EURASIP Journal on Advances in Signal Processing
a SpringerOpen Journal

RESEARCH Open Access

Adaptive lifting scheme with sparse criteria for image coding

Mounir Kaaniche[1*], Béatrice Pesquet-Popescu[1], Amel Benazza-Benyahia[2] and Jean-Christophe Pesquet[3]

Abstract

Lifting schemes (LS) were found to be efficient tools for image coding purposes. Since LS-based decompositions depend on the choice of the prediction/update operators, many research efforts have been devoted to the design of adaptive structures. The most commonly used approaches optimize the prediction filters by minimizing the variance of the detail coefficients. In this article, we investigate techniques for optimizing sparsity criteria by focusing on the use of an ℓ_1 criterion instead of an ℓ_2 one. Since the output of a prediction filter may be used as an input for the other prediction filters, we then propose to optimize such a filter by minimizing a weighted ℓ_1 criterion related to the global rate-distortion performance. More specifically, it will be shown that the optimization of the diagonal prediction filter depends on the optimization of the other prediction filters and vice-versa. Related to this fact, we propose to jointly optimize the prediction filters by using an algorithm that alternates between the optimization of the filters and the computation of the weights. Experimental results show the benefits which can be drawn from the proposed optimization of the lifting operators.

1 Introduction

The discrete wavelet transform has been recognized to be an efficient tool in many image processing fields, including denoising [1] and compression [2]. Such a success of wavelets is due to their intrinsic features: multiresolution representation, good energy compaction, and decorrelation properties [3,4]. In this respect, the second generation of wavelets provides very efficient transforms, based on the concept of lifting scheme (LS) developed by Sweldens [5]. It was shown that interesting properties are offered by such structures. In particular, LS guarantee a lossy-to-lossless reconstruction required in some specific applications such as remote sensing imaging for which any distortion in the decoded image may lead to an erroneous interpretation of the image [6]. Besides, they are suitable tools for scalable reconstruction, which is a key issue for telebrowsing applications [7,8].

Generally, LS are developed for the 1D case and then they are extended in a separable way to the 2D case by cascading vertical and horizontal 1D filtering operators. It is worth noting that a separable LS may not appear always very efficient to cope with the two-dimensional characteristics of edges which are neither horizontal nor vertical [9]. To this respect, several research studies have been devoted to the design of non separable lifting schemes (NSLS) in order to better capture the actual two-dimensional contents of the image. Indeed, instead of using samples from the same rows (resp. columns) while processing the image along the lines (resp. columns), 2D NSLS provide smarter choices in the selection of the samples by using horizontal, vertical and oblique directions at the prediction step [9]. For example, quincunx lifting schemes were found to be suitable for coding satellite images acquired on a quincunx sampling grid [10,11]. In [12], a 2D wavelet decomposition comprising an adaptive update lifting step and three consecutive fixed prediction lifting steps was proposed. Another structure, which is composed of three prediction lifting steps followed by an update lifting step, has also been considered in the nonadaptive case [13,14].

In parallel with these studies, other efforts have been devoted to the design of adaptive lifting schemes. Indeed, in a coding framework, the compactness of a LS-based multiresolution representation depends on the choice of its prediction and update operators. To the best of our knowledge, most existing studies have mainly focused on the optimization of the prediction stage. In general, the goal of these studies is to

* Correspondence: kaaniche@telecom-paristech.fr
[1]Télécom ParisTech, 37-39 rue Dareau 75014 Paris, France
Full list of author information is available at the end of the article

introduce spatial adaptivity by varying the direction of the prediction step [15-17], the length of the prediction filters [18,19] and the coefficient values of the corresponding filters [9,11,15,20,21]. For instance, Gerek and Çetin [16] proposed a 2D edge-adaptive lifting scheme by considering three direction angles of prediction (0°, 45°, and 135°) and by selecting the orientation which leads to the smallest gradient. Recently, Ding et al. [17] have built an adaptive directional lifting structure with perfect reconstruction: the prediction is performed in local windows in the direction of high pixel correlation. A good directional resolution is achieved by employing fractional pixel precision level. A similar approach was also adopted in [22]. In [18], three separable prediction filters with different numbers of vanishing moments are employed, and then the best prediction is chosen according to the local features. In [19], a set of linear predictors of different lengths are defined based on a nonlinear function related to an edge detector. Another alternative strategy to achieve adaptivity aims at designing lifting filters by defining a given criterion. In this context, the prediction filters are often optimized by minimizing the detail signal variance through mean square criteria [15,20]. In [9], the prediction filter coefficients are optimized with a least mean squares (LMS) type algorithm based on the prediction error. In addition to these adaptation techniques, the minimization of the detail signal entropy has also been investigated in [11,21]. In [11], the approach is limited to a quincunx structure and the optimization is performed in an empirical manner using the Nelder-Mead simplex algorithm due to the fact that the entropy is an implicit function of the prediction filter. However, such heuristic algorithms present the drawback that their convergence may be achieved at a local minimum of entropy. In [21], a generalized prediction step, viewed as a mapping function, is optimized by minimizing the detail signal energy given the pixel value probability conditioned to its neighbor pixel values. The authors show that the resulting mapping function also minimizes the output entropy. By assuming that the signal probability density function (pdf) is known, the benefit of this method has firstly been demonstrated for lossless image coding in [21]. Then, an extension of this study to sparse image representation and lossy coding contexts has been presented in [23]. Consequently, an estimation of the pdf must be available at the coder and the decoder side. Note that the main drawback of this method as well as those based on directional wavelet transforms [15,17,22,24,25] is that they require to transmit losslessly a side information to the decoder which may affect the whole compression performance especially at low bitrates. Furthermore, such adaptive methods lead to an increase of the computational load required for the selection of the best direction of prediction.

It is worth pointing out that, in practical implementations of compression systems, the *sparsity* of a signal, where a portion of the signal samples are set to zero, has a great impact on the ultimate rate-distortion performance. For example, embedded wavelet-based image coders can spend the major part of their bit budget to encode the significance map needed to locate non-zero coefficients within the wavelet domain. To this end, sparsity-promoting techniques have already been investigated in the literature. Indeed, geometric wavelet transforms such as curvelets [26] and contourlets [27] have been proposed to provide sparse representations of the images. One difficulty of such transforms is their redundancy: they usually produce a number of coefficients that is larger than the number of pixels in the original image. This can be a main obstacle for achieving efficient coding schemes. To control this redundancy, a mixed contourlet and wavelet transform was proposed in [28] where a contourlet transform was used at fine scales and the wavelet transform was employed at coarse scales. Later, bandlet transforms that aim at developing sparse geometric representations of the images have been introduced and studied in the context of image coding and image denoising [29]. Unlike contourlets and curvelets which are fixed transforms, bandelet transforms require an edge detection stage, followed by an adaptive decomposition. Furthermore, the directional selectivity of the 2D complex dual-tree discrete wavelet transforms [30] has been exploited in the context of image [31] and video coding [32]. Since such a transform is redundant, Fowler et al. applied a noise-shaping process [33] to increase the sparsity of the wavelet coefficients.

With the ultimate goal of promoting sparsity in a transform domain, we investigate in this article techniques for optimizing sparsity criteria, which can be used for the design of all the filters defined in a non separable lifting structure. We should note that sparsest wavelet coefficients could be obtained by minimizing an ℓ_0 criterion. However, such a problem is inherently non-convex and NP-hard [34]. Thus, unlike previous studies where prediction has been separately optimized by minimizing an ℓ_2 criterion (i.e., the detail signal variance), we focus on the minimization of an ℓ_1 criterion. Since the output of a prediction filter may be used as an input for other prediction filters, we then propose to optimize such a filter by minimizing a *weighted* ℓ_1 criterion related to the global prediction error. We also propose to *jointly* optimize the prediction filters by using an algorithm that alternates between filter optimization and weight computation. While the minimization of an ℓ_1

criterion is often considered in the signal processing literature such as in the compressed sensing field [35], it is worth pointing out that, to the best of our knowledge, the use of such a criterion for lifting operator design has not been previously investigated.

The rest of this article is organized as follows. In Section 2, we recall our recent study for the design of all the operators involved in a 2D non separable lifting structure [36,37]. In Section 3, the motivation for using an ℓ_1 criterion in the design of optimal lifting structures is firstly discussed. Then, the iterative algorithm for minimizing this criterion is described. In Section 4, we present a weighted ℓ_1 criterion which aims at minimizing the global prediction error. In Section 5, we propose to jointly optimize the prediction filters by using an algorithm that alternates between optimizing all the filters and redefining the weights. Finally, in Section 6, experimental results are given and then some conclusions are drawn in Section 7.

2 2D lifting structure and optimization methods

2.1 Principle of the considered 2D NSLS structure

In this article, we consider a 2D NSLS composed of three prediction lifting steps followed by an update lifting step. The interest of this structure is two-fold. First, it allows us to reduce the number of lifting steps and rounding operations. A theoretical analysis has been conducted in [13] showing that NSLS improves the coding performance due to the reduction of rounding effects. Furthermore, any separable prediction-update LS structure has its equivalent in this form [13,14]. The corresponding analysis structure is depicted in Figure 1.

Let x denote the digital image to be coded. At each resolution level j and each pixel location (m, n), its approximation coefficient is denoted by $x_j(m, n)$ and the associated four polyphase components by $x_{0,j}(m, n) = x_j(2m,2n)$, $x_{1,j}(m,n) = x_j(2m,2n+1)$, $x_{2,j}(m,n) = x_j(2m+1,2n)$, and $x_{3,j}(m,n) = x_j(2m + 1, 2n + 1)$. Furthermore, we denote by $\mathbf{P}_j^{(HH)}$, $\mathbf{P}_j^{(LH)}$, $\mathbf{P}_j^{(HL)}$, and $\mathbf{U}_j$ the three prediction and update filters employed to generate the detail coefficients $x_{j+1}^{(HH)}$ oriented diagonally, $x_{j+1}^{(LH)}$ oriented vertically, $x_{j+1}^{(HL)}$ oriented horizontally, and the approximation coefficients x_{j+1}. In accordance with Figure 1, let us introduce the following notation:

- For the first prediction step, the prediction multiple input, single output (MISO) filter $\mathbf{P}_j^{(HH)}$ can be seen as a sum of three single input, single output (SISO) filters $\mathbf{P}_{0,j}^{(HH)}$, $\mathbf{P}_{1,j}^{(HH)}$, and $\mathbf{P}_{2,j}^{(HH)}$ whose respective inputs are the components $x_{0,j}$, $x_{1,j}$ and $x_{2,j}$.
- For the second (resp. third) prediction step, the prediction MISO filter $\mathbf{P}_j^{(LH)}$ (resp. $\mathbf{P}_j^{(HL)}$) can be seen as a sum of two SISO filters $\mathbf{P}_{0,j}^{(LH)}$ and $\mathbf{P}_{1,j}^{(LH)}$ (resp. $\mathbf{P}_{0,j}^{(HL)}$ and $\mathbf{P}_{1,j}^{(HL)}$) whose respective inputs are the components $x_{2,j}$ and $x_{j+1}^{(HH)}$ (resp. $x_{1,j}$ and $x_{j+1}^{(HH)}$).
- For the update step, the update MISO filter U_j can be seen as a sum of three SISO filters $\mathbf{U}_j^{(HL)}$, $\mathbf{U}_j^{(LH)}$, and $\mathbf{U}_j^{(HH)}$ whose respective inputs are the detail coefficients $x_{j+1}^{(HL)}$, $x_{j+1}^{(LH)}$, and $x_{j+1}^{(HH)}$.

