tiles of size  $64 \times 64$  and boundary parameters  $C=1$  and  $A=16$  with the same source images as those shown in Fig. 2.11. For the first mosaic, there is a very large gain, of 8-10 dB at medium-high rates, and up to 20 dB at low rates (out of the scale of our figure). This

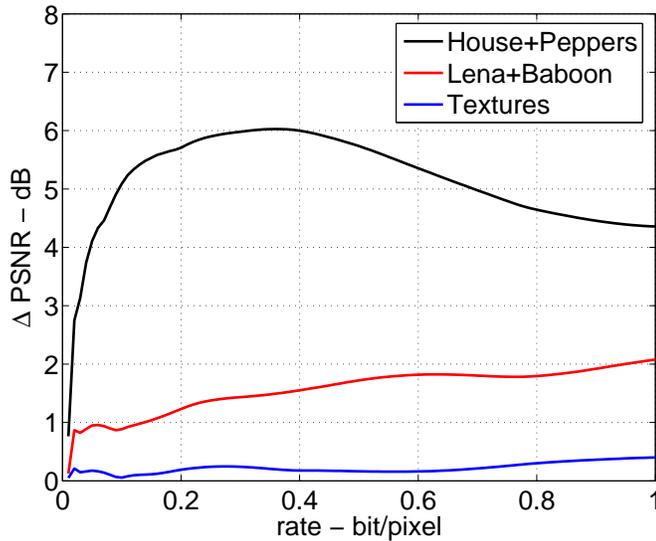


**Figure 2.11:** Some mosaics used in the experiments, with square  $128 \times 128$  tiles: (a) Polynomials, (b) House+Peppers, (c) Lena+Baboon, (d) Textures.

is remarkable but not really surprising, given the smooth sources and the fact that Daubechies wavelets are perfectly fit for polynomial signals.

More interesting are the results obtained with the natural mosaics, with a gain at all bit-rates of about 5 dB in the first case, and almost 2 dB in the second case. Considering that these are natural images, this speaks strongly in favor of the potential of object-based coding, even with all the *caveat* due to the favorable experimental conditions. Also, remember that the observed gain is obtained despite the losses due to the use of SA-WT with small wavy tiles (see again Fig. 2.7). As expected, results are less favorable for the fourth mosaic, where the presence of many high-frequency components within the tiles reduces the gain to the point that it compensates the shape loss but little more.

Fig. 2.13 shows results obtained with the same source images but with square  $128 \times 128$  tiles. The general behavior is very similar to the former case,



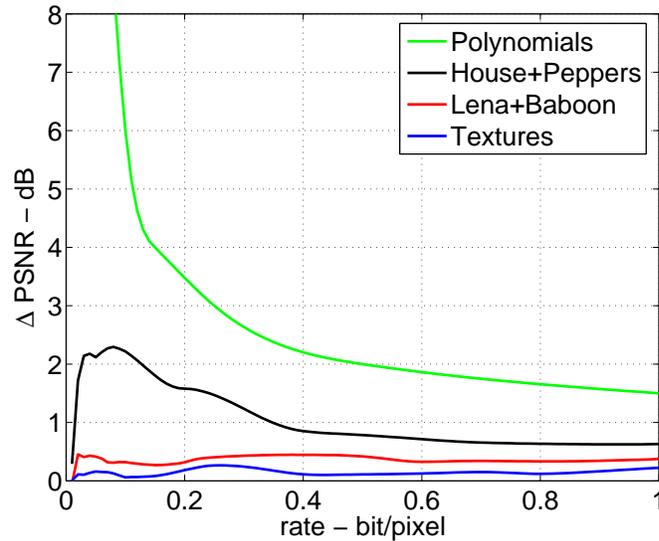
**Figure 2.12:** PSNR gain of OB-coding w.r.t. flat coding for wavy-tile mosaics.

but all gains are now much smaller because of the reduced number of objects and the straight boundaries, and even with the polynomial mosaic there is only a 2 dB gain at high rates.

## 2.5 Performance with real-world images

In order to isolate and analyze in depth the phenomena of interest, the experiments carried out in the preceding sections dealt with ideal and sometimes limiting cases. Now, we focus on the performance of the whole coding scheme in real-world situations, thus including the image segmentation, with all its inaccuracies.

In these experiments, we consider the image Peppers of Fig. 2.3 (c) because its segmentation in a reasonably small number of meaningful objects is somewhat simpler. As a side effect, some objects comprise just one or a few smooth and coherent surfaces, which makes Peppers a more favorable case w.r.t. other, more complex, images. In any case, the choice of what represents an object is somewhat arbitrary, and therefore we use several segmentation

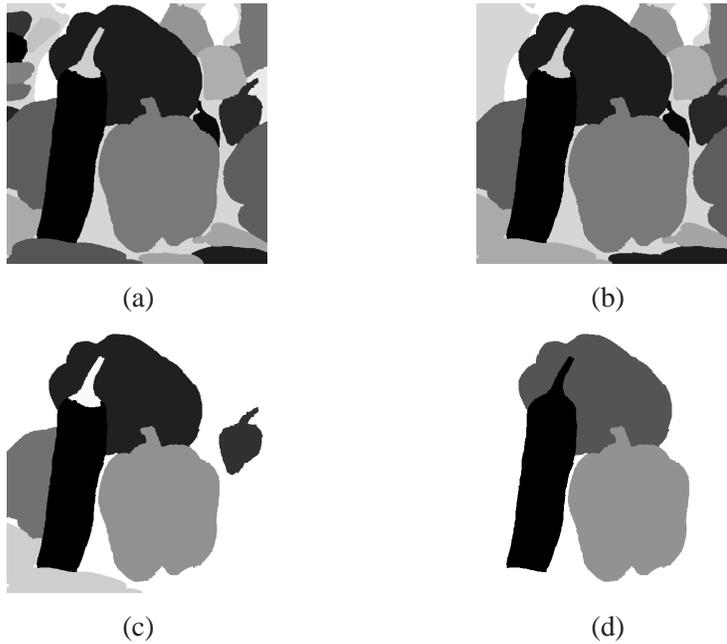


**Figure 2.13:** PSNR gain of OB-coding w.r.t. flat coding for square-tile mosaics.

maps, with a different number of objects, shown in Fig. 2.14 from the most detailed (25 objects) to the simplest one (just 4 objects, including the background).

Our object-based coding scheme provides the RD curves shown in Fig. 2.15 together with the curve for the flat coder. Results might seem a bit disappointing at first, since the flat coder is always the best, but this is easily justified. In fact, even neglecting the unavoidable segmentation inaccuracies, it must be considered that, with ordinary images, the object boundaries are rarely clear-cut, due to the combination of the object 3-d geometry and the illumination, and also to the limited resolution of the sensors that causes some edge smearing. Of course, this erodes the gains of removing strong edges. In addition, when objects have a semantic meaning, their interior is typically not uniform (just think of the bright glares within each pepper), and therefore the WT does not benefit much from the segmentation. On the other hand, when the segmentation map becomes very accurate, so as to single out regions that are actually uniform, the cost of side information increases significantly.

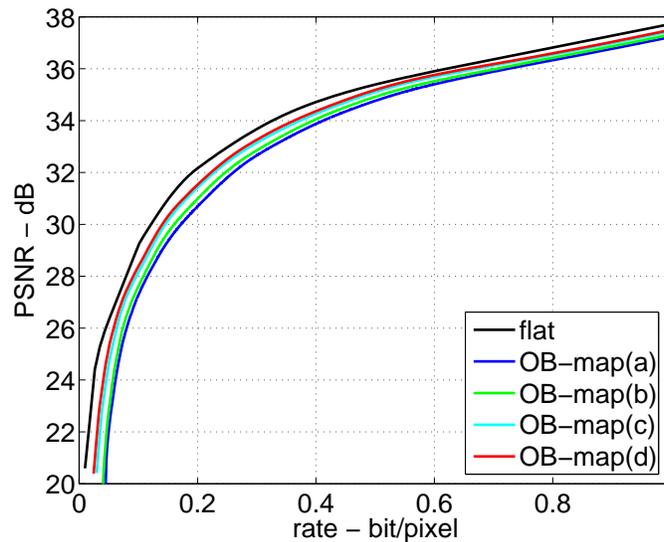
In this light, the object-based RD curves of Fig. 2.15 can be considered



**Figure 2.14:** Segmentation maps for image Peppers with 25 (a), 16 (b), 8 (c), and 4 (d) objects.

reasonably good, with a loss of no more than half dB at medium-high rates, and somewhat more at the lower rates, when the cost of side information is proportionally more relevant.

It is also interesting to consider the visual quality of compressed images, and to this end, in Fig. 2.16 we show the image Peppers compressed at 0.05 bit/pixel with WT/SPIHT (a) and with our object-based coder using the simple segmentation map of Fig. 2.14 (b). Such a low rate was selected in order to emphasize the differences of the two approaches, which at higher rates tend to disappear. The first image has a better PSNR (26.3 vs. 25.2 dB), but the second one has a superior perceptual quality, at a first look, because major edges have been better preserved. At a closer inspection, however, the object-based coded image presents a slightly worse texture quality, due to the lower effective rate available, and especially some annoying artifacts at the diagonal boundaries, which appear unnaturally rugged. This last problem could be easily overcome by some directional filtering. Needless to say, if one concentrates most coding



**Figure 2.15:** RD performance of flat and object-based coding for image Peppers.

resources on a single object considered of interest, neglecting the background, the object-based approach shows an overwhelming superiority.

The object-based coder we have analyzed uses what are probably the most well-know and widespread tools in this field, but other object-based coders have been proposed recently, and it is therefore interesting to carry out a performance comparison. We therefore repeated the experiments of Fig. 2.15 using various algorithms: WDR [42], TARP [45], OB-SPECK [43], and BISK [47], implemented in the Qcc library [50] freely available at [51]. All these algorithms are based on a SA-WT [38] very similar to Li and Li's SA-WT, and encode the coefficients by means of embedded bit-plane coding algorithms.

The best performance is exhibited by BISK, based on the shape-adaptive version of SPECK, from which it differs for two main innovations: the use of a more flexible binary rather than quaternary splitting of blocks, and the introduction of a bounding box to help discard nodes outside the object of interest. BISK proves also superior to SA-SPIHT, as appears from the curves of Fig. 2.17, obtained with the map of Fig. 2.14 (d). The gap, however, is partially due to BISK use of arithmetic coding for the output stream. When



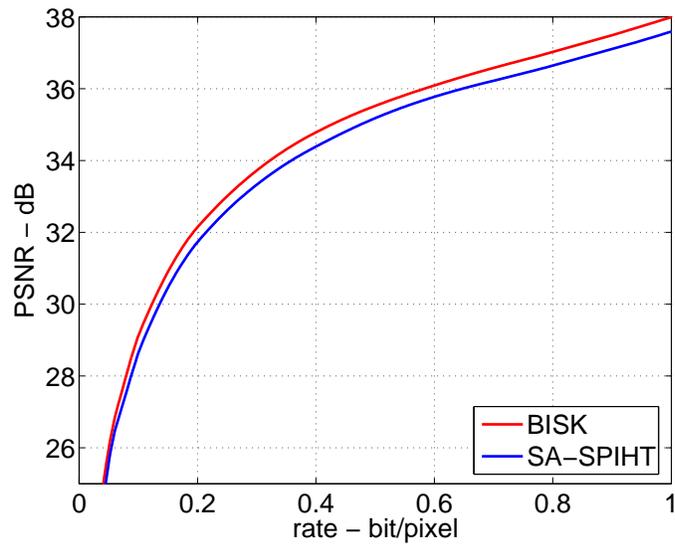
(a)

(b)

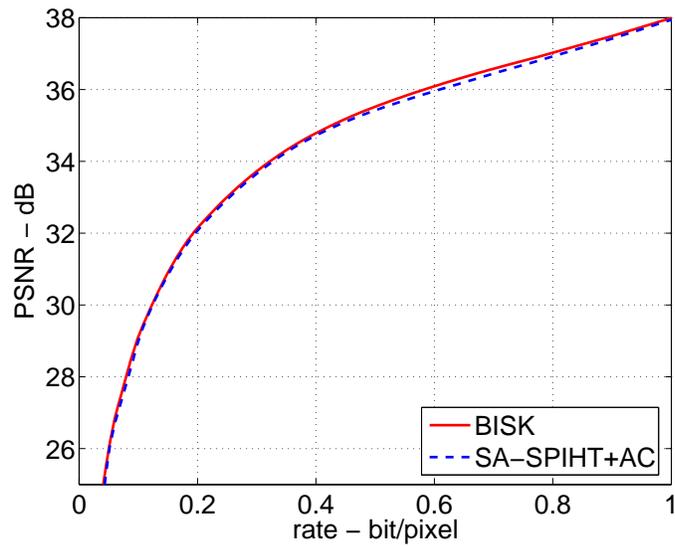
**Figure 2.16:** Image Peppers compressed at 0.05 bit/pixel with flat (a) and object-based (b) coding.

we introduce a similar coding step after SPIHT the difference becomes very limited, Fig. 2.18. This had to be expected, if losses are mostly related, directly or indirectly, to the SA-WT performances, and this is the same for the two coders.

Our analysis showed that the gains can be significant when the image presents sharp edges between relatively homogeneous regions but also that this is rarely the case with real-world images where the presence of smooth contours, and the inaccuracies of segmentation (for a few objects) or its large cost (for many objects) represent serious hurdles towards potential performance gains. For these reasons we want to explore the capability of this coding approach on multispectral images, where the segmentation produces regions with nearly uniform statistics, the cost of the segmentation map is shared among many bands, and hence the conditions are such that object-based coding can actually provide some rate-distortion gains.



**Figure 2.17:** RD performance of BISK and SA-SPIHT for image Peppers.



**Figure 2.18:** RD performance of BISK and SA-SPIHT with Arithmetic coding for image Peppers.

## 2.6 Multispectral images

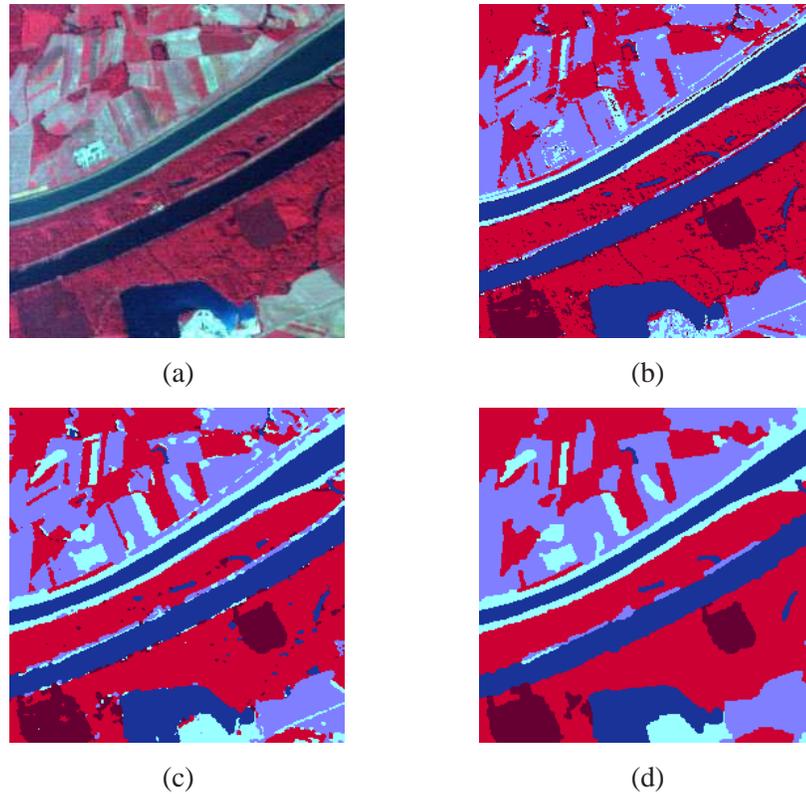
### 2.6.1 Class and region-based approaches

Multispectral images are characterized by better and better spatial, spectral and radiometric resolution, and hence by ever-increasing demands of communication and storage resources. Often, such demands exceed the system capacity, like, for example, in the downlink from satellite to Earth stations, where the channel bandwidth is often much inferior to the intrinsic data-rate of the images, some of which must be discarded altogether. In this situation, as well as in many others, high-fidelity compression of the images represents a very appealing alternative. As a matter of fact, there has been intense research activity on this topic in the last few years [52, 53, 54, 55, 56, 4, 57], focusing especially on transform coding techniques, due to their good performance and limited computational complexity. Now we want to extend object-based approach to this category of images.

As already underlined in the previous sections of this chapter, for object-based coding we mean a scheme that codes separately the different semantic objects present in the image. For multispectral images, the concept of semantic object, already critic for natural images, becomes more ambiguous. A better idea is to focus the attention on class-based [58] and region-based [59], [60] paradigms.

Although class-based and region-based coding techniques share the same general approach, they differ under some important respects, that is, the number of objects (few classes vs. many regions), their statistical homogeneity and their spatial compactness (sparse classes vs compact regions). To gain insight about this difference consider Fig. 2.19, which shows an example image in false colors (a), its pixel-wise segmentation (b), contextual segmentation (c), and a further processing of this latter map (d). The segmentation of Fig. 2.19 (b) is just a clustering of the spectral response vectors associated with each pixel. The output is a set of 5 classes with a strong statistical homogeneity, corresponding to the colors, each one composed by several disjoint regions as well as by many isolated points or small clusters of points that can be hardly regarded as regions. The segmentation of Fig. 2.19 (c) is instead obtained by means of a bayesian contextual technique [61] which penalizes isolated points and provides therefore a smaller number of more compact regions. If one further regularizes such a map, by erasing small regions and smoothing boundaries between different regions, one obtains a map like that of Fig. 2.19 (d), with a limited number of compact regions. For example, in this map there are

just 8 disjoint regions belonging to the same “water basins” class (dark blue).

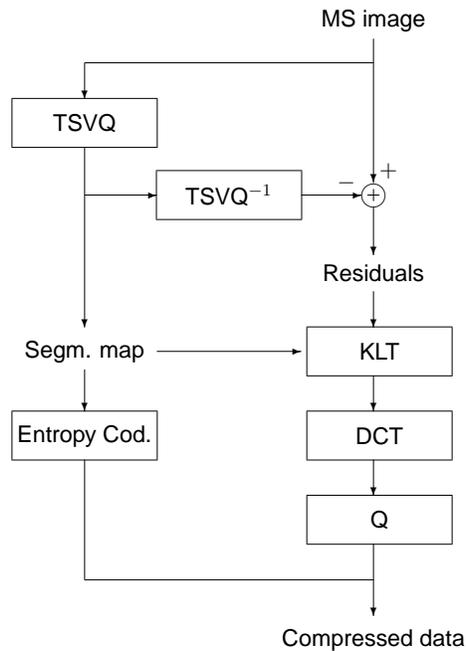


**Figure 2.19:** An example image and its segmentation maps

The number of objects is important if one is interested in adaptive transform coding, because some parameters must be transmitted for each object, and therefore, there is a rapid growth of the side information in the region-based case, which could easily become prohibitive. The statistical homogeneity and the spatial compactness of objects are also important: while classes are singled out mostly on the basis of spectral homogeneity criteria, regions are required to satisfy some additional spatial criteria. For example, an isolated point can hardly be regarded as a region, and must be absorbed in some larger regions with different statistics. Therefore, regions tend to be less “pure” than classes, and a spectral transform is less effective on them. On the other hand, a spatial transform applied to a compact region, rather than to a sparse class, might work better.