Now, it is easy to derive the expressions of the resulting coefficients in the 2D z-transform domain.[a] Indeed,

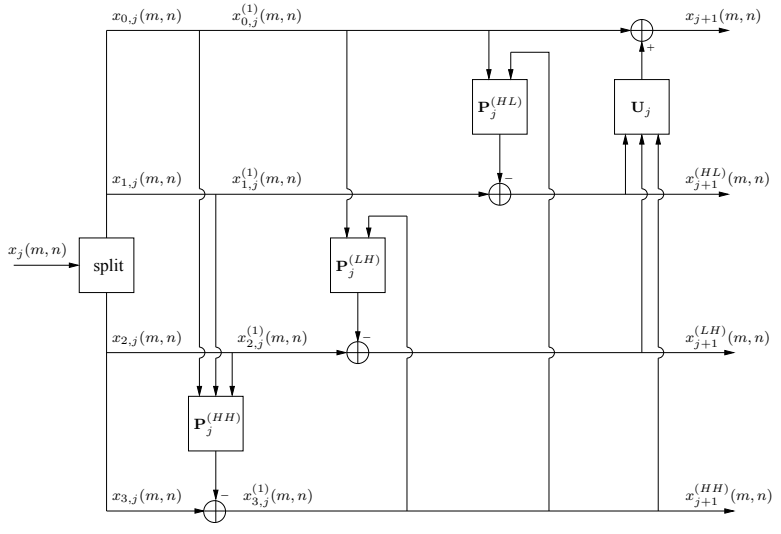


Figure 1 NSLS decomposition structure.

the z-transforms of the output coefficients can be expressed as follows:

$$X_{j+1}^{(HH)}(z_1,z_2) = X_{3,j}(z_1,z_2) - \lfloor P_{0,j}^{(HH)}(z_1,z_2)X_{0,j}(z_1,z_2) + P_{1,j}^{(HH)}(z_1,z_2)X_{1,j}(z_1,z_2) + P_{2,j}^{(HH)}(z_1,z_2)X_{2,j}(z_1,z_2)\rfloor, \quad (1)$$

$$X_{j+1}^{(LH)}(z_1,z_2) = X_{2,j}(z_1,z_2) - \left\lfloor P_{0,j}^{(LH)}(z_1,z_2)X_{0,j}(z_1,z_2) + P_{1,j}^{(LH)}(z_1,z_2)X_{j+1}^{(HH)}(z_1,z_2)\right\rfloor, \quad (2)$$

$$X_{j+1}^{(HL)}(z_1,z_2) = X_{1,j}(z_1,z_2) - \left\lfloor P_{0,j}^{(HL)}(z_1,z_2)X_{0,j}(z_1,z_2) + P_{1,j}^{(HL)}(z_1,z_2)X_{j+1}^{(HH)}(z_1,z_2)\right\rfloor, \quad (3)$$

$$X_{j+1}(z_1,z_2) = X_{0,j}(z_1,z_2) + \lfloor U_j^{(HL)}(z_1,z_2)X_{j+1}^{(HL)}(z_1,z_2) + U_j^{(LH)}(z_1,z_2)X_{j+1}^{(LH)}(z_1,z_2) + U_j^{(HH)}(z_1,z_2)X_{j+1}^{(HH)}(z_1,z_2)\rfloor \quad (4)$$

where, for every polyphase index $i \in \{0,1,2\}$ and orientation $o \in \{HH, HL, LH\}$,

$$P_{i,j}^{(o)}(z_1,z_2) = \sum_{(k,l)\in\mathcal{P}_{i,j}^{(o)}} p_{i,j}^{(o)}(k,l)z_1^{-k}z_2^{-1}, \quad \text{and} \quad U_j^{(o)}(z_1,z_2) = \sum_{(k,l)\in\mathcal{U}_j^{(o)}} u_j^{(o)}(k,l)z_1^{-k}z_2^{-l}.$$

The set $\mathcal{P}_{i,j}^{(o)}$ (resp. $\mathcal{U}_j^{(o)}$) and the coefficients $p_{i,j}^{(o)}(k,l)$ (resp. $u_j^{(o)}(k,l)$) denote the support and the weights of the three prediction filters (resp. of the update filter). Note that in Equations (1)-(4), we have introduced the rounding operations $\lfloor . \rfloor$ in order to allow lossy-to-lossless encoding of the coefficients [7]. Once the considered NSLS structure has been defined, we will focus now on the optimization of its lifting operators.

2.2 Optimization methods

Since the detail coefficients are defined as prediction errors, the prediction operators are often optimized by minimizing the variance of the coefficients (i.e., their ℓ_2-norm) at each resolution level. The rounding operators being omitted, it is readily shown that the minimum variance predictors must satisfy the well-known Yule-Walker equations. For example, for the prediction vector $\mathbf{p}_j^{(HH)}$, the normal equations read

$$\mathbf{E}[\tilde{\mathbf{x}}_j^{(HH)}(m,n)\tilde{\mathbf{x}}_j^{(HH)}(m,n)^{\mathrm{T}}]\mathbf{p}_j^{(HH)} = \mathbf{E}[x_{3,j}(m,n)\tilde{\mathbf{x}}_j^{(HH)}(m,n)] \quad (5)$$

where

- $\mathbf{p}_j^{(HH)} = (\mathbf{p}_{0,j}^{(HH)}, \mathbf{p}_{1,j}^{(HH)}, \mathbf{p}_{2,j}^{(HH)})^{\mathrm{T}}$ is the prediction vector, and, for every $i \in \{0, 1, 2\}$,

$$p_{i,j}^{(HH)} = \left(p_{i,j}^{(HH)}(k,l)\right)_{(k,l)\in\mathcal{P}_{i,j}^{(HH)}},$$

- $\tilde{\mathbf{x}}_j^{(HH)}(m,n) = (\mathbf{x}_{0,j}^{(HH)}(m,n), \mathbf{x}_{1,j}^{(HH)}(m,n), \mathbf{x}_{2,j}^{(HH)}(m,n))^{\mathrm{T}}$ is the reference vector with

$$\mathbf{x}_{i,j}^{(HH)}(m,n) = \left(x_{i,j}(m-k,n-l)\right)_{(k,l)\in\mathcal{P}_{i,j}^{(HH)}}.$$

The other optimal prediction filters $\mathbf{p}_j^{(HL)}$ and $\mathbf{p}_j^{(LH)}$ are obtained in a similar way.

Concerning the update filter, the conventional approach consists of optimizing its coefficients by minimizing the reconstruction error when the detail signal is canceled [20,38]. Recently, we have proposed a new optimization technique which aims at reducing the aliasing effects [36,37]. To this end, the update operator is optimized by minimizing the quadratic error between the approximation signal and the decimated version of the output of an ideal low-pass filter:

$$\tilde{\mathcal{J}}(\mathbf{u}_j) = \mathbf{E}\left[\left(x_{j+1}(m,n) - y_{j+1}(m,n)\right)^2\right] = \mathbf{E}\left[\left(x_{0,j}(m,n) + \sum_{o\in\{HL,LH,HH\}}\sum_{(k,l)\in\mathcal{U}_j^{(o)}} u_j^{(o)}(k,l)x_{j+1}^{(o)}(m-k,n-l) - y_{j+1}(m,n)\right)^2\right] \quad (6)$$

where $y_{j+1}(m,n) = \tilde{y}_j(2m,2n) = (h * x_j)(2m,2n)$. Recall that the impulse response of the 2D ideal low-pass filter is defined in the spatial domain by:

$$\forall (m,n) \in \mathbb{Z}^2, \quad h(m,n) = \frac{1}{4}\,\mathrm{sinc}\left(\frac{m\pi}{2}\right)\mathrm{sinc}\left(\frac{n\pi}{2}\right). \quad (7)$$

Thus, the optimal update coefficients $\mathbf{u}_j$ minimizing the criterion $\tilde{\mathcal{J}}$ are solutions of the following linear system of equations:

$$\mathbf{E}[\mathbf{x}_{j+1}(m,n)\mathbf{x}_{j+1}(m,n)^{\mathrm{T}}]\mathbf{u}_j = \mathbf{E}[y_{j+1}(m,n)\mathbf{x}_{j+1}(m,n)] - \mathbf{E}[x_{0,j}(m,n)\mathbf{x}_{j+1}(m,n)]$$

Where

- $\mathbf{u}_j = \left(u_j^{(o)}(k,l)\right)_{(k,l)\in\mathcal{U}_j^{(o)}, o\in\{HL,LH,HH\}}^{\mathrm{T}}$ is the update weight vector,
- $\mathbf{x}_{j+1}(m,n) = \left(x_{j+1}^{(o)}(m-k,n-l)\right)_{(k,l)\in\mathcal{P}_{i,j}^{(o)}, o\in\{HL,LH,HH\}}^{\mathrm{T}}$ is the reference vector containing the detail signals previously computed at the jth resolution level.

Now, we will introduce a novel twist in the optimization of the different filters: the use of an ℓ_1-based criterion in place of the usual ℓ_2-based measure.

3 From ℓ_2 to ℓ_1 minimization

3.1 Motivation

Wavelet coefficient statistics are often exploited in order to increase image compression efficiency [39]. More precisely, detail wavelet coefficients are often viewed as realizations of a zero-mean continuous random variable whose probability density function f is given by a generalized Gaussian distribution (GGD) [40,41]:

$$\forall x \in \mathbb{R}, \quad f(x;\alpha,\beta) = \frac{\beta}{2\alpha\Gamma\left(\frac{1}{\beta}\right)} e^{-\left(\frac{|x|}{\alpha}\right)^{\beta}} \tag{8}$$

where $\Gamma(z) = \int_0^{+\infty} t^{z-1}e^{-t}dt$ is the Gamma function, $\alpha > 0$ is the scale parameter, and $\beta > 0$ is the shape parameter. We should note that in the particular case when $\beta = 2$ (resp. $\beta = 1$), the GGD corresponds to the Gaussian distribution (resp. the Laplace one). The parameters α and β can be easily estimated by using the maximum likelihood technique [42].

Let us now adopt this probabilistic GGD model for the detail coefficients generated by a lifting structure. More precisely, at each resolution level j and orientation o ($o \in \{HL,LH,HH\}$), the wavelet coefficients $x_{j+1}^{(o)}(m,n)$ are viewed as realizations of random variable $X_{j+1}^{(o)}$ with probability distribution given by a GGD with parameters $\alpha_{j+1}^{(o)}$ and $\beta_{j+1}^{(o)}$. Thus, this class of distributions leads us to the following sample estimate of the differential entropy h of the variable $X_{j+1}^{(o)}$ [11,43]:

$$h(X_{j+1}^{(o)}) \approx \left(\frac{1}{M_j N_j (\alpha_{j+1}^{(o)})^{\beta_{j+1}^{(o)}} \ln(2)}\right) \sum_{m=1}^{M_j}\sum_{n=1}^{N_j} \left|x_{j+1}^{(o)}(m,n)\right|^{\beta_{j+1}^{(o)}} - \log_2\left(\frac{\beta_{j+1}^{(o)}}{2\alpha_{j+1}^{(o)}\Gamma\left(\frac{1}{\beta_{j+1}^{(o)}}\right)}\right) \tag{9}$$

where (M_j,N_j) corresponds to the dimensions of the subband $x_{j+1}^{(o)}$.

Let $\left(\bar{x}_{j+1}^{(o)}(m,n)\right)_{\substack{1\le m\le M_j \\ 1\le n\le N_j}}$ be the outputs of a uniform quantizer with quantization step q driven with the real-valued coefficients $\left(x_{j+1}^{(o)}(m,n)\right)_{\substack{1\le m\le M_j \\ 1\le n\le N_j}}$. The coefficients $\bar{X}_{j+1}^{(o)}$ can be viewed as realizations of a random variable $\bar{X}_{j+1}^{(o)}$ taking its values in {..., -2q, -q, 0, q, 2q, ...}. At high resolution, it was proved in [43] that the following relation holds between the discrete entropy $\bar{X}_{j+1}^{(o)}$ and the differential entropy h of $X_{j+1}^{(o)}$:

$$H(\bar{X}_{j+1}^{(o)}) \approx h(X_{j+1}^{(o)}) - \log_2(q). \tag{10}$$

Thus, from Equation (9), we see [44] that the entropy $H(\bar{X}_{j+1}^{(o)})$ of $\bar{X}_{j+1}^{(o)}$ is (up to a dividing factor and an additive constant) approximatively equal to:

$$\sum_{m=1}^{M_j}\sum_{n=1}^{N_j} \left|x_{j+1}^{(o)}(m,n)\right|^{\beta_{j+1}^{(o)}}.$$

This shows that there exists a close link between the minimization of the entropy of the detail wavelet coefficients and the minimization of their $\ell_{\beta_{j+1}^{(o)}}$-norm. This suggests in particular that most of the existing studies minimizing the ℓ_2-norm of the detail signals aim at minimizing their entropy by assuming a Gaussian model.

Based on these results, we have analyzed the detail wavelet coefficients generated by the decomposition based on the lifting structure NSLS(2,2)-OPT-L2 described in Section 6. Figure 2 shows the distribution of each detail subband for the "einst" image when the prediction filters are optimized by minimizing the ℓ_2-norm of the detail coefficients. The maximum likelihood technique is used to estimate the β parameter.

It is important to note that the shape parameters of the resulting detail subbands are closer to $\beta = 1$ than *to* $\beta = 2$. Further experiments performed on a large dataset of images[b] have shown that the average of β values are

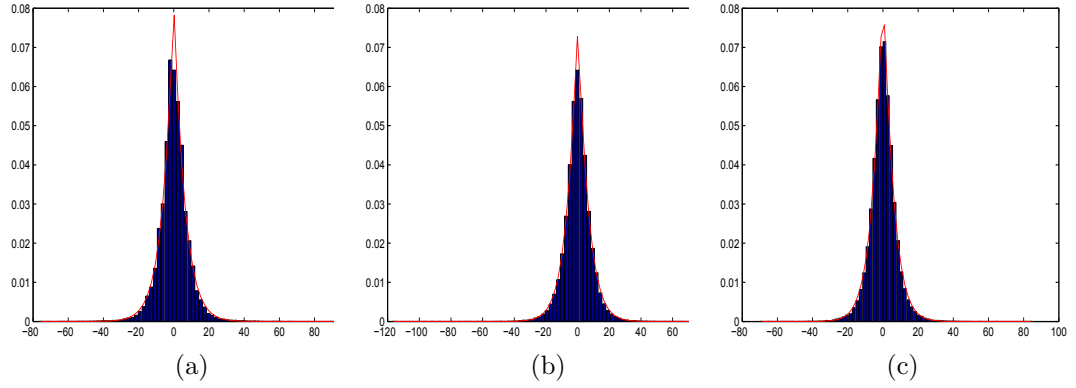


Figure 2 The GGD of the. (a) horizontal detail subband $x_1^{(HL)}(\beta_1^{(HL)} = 1.07)$, **(b)**: vertical detail subband $x_1^{(LH)}(\beta_1^{(LH)} = 1.14)$, (c): diagonal detail subband $x_1^{(HH)}(\beta_1^{(HH)} = 1.15)$. The detail coefficients of the "einst" image are optimized by minimizing their ℓ_2-norm.

closer to 1 (typical values range from 0.5 to 1.5). These observations suggest that minimizing the ℓ_1-norm may be more appropriate than ℓ_2 minimization. In addition, the former approach has the advantage of producing sparse representations.

3.2 ℓ_1 minimization technique

Instead of minimizing the ℓ_2-norm of the detail coefficients $x_{j+1}^{(o)}$ as done in [37], we propose in this section to optimize each of the prediction filters by minimizing the following ℓ_1 criterion:

$$\forall o \in \{HL, LH, HH\}, \forall i \in \{1, 2, 3\}, \quad \mathcal{J}_{\ell_1}(\mathbf{p}_j^{(o)}) = \sum_{m=1}^{M_j} \sum_{n=1}^{N_j} \left| x_{i,j}(m,n) - (\mathbf{p}_j^{(o)})^{\mathrm{T}} \tilde{\mathbf{x}}_j^{(o)}(m,n) \right| \quad (11)$$

where $x_{i,j}(m,n)$ is the $(i+1)^{th}$ polyphase component to be predicted, $\tilde{\mathbf{x}}_j^{(o)}(m,n)$ is the reference vector containing the samples used in the prediction step, $\mathbf{p}_j^{(o)}$ is the prediction operator vector to be optimized (L will subsequently designate its length). Although the criterion in (11) is convex, a major difficulty that arises in solving this problem stems from the fact that the function to be minimized is not differentiable. Recently, several optimization algorithms have been proposed to solve non-smooth minimization problems like (11). These problems have been traditionally addressed with linear programming [45]. Alternatively, a flexible class of proximal optimization algorithms has been developed and successfully employed in a number of applications. A survey on these proximal methods can be found in [46]. These methods are also closely related to augmented Lagrangian methods [47]. In our context, we have employed the Douglas-Rachford algorithm which is an efficient optimization tool for this problem [48].

3.2.1 The Douglas-Rachford algorithm

For minimizing the ℓ_1 criterion, we will resort to the concept of proximity operators [49], which has been recognized as a fundamental tool in the recent convex optimization literature [50,51]. The necessary background on convex analysis and proximity operators [52,53] is given in Appendix A.

Now, we recall that our minimization problem (11) aims at optimizing the prediction filters by minimizing the ℓ_1-norm of the difference between the current pixel $x_{i,j}$ and its predicted value. We note here that $\mathbf{x}_{i,j} = \left(x_{i,j}(m,n)\right)_{\substack{1 \le m \le M_j \\ 1 \le n \le N_j}}$ can be viewed as an element of the Euclidean space $\mathbb{R}^{K_j}$, where $K_j = M_j \times N_j$. Thus, the minimization problem (11) can be rewritten as:

$$\forall o \in \{HL, LH, HH\}, \forall i \in \{1, 2, 3\}, \quad \min_{z_j^{(o)} \in V} \sum_{m=1}^{M_j} \sum_{n=1}^{N_j} \left| x_{i,j}(m,n) - z_j^{(o)}(m,n) \right| \quad (12)$$

where V is the vector space defined as

$$V = \left\{ \mathbf{z}_j^{(o)} = \left(z_j^{(o)}(m,n) \right)_{\substack{1 \le m \le M_j \\ 1 \le n \le N_j}} \in \mathbb{R}^{K_j} | \exists \mathbf{p}_j^{(o)} \in \mathbb{R}^L, \right.$$
$$\left. \forall (m,n) \in \{1, \ldots, M_j\} \times \{1, \ldots, N_j\}, z_j^{(o)}(m,n) = (\mathbf{p}_j^{(o)})^{\mathrm{T}} \tilde{\mathbf{x}}_j^{(o)}(m,n) \right\}.$$

Based on the definition of the indicator function ι_V (see Appendix A), Problem (12) is equivalent to the following minimization problem:

$$\forall o \in \{HL, LH, HH\}, \forall i \in \{1, 2, 3\}, \quad \min_{z_j^{(o)} \in \mathbb{R}^{K_j}} \sum_{m=1}^{M_j} \sum_{n=1}^{N_j} \left| x_{i,j}(m,n) - z_j^{(o)}(m,n) \right| + \iota_V(\mathbf{z}_j^{(o)}). \quad (13)$$

Therefore, Problem (13) can be viewed as a minimization of a sum of two functions f_1 and f_2 defined by:

$$f_1(\mathbf{z}_j^{(o)}) = ||\mathbf{x}_{i,j} - \mathbf{z}_j^{(o)}||_{\ell_1} = \sum_{m=1}^{M_j} \sum_{n=1}^{N_j} \left| x_{i,j}(m,n) - z_j^{(o)}(m,n) \right| \quad (14)$$

$$f_2(\mathbf{z}_j^{(o)}) = \iota_V(\mathbf{z}_j^{(o)}). \quad (15)$$

In this case, the Douglas-Rachford algorithm can be applied to provide an appealing numerical solution to Problem (13) (see Appendix B).