### 2.6.2 Class-based scheme

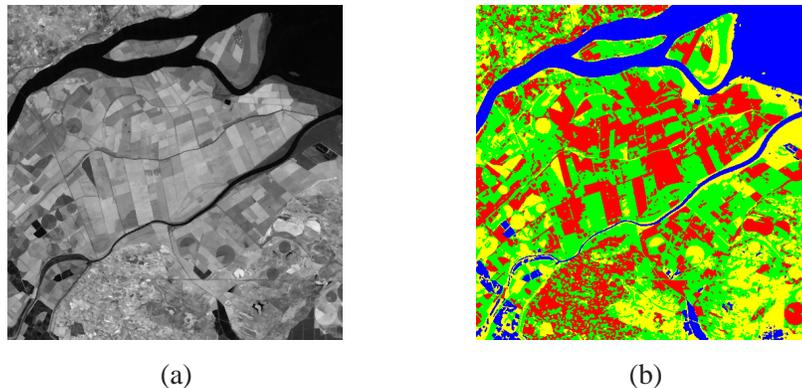
A class-based coder for multispectral images (referred to from now on as CBC) was already proposed by Gelli and Poggi in 1999 [52]. In [52] the authors start from the observation that linear transform coding does not take into account the *nonlinear* dependences existing among different bands of a multispectral images, due to the fact that multiple land covers, each with its own interband statistics, are present in a single image; so they try to address this problem by segmenting the image into several classes, corresponding as much as possible to the different land covers of the scene. As a consequence, within each class, pixels share the same statistics and exhibit only *linear* interband dependencies, which can be efficiently exploited by conventional transform coding.



**Figure 2.20:** The original class-based coding scheme.

The coding scheme is summarized in Fig. 2.20: the segmentation is carried out by means of tree-structured vector quantization (TSVQ) and the resulting map is encoded without loss of information. TSVQ segmentation also provides a rough encoding of the image through the class means, which are subtracted

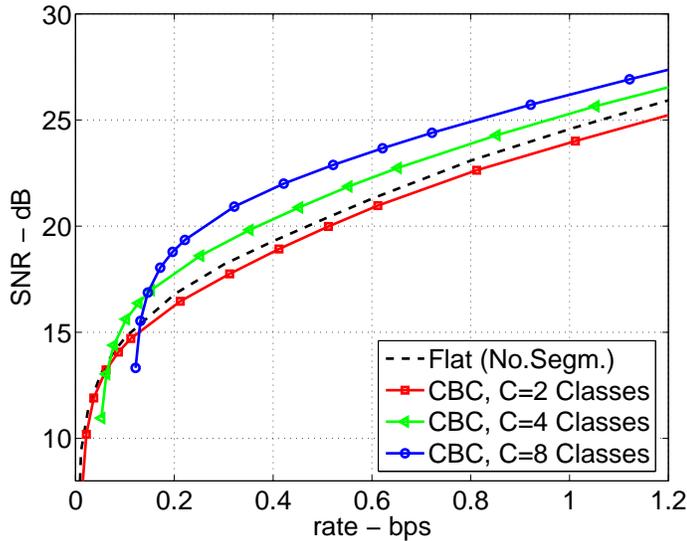
from the data. All residuals of the same class are then pooled together, and are subject to a Karhunen-Loeve transform (KLT) along the spectral dimension, DCT in the spatial domain and finally scalar quantization of the coefficients with optimal bit allocation. To take into account the different spectral statistics of the classes, a different KLT matrix is used for each class, so as to better compact energy in the first few transform bands.



**Figure 2.21:** The Landsat TM test image: (a) band 5, (b) 4-class segmentation.

Experiments on several multispectral test images proved this technique to have an excellent rate-distortion performance. As a running example, we consider a Landsat TM image (6 bands, 512x512 pixels, 8 bit/sample) of a region near Lisbon, a band a which is shown in Fig. 2.21 (a). Fig. 2.22 reports the rate-distortion curves (SNR vs coding rate in bit/sample) obtained by CBC when  $C = 2, 4, 8$  classes are used in the segmentation and subsequent phases. For comparison, the curve obtained without segmentation is also reported. We see that the class-based coder significantly outperforms the reference “flat” coder, especially when a relatively large number of classes is used. CBC curves start from an unfavorable rate-distortion point, due to the cost of the segmentation map, but then the SNR grows rapidly because the classified data are much more homogeneous and easier to encode. It is worth pointing out that the segmentation map is by itself a valuable piece of information for the end user, and is included automatically in the coded stream.

The weakest point of this technique is in the spatial-domain processing, that is, the DCT of KLT bands, and the subsequent encoding of the transform



**Figure 2.22:** RD performance of the original DCT-based coder.

coefficients. In fact, since TSVQ produces segmentation maps with arbitrary geometry, classes are typically composed by a large number of unconnected regions with irregular shape, as shown for example in Fig. 2.21 (b) for the case of 4 classes, and hence one cannot encode them by simple bi-dimensional DCT. After testing the various viable encoding strategies, including shape-adaptive DCT, it resulted that collecting all pixels of a class in a vector and encoding it as a one-dimensional source provided the best performance<sup>5</sup>. Nonetheless, this linearization tends to destroy all residual spatial dependencies in the data, and hence to impair the overall performance.

Given the encouraging results reported in [52] and its limitations, we want to improve the spatial-domain processing phase of class-based coding technique introducing in the scheme more performing tool as SA-WT and SA-SPIHT, already introduced in previous sections.

Since we want to improve the class-oriented transform coding technique proposed in [52] by acting only on the spatial domain processing, the structure of the coding scheme is the same as that of Fig. 2.20 except for the fact that some blocks are updated with the introduction of new technical solu-

<sup>5</sup>Similar results have been later found independently in [55].

tions. We keep using TSVQ to carry out segmentation, because of its very low complexity and good rate-distortion performance. For the map coding, as done in the section Sec 2.2, we will resort to the algorithm proposed in [48] even though this step has a limited impact anyway on the overall performance. Class-adaptive KLT will not be changed, as it stands at the core of the class-oriented approach. The major innovations take place in the two spatial-domain coding blocks, where DCT (that is, data linearization followed by 1d-DCT) is replaced by SAWT, and scalar quantization by shape-adaptive SPIHT coding (SA-SPIHT).

As already pointed out the actual results of object-based schemes using SA-WT and SA-SPIHT depend on the fragmentation of the map, and eventually on the number of objects used in segmentation. Using more classes one obtains more homogeneous sources for subsequent transform coding, but also a more fragmented map which could lead to coding inefficiencies. This also suggests us to leave open the option of modifying the segmentation strategy, once experimental results are available, in order to produce more regular maps.

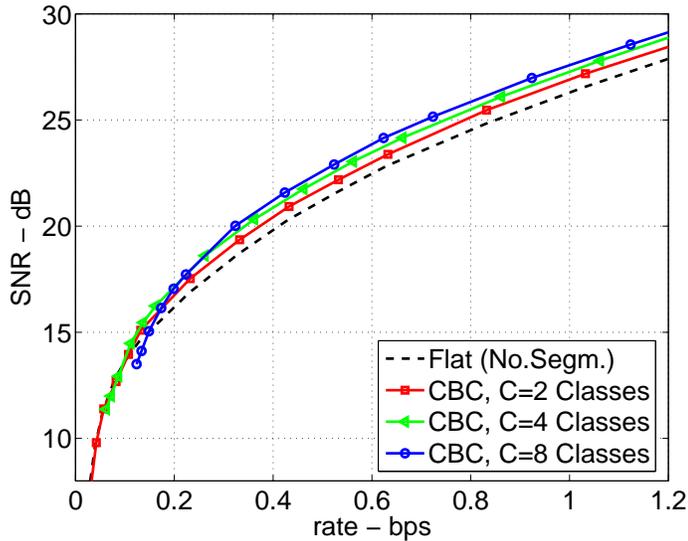
To complete the description of coding scheme, even not shown in the block diagram, a rate allocation block, similar to that described in the Sec.2.2, is used.

### Experimental results

We carried out experiments on several multispectral images and observed always a similar behavior, but results are reported only for the TM image described in the previous section and, later on, for the "Low-Altitude" AVIRIS hyperspectral image (224 bands, 512x512 pixels, 16 bit/sample) available at [62].

In Fig. 2.23 we show the rate-distortion curves obtained with the wavelet-based version of CBC for  $C = 2, 4, 8$  classes and, for comparison, the curve obtained with the the wavelet-based coder without segmentation. It is worth noting that this "flat" coding scheme, originally proposed in [57] can be regarded as a refinement of the 3d-SPIHT coder proposed in [56], improved by using KLT instead of WT in the spectral domain (with better energy compaction) and by resorting to 2d-SPIHT on all spectral bands with explicit rate allocation. It appears that the use of segmentation keeps providing a performance gain w.r.t. the flat coder, although the gap is now more limited, about 1.5 dB at best instead of the 2-3 dB observed with the original coder.

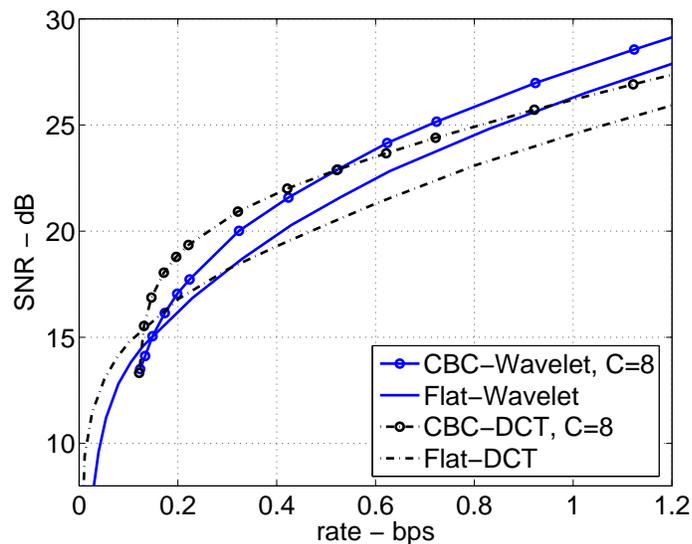
We also want to compare the performance of the new wavelet-based coder with that of the original coder, and hence report in Fig. 2.24 the best rate-



**Figure 2.23:** RD performance of the new wavelet-based coder.

distortion curves for both techniques ( $C = 8$ ) together with those of the two flat reference schemes. First of all, the wavelet-based flat coder consistently outperforms the DCT-based one, as was to be expected given the well-known superior performance of wavelet-based coding for natural images. When we add the segmentation step, that is, consider the class-oriented coders, the comparison is not so clear-cut anymore. In particular, at low bit-rates the DCT-based coder outperforms the wavelet-based one, with an advantage of up to 2 dB at 0.2 bit/sample. The opposite happens at high bit-rates, beyond 0.5 bit/sample, where the wavelet-based coder works better and better, with a gain of 2 dB at 1.2 bit/sample. We can explain this behavior by recalling that shape-adaptive SPIHT is rather inefficient at the beginning, because it deals with a very fragmented map, and must spend many bits to describe significance trees for coefficients scattered over the whole image<sup>6</sup>. In “steady state”, however, it becomes more efficient than 1d-DCT, because spatial relationships among coefficients have been retained and can now be exploited. As a consequence, the slope of its rate-distortion curve is steeper, which makes the new scheme more

<sup>6</sup>To improve performance at low rates, we also experimented with smoother segmentation maps, obtained through morphological filtering of the original maps, but overall results, not reported here for brevity, were disappointing, so we are not going to consider this option anymore.

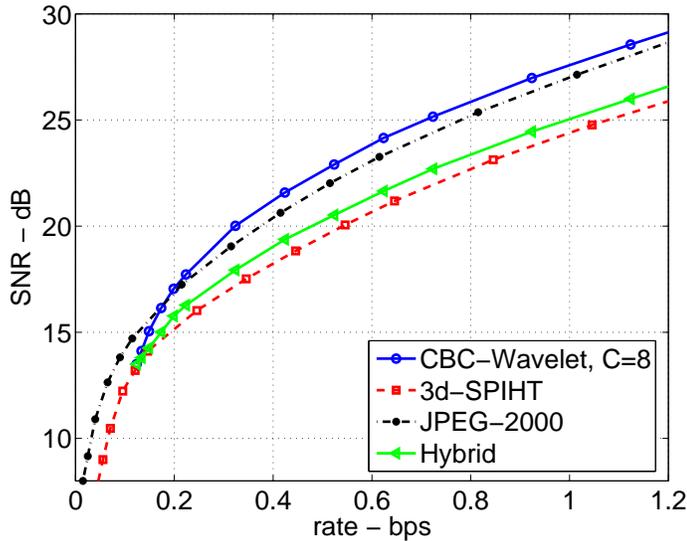


**Figure 2.24:** Performance comparison between wavelet-based and DCT-based CBC.

and more convenient as the available coding resources grow. In summary, the new wavelet-based scheme is certainly preferable when high-quality images are desired (as happens quite often with remote-sensing imagery, where “near-lossless” compression is typically required) while the original scheme is better when resources are very limited.

We complete this analysis by comparing the performance of the proposed class-based coder with those of several significant reference schemes, that is 3d-SPIHT [56] and JPEG-2000 [20], both implemented using the KLT as a spectral transform, and finally an hybrid coding scheme in which the class-based approach is used to perform the spectral KLT, but then the transformed eigenimages are coded by flat JPEG-2000 with proper rate allocation. Results are reported in Fig. 2.25, and show that the proposed wavelet-based CBC outperforms all reference schemes, by less than 1 dB in the case of JPEG-2000, and up to 3 dB for 3d-SPIHT. The hybrid scheme remains well under CBC and JPEG-2000, suggesting that, once segmentation is carried out, one should try to exploit the class information in all instances to make up for its initial cost.

Finally, we present compression results for a different source, the AVIRIS

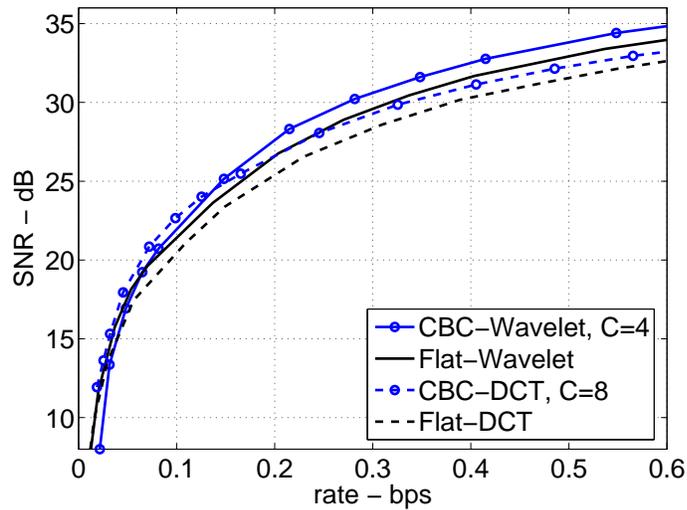


**Figure 2.25:** Performance comparison between wavelet-based CBC and various reference coders.

hyperspectral image “Low Altitude”. We use only 192 out of the total 224 bands, since some bands are clearly useless and must be discarded altogether. For this image, it is also convenient to normalize bands to zero mean and unit variance before encoding, since the rate allocation procedure would otherwise give little or no resources to low-power but informative bands, with a detrimental effect on the image quality<sup>7</sup>. In Fig. 2.26 we report results for the CBC coder, the flat reference scheme, and JPEG2000 multicomponent, and observe a behavior quite similar to that observed for the Landsat TM image. Note that there are just a few distinct information classes in this image, and therefore the wavelet-based CBC provides the best performance with 4 classes rather than 8. JPEG-2000 is quite competitive, if KLT is used to decorrelate bands prior to encoding, with a performance very close to that of wavelet-based CBC, losing just 0.6 dB at high rate. On the contrary, JPEG-2000 with WT in the spectral domain exhibits a performance gap of about 4 dB w.r.t. the other techniques.

In conclusion, the class-based coding approach guarantees always a very

<sup>7</sup>Note that the SNR obtained with this pre-processing is necessarily lower than that obtained without normalization.



**Figure 2.26:** RD curves for the AVIRIS image.

good performance, because the KLT is much more effective on homogeneous sources than on the whole image. Its weak point is the spatial transform and coding, since land covers, and hence classes, are usually scattered over the whole image. The use of shape-adaptive wavelet transform and shape-adaptive SPIHT in place of 1d-DCT, partially overcomes this problem, and allows one to improve performance at medium and high bit-rates, that is, for the high-quality compression typically required by the end users.

### 2.6.3 Region-based scheme

One of the major problems for the final user of multispectral images is to locate them effectively. This is the main reason towards the use of a region-based, rather than pixel-based, description of remote sensing images, so that the user can retrieve the image of interest based on the shape or the synthetic statistics of some of its regions. In other scenarios, the user might even be interested in downloading only the image segmentation map, or just a few regions of interest, with huge bandwidth saving in both cases. So the major interest of region-approach is that it allows a semantics-based access to image content, nevertheless, we are interested to the effect that the region-based paradigm has

#	No	Mean	KLT	Cod.	SI TM	SI AVI
1	Yes	G	G	G	$\approx 0$	0.003
2	Yes	G	G	C	0.012	0.003
3	Yes	C	C	C	0.012	0.006
4	Yes	G	G	R	0.012	0.003
5	Yes	C	C	R	0.012	0.006
6	Yes	R	R	R	0.026	0.031

**Table 2.3:** Reference and region-based coding schemes, and side information (bit/sample) for Landsat and AVIRIS images. G=global, C=class-based, R=region-based.

on rate-distortion performance.

The coding scheme is similar to that described in the previous section except for the different type of segmentation (see Sec. 2.6.1). As in Sec. 2.6.2 we want to use a different KLT for every region but in this context this choice is more critical. In fact, in class-adaptive KLT, a different transform matrix is computed (and hence transmitted) for each class; this increases the cost of side information, but not very much, since only a few classes are typically present. In region-based KLT, instead, a different matrix is transmitted for each connected region; this allows for a better compaction of signal energy [60], but the cost of side information can become prohibitive. An intermediate solution, which saves some side information, is to send the mean vector for each region, but then use a single KLT matrix for all regions of the same class.

## Experimental results

In order to study the potential of region-based approaches, we select three such schemes and three reference schemes, as summarized in Tab. 2.3, and apply them to two quite different test images, a 6-band,  $512 \times 512$  pixel section of a Landsat TM image (already used in previous section Fig. 2.21 (a)), and a 32-band,  $512 \times 512$  pixel section of an AVIRIS image of Moffett-fiel (Fig. 2.27). We used a map with 3 classes and 59 regions for the TM image and 4 classes and 28 regions for the AVIRIS image.

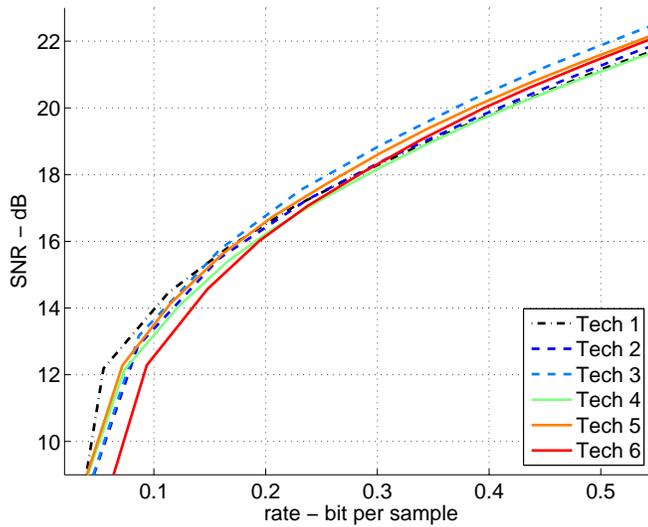
The first reference scheme is just a conventional “flat coder”, with spectral KLT, spatial WT, and band-by-band SPIHT with explicit rate allocation. The following two schemes are class-based coders, which differ in the amount of adaptivity offered and side information required. The first one (#2) uses global

KLT after removing the global mean vector; #3 uses a different KLT matrix for each class, after removing the class mean vectors. Coding scheme #6 is the only “pure” region-based coder, as it carries out all transform and coding steps on a per-region basis. The other region-based coders resort to some compromise to reduce side information: #4 performs a global KLT, while #5 performs a class-based KLT, removing means class-by-class. The last two columns of the table show the corresponding side information for the Landsat TM and the AVIRIS image, expressed in bit per sample. The side information mainly accounts for the KLT matrices and the segmentation map information. For the 6-band TM images the map cost is prominent, as it is shared among a relatively small number of bands. On the contrary, for the 32-band AVIRIS image, the main contribution to the side information is due to the KLT matrices. Anyway it becomes significant only for the last technique, when we need a different KLT matrix for each region, but even in this case this cost is quite low – about 0.03 bit per sample.



**Figure 2.27:** AVIRIS image, false colors

Fig. 2.28 and Fig. 2.29 report the rate-distortion curves of all these schemes for the Landsat and, respectively, the AVIRIS test image. We observe that at low bit-rates, flat and class-based schemes have the best performances, thanks to their low side-information requirements. However #2 is always worse than #3, meaning that classified KLT is worth its cost with respect to global KLT. This is still true for the region-based techniques, where #4 is worse than #5 and #6 (but for low rates). At medium rates, the flat scheme is overcome by the class-based and region-based techniques; moreover, we see that region-based techniques #5 and #6 surpass class based techniques #2 and approach performances of #3, which however can be considered the best technique overall. We can conclude that the more complex description of image objects carried

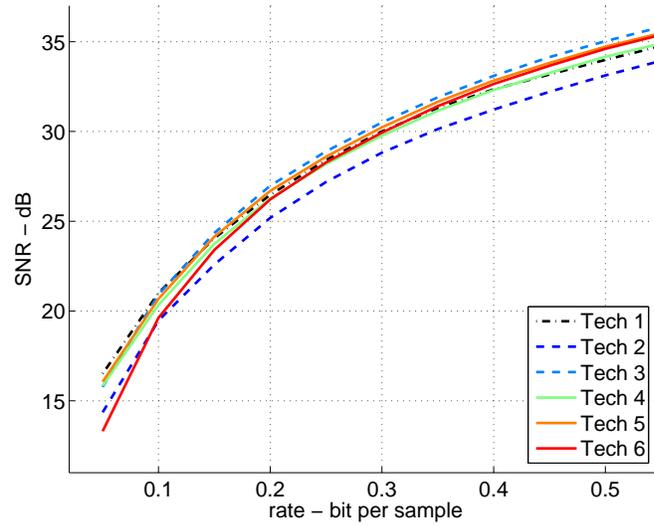


**Figure 2.28:** Performance of test techniques on the TM (Lisbon) image.

by class-based and region-based techniques causes a global improvement of performances with respect to a completely flat technique as #1. The best compromise between cost and effectiveness of shape information seems to be the class-based technique #3, but the region-based techniques (namely #5 and #6) have quite close performance, except for very low bit-rates.

Similar considerations can be made for performances on the AVIRIS image, see Fig. 2.29. The main difference with respect to the previous case is that now, region-based techniques performance is even closer to the best class-based technique #3. We ascribe this behavior to the smaller number of objects, that reduce side-information, and the higher number of bands, that makes more important the effectiveness of a precise spectral transform.

Our experiments lead to the conclusion that the region-based approach does not penalize significantly RD performance with respect to classical, non object-based schemes; on the contrary, their performance is usually better than that of completely flat approaches, and always quite close to the class-based one, which optimize the trade-off among cost and effectiveness of object-based transforms.



**Figure 2.29:** Performance of test techniques on the AVIRIS (Moffett-field) image.

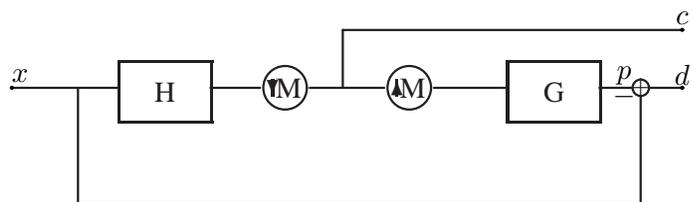
## Chapter 3

# Coding with contourlets

The contourlet transform was recently proposed to overcome the limited ability of wavelet to represent image edges and discontinuities. Besides retaining the desirable characteristics of wavelet transform, such as multiresolution and localization, it has two additional important features: directionality and anisotropy. In this chapter, after presenting principal characteristics of contourlet transform and introducing the motivation for an hybrid contourlet-wavelet decomposition, we propose a new image coding technique based on this transform and built upon the well-known SPIHT algorithm. Numerical results and visual quality are reported to confirm the potential of this approach, especially for images with high texture content.

### 3.1 Contourlets

In the Chapter 1, we noted how recent researches have identified in wavelet lack of directionality and anisotropy the main cause of its inefficiency in describing bi-dimensional smooth contours, and how new transforms, instead, aspire to these features. In the last years a lot of different transforms have been proposed (contourlets [6], directionlets [7], curvelets [2], bandelets [8], etc.), that overcome wavelet limits in representing images contours. The results are mostly theoretical, focusing only NLA asymptotic rate, but they are quite promising, and stimulate the quest for actual coding algorithms based on these new tools. We chose the contourlet transform [6] for several reasons: it has a number of desirable characteristics (directionality, anisotropy, etc.) and an almost optimal NLA behavior for simple classes of images; in addition, unlike other transforms, it is easily implemented by a filter bank. Its main



**Figure 3.1:** Laplacian pyramid.

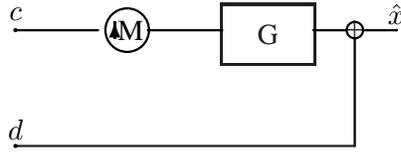
drawback is a slight redundancy which, however, is not really a problem in the context of low bit-rate coding [28].

In the first place, it is worthwhile to recall some basic rudiments about contourlets. Contourlet transform was introduced by Do and Vetterli in 2005 [6], it comprises two blocks, a Laplacian pyramid and a directional filter bank (DFB).

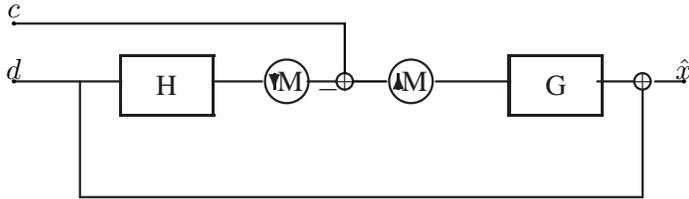
The Laplacian pyramid (LP) was proposed by Burt and Adelson [63] in 1983 as a multiresolution image representation. In the first stage of the decomposition, the original image is transformed into a coarse signal by mean of a lowpass filtering and a downsampling. This coarse version is then upsampled and filtered to predict the original image. The prediction residual constitutes the detail signal (see Fig. 3.1). This procedure can be repeated iteratively in order to obtain a multiresolution decomposition.



**Figure 3.2:** Laplacian decomposition of image Lena.



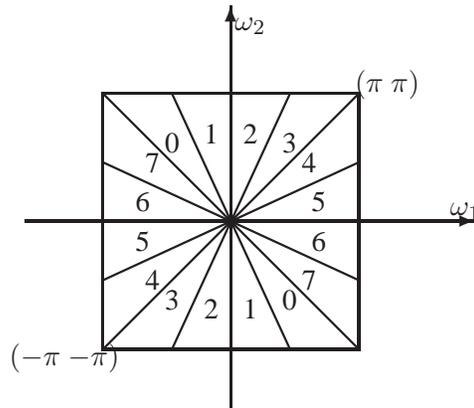
**Figure 3.3:** Simple reconstruction scheme for the Laplacian pyramid.



**Figure 3.4:** Do-Vetterli reconstruction scheme for the Laplacian pyramid.

As we can see in Fig. 3.2 LP decomposition is a redundant representation so it is natural to wonder why in contourlet it has been preferred to critically sampled filterbanks as discrete wavelet transforms. The motivation must be detected in the successive use of Laplacian pyramid; in fact, in contourlet decomposition, a directional filtering is performed on the bandpass versions of input signal. So it needs a decomposition that permits further subband decomposition of its bandpass images. To this target the LP has two advantages over the critically sampled wavelet scheme: first, it generates only one bandpass version, second, it does not suffer from the frequencies “scrambling”. This problem arises in the critical sampling filter banks when, because of down-sampling, the highpass channel is folded back into the low frequency band and its spectrum is reflected [64]. This problem is overcome in LP downsampling only the lowpass channel.

A peculiarity of LP used in contourlet transform is the reconstruction structure. In fact, most applications employ the simple synthesis operator shown in Fig. 3.3 to reconstruct the image from the LP, but this simple synthesis operator is not optimal in terms of minimizing the distortion propagation from the subbands of the LP to the reconstructed image. In contourlet scheme, Do and Vetterli use, instead, a structure which implements the dual frame reconstruc-



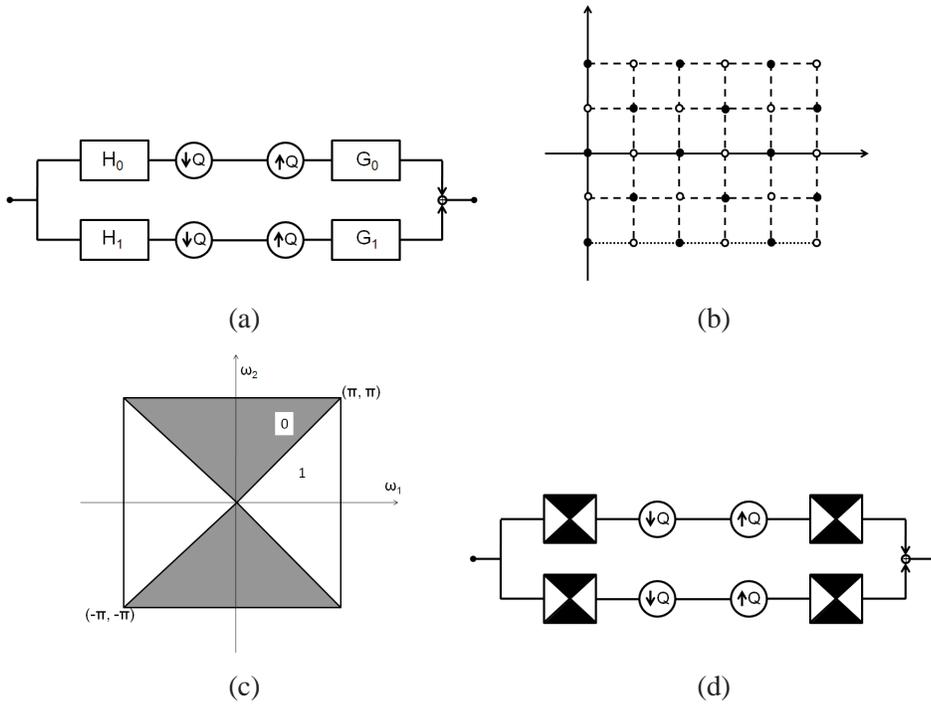
**Figure 3.5:** DFB frequency partitioning with  $l = 3$ .

tion (Fig. 3.4)(LP multiresolution pyramid is in practice a frame expansion), because this is the optimal choice in presence of noise [65].

The second block of contourlet decomposition is a directional filter bank that singles out directional components, with a number of directions that can vary as a power of two. Bamberger and Smith ([66]) introduced a perfect reconstruction directional filter banks (DFB), that can be maximally decimated, implemented via a  $l$ -level tree-structured decomposition that leads to  $2^l$  subbands with wedge-shaped frequency partition. Fig. 3.5 shows an example of DFB frequency partitioning with  $l = 3$ , the subbands 0-3 correspond to the mostly horizontal directions, while subbands 4-7 correspond to the mostly vertical directions.

The version of the DFB used in contourlet is a simplified formulation of Bamberger and Smith filter based only on the quincunx filter banks (QFB) with fan filters. The structure of a QFB is shown in Fig.3.6 (a), its name derives from the type of subsampling (Fig.3.6 (b)). Using a fan filters (Fig.3.6 (c)) the QFB can be used to split the frequency spectrum of the input signal into an horizontal and a vertical channel (Fig.3.6 (d)). The frequency partition of the DFB is realized by an appropriate combination of directional frequency splitting by the fan QFB's and the "rotation" operations done by resampling [64]. In conclusion the entire scheme of contourlet transform is shown in Fig. 3.7.

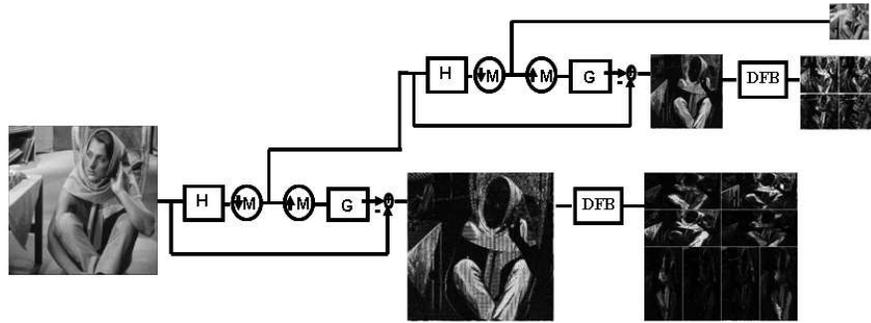
Theoretically, the number of directions in which one can divide the band-pass subbands at each level of decomposition is a free parameter, but, to make



**Figure 3.6:** (a) QFB. (b) Quincunx sampling. (c) Ideal frequency support of the fan filter. (d) QFB with fan filter.

contourlet basis to be anisotropic, as well as directional, a condition must be imposed. In [2], Candès and Donoho demonstrated that basis function supports must respect the parabolic scaling rule,  $width \propto length^2$ , in order to provide a good representation of regular contours. Contourlet expansions respect such a rule if the number of directions doubles at every other finer scale [6]. In addition, if it has enough vanishing moments, its NLA behavior for a  $C^2/C^2$  image model (see. Sec. 1.5), is almost optimal and, more precisely, the NLA approximation error can be bounded as

$$\|x - \hat{x}_M^{(contourlet)}\|_2^2 \leq C(\log M)^3 M^{-2}. \quad (3.1)$$



**Figure 3.7:** Contourlet filter bank. First, a multiscale decomposition by the LP is computed, and then a DFB is applied to each bandpass channel.

## 3.2 Hybrid contourlet-wavelet decomposition

In precedent section, we observed that contourlet transform has all the characteristics that prevent wavelet from representing well image boundaries and that, under certain hypotheses, the contourlet has an almost optimal asymptotic NLA behavior. Despite this, when dealing with typical low-resolution digital images NLA results are not so exciting. We carried out some preliminary experiments on well-known test images (shown in Fig. 3.8), comparing NLA errors for the wavelet and contourlet transforms (black curves vs blue curves). We used 5 decomposition levels in both cases, partitioning the high-pass contourlet subbands (from coarsest to finest) in 4, 4, 8, 8, and 16 components respectively, and adopting the 9/7 biorthogonal filters for wavelet and contourlet LP stage, and the 23/45 biorthogonal quincunx filters [67] for the DFB. Results are reported in Fig. 3.9- 3.12 in terms of PSNR vs. number of coefficients, and show that only for highly textured images contourlets provide a significant gain, while wavelets are superior for all smooth images. Such results, however, are not really surprising, because the parabolic scaling rule assures only asymptotic results for NLA. In addition, the model is less and less accurate as the image resolution reduces through the Laplacian pyramid, since contours are less and less similar to  $C^2$  curves.

Indeed, it has been noted [68] that an hybrid contourlet-wavelet decomposition has often a better NLA behavior on real images than a pure contourlet with parabolic scaling. This observation is confirmed by our experimental results (red curves in Fig. 3.9- 3.12), where we implemented the hybrid scheme



(a)



(b)



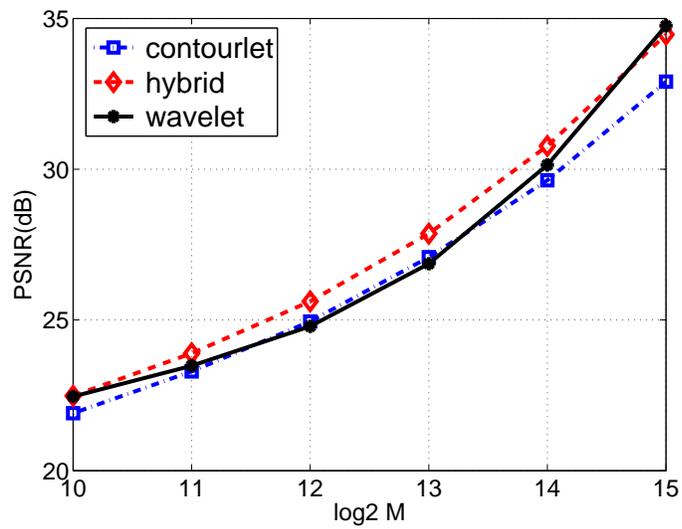
(c)



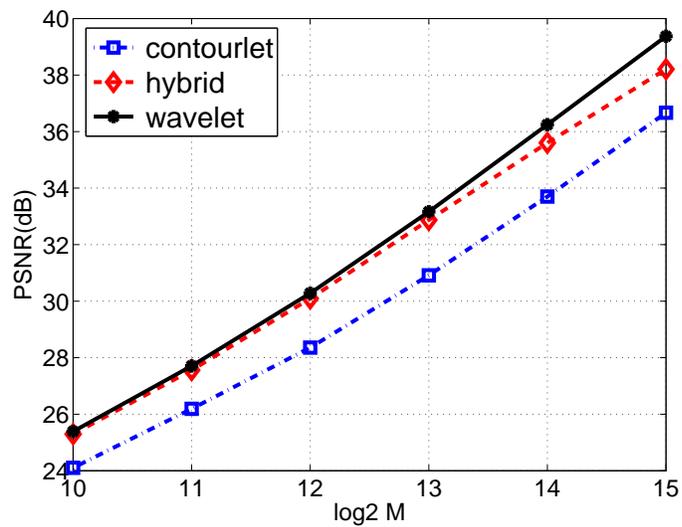
(d)

**Figure 3.8:** Test images.

with wavelets in the coarsest subbands and a contourlet decomposition only in the one or two finest subbands (in particular, for images Lena, Barbara and Goldhill, we have utilized one only level of 32-directions contourlet, while, for image bench, we have utilized a two contourlet levels with 32 and 16 directions). This solution guarantees several advantages: the redundancy is slightly reduced and, since contourlets are used only in the larger bands, contours are more similar to the  $C^2$  curve model and there are less artifacts due to the directional filter length [69]. In the proposed coder we will always consider this solution.



**Figure 3.9:** NLA with wavelets and contourlets for the image Barbara.



**Figure 3.10:** NLA with wavelets and contourlets for the image Lena.

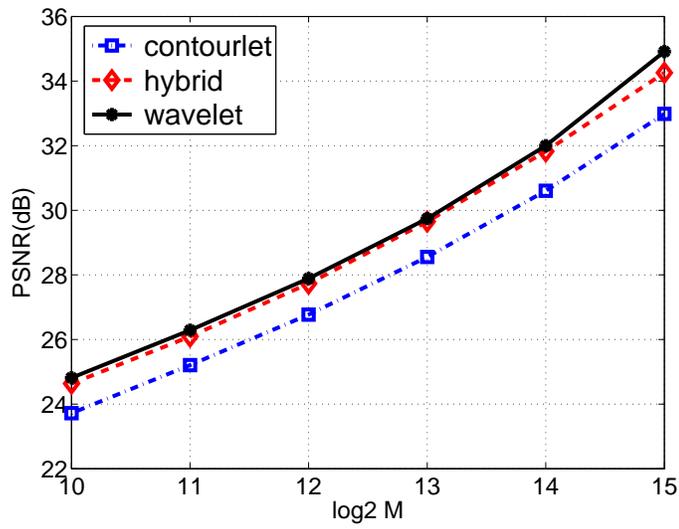


Figure 3.11: NLA with wavelets and contourlets for the image Goldhill.

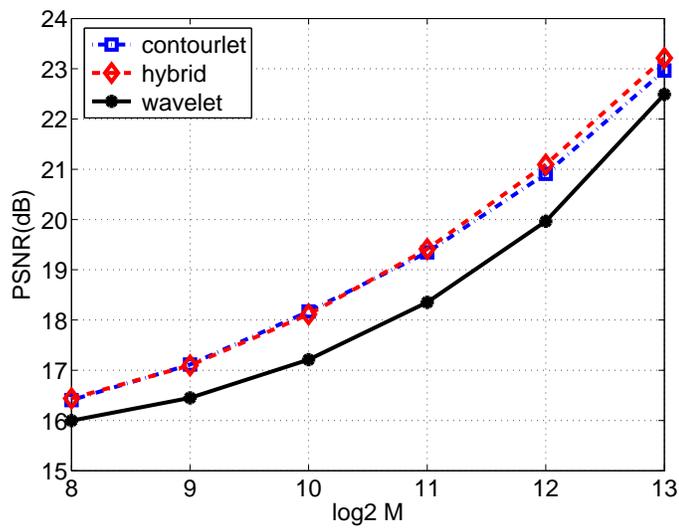


Figure 3.12: NLA with wavelets and contourlets for the image Bench.