Although it is an iterative algorithm, we have observed experimentally that the convergence of the Douglas-Rachford algorithm is generally ensured after a small number of iterations (often between 30 et 60 iterations). As an example, we plot in Figure 3a (resp. 3b) the evolution of the criterion $\mathcal{J}_{\ell_1}(\mathbf{p}_0^{(HH)})$ (resp. $\mathcal{J}_{\ell_1}(\mathbf{p}_0^{(LH)})$) w.r. t the iteration number for this algorithm.

Once the different terms involved in the iterative algorithm (33) are defined, this one can be applied and further extended to optimize all the prediction filters.

4 Global prediction error minimization technique

4.1 Motivation

Up to now, each prediction filter $\mathbf{p}_j^{(o)}(o \in \{HL, LH, HH\})$ has been separately optimized by minimizing the ℓ_1-norm of the corresponding detail signal $x_{j+1}^{(o)}$ which seems appropriate to determine $\mathbf{p}_j^{(LH)}$ and $\mathbf{p}_j^{(HL)}$. However, it can be noticed from Figure 1 that the diagonal detail signal $x_{j+1}^{(HH)}$ is also used through the second and the third prediction steps to compute the vertical and the horizontal detail signals respectively. Therefore, the solution $\mathbf{p}_j^{(HH)}$ resulting from the previous optimization method may be suboptimal. As a result, we propose to optimize the prediction filter $\mathbf{p}_j^{(HH)}$ by minimizing the global prediction error, as described in detail in the next section.

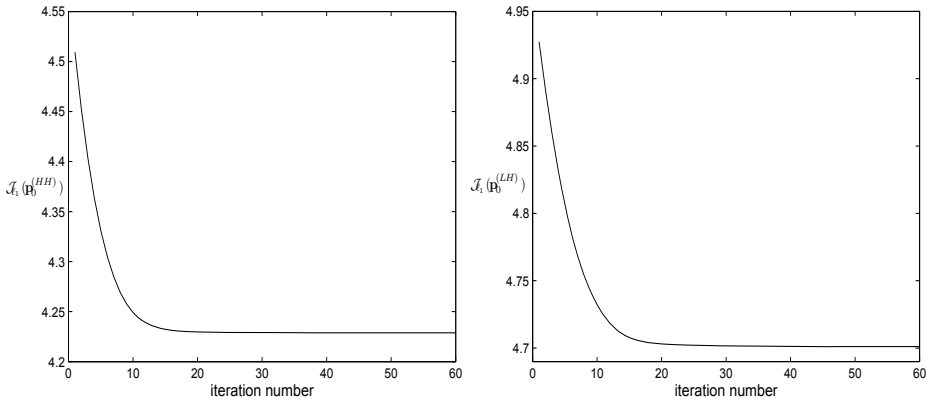


Figure 3 Convergence of the Douglas Rachford algorithm w.r.t the iteration number: (a) evolution of $\mathcal{J}_{\ell_1}(\mathbf{p}_0^{(HH)})$, (b) evolution of $\mathcal{J}_{\ell_1}(\mathbf{p}_0^{(LH)})$.

4.2 Optimization of the prediction filter $\mathbf{p}_j^{(HH)}$

More precisely, instead of minimizing the ℓ_1-norm of $x_{j+1}^{(HH)}$, the filter $\mathbf{p}_j^{(HH)}$ will be optimized by minimizing the sum of the ℓ_1-norm of the three detail subbands $x_{j+1}^{(o)}$. To this respect, we will consider the minimization of the following weighted ℓ_1 criterion:

$$\mathcal{J}_{w\ell_1}(\mathbf{p}_j^{(HH)}) = \sum_{o\in\{HL,LH,HH\}} \sum_{m,n} \kappa_{j+1}^{(o)} \left| x_{j+1}^{(o)}(m,n) \right| \tag{16}$$

where $\kappa_{j+1}^{(o)}$, $o \in \{HL, LH, HH\}$, are strictly positive weighting terms.

Before focusing on the method employed to minimize the proposed criterion, we should first express $\mathcal{J}_{w\ell_1}$ as a function of the filter $\mathbf{p}_j^{(HH)}$ to be optimized.

Let $\left(x_{i,j}^{(1)}(m,n)\right)_{i\in\{0,1,2,3\}}$ be the four outputs obtained from $\left(x_{i,j}(m,n)\right)_{i\in\{0,1,2,3\}}$ following the first prediction step (see Figure 1). Although $x_{i,j}^{(1)}(m,n) = x_{i,j}(m,n)$ for all $i \in \{0, 1, 2\}$, the use of the superscript will make the presentation below easier. Thus $x_{j+1}^{(o)}$ can be expressed as:

$$\begin{aligned} x_{j+1}^{(o)}(m,n) &= \sum_{i\in\{0,1,2,3\}} \sum_{k,l} h_{i,j}^{(o,1)}(k,l) x_{i,j}^{(1)}(m-k,n-l) \\ &= \sum_{i\in\{0,1,2\}} \sum_{k,l} h_{i,j}^{(o,1)}(k,l) x_{i,j}^{(1)}(m-k,n-l) + \sum_{k,l} h_{3,j}^{(o,1)}(k,l) x_{3,j}^{(1)}(m-k,n-l) \end{aligned} \tag{17}$$

where $h_{i,j}^{(o,1)}$ is a filter which depends on the prediction coefficients of $\mathbf{p}_j^{(LH)}$ and $\mathbf{p}_j^{(HL)}$.

Knowing that

$$x_{3,j}^{(1)}(m,n) = x_{3,j}(m,n) - (\mathbf{p}_j^{(HH)})^{\mathrm{T}} \tilde{\mathbf{x}}_j^{(HH)}(m,n) \tag{18}$$

where $\tilde{\mathbf{x}}_j^{(HH)}(m,n) = \left(x_{i,j}(m-r,n-s)\right)_{(r,s)\in\mathcal{P}_j^{(HH)}}(\mathcal{P}_j^{(HH)}$ $_{i\in\{0,1,2\}}$ is the support of the predictor $\mathbf{p}_j^{(HH)}$), we thus obtain, after some simple calculations,

$$\forall o \in \{HH, LH, HL\}, \quad x_{j+1}^{(o)}(m,n) = y_j^{(o,1)}(m,n) - (\mathbf{p}_j^{(HH)})^{\mathrm{T}} \mathbf{x}_j^{(o,1)}(m,n) \tag{19}$$

Where

$$y_j^{(o,1)}(m,n) = \sum_{i\in\{0,1,2\}} \sum_{k,l} h_{i,j}^{(o,1)}(k,l) x_{i,j}^{(1)}(m-k,n-l) + \sum_{k,l} h_{3,j}^{(o,1)}(k,l) x_{3,j}(m-k,n-l), \tag{20}$$

$$\mathbf{x}_j^{(o,1)}(m,n) = \left(\sum_{k,l} h_{3,j}^{(o,1)}(k,l) x_{i,j}(m-k-r,n-l-s) \right)_{\substack{(r,s)\in\mathcal{P}_j^{(HH)} \\ i\in\{0,1,2\}}}. \tag{21}$$

Consequently, the proposed weighted ℓ_1 criterion (Equation (16)) can be expressed as:

$$\mathcal{J}_{w\ell_1}(\mathbf{p}_j^{(HH)}) = \sum_{o\in\{HL,LH,HH\}} \sum_{m,n} \kappa_{j+1}^{(o)} \left| y_j^{(o,1)}(m,n) - (\mathbf{p}_j^{(HH)})^{\mathrm{T}} \mathbf{x}_j^{(o,1)}(m,n) \right|. \tag{22}$$

It is worth noting that in practice, the determination of $y_j^{(o,1)}(m,n)$ and $\mathbf{x}_j^{(o,1)}(m,n)$ does not require to find the explicit expressions of $h_{i,j}^{(o,1)}$ and these signals can be determined numerically as follows:

- The first term (resp. the second one) in the expression of $y_j^{(o,1)}(m,n)$ in Equation (20) can be found by computing $x_{j+1}^{(o)}(m,n)$ from the components

$\mathcal{J}_{\ell_1}(\mathbf{p}_0^{(LH)})$ while setting $x_{3,j}^{(1)}(m,n) = 0$ (resp. while setting $x_{i,j}^{(1)}(m,n) = 0$ for $i \in \{0,1,2\}$ and $x_{3,j}^{(1)}(m,n) = x_{3,j}(m,n)$).

• The vector $\mathbf{x}_j^{(o,1)}(m,n)$ in Equation (21) can be found as follows. For each $i \in \{0,1,2\}$, the computation of its component $\sum_{k,l} h_{3,j}^{(o,1)}(k,l) x_{i,j}(m-k,n-l)$ requires to compute $x_{j+1}^{(o)}(m,n)$ by setting $x_{3,j}^{(1)}(m,n) = x_{i,j}(m,n)$ and $x_{i',j}^{(1)}(m,n) = 0$ for $i' \in \{0,1,2\}$. The result of this operation has to be considered for different shift values (r, s) (as can be seen in Equation (21)).

Once the different terms involved in the proposed weighted criterion in Equation (22) are defined (the constant values $\kappa_{j+1}^{(o)}$ are supposed to be known), we will focus now on its minimization. Indeed, unlike the previous criterion (Equation 11), which consists only of an ℓ_1 term, the proposed criterion is a sum of three ℓ_1 terms. To minimize such a criterion (22), one can still use the Douglas-Rachford algorithm through a formulation in a product space [46,54].

4.2.1 Douglas-Rachford algorithm in a product space

Consider the ℓ_1 minimization problem:

$$\min_{\mathbf{p}_j^{(HH)}} \sum_{o\in\{H,L,LH,HH\}} \sum_{m,n} k_{j+1}^{(o)} \left| y_j^{(o,1)}(m,n) - (\mathbf{p}_j^{(HH)})^T \mathbf{x}_j^{(o,1)}(m,n) \right| \quad (23)$$

where $\kappa_{j+1}^{(o)}$, $o \in \{HL,LH,HH\}$, are positive weights.