### 3.3 A sphit-like algorithm for Contourlets

Just a few contourlet-based coders have been proposed in the literature so far: a bit plane coder based on run-length is described in [70], while [71] uses multistage vector quantization, and [72] focuses on rate-distortion optimization based on a graph-cut algorithm. Here we propose a SPIHT-like coder, using a hybrid wavelet-contourlet decomposition and suitable tree structures that account for the various kinds of dependency existing among transform coefficients [73].

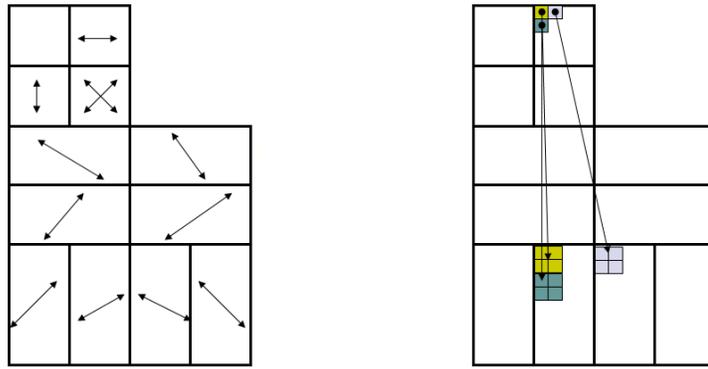
Our coding algorithm is based upon some observations on the properties of contourlet transform coefficients. As noted by Duncan and Do [74], contourlet coefficients exhibit a significant correlation across different subbands, just as it happens for wavelet coefficients, which is why we based our coder on a SPIHT-like engine. As a matter of fact, the coding algorithm is exactly the same used in SPIHT [5], which is well known and will not be discussed here. On the contrary, the *spatial orientation trees* used in SPIHT do not fit the structure and, likewise, given our hybrid decomposition, we cannot rely only on the trees considered in [74] for a pure contourlet decomposition. Therefore the significance trees must be designed anew, and more complex structures are necessary.

Indeed, in [74] it was also observed that there are various types of dependency among coefficients: across scale, space and direction. Therefore, we should define tree structures that do not consider only correlation across scales, but also among spatial neighbors in the same subbands, and homologous coefficients in subbands with similar directions. In more detail, given the selected hybrid decomposition (contourlets in the two finest levels, wavelet in the others) we have to define the father-children relationships for the following situations

1. wavelet-wavelet;
2. wavelet-contourlet;
3. contourlet-contourlet, with  $n - n$  directions;
4. contourlet-contourlet, with  $n - 2n$  directions;

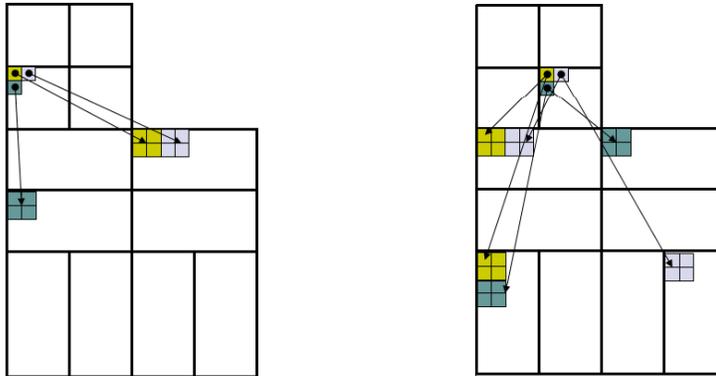
where the last two cases account for the fact that the number of directions in the last level can be the same or double that of the preceding one.

For the first case, our coder resorts to the conventional SPIHT tree. The second case is the most complex because the relationship between the high-pass subbands of a wavelet decomposition and the contourlet directional subbands is not trivial, and even the ratio between the number of coefficients is



(a) Direction of details.

(b) Tree for horizontal details.



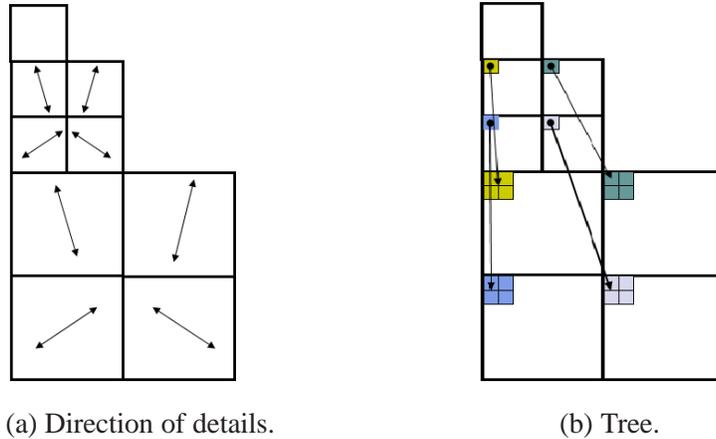
(c) Tree for vertical details.

(d) Tree for oblique details.

**Figure 3.13:** Wavelet-contourlet case with 8 directions.

3:16 rather than the typical 1:4. For the sake of simplicity, such a relationship is described with reference to an 8-direction contourlet decomposition: the cases with more directions, *e.g.*, 32, more interesting for actual coding, present obvious differences. In Fig.3.13(a) the involved subbands (wavelet or contourlet) are symbolically shown, with arrows characterizing their dominant directions. For example, the high-pass wavelet bands are characterized by horizontal, vertical, and diagonal (45 and 135 degrees) arrows, while the contourlet bands have more precise directions. Fig.3.13(b)-(d), instead, show examples of the father-children relationships when the father belongs to horizontal, vertical and diagonal wavelet subbands. Note that this tree-structure takes into account

all types of dependency mentioned before, first of all, the directionality, since we link the horizontal [vertical/diagonal] wavelet details with mostly horizontal [vertical/diagonal] contourlet details. Moreover, we consider correlation across scales by choosing father and children with the same spatial location, and correlation in space, by grouping children in sets of 2 coefficients.



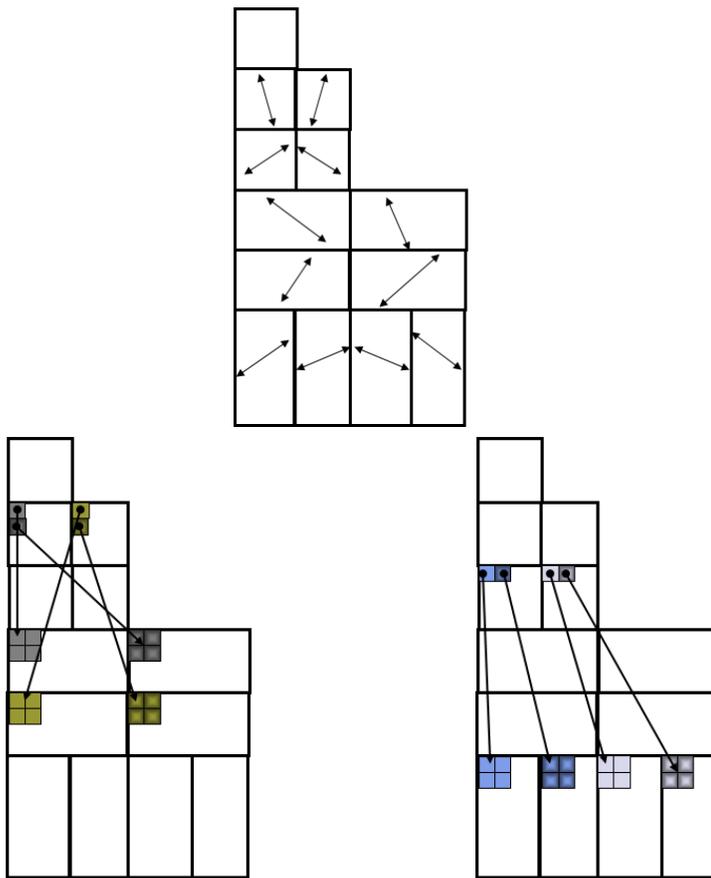
**Figure 3.14:** Contourlet-contourlet with 4-4 directions.

The trees for the contourlet-contourlet case are simpler and follow the same principles: we link coefficients with same spatial location and same direction (fig.3.14). In the last case the tree is only slightly more complex: we connect the coefficients of a direction in the coarser scale with those of the two most similar directions in the finer one (Fig.3.15).

### 3.3.1 Experimental Results

In Fig. 3.17-3.20 we report the rate-distortion curves obtained on the test images using the proposed coder and the conventional wavelet/SPIHT coder. In both cases, an arithmetic coder is also used, similar to that described in [5]. For Barbara and Bench, our coder outperforms the conventional one at low rates (up to 0.25 bpp) and is comparable at higher rates, while its performance is always slightly worse for Lena and Goldhill.

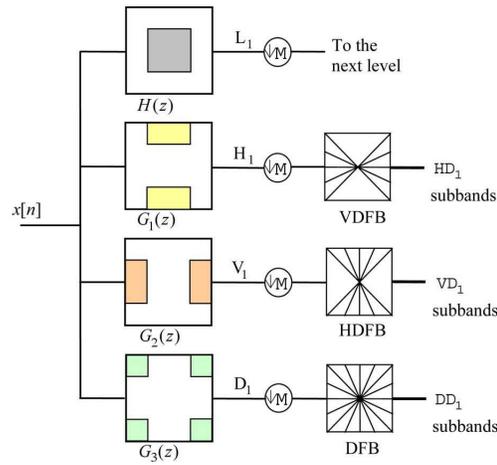
At first, such results might seem disappointing, but it is worth pointing out that our rate-distortion curves follow closely the NLA curves reported in Fig. 3.9-3.12. This suggests that the proposed SPIHT-like coder does actually



**Figure 3.15:** Contourlet-contourlet case with 4-8 directions.

a good job in compacting information, even in the presence of a slightly redundant transform, and the main source of inefficiency is the transform itself, probably because the low resolution test images are quite far from the idealized  $C^2/C^2$  model.

This positive judgement is reinforced by results obtained with a different coder based on the HWD (hybrid wavelets and directional filter banks) transform [69]. The HWD structure is very similar to contourlet one with the difference that LP pyramid is replaced by the separable wavelet decomposition. In this way a critical sampling transform is obtained but at the price of a major presence of pseudo-Gibbs phenomena artifacts due to the problem of



**Figure 3.16:** The HWD-half transform.

frequency scrambling. To address this problem two strategies are suggested in [69]. First of all, directional filtering is performed only on the one-two finest level of wavelet transform, second, in one of its versions called half-HWD, different directions are foreseen on different wavelet subbands [69]: only horizontal directions on vertical wavelet subband, vertical directions on wavelet horizontal subband and a full DFB on oblique wavelet details (Fig 3.16). Another version of HWD, named full-HWD, exists; in it only the first strategy is used and a complete DFB is employed on all wavelet subbands.

In our experiment we used the full-HWD, because, on image Barbara, it is more performing, being the problem of pseudo-Gibbs artifacts more marked on smooth images while Barbara is a very directional image. It is possible to notice that the NLA quality of HWD transform is superior to that of the hybrid contourlet used here (see Fig.3.22 referring to Barbara), but rate-distortion results are comparable to ours, suggesting that a better coding algorithm than that of proposed in [69] is probably at hand.

Finally, it is worth remembering that a major motivation for using the contourlet transform is the higher emphasis given to contours and textural components, leading to a visual quality that is often superior to that of wavelet/SPIHT even for the same objective performance. Indeed, this behavior is confirmed for our coder as is clear by the details shown in Fig.3.21, where a superior visual quality is apparent not only for the highly textured Barbara, but also for the much smoother Goldhill.

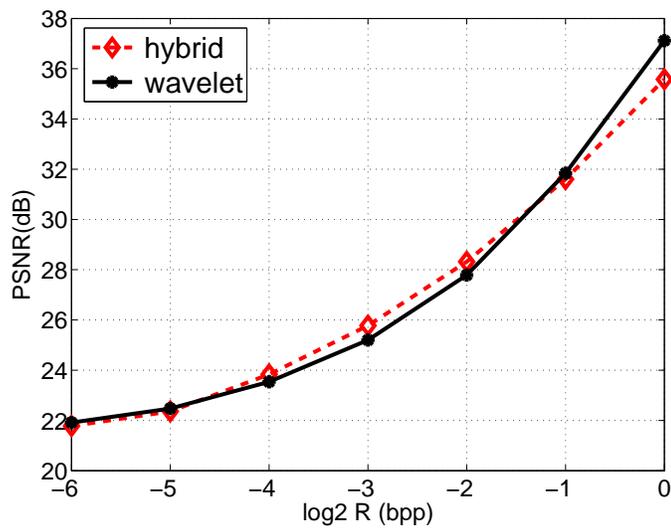


Figure 3.17: Rate-distortion performance for the image Barbara.

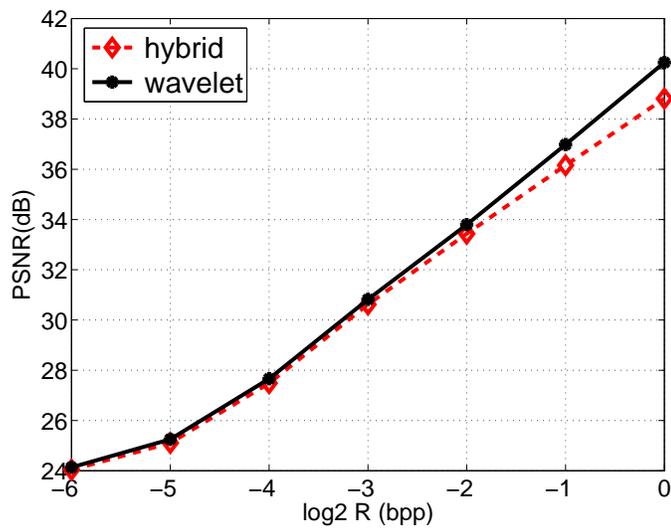


Figure 3.18: Rate-distortion performance for the image Lena.

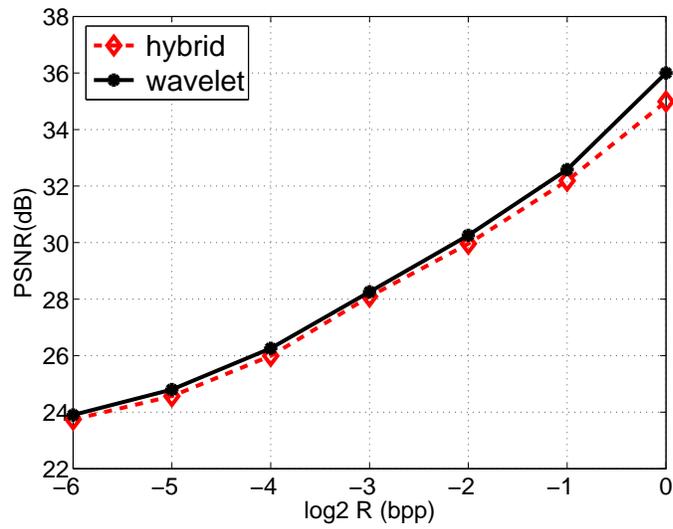


Figure 3.19: Rate-distortion performance for the image Goldhill.

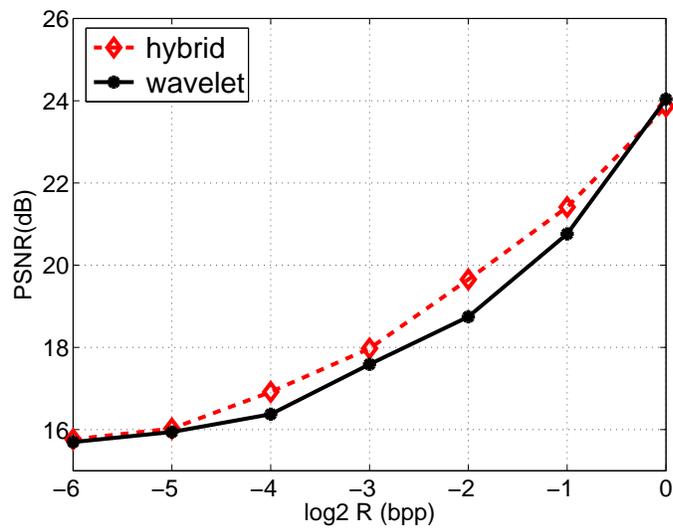


Figure 3.20: Rate-distortion performance for the image Bench.



(a) Contourlet at 0.125 bpp



(b) Wavelet at 0.125 bpp

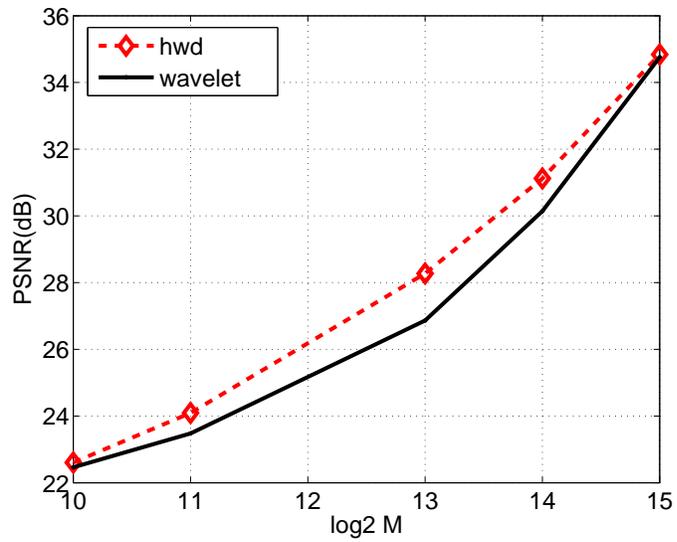


(c) Contourlet at 0.25 bpp

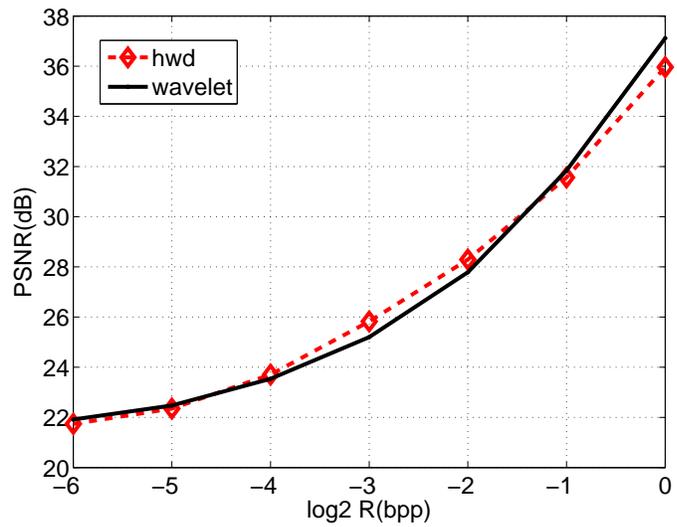


(d) Wavelet at 0.25 bpp

**Figure 3.21:** Details of reconstructed images.



(a) NLA results.



(b) Coding results.

**Figure 3.22:** HWD results for barbara.

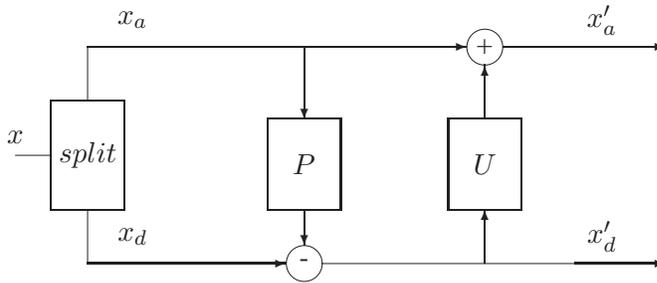
## Chapter 4

# Distortion evaluation in adaptive lifting schemes

The lifting scheme represents an easy way of implementing the wavelet transform and to construct new adapted-content wavelet transforms. So it can be considered another approach to the problem of contours in image coding. However, the adaptive lifting schemes can result in strongly non-isometric transforms. This can be a major limitation, since all most successful coding techniques rely on the distortion estimation in the transform domain. In this chapter, after introducing adapting lifting schemes, we examine the problem of evaluating the reconstruction distortion in the wavelet domain when a non isometric adaptive lifting scheme is used, focusing our analysis on two interesting classes of adaptive lifting scheme. The problem is that these transforms are nonlinear, which prevents using common techniques for distortion evaluation. However we show the equivalence of these nonlinear schemes with time-varying linear filters, and we generalize the distortion computation technique to it. Experiments show that the proposed method allows a reliable estimation of the distortion in the transform domain. This results in improved coding performance.

### 4.1 Adaptive lifting scheme

Lifting structure was originally introduced by Sweldens [75] to design wavelets on complex geometrical surfaces (the, so called, second generation wavelets) but, at same time, it offers an efficient implementation of classic wavelet transforms. In fact, as shown in [9], every wavelet can be implemented



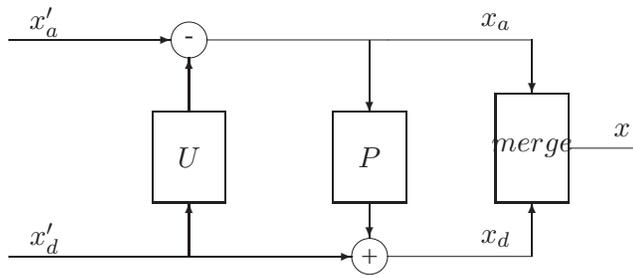
**Figure 4.1:** Classical lifting scheme. Analysis.

by a sequence of lifting steps. One of the main merits of the lifting scheme is to provide a totally time domain interpretation of the wavelet transform and this characteristic makes very simple to design new wavelets.

The blocks that compose the lifting scheme are illustrated in Fig.4.1. The first block splits the input signal  $x$  into an approximation signal  $x_a$  and a detail signal  $x_d$  by the mean of an existing wavelet transform or a simple polyphase decomposition (called lazy wavelet). Then, a prediction operator  $P$  is used in order to predict the current polyphase component from a linear combination of samples of the other component. In practice, the prediction operator  $P$  is chosen such that it is an estimate of  $x_d$  and hence the new signal  $x'_d = x_d - P(x_a)$  is smaller than  $x_d$ . Finally, the third block, the update operator  $U$ , acting on  $x'_d$  modifies  $x_a$ , resulting in an approximation signal  $x'_a = x_a + U(x'_d)$ . Generally, the update operator is chosen in such a way that the approximation signal  $x'_a$  satisfies a certain constraint such as preserving the average of the input  $x$ .

It is interesting to notice that lifting scheme is a tool to construct better wavelet from existing ones, in fact, with a proper combination of lifting steps it is possible to impose new property on the resulting decomposition in order to have an improved wavelet. For example, the lifted wavelet may have more vanishing moments than the original one.

As for the synthesis scheme, shown in Fig. 4.2, it is worthwhile to underline that, since the original signal is reconstructed simply by reversing the lifting steps, perfect reconstruction is assured by the intrinsic structure of the scheme and does not require any particular assumptions on the operators  $P$  and



**Figure 4.2:** Classical lifting scheme. Synthesis.

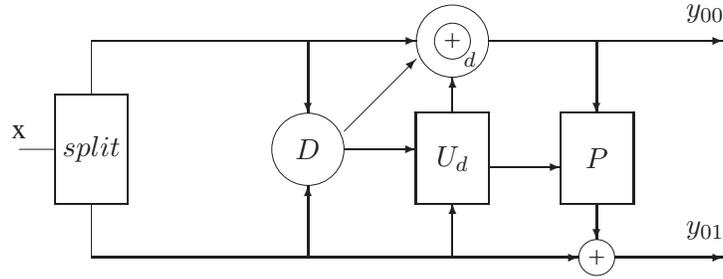
$U$ . In general, even the operators ‘+’ and ‘-’ used in the scheme of Fig. 4.2 can be replaced by any pair of invertible operators.

This great flexibility of lifting scheme offers the possibility to replace linear filters by nonlinear ones and, in particular, to utilize content-adaptive filters [15, 13, 76] similar to that described in the following. In this chapter we will consider two different families of adaptive lifting scheme:

- Adaptive update lifting scheme (AULS)
- Adaptive prediction lifting scheme (APLS)

#### 4.1.1 Adaptive update lifting scheme

The first type of lifting scheme considered has been proposed by and Heijmans, Piella and Pesquet-Popescu [14], [77], [78] and uses seminorms of local features of images in order to build a decision map that determines the lifting update step, while the prediction step is fixed. The general scheme is shown in Fig. 4.3: the polyphase components of the input signal  $x$  are analyzed in order to determine a decision map  $d(k)$ . According to it, different update steps can be performed: for example, when the decision map highlights important features like contours or singularities, a weaker filter (or no filtering at all) can be used. This type of choice could be a useful in a large number of applications in which it is desirable to have multiresolution representations in which important signal features, as discontinuities and singularities, are not oversmoothed.



**Figure 4.3:** Lifting scheme with adaptive update first.

One of the most interesting features of this adaptive transform is that it does not require the transmission of side information, since the decision on the update step can be made with the information available at the synthesis stage. In fact, in [14] authors describe sufficient conditions for this transform to be perfectly reversible without having to send the decision map, which actually can be recovered from the transformed subbands.

We introduce the following notation:  $x$  is the original signal;  $y_{ij}$  is the generic wavelet subband, where  $i \in \mathcal{I}$  identifies the decomposition level starting from 0, and  $j \in \mathcal{J}$  identifies the channel. Usually  $\mathcal{J} = \{0, 1\}$ , with 0 used for the low-pass and 1 for the high-pass channel, but more channels can be used, for example in the case of multi-dimensional transforms. The subbands produced by a single-level decomposition are called  $y_{00}$  and  $y_{01}$ , like in Fig. 4.3. For an AULS, the decomposition is described by the following equations:

$$y_{00}(k) = \alpha_{d(k)}x(2k) + \sum_{n \in \mathbb{Z}} \beta_{d(k)}(n)x(2k+1-2n) \quad (4.1)$$

$$y_{01}(k) = x(2k+1) - \sum_{n \in \mathbb{Z}} \gamma(n)y_{00}(k-n), \quad (4.2)$$

where  $x(k)$  is the input signal and  $d(k)$  is the decision map, which in general can assume  $D$  values in the set  $\mathcal{D} = \{0, 1, \dots, D-1\}$ .

From the previous equations it is easy to find out the synthesis equations

for AULS:

$$x(2k+1) = y_{01}(k) + \sum_{n \in \mathbb{Z}} \gamma(n) y_{00}(k-n) \quad (4.3)$$

$$x(2k) = \alpha'_{d(k)} y_{00}(k) - \sum_{n \in \mathbb{Z}} \beta'_{d(k)}(n) x(2k+1-2n), \quad (4.4)$$

where we used the shorthand symbols  $\alpha'_{d(k)} = 1/\alpha_{d(k)}$  and  $\beta'_{d(k)} = \beta_{d(k)}/\alpha_{d(k)}$ ,

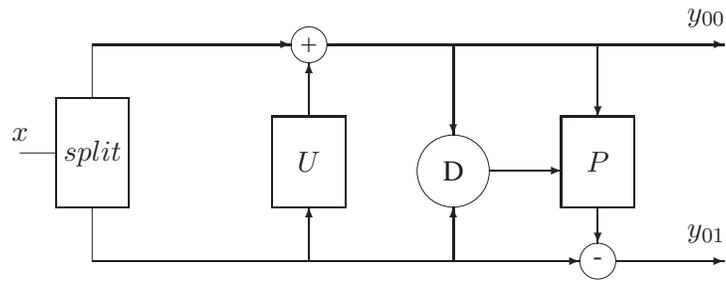
Multiple decomposition levels and wavelet packets can be obtained by applying the same transform of Eqs. (4.1), (4.2) to any subband. We consider only the case of dyadic decompositions (*i.e.* only the low-pass channel is further decomposed) because it is more popular, but our analysis can be easily extended to any decomposition scheme

### 4.1.2 Adaptive prediction lifting scheme

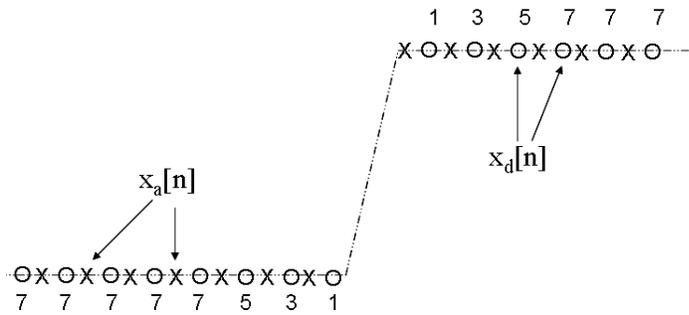
The second class of lifting scheme that we want to analyze is inspired by Claypoole *et al.* lifting scheme [13], because it is well known and achieves good performance. Although our reference is a specified prediction lifting scheme, and not a family, we will consider, where possible, a general form of the adaptive prediction lifting scheme (APLS) that is described in Fig. 4.4 and has the following characteristics:

- update step first;
- the adaptivity involves only the prediction step.

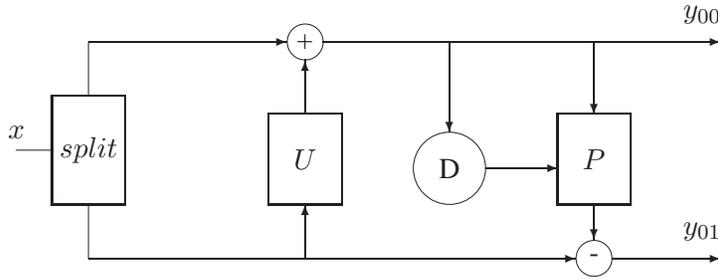
The philosophy of Claypoole's APLS consists in switching between different linear predictors at the predict stage: higher order predictors where the image is locally smooth and lower order predictors near edges to avoid prediction across discontinuities (see Fig. 4.5). In order to guarantee perfect reconstruction (in absence of quantization) at the synthesis stage it is important that the decoder can reproduce all the encoder decisions. To obtain this goal, without sending side information, the update stage is applied first and the decision is based on the approximation signal only. So, even if our theoretical analysis does not depend on the particular type of adaptivity, in the following, we will consider an adaptivity like that shown in Fig. 4.6, namely the decision rely only on one component.



**Figure 4.4:** Lifting scheme with update first and adaptive prediction.



**Figure 4.5:** Predictor selection at an ideal step edge. Numbers indicate the order of the predictors used. The closer to the edge, the lower the order of the predictor.



**Figure 4.6:** Lifting scheme with update first and adaptive prediction.

For an APLS, using the same notation adopted in the previous sections, the decomposition is described by the following equations:

$$y_{00}(k) = x(2k) + \sum_{n \in \mathbb{Z}} \beta(n)x(2k+1-2n) \quad (4.5)$$

$$y_{01}(k) = x(2k+1) - \sum_{n \in \mathbb{Z}} \gamma_{d(k)}(n)y_{00}(k-n), \quad (4.6)$$

while the synthesis is described by:

$$x(2k+1) = y_{01}(k) + \sum_{n \in \mathbb{Z}} \gamma_{d(k)}(n)y_{00}(k-n) \quad (4.7)$$

$$x(2k) = y_{00}(k) - \sum_{n \in \mathbb{Z}} \beta(n)x(2k+1-2n). \quad (4.8)$$

Observing Eqs. (4.1)-(4.2) and (4.5)-(4.6), we can note that in both AULS and APLS, according to the value of the decision map at time  $k$ , we use one out of  $D$  linear update/prediction filters. However, since the decision map depends at its turn on the input signal, the whole systems are inherently nonlinear. Typically, the decision map accounts for the local behavior of the signal, allowing to discriminate low-activity signal segments from highly variable parts. For example in AULS described in [78] the decision map is a threshold function of the local gradient seminorm, while in Claypoole's work [13] the decision is based on the distance of current sample from the discontinuities, which are detected by a modified version of algorithm [79].

## 4.2 Distortion evaluation problem for adaptive lifting

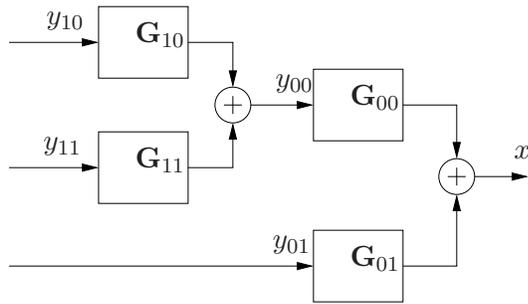
In the adaptive lifting schemes, different filters are used in different parts of image, so the entire transform can be strongly non-isometric. This can be a serious obstacle, since all most successful coding techniques rely on the distortion estimation in the transform domain. For example, the EBCOT [20] algorithm, at the basis of the JPEG2000 standard [80], explicitly uses the wavelet coefficient distortion as an estimation of the reconstructed image distortion in order to compute the resource allocation. Likewise, popular zero-tree based algorithms like SPIHT [5] and EZW [26] perform an implicit resource allocation by encoding first the most significant bits of transformed coefficients: this is efficient only if the distortion estimated in the transform domain is a good approximation of the distortion for the reconstructed image.

From these observations we conclude that, in order to efficiently use the adaptive lifting scheme for image compression, we need to correctly estimate the distortion directly from the transform coefficients in this case as well. Usevitch showed how this can be done for generic linear wavelet filter banks [81]. In particular, it demonstrated that once the equivalent polyphase representation of a generic lifting scheme has been found, the distortion  $D$  in the original domain is related to the distortion  $D_{ij}$  in the wavelet subband  $y_{ij}$  by the relation:

$$D = \sum_{ij} w_{ij} D_{ij}, \quad (4.9)$$

where the weights  $w_{ij}$  are computed based on the reconstruction polyphase matrix of subband  $y_{ij}$ .

The problem is that the AULS and APLS are nonlinear systems, therefore no polyphase representation of them can exist. However, if we forget about the dependence of  $d(k)$  on  $x$  and just look at Eqs. (4.4) and (4.8), we can see them as a linear, time-varying systems. The representation of the AULS and APLS as a linear time-varying systems allows us to find out the relationship between the distortion in the transform domain and in the original domain, using Usevitch tools. Since the nonlinearity of the systems depends on  $d$  rather than on the whole input signal  $x$ , we will find that the weights depend in general only on  $d$ . An even simpler result is found for the one-level decomposition case for update adaptive lifting schemes [82] and for a simplified form of prediction adaptive lifting schemes, as we show in the sequel .



**Figure 4.7:** Equivalent filter bank for synthesis ALS, two decomposition levels.

### 4.3 Computation of polyphase matrices

Let us now compute the synthesis polyphase matrices, starting from the mono-dimensional case, then we will show how to extend this analysis to the bi-dimensional case.

We call  $\mathbf{G}_{00}$  and  $\mathbf{G}_{01}$  the synthesis matrices (see Fig. 4.7). The reconstruction process amounts to obtaining  $\mathbf{x}$  from  $\mathbf{y}_{00}$  and  $\mathbf{y}_{01}$ :

$$\mathbf{x} = \mathbf{G}_{00}\mathbf{y}_{00} + \mathbf{G}_{01}\mathbf{y}_{01}, \quad (4.10)$$

where the bold font refers to the vector form of the reconstructed signal and of the wavelet sub-bands. This equation tells us that the  $2k$ -th [resp.,  $(2k+1)$ -th] row of  $\mathbf{G}_{00}$  is the contribution of the low-pass channel to the even [odd] sample  $x(2k)$  [ $x(2k+1)$ ]. The  $2k$ -th [ $2k+1$ ] row of  $\mathbf{G}_{01}$  is likewise the contribution of the high-pass channel to the even [odd] sample  $x(2k)$  [ $x(2k+1)$ ].

#### 4.3.1 Adaptive update lifting scheme

To find the expression of  $\mathbf{G}_{00}$  and  $\mathbf{G}_{01}$  for AULS, let us start to rewrite the Eq. (4.3) that describes the synthesis equation for the odd samples of signal for an AULS:

$$x(2k+1) = y_{01}(k) + \sum_{n \in \mathbb{Z}} \gamma(n)y_{00}(k-n)$$

...	...	...	...	...	...
...	$-\sum_m \gamma(m) \cdot \beta'_{d(k)}(-m+1)$	$\alpha'_{d(k)-} \sum_m \gamma(m) \cdot \beta'_{d(k)}(-m)$	$-\sum_m \gamma(m) \cdot \beta'_{d(k)}(-m-1)$	$-\sum_m \gamma(m) \cdot \beta'_{d(k)}(-m-2)$	...
...	$\gamma(1)$	$\gamma(0)$	$\gamma(-1)$	$\gamma(-2)$	...
...	$-\sum_m \gamma(m) \cdot \beta'_{d(k+1)}(-m+2)$	$-\sum_m \gamma(m) \cdot \beta'_{d(k+1)}(-m+1)$	$\alpha'_{d(k+1)-} \sum_m \gamma(m) \cdot \beta'_{d(k+1)}(-m)$	$-\sum_m \gamma(m) \cdot \beta'_{d(k+1)}(-m-1)$	...
...	$\gamma(2)$	$\gamma(1)$	$\gamma(0)$	$\gamma(-1)$	...
...	...	...	...	...	...

**Figure 4.8:** Structure of the matrix  $\mathbf{G}_{00}$ . AULS. Highlighted cell is in position  $(2k, k)$ .

From this relation and Eq. (4.10) we observe that the odd rows of  $\mathbf{G}_{00}$  and  $\mathbf{G}_{01}$  can be expressed as:

$$\mathbf{G}_{00}(2k+1, n) = \gamma(k-n) \quad (4.11)$$

$$\mathbf{G}_{01}(2k+1, n) = \delta_{k-n}, \quad (4.12)$$

where  $\delta_k$  is the Kronecker symbol:

$$\delta_k = \begin{cases} 1, & \text{if } k = 0 \\ 0, & \text{otherwise.} \end{cases}$$

As far as the even rows are concerned, we develop the expression of  $x(2k)$  from Eq. (4.4). It is easy to find that:

$$\begin{aligned} x(2k) &= -\sum_n \beta'_{d(k)}(k-n)y_{01}(n) + \alpha'_{d(k)}y_{00}(k) \\ &\quad - \sum_n y_{00}(n) \sum_m \beta'_{d(k)}(k-n-m)\gamma(m). \end{aligned} \quad (4.13)$$

From the last equation we obtain the expression of the generic element on an even row of  $\mathbf{G}_{00}$  and  $\mathbf{G}_{01}$ :

$$\mathbf{G}_{00}(2k, n) = \alpha'_{d(k)}\delta_{n-k} - \sum_m \beta'_{d(k)}(k-n-m)\gamma(m) \quad (4.14)$$

$$\mathbf{G}_{01}(2k, n) = -\beta'_{d(k)}(k-n). \quad (4.15)$$

The structure of the reconstruction polyphase matrices is summarized in Fig. 4.8 and 4.9. We note that the decision map  $d(\cdot)$  influences only the  $2k$ -th row in both reconstruction matrices. Therefore, the even rows of the reconstruction matrices differ from one another only for the corresponding value of  $d$ , besides the fact that there is an horizontal shift of the coefficients.

...	...	...	...	...	...
...	$-\beta'_{d(k)}(1)$	$-\beta'_{d(k)}(0)$	$-\beta'_{d(k)}(-1)$	$-\beta'_{d(k)}(-2)$	...
...	0	1	0	0	...
...	$-\beta'_{d(k+1)}(2)$	$-\beta'_{d(k+1)}(1)$	$-\beta'_{d(k+1)}(0)$	$-\beta'_{d(k+1)}(-1)$	...
...	0	0	1	0	...
...	...	...	...	...	...