Since the Douglas-Rachford algorithm described hereabove is designed for the sum of two functions, we can reformulate (23) under this form in the 3-fold product space $\mathbb{H}_j$

$$\mathbb{H}_j = \mathbb{R}^{K_j} \times \mathbb{R}^{K_j} \times \mathbb{R}^{K_j}. \quad (24)$$

If we define the vector subspace U as

$$\begin{aligned} U = \Big\{ \mathbf{Z}_j = \begin{pmatrix} \mathbf{z}_j^{(HH,1)} \\ \mathbf{z}_j^{(LH,1)} \\ \mathbf{z}_j^{(HL,1)} \end{pmatrix} \in \mathbb{H}_j | \exists\, \mathbf{p}_j^{(HH)} \in \mathbb{R}^L, \forall o \in \{HH, LH, HL\}, \\ \forall (m,n) \in \{1,2,\ldots,M_j\} \times \{1,2,\ldots,N_j\}, z_j^{(o,1)}(m,n) = (\mathbf{p}_j^{(HH)})^T \mathbf{x}_j^{(o,1)}(m,n) \Big\} \\ = \Big\{ \mathbf{Z}_j = \begin{pmatrix} \mathbf{z}_j^{(HH,1)} \\ \mathbf{z}_j^{(LH,1)} \\ \mathbf{z}_j^{(HL,1)} \end{pmatrix} \in \mathbb{H}_j | \exists\, \mathbf{p}_j^{(HH)} \in \mathbb{R}^L, \forall (m,n) \in \{1,2,\ldots,M_j\} \times \{1,2,\ldots,N_j\}, \\ \mathbf{Z}_j(m,n) = \mathbf{X}_j(m,n)^T \mathbf{p}_j^{(HH)} \text{ with } \mathbf{X}_j(m,n) = \left(\mathbf{x}_j^{(HH,1)}(m,n), \mathbf{x}_j^{(LH,1)}(m,n), \mathbf{x}_j^{(HL,1)}(m,n) \right) \Big\}, \end{aligned} \quad (25)$$

the minimization problem (Equation 23) is equivalent to

$$\min_{\mathbf{z}_j \in \mathbb{H}_j} f_3(\mathbf{z}_j) + f_4(\mathbf{z}_j) \quad (26)$$

where

$$\begin{aligned} f_3(\mathbf{z}_j) &= \sum_{o\in\{HL,LH,HH\}} \sum_{m,n} \kappa_{j+1}^{(o)} \left| y_j^{(o,1)}(m,n) - z_j^{(o,1)}(m,n) \right| \\ f_4(\mathbf{z}_j) &= \iota_U(\mathbf{z}_j). \end{aligned} \quad (27)$$

We are thus back to a problem involving two functions in a larger space, which is the product space $\mathbb{H}_j$. So, the Douglas-Rachford algorithm can be applied to solve our minimization problem (see Appendix C). Finally, once the prediction filter $\mathbf{p}_j^{(HH)}$ is optimized and fixed, it can be noticed that the other prediction filters $\mathbf{p}_j^{(HL)}$ and $\mathbf{p}_j^{(LH)}$ can be separately optimized by minimizing $\mathcal{J}_{\ell_1}(\mathbf{p}_j^{(HL)})$ and $\mathcal{J}_{\ell_1}(\mathbf{p}_j^{(LH)})$ as explained in Section 3. This is justified by the fact that the inputs of the filter $\mathbf{p}_j^{(HL)}$ (resp. $\mathbf{p}_j^{(LH)}$) are independent of the output of the filter $\mathbf{p}_j^{(LH)}$ (resp. $\mathbf{p}_j^{(HL)}$).

5 Joint optimization method

5.1 Motivation

From Equations (20) and (21), it can be observed that $y_j^{(o,1)}$ and $\mathbf{x}_j^{(o,1)}$, which are used to optimize $\mathbf{p}_j^{(HH)}$, depend on the coefficients of the prediction filters $\mathbf{p}_j^{(HL)}$ and $\mathbf{p}_j^{(LH)}$. On the other hand, since $\mathbf{p}_j^{(HL)}$ and $\mathbf{p}_j^{(LH)}$ use $x_{j+1}^{(HH)}$ as reference signal in the second and the third prediction steps, their optimal values will depend on the optimal prediction filter $\mathbf{p}_j^{(HH)}$. Thus, we conclude that the optimization of the filters ($\mathbf{p}_j^{(HL)}$, $\mathbf{p}_j^{(LH)}$) depends on the optimization of the filter $\mathbf{p}_j^{(HH)}$ and vice-versa.

A *joint* optimization method can therefore be proposed which iteratively optimizes the prediction filters $\mathbf{p}_j^{(HL)}$, $\mathbf{p}_j^{(HL)}$, and $\mathbf{p}_j^{(LH)}$.

5.2 Proposed algorithms

While the optimization of the prediction filters $\mathbf{p}_j^{(HL)}$ and $\mathbf{p}_j^{(LH)}$ is simple, the optimization of the prediction filter $\mathbf{p}_j^{(HH)}$ is less obvious. Indeed, if we examine the criterion $\mathcal{J}_{w\ell_1}$, the immediate question that arises is: which values of the weighting parameters will produce the sparsest decomposition?

A simple solution consists of setting all the weights $\kappa_{j+1}^{(o)}$ to one. Then, we are considering the particular case

of the unweighted ℓ_1 criterion, which simply represents the sum of the ℓ_1-norm of the three details subbands $x_{j+1}^{(o)}$. In this case, the joint optimization problem is solved by applying the following simple iterative algorithm at each resolution level *j*.

5.2.1 First proposed algorithm

① Initialize the iteration number *it* to 0.

- Optimize separately the three prediction filters as explained in Section 3. The resulting filters will be denoted respectively by $\mathbf{p}_j^{(HH,0)}$, $\mathbf{p}_j^{(LH,0)}$, and $\mathbf{p}_j^{(HL,0)}$.
- Compute the resulting global unweighted prediction error (i.e., the sum of the ℓ_1-norm of the three resulting details subbands).

② for *it* = 1,2,3,

- Set $\mathbf{p}_j^{(LH)} = \mathbf{p}_j^{(LH,it-1)}$, $\mathbf{p}_j^{(HL)} = \mathbf{p}_j^{(HL,it-1)}$, and optimize $\mathbf{P}_j^{(HH)}$ by minimizing $\mathcal{J}_{w\ell_1}(\mathbf{p}_j^{(HH)})$ (while setting $\kappa_{j+1}^{(o)} = 1$). Let $\mathbf{p}_j^{(HH,it)}$ be the new optimal filter at iteration *it*.
- Set $\mathbf{p}_j^{(HH)} = \mathbf{p}_j^{(HH,it)}$, and optimize $\mathbf{P}_j^{(LH)}$ by minimizing $\mathcal{J}_{\ell_1}(\mathbf{p}_0^{(LH)})$. Let $\mathbf{p}_j^{(LH,it)}$ be the new optimal filter.
- Set $\mathbf{p}_j^{(HH)} = \mathbf{p}_j^{(HH,it)}$, and optimize $\mathbf{P}_j^{(HL)}$ by minimizing $\mathcal{J}_{\ell_1}(\mathbf{p}_j^{(HL)})$. Let $\mathbf{p}_j^{(HL,it)}$ be the new optimal filter.

Once the prediction filters are optimized, the update filter is finally optimized as explained in Section 2. However, in practice, once all the filters are optimized and the decomposition is performed, the different generated wavelet subbands $x_{j+1}^{(o)}$ are weighted before the entropy encoding (using JPEG2000 encoder) in order to obtain a distortion in the spatial domain which is very close to the distortion in the wavelet domain.

More precisely, as we can see in Figure 4, each wavelet subband is multiplied by $\sqrt{w_{j+1}^{(o)}}$, where $w_{j+1}^{(o)}$ represents the weight corresponding to $x_{j+1}^{(o)}$. Generally, these weights are computed based on the wavelet filters used for the reconstruction process as indicated in [55,56]. A simple weight computation procedure based on the following assumption can be used. As shown in [55], if the error signal in a subband (i.e., the quantization noise) is white and uncorrelated to the other subband errors, the reconstruction distortion in the spatial domain is a weighted sum of the distortion in each wavelet subband. Therefore, for each subband $x_{j+1}^{(o)}$, a white Gaussian noise of variance $(\sigma_{j+1}^{(o)})^2$ is firstly added while keeping the remaining subbands noiseless. Then, the resulting distortion in the spatial domain $\hat{D}_s$ is evaluated by taking the inverse transform. Finally, the corresponding subband weight can be estimated as follows:

$$w_{j+1}^{(o)} = \frac{\hat{D}_s \times 4^{j+1}}{(\sigma_{j+1}^{(o)})^2}. \tag{28}$$

This weighting step is very important since standard bit allocation algorithms assume that the quadratic distortion in the wavelet domain is equal to that in the spatial domain, which is not true in the case of biorthogonal wavelets [55]. Therefore, the filters resulting from the first choice of $\kappa_{j+1}^{(o)}$ are suboptimal in the sense that they do not take into account the weighting procedure.

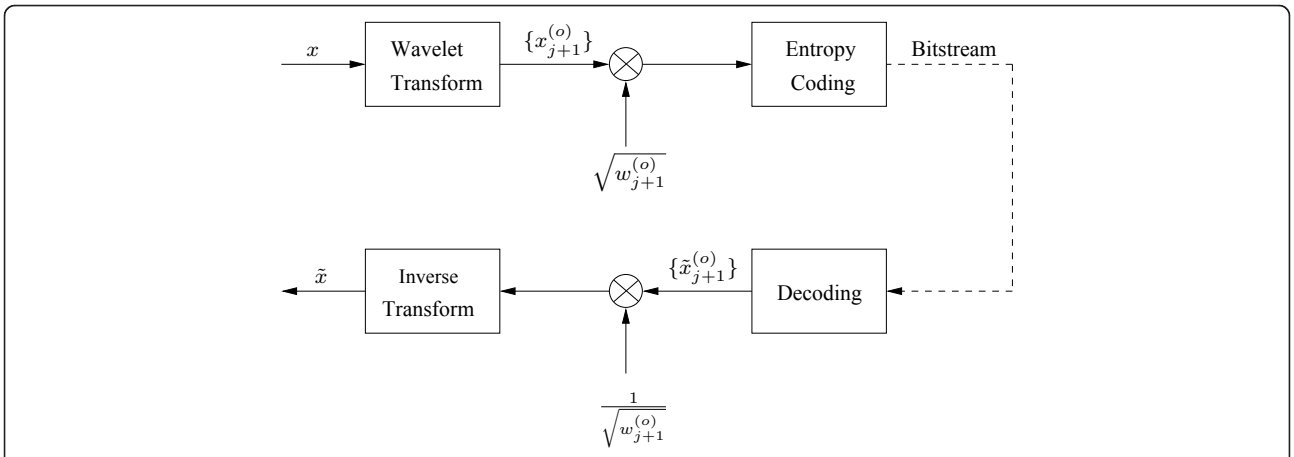


Figure 4 Wavelet-based compression procedure involving a weighting prior the encoding stage.

For this reason, it has been noticed on some experiments (as it can be seen in Section 6) that the basic optimization technique does not achieve the best coding performances.