**Figure 4.9:** Structure of  $\mathbf{G}_{01}$ . AULS. Highlighted cell is in position  $(2k, k)$ .

### 4.3.2 Adaptive prediction lifting scheme

As in adaptive update case, let us compute the synthesis polyphase matrices. Starting from the Eq. (4.10) and from the equation (Eq. (4.7)):

$$x(2k+1) = y_{01}(k) + \sum_{n \in \mathbb{Z}} \gamma_{d(k)}(n) y_{00}(k-n)$$

it easy to find that the odd rows of  $\mathbf{G}_{00}$  and  $\mathbf{G}_{01}$  can be expressed as:

$$\mathbf{G}_{00}(2k+1, n) = \gamma_{d(k)}(k-n) \quad (4.16)$$

$$\mathbf{G}_{01}(2k+1, n) = \delta_{k-n}, \quad (4.17)$$

As for the even rows, we develop the expression of  $x(2k)$  from Eq. (4.8), and with a little of computation we find that:

$$\begin{aligned} x(2k) &= y_{00}(k) - \sum_n \beta(k-n) y_{01}(n) \\ &\quad - \sum_n y_{00}(n) \sum_m \beta(k-n-m) \gamma_{d(m+n)}(m). \end{aligned} \quad (4.18)$$

So we obtain the expression of the generic element on an even row of  $\mathbf{G}_{00}$  and  $\mathbf{G}_{01}$ :

$$\mathbf{G}_{00}(2k, n) = \delta_{n-k} - \sum_m \beta(k-n-m) \gamma_{d(m+n)}(m) \quad (4.19)$$

$$\mathbf{G}_{01}(2k, n) = -\beta(k-n). \quad (4.20)$$

Note that the expressions for  $\mathbf{G}_{00}$  and  $\mathbf{G}_{01}$  rows are more complicated compared with the one obtained in the case of AULS. In particular, the generic  $2k$ -th row of  $\mathbf{G}_{00}$  does not depend simply on the value of  $d$  at instant  $k$ , as in AULS, but it depends by different values of  $d$  (see eq. 4.19). This aspect does not prevent us to compute the equivalent polyphase matrix but impedes

...	...	...	...	...	...
...	$-\gamma_{d(k)}(1)$	$1 - \gamma_{d(k)}(0)$	$-\gamma_{d(k)}(-1)$	$-\gamma_{d(k)}(-2)$	...
...	$\gamma_{d(k)}(1)$	$\gamma_{d(k)}(0)$	$\gamma_{d(k)}(-1)$	$\gamma_{d(k)}(-2)$	...
...	$\gamma_{d(k+1)}(2)$	$\gamma_{d(k+1)}(1)$	$1 - \gamma_{d(k+1)}(0)$	$\gamma_{d(k+1)}(-1)$	...
...	$\gamma_{d(k+1)}(2)$	$\gamma_{d(k+1)}(1)$	$\gamma_{d(k+1)}(0)$	$\gamma_{d(k+1)}(-1)$	...
...	...	...	...	...	...

**Figure 4.10:** Structure of the matrix  $\mathbf{G}_{00}$ . APLS. Highlighted cell is in position  $(2k, k)$ .

...	...	...	...	...	...
...	0	-1	0	0	...
...	0	1	0	0	...
...	0	0	-1	0	...
...	0	0	1	0	...
...	...	...	...	...	...

**Figure 4.11:** Structure of the matrix  $\mathbf{G}_{01}$ . APLS. Highlighted cell is in position  $(2k, k)$ .

the possibility to establish a simple connection between adaptivity and matrix expression.

Luckily, in the lifting scheme that we are going to consider (Claypoole), the expressions of  $2k$ -th row of  $\mathbf{G}_{00}$  and of  $\mathbf{G}_{01}$  are simplified because  $\beta(n) = \delta_n$ , so the Eq. 4.19 and Eq. 4.20 become:

$$\begin{aligned}\mathbf{G}_{00}(2k, n) &= \delta_{k-n} - \gamma_{d(k)}(k-n) \\ \mathbf{G}_{01}(2k, n) &= -\delta_{k-n}.\end{aligned}$$

The structure of the reconstruction polyphase matrices, in this simplified hypothesis, is summarized in Fig. 4.10 and 4.11.

## 4.4 Weight computation

### 4.4.1 One level of decomposition

Once polyphase matrices has been computed, we can find the expression for corrective weights simply following Usevitch approach [81]. It is interesting to develop the calculus in the hypothesis of one dimensional signal and one level

of decomposition because they provide a simple and intuitive relation between weights in adaptive case and weights in non-adaptive case.

Now let us introduce the matrices:

$$\mathbf{G}_{0i}^{(h)} = \mathbf{G}_{0i} \big|_{\mathbf{d}=[h \ h \ \dots \ h]} \quad (4.21)$$

For example  $\mathbf{G}_{00}^{(0)}$  is the low-pass channel reconstruction matrix that we would have if the decision map was always equal to zero. We can compute the weights associated with these matrices: they are the weights that we should apply when considering a non-adaptive LS. From [81], we have:

$$w_{0i}^{(h)} = \frac{2}{N} \sum_{n,m} \mathbf{G}_{0i}^{(h)}(n, m)^2.$$

We can develop it as:

$$\begin{aligned} w_{0i}^{(h)} &= \frac{2}{N} \sum_n \left[ \sum_m \mathbf{G}_{0i}^{(h)}(2n, m)^2 + \sum_m \mathbf{G}_{0i}^{(h)}(2n+1, m)^2 \right] \\ &= \sum_m \mathbf{G}_{0i}^{(h)}(0, m)^2 + \sum_m \mathbf{G}_{0i}^{(h)}(1, m)^2. \end{aligned} \quad (4.22)$$

The last equation takes into account the fact that all even [resp., odd] rows are equal but for a shift, so the sum of their squared values can be obtained from any even [resp., odd] row.

In the adaptive case we have:

$$\begin{aligned} w_{0i} &= \frac{2}{N} \sum_{n,m} \mathbf{G}_{0i}(n, m)^2 \\ &= \frac{2}{N} \sum_n \left[ \sum_m \mathbf{G}_{0i}(2n, m)^2 + \sum_m \mathbf{G}_{0i}(2n+1, m)^2 \right]. \end{aligned}$$

We know that the values of the reconstruction matrix on the couple of rows  $2n$  and  $2n+1$  only depend on  $d(n)$ :

$$\begin{aligned} \mathbf{G}_{0i}(2n, m) &= \mathbf{G}_{0i}^{(d(n))}(2n, m) \\ \mathbf{G}_{0i}(2n+1, m) &= \mathbf{G}_{0i}^{(d(n))}(2n+1, m). \end{aligned}$$

So we can write:

$$\begin{aligned}
 w_{0i} &= \frac{2}{N} \sum_n \left[ \sum_m \mathbf{G}_{0i}^{(d(n))}(2n, m)^2 + \sum_m \mathbf{G}_{0i}^{(d(n))}(2n+1, m)^2 \right] \\
 &= \frac{2}{N} \sum_n \left[ \sum_m \mathbf{G}_{0i}^{(d(n))}(0, m)^2 + \sum_m \mathbf{G}_{0i}^{(d(n))}(1, m)^2 \right] \\
 &= \frac{2}{N} \sum_n w_{0i}^{d(n)},
 \end{aligned}$$

where we used Eq. (4.22). If we denote by  $N_h$  the number of occurrences of the value  $h$  in the decision map, we can write:

$$w_{0i} = \sum_{h=0}^{D-1} \frac{2N_h}{N} w_{0i}^{(h)}. \quad (4.23)$$

In other words, the weight of each subband depends only on the relative frequency of the various symbols in the decision map. The relative frequencies are used as multiplicative coefficients in order to find the adaptive weight as a function of the “non-adaptive” ones. It is interesting to see that even though the AULS and APLS are inherently nonlinear, we can find such a simple and intuitive relationship between their weights and those of linear lifting schemes. Unfortunately, the relationship becomes more complex when more than one decomposition level is performed.

#### 4.4.2 Multiple levels of decomposition

In this subsection we show how to compute the weights for an ALS when more than one decomposition level is used. Coherently with the notation used for the wavelet subbands, we define  $\mathbf{G}_{ij}$  as the reconstruction matrix for the decomposition level  $i$  and for the channel  $j$  (see Fig. 4.7). For example, the low-pass subband at level  $i-1$  can be obtained from the subbands at level  $i$  via the matrices  $\mathbf{G}_{ij}$ :

$$\mathbf{y}_{i-1,0} = \sum_{j \in \mathcal{J}} \mathbf{G}_{ij} \mathbf{y}_{ij} \quad (4.24)$$

It is obvious that  $\mathbf{G}_{ij}$  has the same structure as  $\mathbf{G}_{0j}$ , except that we have to use the appropriate decision map at level  $i$ , denoted by  $d_i(\cdot)$ . Let us now introduce  $\mathbf{d}_i^{(h)}$  as a vector whose  $k$ -th component is:

$$\mathbf{d}_i^{(h)}(k) = \begin{cases} 1 & \text{if } \mathbf{d}_i \left( \lfloor \frac{k}{2} \rfloor \right) = h \\ 0 & \text{otherwise.} \end{cases} \quad (4.25)$$

Finally, let us define  $\mathbf{D}_j^{(h)} = \text{diag}(\mathbf{d}_j^{(h)})$ . It is easy to see that:

$$\mathbf{G}_{ij} = \sum_{h=0}^{D-1} \mathbf{D}_i^{(h)} \mathbf{G}_{ij}^{(h)}, \quad (4.26)$$

where  $\mathbf{G}_{ij}^{(h)}$  is defined similarly to  $\mathbf{G}_{i0}^{(h)}$  in Eq. (4.21). In other words, the synthesis matrix (at any decomposition level) for the ALS is composed by selecting the  $2k$ -th and  $(2k+1)$ -th rows of the non-adaptive matrix determined by the map value  $\mathbf{d}_i(k)$ .

It is easy to remark that the reconstructed signal can be expressed using recursively Eq. (4.24). We obtain:

$$\mathbf{x} = \sum_{ij} \mathbf{A}_{ij} \mathbf{y}_{ij},$$

where  $(i, j) \in \{(0, 1), (1, 1), (2, 1), \dots, (N-1, 1), (N-1, 0)\}$ . The reconstruction matrices can be computed as:

$$\mathbf{A}_{01} = \mathbf{G}_{01} \quad (4.27)$$

$$\mathbf{A}_{i1} = \mathbf{G}_{i1} \prod_{\ell=0}^{i-1} \mathbf{G}_{\ell 0}, \quad \forall i \in \{1, \dots, I-1\} \quad (4.28)$$

$$\mathbf{A}_{I-1,0} = \prod_{i=0}^{I-1} \mathbf{G}_{i0}. \quad (4.29)$$

We observe that  $\mathbf{A}_{ij}$  is the product of the matrices corresponding to the filters between the subband  $y_{ij}$  and the reconstructed signal  $x$ . This is still true when the decomposition is non-dyadic or more than two channels are used.

In conclusion, in order to get the weight for the  $y_{ij}$  subband, we have to:

1. Compute all the matrices  $\mathbf{G}_{\ell k}$  needed to build  $\mathbf{A}_{ij}$ ;
2. Compute  $\mathbf{A}_{ij}$  using the appropriate equation among (4.27), (4.28), and (4.29);
3. Obtain  $w_{ij}$  as the average of the column norms of  $\mathbf{A}_{ij}$ .

Unfortunately, the simple interpretation of the adaptive lifting scheme weights obtained for the one-level decomposition does not hold anymore when more levels are used, because of the matrix product in Eq. (4.28) or Eq. (4.29).

$x(n-1,m-1)$	$x(n-1,m)$	$x(n,m+1)$
$x_3(k-M-1)$	$x_2(k-M)$	$x_3(k-M)$
$y_{03}(k-M-1)$	$y_{02}(k-M)$	$y_{03}(k-M)$
$x(n,m-1)$	$x(n,m)$	$x(n,m+1)$
$x_1(k-1)$	$x_0(k)$	$x_1(k)$
$y_{01}(k-1)$	$y_{00}(k)$	$y_{01}(k)$
$x(n+1,m-1)$	$x(n+1,m)$	$x(n+1,m+1)$
$x_3(k-1)$	$x_2(k)$	$x_3(k)$
$y_{03}(k-1)$	$y_{02}(k)$	$y_{03}(k)$

**Figure 4.12:** The bi-dimensional signal  $x$  represented via four channels;  $x$  has  $2M$  columns, and  $k = Mn + m$ .

## 4.5 Extension to multi-dimensional case

Both AULS and APLS can be extended to the bi-dimensional case in order to obtain adaptive transforms of images. The extension could be done in the same manner for both but we prefer to treat separately the two types of lifting scheme. The reason of our choice resides to the fact that the specific 2D lifting schemes, belonging to these two families, that we want to analyze come from a different type of extension. In fact, for AULS we consider directly a 2D non separable version of the scheme, because these type of scheme has been proposed in the literature; for APLS, instead, being our reference Claypoole's work, we start from a 2D separable extension of transform to come to a 2D non separable extension.

For AULS, we consider a number of non-separable bi-dimensional transforms presented in [78]. This case can be treated as the mono-dimensional one, with the difference that more than two channels are used at each level. The input signal  $x$  is divided into  $J = 4$  channels, as shown in Fig. 4.12. The

AULS analysis equations are the following:

$$y_{00}(k) = \alpha_{d(k)}x_0(k) + \sum_{n \in \mathbb{Z}} \beta_{1,d(k)}(n)x_1(k-n) \\ + \sum_{n \in \mathbb{Z}} \beta_{2,d(k)}(n)x_2(k-n) + \sum_{n \in \mathbb{Z}} \beta_{3,d(k)}(n)x_3(k-n) \quad (4.30)$$

$$y_{01}(k) = x_1(k) - \sum_{n \in \mathbb{Z}} \gamma_{1,0}(n)y_{00}(k-n), \quad (4.31)$$

$$y_{02}(k) = x_2(k) - \sum_{n \in \mathbb{Z}} \gamma_{2,0}(n)y_{00}(k-n) - \sum_{n \in \mathbb{Z}} \gamma_{2,1}(n)y_{01}(k-n), \quad (4.32)$$

$$y_{03}(k) = x_3(k) - \sum_{n \in \mathbb{Z}} \gamma_{3,0}(n)y_{00}(k-n) - \sum_{n \in \mathbb{Z}} \gamma_{3,1}(n)y_{01}(k-n) \\ - \sum_{n \in \mathbb{Z}} \gamma_{3,2}(n)y_{02}(k-n), \quad (4.33)$$

and its structure is shown in Fig. 4.13.

Actually, we will use a simplified form, in fact, the predict operators proposed in [78] are the following:

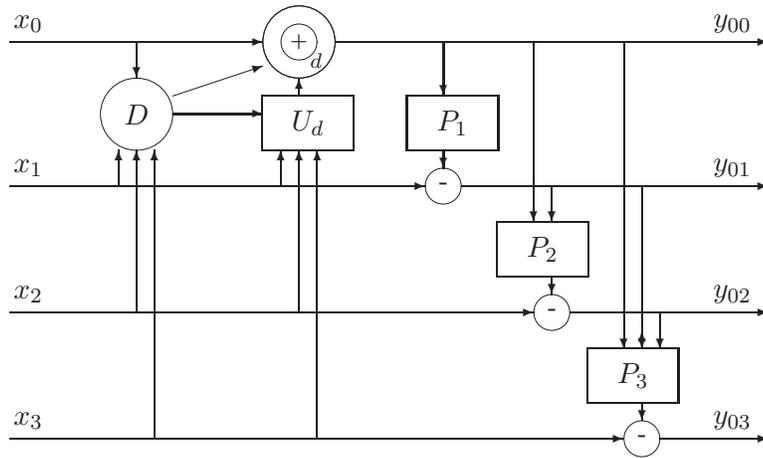
$$y_{01}(k) = x_1(k) - y_{00}(k) \quad (4.34)$$

$$y_{02}(k) = x_2(k) - y_{00}(k) \quad (4.35)$$

$$y_{03}(k) = x_3(k) - y_{00}(k) - y_{01}(k) - y_{02}(k) \quad (4.36)$$

So, in the following, we refer to this simplified scheme.

If we would construct the 2D version of APLS for similarity to that of



**Figure 4.13:** 2D AULS.

AULS, we obtain the scheme, shown in Fig.4.14, described by the equation:

$$y_{00}(k) = x_0(k) + \sum_{n \in \mathbb{Z}} \beta_1(n)x_1(k-n) + \sum_{n \in \mathbb{Z}} \beta_2(n)x_2(k-n) + \sum_{n \in \mathbb{Z}} \beta_3(n)x_3(k-n) \quad (4.37)$$

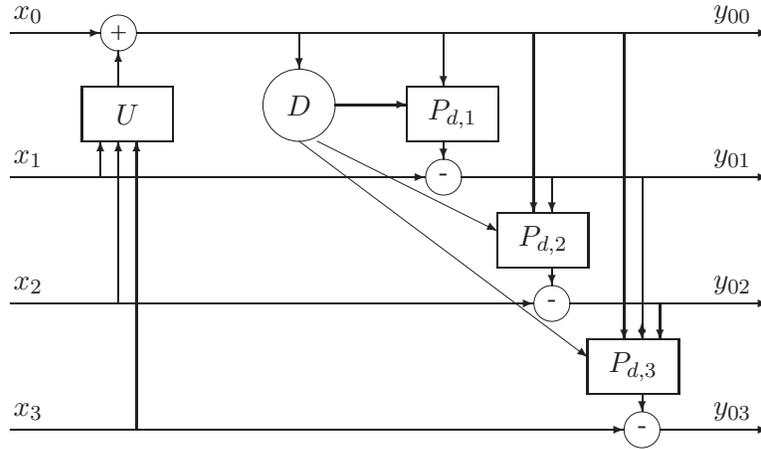
$$y_{01}(k) = x_1(k) - \sum_{n \in \mathbb{Z}} \gamma_{1,0,d(k)}(n)y_{00}(k-n), \quad (4.38)$$

$$y_{02}(k) = x_2(k) - \sum_{n \in \mathbb{Z}} \gamma_{2,0,d(k)}(n)y_{00}(k-n) - \sum_{n \in \mathbb{Z}} \gamma_{2,1,d(k)}(n)y_{01}(k-n), \quad (4.39)$$

$$y_{03}(k) = x_3(k) - \sum_{n \in \mathbb{Z}} \gamma_{3,0,d(k)}(n)y_{00}(k-n) - \sum_{n \in \mathbb{Z}} \gamma_{3,1,d(k)}(n)y_{01}(k-n) - \sum_{n \in \mathbb{Z}} \gamma_{3,2,d(k)}(n)y_{02}(k-n), \quad (4.40)$$

The problem is that this scheme does not fit Claypoole's lifting scheme, but a more complex structure is needed.

In Claypoole's original work, the extension of the transform to 2D case is



**Figure 4.14:** Theoretical 2D APLS.

done in a separable manner. This is a problem because for weights computation we need to have a polyphase matrix that describes the link between input and output signals. In non adaptive case, when we consider a 2D separable transform, this is possible because we use the same polyphase matrix for all rows and for all columns [81] but, in adaptive case, different rows/columns mean different decision maps, and different decision maps mean different polyphase matrices. So we can not express input-output relation by a polyphase matrix. To overcome this problem we have to compute a 2D non separable version of Claypoole's lifting scheme.

As mentioned in Sec.4.1.2, Claypoole's transform switches between different linear filters (acting on predict stage) according to the local behavior of the mono-dimensional input signal (a row or a column for the images). In particular it uses four different filters. To construct a 2D non separable Claypoole's transform, we have to consider all sixteen 2D non separable filters obtained by the combination of all mono-dimensional filters and then switch among them according to the behavior both of the current row and the current column. Doing this operation, the structure of 2D non separable lifting scheme obtained is shown in Fig.4.15 and correspond to the following expressions:

$$\begin{aligned}
y_{00}(k) &= x_0(k) + \sum_{n \in \mathbb{Z}} \beta_1(n)x_1(k-n) \\
&\quad + \sum_{n \in \mathbb{Z}} \beta_2(n)x_2(k-n) + \sum_{n \in \mathbb{Z}} \beta_3(n)x_3(k-n), \tag{4.41}
\end{aligned}$$

$$\begin{aligned}
y_{01}(k) &= x_1(k) - \sum_{n \in \mathbb{Z}} \gamma_{1,0,d(k)}(n)y_{00}(k-n) \\
&\quad + \sum_{n \in \mathbb{Z}} \sigma_1(n)x_3(k-n), \tag{4.42}
\end{aligned}$$

$$\begin{aligned}
y_{02}(k) &= x_2(k) - \sum_{n \in \mathbb{Z}} \gamma_{2,0,d(k)}(n)y_{00}(k-n) - \sum_{n \in \mathbb{Z}} \gamma_{2,1,d(k)}(n)y_{01}(k-n) \\
&\quad + \sum_{n \in \mathbb{Z}} \sigma_2(n)x_3(k-n), \tag{4.43}
\end{aligned}$$

$$\begin{aligned}
y_{03}(k) &= x_3(k) - \sum_{n \in \mathbb{Z}} \gamma_{3,0,d(k)}(n)y_{00}(k-n) - \sum_{n \in \mathbb{Z}} \gamma_{3,1,d(k)}(n)y_{01}(k-n) \\
&\quad - \sum_{n \in \mathbb{Z}} \gamma_{3,2,d(k)}(n)y_{02}(k-n). \tag{4.44}
\end{aligned}$$

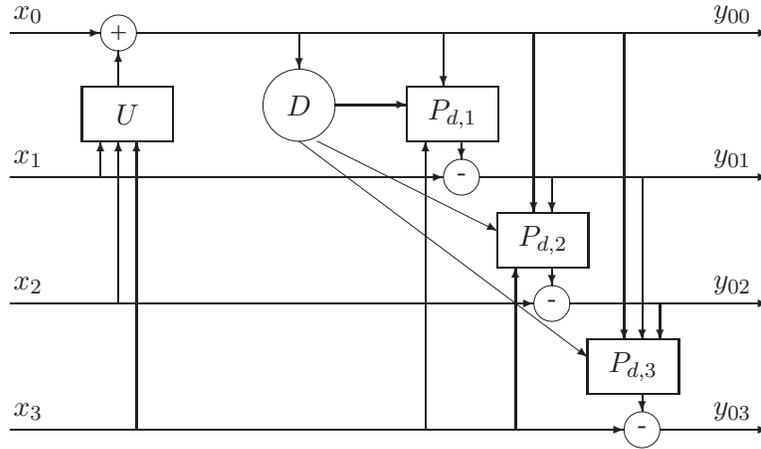
In our case this general expression can be simplified because, as already noted, we use a very simple update <sup>1</sup>:

$$y_{00}(k) = x_0(k) + x_1(k) + x_2(k) + x_3(k) \tag{4.45}$$

Once derived the expression of 2D AULS and APLS, the synthesis equations can be easily obtained from the analysis ones. Then, the equivalent polyphase matrix for reconstruction,  $\mathbf{G}_{0j}$ , can be obtained by evaluating the contribution of the wavelet subband  $y_{0j}$  to the channel  $x_i$  for  $i = 0, 1, 2, 3$ . This process is perfectly analogous to the one described in Section 4.3. However, here we do not report the computation of the reconstruction matrices for the sake of simplicity. The result is that the  $\mathbf{G}_{0j}$  matrices are composed of blocks of  $J$  rows, from the  $Jk$ -th to the  $(Jk + J - 1)$ -th row that depend on the  $k$ -th value of the decision map. As in the 1-D case, the adaptive lifting scheme  $w_{0j}$  (one-level decomposition) can be obtained as weighted average of non-adapted weights.

---

<sup>1</sup>we omit a constant equal to  $\frac{1}{4}$



**Figure 4.15:** Claypoole inspired 2D APLS.

Once one has obtained the first level decomposition matrices, the weights  $w_{ij}$  can be computed recursively as in the 1-D case.

## 4.6 Experimental results

### 4.6.1 Test Lifting Schemes

In this section we validate the results previously obtained for some simple AULS and for our 2D APLS inspired by Claypoole *et al.* lifting scheme.

For AULS we consider four bi-dimensional non-separable adaptive lifting schemes, three presented in [78] and one presented in [83].

The first three are binary AULS, in the sense that one out of two update filters is chosen at each time. In all the three cases, when  $d = 1$ , the update step does not perform any filtering, that is  $\alpha_1 = 1$  and  $\beta_{j,1}(n) = 0$  for all  $j$  and  $n$ . This happens when discontinuities are detected, so that they are preserved at low resolution levels without smoothing. The three filters differ for the update step in homogeneous regions (besides the way the decision map is computed, see [78] for details). The first one, which we will refer to as AULS A, when

$d = 0$  has the following update operator:

$$y_{00}(k) = \frac{1}{2} [x_0(k) + x_1(k) + x_2(k) - x_3(k)].$$

The second one is denoted by AULS B. When  $d = 0$ , it uses the following update:

$$\begin{aligned} y_{00}(k) &= \frac{1}{2}x_0(k) \\ &+ \frac{1}{4} [x_1(k) + x_2(k) + x_1(k-1) + x_2(k-M)] \\ &- \frac{1}{8} [x_3(k) + x_3(k-1) + x_3(k-M) + x_3(k-M-1)], \end{aligned}$$

where the input signal has  $2M$  columns (see Fig. 4.12). Finally, we consider an AULS that we call AULS C, whose update step for  $d = 0$  is:

$$\begin{aligned} y_{00}(k) &= \frac{1}{2}x_0(k) \\ &+ \frac{1}{8} [x_1(k) + x_2(k) + x_1(k-1) + x_2(k-M)], \end{aligned}$$

In [78] it is shown that AULS A [resp., AULS B] corresponds to a decision map which is insensitive to first [resp., second] degree polynomials. This means that the first 2 AULS respond to higher degree polynomials by adapting the update. The third AULS is sensitive to high values of the discrete Laplacian of  $x$ .

The fourth AULS considered is proposed in [83], it is not binary but it uses multiple criteria for choosing the update filters giving rise to multi-valued decision. In the lifting schemes proposed in [83], local gradient seminorms computed along different directions are compared in order to discriminate between different geometric structures and to capture the directional nature of images. We refer to these lifting scheme as combining seminorms (CS) AULS [83]. In the example that we consider we have three different seminorms:  $p_0$ , relative to the horizontal direction;  $p_1$ , relative to vertical direction; and  $p_3$ , relative to oblique direction. The filtering is performed along the direction with the smallest seminorm, furthermore the latter is compared with a threshold and, if it is found greater, non filtering at all is performed, because, with high probability, there is an edge along all directions. The expression of the

considered lifting scheme is the following:

$$y_{00}(k) = \begin{cases} \frac{1}{2}x_0(k) + \frac{1}{4}[x_1(k) + x_1(k-1)], & \text{if } d = 0; \\ x_0(k), & \text{if } d = 1; \\ \frac{1}{2}x_0(k) + \frac{1}{4}[x_2(k) + x_2(k-M)], & \text{if } d = 2; \\ x_0(k), & \text{if } d = 3; \\ \frac{1}{2}x_0(k) + \frac{1}{8}[x_1(k) + x_1(k-1) + x_2(k) + x_2(k-M)], & \text{if } d = 4; \\ x_0(k), & \text{if } d = 5. \end{cases}$$

where:

$$\begin{aligned} d = 0 &\Leftrightarrow p_0 = \min(p_0, p_1, p_2) \text{ and } p_0 \leq T_0; & (4.46) \\ d = 1 &\Leftrightarrow p_0 = \min(p_0, p_1, p_2) \text{ and } p_0 > T_0; \\ d = 2 &\Leftrightarrow p_1 = \min(p_0, p_1, p_2) \text{ and } p_1 \leq T_1; \\ d = 3 &\Leftrightarrow p_1 = \min(p_0, p_1, p_2) \text{ and } p_1 > T_1; \\ d = 4 &\Leftrightarrow p_2 = \min(p_0, p_1, p_2) \text{ and } p_2 \leq T_2; \\ d = 5 &\Leftrightarrow p_2 = \min(p_0, p_1, p_2) \text{ and } p_2 > T_2. \end{aligned}$$

While for AULS A, B and CS the prediction is performed with Eq. (4.34)-(4.36), for the AULS C, the last equation is simplified to:

$$y_{03}(k) = x_3(k) - y_{00}(k).$$

Several other binary and combining seminorms AULS are described in [78], [83] and in related works, the results for these other schemes are similar to those reported in the following.

With reference to APLS, we consider the 2D non separable extension of Claypoole's work described in the Sec.4.5 Our lifting scheme is very similar to that described in [13] except for the fact that:

- our adaptivity does not choose between four mono-dimensional filters according to the behavior of the current row or column, but between sixteen bi-dimensional filters according the behavior of both row and column;
- our way of individuating discontinuity is based upon the thresholding of Sobel operator instead on a modified version of [79].

	Number of decomposition levels		
	1	2	3
AULS A No weights	59.77%	74.69%	81.45%
AULS A Weighted	0.23%	0.36%	0.56%
AULS B No weights	61.84%	77.09%	83.70%
AULS B Weighted	0.31%	1.53%	2.93%
AULS C No weights	42.87%	60.01%	69.26%
AULS C Weighted	0.17%	0.27%	0.44%
AULS CS No weights	63.39%	79.25%	85.98%
AULS CS Weighted	0.28%	0.60%	2.52%
APLS No weights	76.41%	87.18%	91.31%
APLS Weighted	0.14%	0.15%	0.35%

**Table 4.1:** Relative error of the energy estimation.

#### 4.6.2 Distortion evaluation in transformed domain

A first experiment is conducted in order to validate the weights computed with the proposed method. As shown in [81], if the error signal in a subband  $y_{ij}$  (*i.e.* the quantization noise) is white and uncorrelated to the other subband errors, the distortion in the original domain  $D$  is related to the distortion in the wavelet domain by Eq. (4.9). In order to verify this relationship we generate white Gaussian noise for the coefficients in each transform subband. Then we estimate the distortion in the wavelet domain as the energy of the error signal. We consider two cases: in the first one we use the weights as in Eq. (4.9); in the second one we use  $w_{ij} = 1$  for all subbands. This means that we estimate the distortion in the wavelet domain without using weights. Then the two distortion estimations are compared to the real distortion, obtained as energy of the error signal after the inverse transform. The per cent relative errors of the two estimations are reported in Tab. 4.1 for our five adaptive lifting schemes.

These results show that, on one hand, these lifting scheme are quite far from orthogonal, so the distortion in the transform domain is a poor estimation of the actual distortion. On the other hand, with the weights computed with the proposed method, the distortion estimation becomes much more reliable.

	AULS A	AULS B	AULS	AULS CS	APLS
<i>Lena</i>	0.7dB	1.6dB	1.2dB	1.5dB	0.6dB
<i>House</i>	0.6dB	0.6dB	0.9dB	0.4dB	0.4dB
<i>Peppers</i>	0.3dB	0.7dB	1.0dB	1.4dB	0.4dB
<i>Cameraman</i>	1.0dB	0.6dB	0.7dB	0.7dB	0.3dB
<i>Barbara</i>	1.5dB	1.3dB	1.1dB	1.4dB	0.8dB

**Table 4.2:** PSNR improvements at 0.5 bpp for test adaptive lifting scheme compared with no weights.

### 4.6.3 Bit-rate allocation

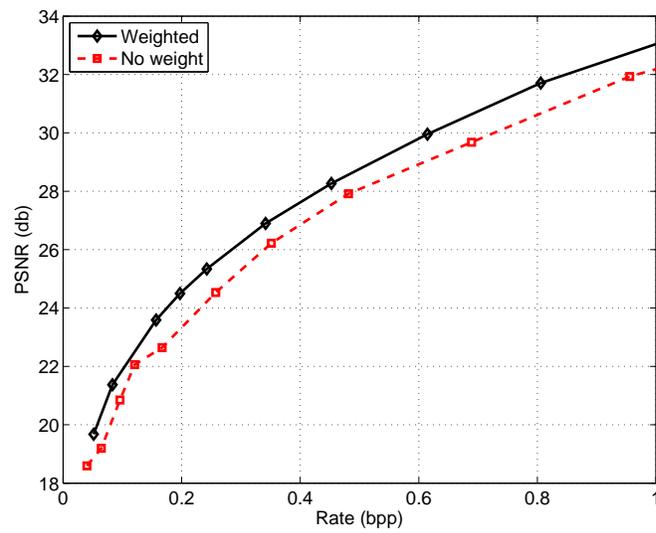
The ability of reliably estimating the distortion in the transform domain gives consistent benefits in a compression scheme. In this section we show some quantitative results about the improvement that a correct use of weights gives w.r.t. not using any weights at all.

To this end, we use a simple compression scheme, which is very similar to the original one proposed in [78]. The input image is transformed with one of the five test adaptive lifting scheme and quantized with a dead-zone quantizer. An optimal bit-rate allocation algorithm is ran [84] to choose the quantization step for each subband such that the spatial domain distortion expressed via Eq. (4.9) is minimized for the assigned target rate. Then the inverse transform is applied on quantized coefficients, and the resulting distortion is computed. In order to assess the effect of weights, we carry out the same compression scheme using unitary weights for all subbands. Finally, we compare the rate/distortion curves for the two schemes.

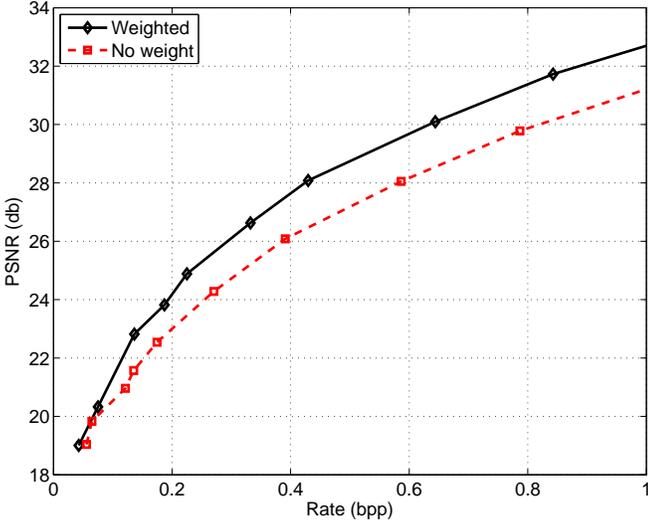
We performe this experiment on the images Lena, House, Peppers, Cameraman, and Barbara. The RD curves for Lena are reported in Fig. 4.16- 4.20 and the PSNR improvements compared with no weights over the five images at 0.5 are reported in Tab. 4.2 and at 1.0 bpp in Tab. 4.3. We see that using weights brings a gain, a little more consistent for AULS; however the improvement is remarkable even for APLS.

	AULS A	AULS B	AULS	AULS CS	APLS
<i>Lena</i>	0.9dB	1.5dB	1.0dB	1.5dB	0.4dB
<i>House</i>	0.6dB	0.8dB	0.9dB	0.9dB	0.6dB
<i>Peppers</i>	0.4dB	0.7dB	0.8dB	1.4dB	0.4dB
<i>Camerman</i>	0.8dB	0.8dB	0.5dB	0.6dB	0.4dB
<i>Barbara</i>	1.4dB	1.0dB	1.2dB	1.0dB	1.0dB

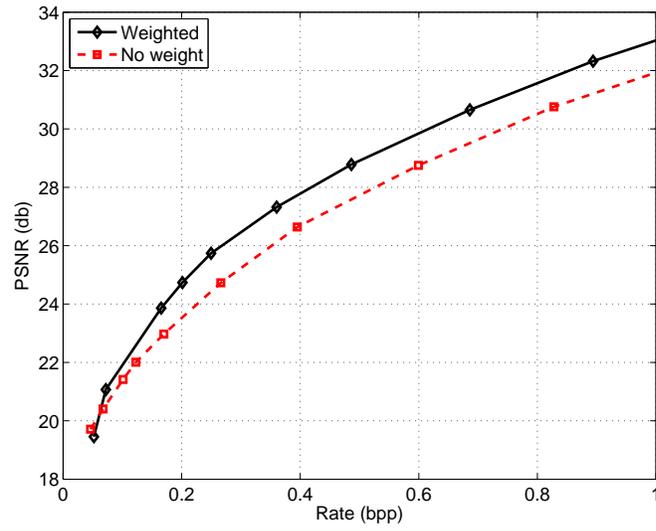
**Table 4.3:** PSNR improvements at 1 bpp for test adaptive lifting scheme compared with no weights.



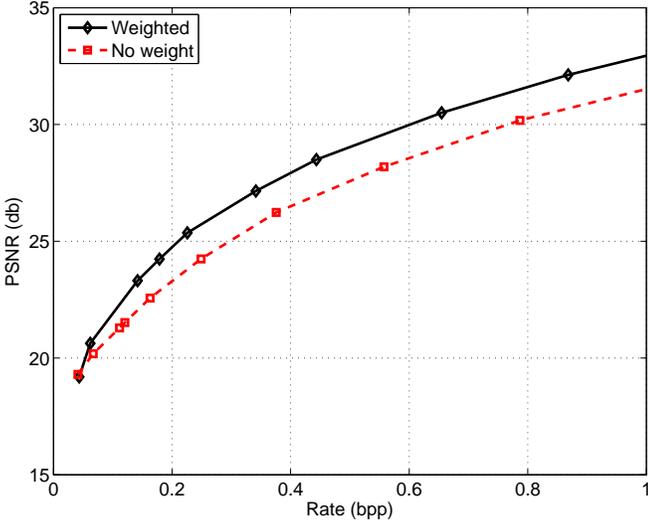
**Figure 4.16:** Rate-distortion curves for Lena with and without weights. AULS A.



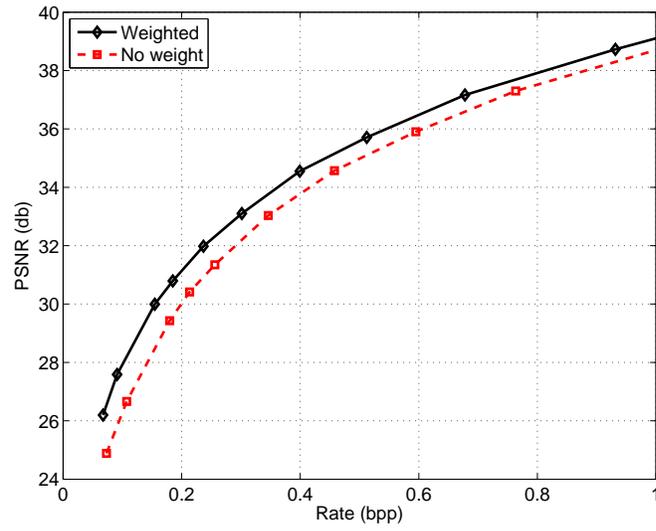
**Figure 4.17:** Rate-distortion curves for Lena with and without weights. AULS B.



**Figure 4.18:** Rate-distortion curves for Lena with and without weights. AULS C.



**Figure 4.19:** Rate-distortion curves for Lena with and without weights. AULS CS.



**Figure 4.20:** Rate-distortion curves for Lena with and without weights. APLS.

# Conclusions

Object contours contribute to a large extent to the perceived quality of an image but are typically quite hard to compress. As a matter of facts, many coding algorithms fail to describe efficiently this information. In this thesis, we discussed this issue explaining the relation between this problem and the most effective and widespread transform for image compression: the wavelet. Referring to recent studies on harmonic analysis, we explained the reasons for wavelet efficiency but also for its sub-optimality when dealing with bi-dimensional discontinuities. To overcome wavelet limits we considered three different solutions (object-based coding, new directional transforms and adaptive lifting scheme) and in each of these scenarios we brought our original contribute.