Thus, a more judicious choice of $\kappa_{j+1}^{(o)}$ should take into account the weighting procedure applied to the wavelet coefficients before the entropy encoding process. Furthermore, if in the general formula in Equation (9), we consider the case of $\beta_{j+1}^{(o)} = 1$, the differential entropy of $X_{j+1}^{(o)}$ multiplied by $\sqrt{w_{j+1}^{(o)}}$ becomes:

$$\frac{1}{M_j N_j \alpha_{j+1}^{(o)} \ln(2)} \sum_{m=1}^{M_j} \sum_{n=1}^{N_j} \left| x_{j+1}^{(o)}(m,n) \right| + \log_2 \left(2\alpha_{j+1}^{(o)} \sqrt{w_{j+1}^{(o)}} \right) \quad (29)$$

where $\alpha_{j+1}^{(o)}$ can be estimated by using a classical maximum likelihood estimate. Thus, it can be observed from Equation (29) that the first term of the resulting entropy, which corresponds to a weighted ℓ_1-norm of $x_{j+1}^{(o)}$, is inversely proportional to $\alpha_{j+1}^{(o)}$. Consequently, in order to obtain a criterion (Equation 16) that results in a good approximation of the entropy (29), a more reasonable choice of $\kappa_{j+1}^{(o)}$ will be as follows:

$$\kappa_{j+1}^{(o)} = \frac{1}{\alpha_{j+1}^{(o)}}. \quad (30)$$

Since the resulting entropy of each subband uses weights which also depend on the prediction filters (as mentioned above), we propose an iterative algorithm that alternates between optimizing all the filters and redefining the weights. This algorithm, which is performed for each resolution level j, is as follows.

5.2.2 Second proposed algorithm

① Initialize the iteration number *it* to 0.

- Optimize separately the three prediction filters as explained in Section 3. The resulting filters will be denoted respectively by $\mathbf{p}_j^{(HH,0)}$, $\mathbf{p}_j^{(LH,0)}$, and $\mathbf{p}_j^{(HL,0)}$.
- Optimize the update filter (as explained in Section 2).
- Compute the weights $w_{j+1}^{(o,0)}$ of each detail subband as well as the constant values $\kappa_{j+1}^{(o,0)}$.

② for *it* = 1,2,3,...

- Set $\mathbf{p}_j^{(LH)} = \mathbf{p}_j^{(LH,it-1)}$, $\mathbf{p}_j^{(HL)} = \mathbf{p}_j^{(HL,it-1)}$, and optimize $\mathbf{P}_j^{(HH)}$ by minimizing $\mathcal{J}_{w\ell_1}(\mathbf{p}_j^{(HH)})$. Let $\mathbf{p}_j^{(HH,it)}$ be the new optimal filter.
- Set $\mathbf{p}_j^{(HH)} = \mathbf{p}_j^{(HH,it)}$, and optimize $\mathbf{P}_j^{(LH)}$ by minimizing $\mathcal{J}_{\ell_1}(\mathbf{p}_j^{(LH)})$. Let $\mathbf{p}_j^{(LH,it)}$ be the new optimal filter.
- Set $\mathbf{p}_j^{(HH)} = \mathbf{p}_j^{(HH,it)}$, and optimize $\mathbf{P}_j^{(HL)}$ by minimizing $\mathcal{J}_{\ell_1}(\mathbf{p}_j^{(HL)})$. Let $\mathbf{p}_j^{(HL,it)}$ be the new optimal filter.
- Optimize the update filter (as explained in Section 2).
- Compute the new weights $w_{j+1}^{(o,it)}$ as well as $\kappa_{j+1}^{(o,it)}$.

Let us now make some observations concerning the convergence of the proposed algorithm. Since the goal of the second weighting procedure is to better approximate the entropy, we have computed at the end of each iteration number *it* the differential entropy of the three resulting details subbands. More precisely, the evaluated criterion, obtained from Equation (29) by setting $\alpha_{j+1}^{(o)} = \frac{1}{\kappa_{j+1}^{(o)}}$ and performing the sum over the three details subbands, is given by:

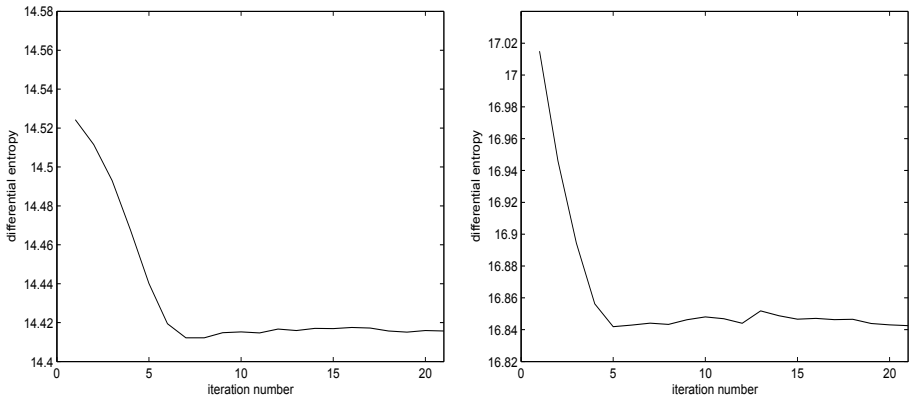
$$\sum_{o\in\{HL,LH,HH\}} \left(\frac{\kappa_{j+1}^{(o,it)}}{M_j N_j \ln(2)} \sum_{m=1}^{M_j} \sum_{n=1}^{N_j} |x_{j+1}^{(o)}(m,n)| + \log_2 \left(\frac{2\sqrt{w_{j+1}^{(o,it)}}}{\kappa_{j+1}^{(o,it)}} \right) \right). \quad (31)$$

Figure 5 illustrates the evolution of this criterion w.r.t the iteration number of the algorithm. It can be noticed that the decrease of the criterion is mainly achieved during the early iterations (about after 7 iterations).

6 Experimental results

Simulations were carried out on two kinds of still images originally quantized over 8 bpp which are either single views or stereoscopic ones. A large dataset composed of 50 still images[b] and 50 stereo images[c] has been considered. The gain related to the optimization of the NSLS operators, using different minimization criteria, was evaluated in these contexts. In order to show the benefits of the proposed ℓ_1 optimization criterion, we provide the results for the following decompositions carried out over three resolution levels:

- The first one is the LS corresponding to the 5/3 transform, also known as the (2,2) wavelet transform [7]. In the following, this method will be designated by NSLS(2,2).
- The second method consists of optimizing the prediction and update filters as proposed in [20,38]. More precisely, the prediction filters are optimized



(a) (b)

Figure 5 Convergence of the optimization algorithm w.r.t the iteration number *it* when it is performed at: (a) *j* = 1, (b) *j* = 2.

by minimizing the ℓ_2-norm of the detail coefficients whereas the update filter is optimized by minimizing the reconstruction error. This optimization method will be designated by NSLS(2,2)-OPT-GM.

• The third approach corresponds to our previous method presented recently in [37]. While the prediction filters are optimized in the same way as the second method, the update filter is optimized by minimizing the difference between the approximation signal and the decimated version of the output of an ideal low-pass filter. We emphasize here that the prediction filters are optimized *separately*. This method will be denoted by NSLS(2,2)-OPT-L2.

• The fourth method modifies the optimization stage of the prediction filters by using the ℓ_1-norm instead of the ℓ_2-norm. The optimization of the update filter is similar to the technique used in the third method. In what follows, this method will be designated by NSLS(2,2)-OPT-L1.

• The fifth method consists of *jointly* optimizing the prediction filters by using the proposed weighted ℓ_2 minimization technique where the weights $\kappa_{j+1}^{(o)}$ are set to $\frac{1}{\alpha_{j+1}^{(o)}}$. The optimization of the update filter is similar to the technique used in the third and fourth methods. This optimization method will be designated by NSLS(2,2)-OPT-WL1. We have also tested this optimization method when the weights $\kappa_{j+1}^{(o)}$ are set to 1. In this case, the method will be denoted by NSLS(2,2)-OPT-WL1 ($\kappa_{j+1}^{(o)} = 1$).

Figures 6 and 7 show the scalability in quality of the reconstruction procedure by providing the variations of the PSNR versus the bitrate for the images "castle" and "straw" using JPEG2000 as entropy codec. A more exhaustive evaluation was also performed by applying the different methods to 50 still images[b]. The *average* PSNR per-image is illustrated in Figure 8.

These plots show that NSLS(2,2)-OPT-L2 outperforms NSLS(2,2) by 0.1-0.5 dB. It can also be noticed that NSLS(2,2)-OPT-L2 and NSLS(2,2)-OPT-GM perform similarly in terms of quality of reconstruction. An improvement of 0.1-0.3 dB is obtained by using the ℓ_1 minimization technique instead of the ℓ_2 one. Finally, the *joint* optimization technique (NSLS(2,2)-OPT-WL1) outperforms the *separate* optimization technique (NSLS (2,2)-OPT-L1) and improves the PSNR by 0.1-0.2 dB. The gain becomes more important (up to 0.55 dB) when compared with NSLS(2,2)-OPT-L2. It is important to note here that setting the weights $\kappa_{j+1}^{(o)}$ to 1 (NSLS (2,2)-OPT-WL1 ($\kappa_{j+1}^{(o)} = 1$)) can yield to a degradation of about 0.1-0.25 dB compared with NSLS(2,2)-OPT-WL1 on some images.

Figures 9 and 10 display the reconstructed images of "lena" and "einst". In addition to PSNR and SSIM metrics, the quality ofthe reconstructed images are also compared in terms of VSNR (Visual Signal-to-Noise ratio) which was found to be an efficient metric for quantifying the visual fidelity of natural images [57]: it is based on physical luminances and visual angle (rather than on digital pixel values and pixel-based dimensions) to accommodate different viewing conditions. It can be observed that the weighted ℓ_1 minimization technique significantly improves the visual quality of reconstruction. The difference in VSNR (resp. PSNR) between NSLS(2,2)-OPT-L2 and NSLS(2,2)-OPT-WL1 ranges from 0.35 dB to 0.6 dB (resp. 0.25 dB to 0.3 dB). Comparing Figure 9c (resp. Figure 10c) with Figure 9d (resp. Figure 10d), the visual improvement achieved by our

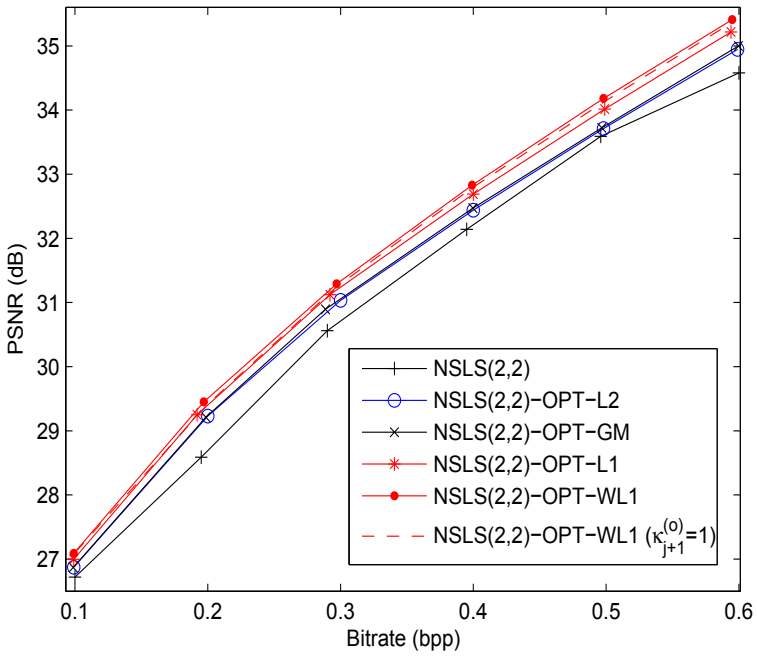


Figure 6 PSNR (in dB) versus the bitrate (bpp) after JPEG2000 progressive encoding for the "castle" image.

method can be mainly seen in the hat and face of Lena (resp. in Einstein's face).