As regards the object-based image coding paradigm, we analyzed costs and advantages of an object-based scheme based on Li and Li's wavelet shape-adaptive (SA-WT) and shape adaptive SPIHT. Our aim was to assess the rate-distortion performance of such an object-based coder by means of numerical experiments in typical situations of interest, and single out, to the extent possible, the individual phenomena that contribute to the overall losses and gains. Since the usual coding gain does not make sense for Li and Li's SA-WT we measured its compaction ability by analyzing the RD performance of a virtual oracle coder which spends bits only for quantization. This was a very important step because SA-WT losses turned out to be quite significant, especially at low rates. Although the quantization by itself account only for a small fraction of the total cost, the reduced efficiency of SA-WT has a deep effect also on the subsequent coding phase, the sorting pass of SPIHT. In fact, our experiments revealed this to be the main cause of SPIHT losses, while the presence of incomplete trees plays only a minor role. As for the gains, our analysis showed that they can be significant when the image presents sharp edges between relatively homogeneous regions but also that this is rarely the case with real-world images where the presence of smooth contours, and the inaccu-

cies of segmentation (for a few objects) or its large cost (for many objects) represent serious hurdles towards potential performance gains. Hence, for natural image, the advantages do not balance the costs and performance gains are currently achievable only for some specific source, like multispectral images.

For multispectral images we introduced two different object-based paradigms, region-based and class-based coding, that differ only for the type of segmentation used. While classes are singled out mostly on the basis of spectral homogeneity criteria, regions are required to satisfy some additional spatial constraints. Therefore, regions tend to be less “pure” than classes, and a spectral transform is less effective on them. On the other hand, a spatial transform applied to a compact region, rather than to a sparse class, might work better. For both approaches we experimented a coding composed of: a segmentation block; a Karhunen-Loeve transform (KLT) along the spectral dimension; a wavelet transform in the spatial domain; a scalar quantizer; and a block of resource allocation. To better compact energy in the first few transform bands, a different KLT matrix is used for each class/region. Our experiments led to the conclusion that the class-based coding approach guarantees always a very good performance, because optimizes the trade-off among costs and gains. The performance of the region-based approach, instead, are a little bit worse, but still better than that of completely flat approaches.

The second scenario refers to the new directional transforms. In this context, we presented a new compression technique based on the contourlet transform. The choice of this transform was based on the fact that it has an almost optimal NLA (nonlinear approximation error) behavior and it is easily implemented by a filter bank. Preliminary results on NLA quality led us to use actually a hybrid wavelet-contourlet decomposition. Then, the SPIHT coder was adapted to the new transform, with the main design problem being the definition of suitable significance trees that took into account the correlation of coefficients across scales, space and directions. Even if the transform is slightly redundant the rate-distortion performance is good, especially for highly textured images, and the visual quality of directional details is better than that of the conventional wavelet/SPIHT coder.

The last solution analyzed is the adaptive lifting scheme. We showed how to estimate the coding distortion in the transform domain for two interesting classes of adaptive lifting schemes. The basic idea is that the nonlinearity of these schemes can be seen as a time-variable behavior. In this way, we can compute the weights allowing us to estimate the distortion in the transform domain via a weighted average of subband distortions. The method we proposed

can be used with a large number of adaptive update/prediction lifting schemes. Experimental results show that by using these weights the distortion assessment becomes very reliable. As a consequence, coding techniques based on distortion minimization benefit from a better distortion estimation and provide better performance.



# Bibliography

- [1] D. Gabor, “Guest editorial,” *IRE Trans. Info. Theory*, Sep. 1959.
- [2] E. J. Candès and D. L. Donoho, “Curvelets—a surprisingly effective non-adaptive representation for objects with edges,” *Curve and Surface Fitting*, 1999.
- [3] S. Li and W. Li, “Shape-adaptive discrete wavelet transforms for arbitrarily shaped visual object coding,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 10, no. 5, pp. 725–743, May 2000.
- [4] M. Cagnazzo, G. Poggi, L. Verdoliva, and A. Zinicola, “Region-oriented compression of multispectral images by shape-adaptive wavelet transform and SPIHT,” in *Proceedings of IEEE International Conference on Image Processing*, vol. 4, Singapore, Oct. 2004, pp. 2459–2462.
- [5] A. Said and W. Pearlman, “A new, fast and efficient image codec based on set partitioning in hierarchical trees,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 6, no. 3, pp. 243–250, Jun. 1996.
- [6] M. Do and M. Vetterli, “The contourlet transform: An efficient directional multiresolution image representation,” *IEEE Transactions on Image Processing*, vol. 14, no. 12, pp. 2091–2106, Dec. 2005.
- [7] M. V. V. Velisavljevic, B. Beferull-Lozano and P. Dragotti, “Directionlets: anisotropic multi-directional representation with separable filtering,” *IEEE Transactions on Image Processing*, vol. 15, no. 7, pp. 1916–1933, Jul. 2006.
- [8] E. L. Pennec and S. Mallat, “Sparse geometric image representation with bandelets,” *IEEE Transactions on Image Processing*, vol. 14, no. 4, pp. 423–438, Apr. 2005.

- 
- [9] I. Daubechies and W. Sweldens, “Factoring wavelet transforms into lifting steps,” *J. Fourier Anal. Appl.*, vol. 4, no. 3, pp. 245–267, 1998.
- [10] C. Chang and B. Girod, “Direction-adaptive discrete wavelet transform for image compress,” *IEEE Transactions on Image Processing*, vol. 16, no. 5, pp. 1289–1302, May 2007.
- [11] O. N. Gerek and A. E. Cetin, “A 2d orientation-adaptive prediction filter in lifting structures for image coding,” *IEEE Transactions on Image Processing*, no. 1, Jan. 2006.
- [12] X. W. S. L. W. Ding, F. Wu and H. Li, “Adaptive directional lifting-based wavelet transform for image coding,” *IEEE Transactions on Image Processing*, vol. 16, no. 2, pp. 416–427, Feb. 2007.
- [13] R. L. Claypoole, G. M. Davis, W. Sweldens, and R. G. Baraniuk, “Non-linear wavelet transforms for image coding via lifting,” *IEEE Transactions on Image Processing*, vol. 12, no. 12, pp. 1449–1459, Dec. 2003.
- [14] H. J. A. M. Heijmans, B. Pesquet-Popescu, and G. Piella, “Building nonredundant adaptive wavelets by update lifting,” *Applied Computational Harmonic Analysis*, no. 18, pp. 252–281, May 2005.
- [15] O. N. Gerek and A. E. Çetin, “Adaptive polyphase subband decomposition structures for image compression,” *IEEE Transactions on Image Processing*, vol. 9, no. 10, pp. 1649–1659, Oct. 2000.
- [16] J. Y. Huang and P. M. Schultheiss, “Block quantization of correlated gaussian random variables,” *IEEE Transactions on Communications*, vol. 11, pp. 289–296, Sep. 1963.
- [17] V.K.Goyal, “Theoretical foundations of transform coding,” *IEEE Signal Processing Magazine*, vol. 18, no. 5, pp. 9–21, Sep. 1998.
- [18] D. L. III, “Fundamental limits of low-rate transform codes,” Ph.D. dissertation, Univ. Michigan, Ann Arbor,MI, 1992.
- [19] *Digital Compression and Coding of Continuous-Tone Still Images*, ISO/IEC JTC1, ITU-T, Sep. 1992, ISO/IEC 10918-1 — ITU-T Recommendation T.81.

- 
- [20] D. Taubman, "High performance scalable image compression with EBCOT," *IEEE Transactions on Image Processing*, vol. 9, no. 7, pp. 1158–1170, Jul. 2000.
- [21] S. Mallat, *A Wavelet Tour of Signal Processing*. Boston, MA: Academic, 1998.
- [22] I. Daubechies, *Ten Lectures on Wavelets*. Society for Industrial and Applied Mathematics, 1992.
- [23] M. Vetterli and J. Kovačević, *Wavelets and Subband Coding*. Englewood Cliffs, NJ: Prentice-Hall, 1995.
- [24] G. Strang and T. Nguyen, *Wavelets and Filter Banks*. Cambridge, MA: Wellesley-Cambridge, 1996.
- [25] B. Usevitch, "A tutorial on modern lossy wavelet image compression: foundations of JPEG2000," *IEEE Signal Processing Magazine*, vol. 18, no. 5, pp. 22–35, Sep. 2001.
- [26] J. M. Shapiro, "Embedded image coding using zerotrees of wavelets coefficients," *IEEE Transactions on Signal Processing*, vol. 41, pp. 3445–3462, Dec. 1993.
- [27] *Information Technology — JPEG 2000 Image Coding System*, International Standardization Organization, Jul. 2002, ISO/IEC 15444.
- [28] S. Mallat and F. Falzon, "Analysis of low bit rate image transform coding," *IEEE Transactions on Signal Processing*, vol. 46, no. 4, pp. 1027–1042, Apr. 1998.
- [29] O. A. Cohen, I. Daubechies and M. T. Orchard, "On the importance of combining wavelet-based nonlinear approximation with coding strategies," *IEEE Transactions on Information Theory*, vol. 48, no. 7, pp. 1895–1921, Jul. 2002.
- [30] M. Vetterli, "Wavelets, approximation and compression," *IEEE Signal Processing Magazine*, vol. 18, no. 5, pp. 59–73, Sep. 2001.
- [31] D. Donoho, M. Vetterli, R. A. DeVore, and I. Daubechies, "Data compression and harmonic analysis," *IEEE Transactions on Information Theory*, vol. 44, no. 6, pp. 2435–2476, 1998.

- 
- [32] D. H. Hubel and T. N. Wiesel, "Receptive fields, binocular interaction and functional architecture in the cat's visual cortex," *Journal of Physiology*, pp. 106–154, 1962.
- [33] J. Daugman, "Two-dimensional spectral analysis of cortical receptive field profile," *Vision Research*, pp. 847–856, 1980.
- [34] M. M. Kunt and R. Leonardi, "Recent results in high-compression image coding," *IEEE Transactions on Circuits and Systems*, vol. 34, no. 11, pp. 1306–1336, Nov. 1987.
- [35] *Coding of audio-visual objects—Part 2: Visual*, iSO/IEC 14496-2 (MPEG-4 Visual), ISO/IEC JTC 1, Version 1: Apr. 1999, Version 3: May 2004.
- [36] M. Madhavi and J. E. Fowler, "Unequal error protection of embedded multimedia objects for packet-erasure channels," in *Proceedings of IEEE Workshop on Multimedia Signal Processing*, St. Thomas, Virgin Islands, USA, Dec. 2002, pp. 61–64.
- [37] T. Gan and K.-K. Ma, "Weighted unequal error protection for transmitting scalable object-oriented images over packet-erasure networks," *IEEE Transactions on Image Processing*, vol. 14, no. 2, pp. 189–199, Feb. 2005.
- [38] J. E. Fowler and D. N. Fox, "Wavelet-based coding of three-dimensional oceanographic images around land masses," in *Proceedings of IEEE International Conference on Image Processing*, Vancouver, BC, Canada, Sep. 2000, pp. 431–434.
- [39] M. Cagnazzo, S. Parrilli, G. Poggi, and L. Verdoliva, "Costs and advantages of object-based image coding with shape-adaptive wavelet transform," *EURASIP Journal on Image and Video Processing*, vol. 2007, pp. Article ID 78 323, 13 pages, 2007, doi:10.1155/2007/78323.
- [40] A. Kawanaka and V. R. Algazi, "Zerotree coding of wavelet coefficients for image data on arbitrarily shaped support," in *Proceedings of Data Compression Conference*, Snowbird, USA, Mar. 1999.
- [41] A. G. Minami, Z. Xiong and S. Mehrotra, "3-d wavelet coding of video with arbitrary regions of support," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 11, no. 9, pp. 1063–1068, Sep. 2001.

- 
- [42] J.Tian and R.Wells Jr., "Embedded image coding using wavelet difference reduction," in *Wavelet Image and Video Compression*, Topiwala, Ed. Kluwer Academic, Norwell, Mass, USA, 1998.
- [43] Z.Lu and W.A.Pearlman, "Wavelet video coding of video object by object-based speck algorithm," in *Proceedings of Picture Coding Symposium*, Seoul, Korea, Apr. 2001, pp. 413–416.
- [44] Z. Z.Liu, J.Hua and K.Castleman, "Lossy to lossless roi coding of chromosome images using modified spiht and ebcot," in *Proceedings of IEEE International Symposium on Biomedical Imaging*, Washington, DC, USA, Jul. 2002, pp. 317–320.
- [45] J.E.Fowler, "Shape adaptive tarp coding," in *Proceedings of IEEE International Conference on Image Processing*, vol. 1, Barcelona, Spain, Sep. 2003, pp. 621–624.
- [46] P.Prandoni and M.Vetterli, "Approximation and compression of piecewise smooth functions," *Philosophical Transaction: Mathematical, Physical and Engineering Science*, vol. 357, no. 1760, pp. 2573–2591, 1999.
- [47] J.E.Fowler, "Shape adaptive coding using binary set splitting with k-d trees," in *Proceedings of IEEE International Conference on Image Processing*, vol. 5, Singapore, Oct. 2004, pp. 1301–1304.
- [48] V. Ratnakar, "Rapp: Lossless image compression with runs of adaptive pixel patterns," in *32nd Asilomar Conf. Signals, System and Computers*, vol. 2, Nov. 1998, pp. 1251–1255.
- [49] [Online]. Available: <http://sipi.usc.edu/database/>
- [50] J.E.Fowler, "Qccpack: An open-source software library for quantization, compression, and coding," in *Applications of Digital Image Processing XXIII, vol. 4115 of Proceedings of SPIE*, San Diego, CA, USA, Jul. 2000, pp. 294–301.
- [51] [Online]. Available: <http://qccpack.sourceforge.net/>
- [52] G.Gelli and G.Poggi, "Compression of multispectral images by spectral classification and transform coding," *IEEE Transactions on Image Processing*, vol. 8, no. 4, pp. 476–489, Apr. 1999.

- 
- [53] G. P.L.Dragotti and A.R.P.Ragozini, "Compression of multispectral images by three-dimensional spihit algorithm," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 38, no. 1, pp. 416–428, Jan. 2000.
- [54] J.E.Fowler and D.N.Fox, "Embedded wavelet-based coding of three-dimensional oceanographic images with land masses," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 39, no. 2, pp. 284–290, Feb. 2001.
- [55] S. M.Petrou, P.Hou and C.I.Underwood, "Region-based image coding with multiple algorithms," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 39, no. 3, pp. 562–570, Mar. 2001.
- [56] W. X.Tang and J.W.Modestino, "Hyperspectral image compression using three-dimensional wavelet coding," in *Proceedings of the SPIE*, 2003, pp. 1037–1047.
- [57] M. Cagnazzo, G. Poggi, and L. Verdoliva, "A comparison of flat and object-based transform coding techniques for the compression of multispectral images," in *Proceedings of IEEE International Conference on Image Processing*, vol. 1, Genova, Italy, Sep. 2005, pp. 657–660.
- [58] M. Cagnazzo, S. Parrilli, G. Poggi, and L. Verdoliva, "Improved class-based coding of multispectral images with shape-adaptive wavelet transform," *IEEE Geoscience and Remote Sensing Letters*, vol. 4, no. 4, pp. 566–570, Oct. 2007, 10.1109/LGRS.2007.900696.
- [59] M. Cagnazzo, R. Gaetano, S. Parrilli, and L. Verdoliva, "Adaptive region-based compression of multispectral images," in *Proceedings of IEEE International Conference on Image Processing*, Atlanta, GA (USA), Oct. 2006, pp. 3249–3252.
- [60] —, "Region based compression of multispectral images by classified KLT," in *Proceedings of European Signal Processing Conference*, Florence, Italy, Sep. 2006.
- [61] G. C.D'Elia and G.Scarpa, "A tree-structured markov random field model for bayesian image segmentation," *IEEE Transactions on Image Processing*, pp. 1259–1273, Oct. 2003.
- [62] "Free aviris standard data products web page of the jet propulsion laboratory." [Online]. Available: <http://aviris.jpl.nasa.gov/html/aviris.freedata.html>

- 
- [63] P. J. Burt and E. H. Adelson, "The laplacian pyramid as a compact image code," *IEEE Transactions on Communications*, vol. 31, pp. 532–540, Apr. 1983.
- [64] M. N. Do, "Contourlets and sparse image expansions," Ph.D. dissertation, Department of Electrical and Computer Engineering University of Illinois, Urbana IL, 2003.
- [65] M. N. Do and M. Vetterli, "Framing pyramids," *IEEE Transactions on Signal Processing*, no. 9, pp. 2329–2342, Sep. 2003.
- [66] R.H.Bamberger and M.J.T.Smith, "Directional decomposition of images: Theory and design," *IEEE Transactions on Signal Processing*, vol. 4, no. 4, pp. 882–893, Apr. 1992.
- [67] P. P. V. S.M. Phoong, C. W. Kim and R. Ansari, "A new class of two-channel biorthogonal filter banks and wavelet bases," *IEEE Transactions on Signal Processing*, vol. 43, no. 3, pp. 649–665, Mar. 1995.
- [68] C. G. Vivien Chappelier and S. Marinković, "Image coding with iterated contourlet and wavelet transforms," in *Proceedings of IEEE International Conference on Image Processing*, vol. 5, Oct. 2004.
- [69] R.Eslami and H.Radha, "A new family of nonredundant transform using hybrid wavelets and directional filter banks," *IEEE Transactions on Image Processing*, vol. 16, no. 4, pp. 1152–1167, Apr. 2007.
- [70] —, "On low bit-rate coding using the contourlet transform," in *Conf. Record of the Thirty-Sixth Asilomar Conference on Signals, Systems and Computers*, Nov. 2003.
- [71] V. M. S.Esakkirajan, T.Veerakumar and R.Sudhakar, "Image compression using contourlet transform and multistage vector quantization," *GVIP Journal*, vol. 6, no. 1, Jul. 2006.
- [72] P. Harremoës and N. Tishby, "The information bottleneck revisited or how to choose a good distortion measure," in *Proceedings of IEEE International Symposium on Information Theory*, Nice (France), Jun. 2007, pp. 566–570.
- [73] S. Parrilli, L. Verdoliva, and G. Poggi, "A spiht-like image coder based on the contourlet transform," in *Proceedings of IEEE International Conference on Image Processing*, San Diego, CA (USA), Oct. 2008.

- [74] D.D.Y.Po and M.N.Do, "Directional multiscale modeling of images using the contourlet transform," *IEEE Transactions on Image Processing*, no. 6, Jun. 2006.
- [75] W. Sweldens, "The lifting scheme: A construction of second generation wavelets," *SIAM Journal on Mathematical Analysis*, vol. 29, no. 2, pp. 511–546, 1997.
- [76] N. Mehrseresht and D. Taubman, "Spatially continuous orientation adaptive discrete packet wavelet decomposition for image compression," in *Proceedings of IEEE International Conference on Image Processing*, Atlanta, GA (USA), Oct. 2006, pp. 1593–1596.
- [77] G. Piella, B. Pesquet-Popescu, and H. J. A. M. Heijmans, "Adaptive update lifting with a decision rule based on derivative filters," *IEEE Signal Processing Letters*, vol. 9, pp. 329–332, Oct. 2002.
- [78] ———, "Gradient-driven update lifting for adaptive wavelets," *Signal Processing: Image Communication (Elsevier Science)*, vol. 20, no. 9-10, pp. 813–831, Oct.-Nov. 2005.
- [79] K. Jensen and D. Anastassiou, "Subpixel edge localization and the interpolation of still images," *IEEE Transactions on Image Processing*, vol. 4, no. 3, pp. 285–295, Mar. 1995.
- [80] D. Taubman and M. W. Marcellin, *JPEG2000: Image Compression Fundamentals, Standards and Practice*. Kluwer, 2002.
- [81] B. Usevitch, "Optimal bit allocation for biorthogonal wavelet coding," in *Proceedings of Data Compression Conference*, Snowbird, USA, Mar. 1996, pp. 387–395.
- [82] S. Parrilli, M. Cagnazzo, and B. Pesquet-Popescu, "Distortion evaluation in transform domain for adaptive lifting schemes," in *IEEE Workshop on Multimedia Signal Processing*, Cairns, Australia, Oct. 2008.
- [83] B. Pesquet-Popescu, G. Piella, H. J. A. M. Heijmans, and G. Pau, "Combining seminorms in adaptive lifting schemes and applications to image analysis and compression," *J. Math. Imaging Vis.*, vol. 25, pp. 203–226, Sep. 2006.
- [84] A. Gersho and R. M. Gray, *Vector Quantization and Signal Compression*. Kluwer Academic, Jan. 1992.