The second part of the experiments is concerned with stereo images. Most of the existing studies in this field rely on disparity compensation techniques [58,59]. The basic principles involved in this technique first consists of estimating the disparity map. Then, one image is considered as a reference image and the other is predicted in order to generate a prediction error referred to as a residual image. Finally, the disparity field, the reference image and the residual one are encoded [58,60]. In this context, Moellenhoff and Maier [61] analyzed the

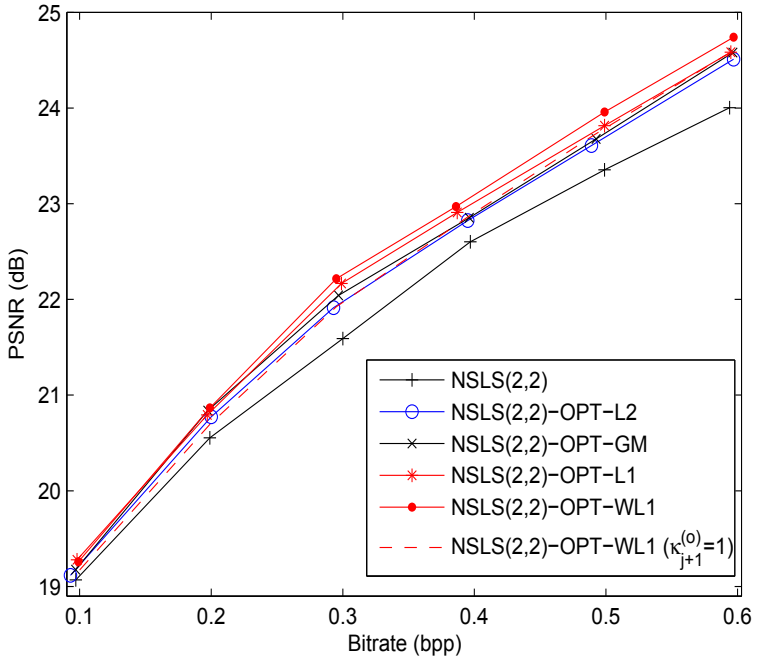


Figure 7 PSNR (in dB) versus the bitrate (bpp) after JPEG2000 progressive encoding for the "straw" image.

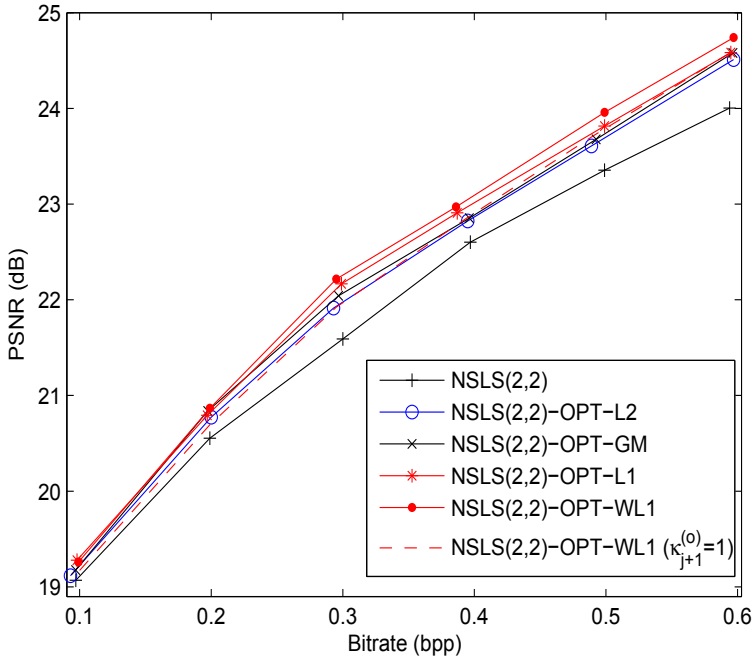


Figure 8 Average PSNR (in dB) computed over 50 still images versus the bitrate (in bpp) after JPEG2000 progressive encoding.

characteristics of the residual image and proved that such images have properties different from natural images. This suggests that transforms that work well for natural images may not be as well-suited for residual images. For this reason, we also proposed to apply these optimization methods for encoding the reference image and the residual one. The resulting rate-distortion curves for the "white house" and "pentagon" stereo images are illustrated in Figures 11 and 12. A more exhaustive evaluation was also performed by applying the different methods to 50 stereo images[c]. The *average* PSNR per-image is illustrated in Figure 13. Figure 14 displays the reconstructed target image of the "pentagon" stereo pair. It can be observed that the proposed *joint* optimization method leads to an improvement of 0.35 dB (resp. 0.016) in VSNR (resp. SSIM) compared with the decomposition in which the prediction filters are optimized *separately*. For instance, it can be noticed that the edges of the pentagon's building as well as the roads are better reconstructed in Figure 14d.

For completeness, the performance of the proposed method (NSLS(2,2)-OPT-WL1) has also been compared with the 9/7 transform retained for the lossy mode of JPEG2000 standard. Table 1 shows the performance of the latter methods in terms of PSNR, SSIM, and VSNR. Since the human eye cannot always distinguish the subjective image quality at middle and high bitrate, the results were restricted to the lower bitrate values.

While the proposed method is less performant in terms of PSNR than the 9/7 transform for some images, it can be noticed from Table 1 that better results are obtained in terms of perceptual quality. For instance, Figures 15 and 16 illustrate some reconstructed images. It can be observed that the proposed method (NSLS (2,2)-OPT-WL1) achieves a gain of about 0.2-0.4 dB (resp. 0.01-0.013) in terms of VSNR (resp. SSIM). Furthermore, Figures 17 and 18 display the reconstructed target image for the stereo image pairs "shrub" and "spot5". While NSLS(2,2)-OPT-WL1 and 9/7 transform show similar visual quality for the "spot5" pair, the proposed method leads to better quality of reconstruction than the 9/7 transform for the "shrub" stereo images.

Before concluding the article, let us now study the complexity of the proposed sparsity criteria for the optimization of the prediction filters. Table 2 gives the iteration number and the execution time for the ℓ_1 and weighted ℓ_1 minimization techniques when considering different image sizes. These results have been obtained with a Matlab implementation on an Intel Core 2 (2.93 GHz) architecture. It is clear that the execution time increases with the image size. Furthermore, we note that the ℓ_1 minimization technique is very fast whereas the weighted ℓ_1 technique needs an additional time of about 0.3-2.6 seconds. This increase is due to the fact that the algorithm is reformulated in a three-fold product space

(a): Original image

(b): PSNR=30.44 dB, SSIM=0.844, VSNR=22.96 dB

(c): PSNR=30.93 dB, SSIM=0.845, VSNR=23.46 dB

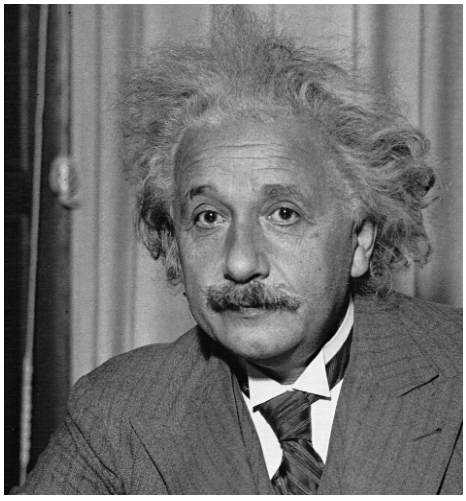
(d): PSNR=31.25 dB, SSIM=0.851, VSNR=24.06 dB

Figure 9 Reconstructed image at 0.15 bpp using: (a) Original "lena" image. (b) NSLS(2,2), (c) NSLS(2,2)-OPT-L2, (d) NSLS(2,2)-OPT2-WL1.

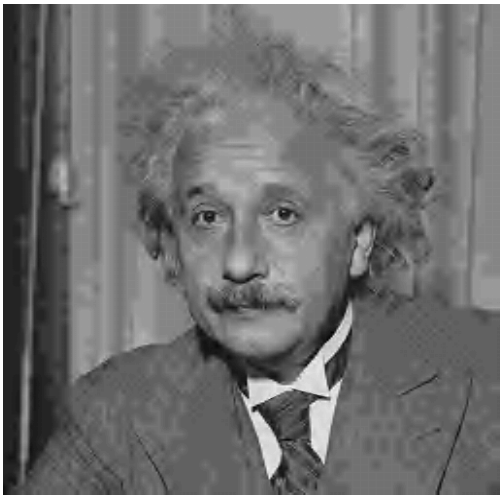
as explained in Section 4.2. However, since the Douglas-Rachford algorithm in a product space has some blocks which can be implemented in a parallel way, the complexity can be reduced significantly (up to three times) when performing an appropriate implementation on a multicore architecture. These results as well as the good compression performance in terms of reconstruction quality confirm the effectiveness of the proposed sparsity criteria.

7 Conclusion

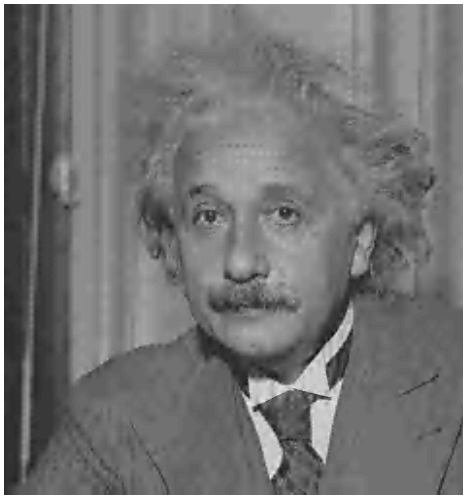
In this article, we have studied different optimization techniques for the design of filters in a NSLS structure. A new criterion has been presented for the optimization of the prediction filters in this context. The idea consists of *jointly* optimizing these filters by minimizing iteratively a weighted ℓ_1 criterion. Experimental results carried out on still images and stereo images pair have illustrated the benefits which can be drawn from the



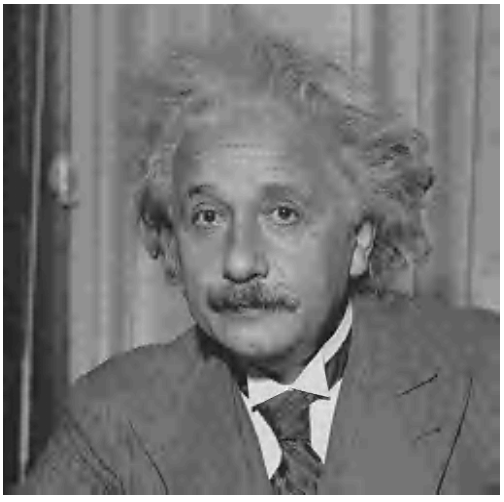
(a): Original image



(b): PSNR=28.55 dB, SSIM=0.648, VSNR=17.82 dB



(c): PSNR=28.94 dB, SSIM=0.649, VSNR=18.24 dB



(d): PSNR=29.12 dB, SSIM=0.654, VSNR=18.62 dB

Figure 10 Reconstructed image at 0.1 bpp using: (a) Original "einst" image, (b) NSLS(2,2), (c) NSLS(2,2)-OPT-L2, (d) NSLS(2,2)-OPT2-WL1.

proposed optimization technique. In future study, we plan to extend this optimization method to LS with more than two stages like the P-U-P and P-U-P-U structures.

Appendix

A Some background on convex optimization

The main definitions which will be useful to understand our optimization algorithms are briefly summarized below:

- $\mathbb{R}^K$ is the usual K-dimensional Euclidean space with norm $||.||$.
- The distance function to a nonempty set $C \subset \mathbb{R}^K$ is defined by

$$\forall \mathbf{x} \in \mathbb{R}^K, \quad d_C(\mathbf{x}) = \inf_{\mathbf{y} \in C} ||\mathbf{x} - \mathbf{y}||.$$

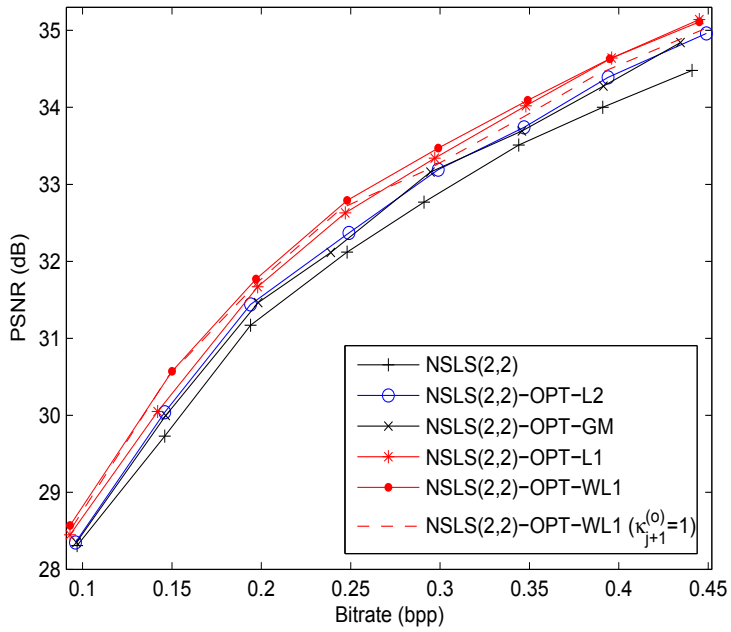


Figure 11 PSNR (in dB) versus the bitrate (bpp) after JPEG2000 progressive encoding for the "white house" stereo images.

• The projection of $\mathbf{x} \in \mathbb{R}^K$ onto a nonempty closed convex set $C \subset \mathbb{R}^K$ is the unique point $P_C(\mathbf{x}) \in C$ such that $d_C(\mathbf{x}) = ||\mathbf{x} - P_C(\mathbf{x})||$.

• The indicator function of C is given by

$$\forall \mathbf{x} \in \mathbb{R}^K, \quad \iota_C(\mathbf{x}) = \begin{cases} 0 & \text{if } \mathbf{x} \in C, \\ +\infty & \text{otherwise.} \end{cases} \tag{32}$$

• $\Gamma_0(\mathbb{R}^K)$ is the class of functions from $\mathbb{R}^K$ to $]-\infty, +\infty]$ which are lower semi-continuous, convex, and not identically equal to $+\infty$.

• The proximity operator of $f \in \Gamma_0(\mathbb{R}^K)$ is $\text{prox}_f : \mathbb{R}^K \to \mathbb{R}^K : \mathbf{x} \mapsto \arg\min_{\mathbf{y} \in \mathbb{R}^K} f(\mathbf{y}) + \frac{1}{2}||\mathbf{x} - \mathbf{y}||^2$. It is important to note that the proximity operator

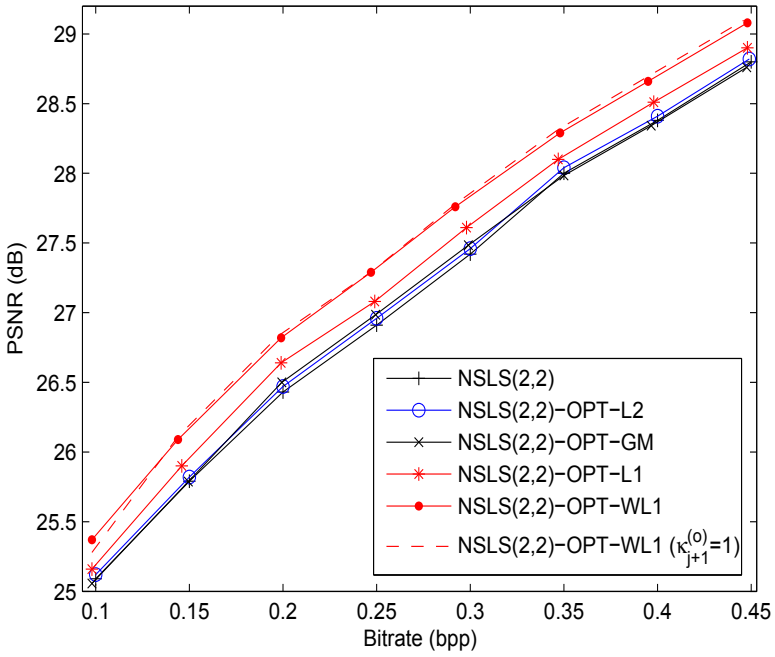


Figure 12 PSNR (in dB) versus the bitrate (bpp) after JPEG2000 progressive encoding for the "pentagon" stereo images.

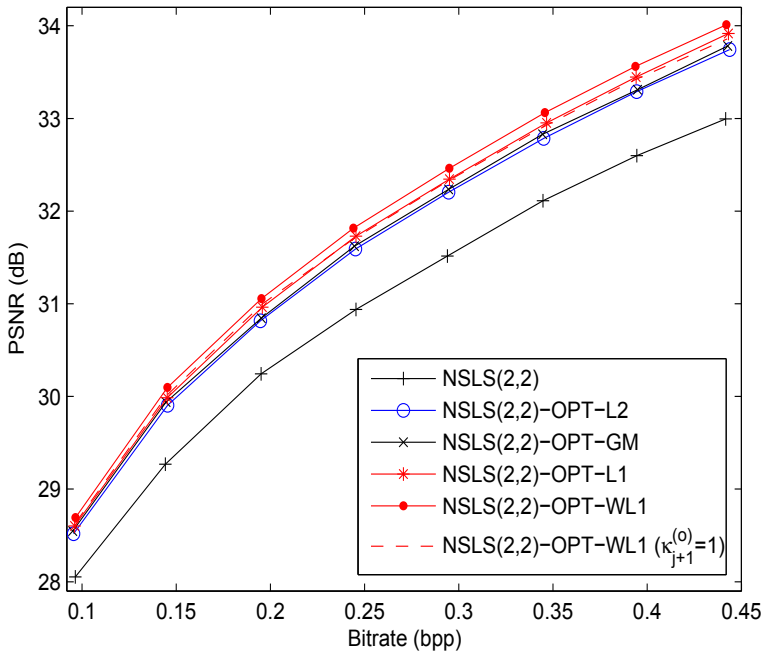


Figure 13 Average PSNR (in dB) computed over 50 stereo images versus the bitrate (in bpp) after JPEG2000 progressive encoding.

(a): Original image

(b): PSNR=26.44 dB, SSIM=0.693, VSNR=12.17 dB

(c): PSNR=26.56 dB, SSIM=0.691, VSNR=12.49 dB

(d): PSNR=26.90 dB, SSIM=0.697, VSNR=13.06 dB

Figure 14 Reconstructed target image at 0.15 bpp using: (a) Original target image for the "pentagon" stereo images. (b) NSLS(2,2) (c) NSLS(2,2)-OPT-L2 (d) NSLS(2,2)-OPT2-WL1.

Table 1 Performance of the proposed method vs the 9/7 transform

		0.05 bpp		0.1 bpp		0.15 bpp		0.2 bpp	
		NSLS(2,2)-OPT-WL1	9/7	NSLS (2,2)-OPT-WL1	9/7	NSLS (2,2)-OPT-WL1	9/7	NSLS (2,2)-OPT-WL1	9/7
elaine	PSNR	**27.85**	27.75	30.25	**30.31**	31.23	**31.35**	31.76	**31.92**
	SSIM	**0.669**	0.659	**0.716**	0.715	**0.739**	0.739	0.754	**0.756**
	VSNR	**18.44**	18.09	**23.10**	23.05	**25.60**	25.50	27.28	**27.42**
castle	PSNR	**25.10**	25.09	27.08	**27.18**	28.36	**28.51**	29.51	**29.58**
	SSIM	**0.725**	0.712	**0.790**	0.780	**0.825**	0.821	**0.855**	0.851
	VSNR	**17.54**	17.22	**21.55**	21.10	**23.74**	23.40	**25.80**	25.32
einst	PSNR	27.51	**27.58**	29.12	**29.24**	29.92	**30.12**	30.50	**30.70**
	SSIM	**0.603**	0.601	0.654	**0.655**	0.687	**0.689**	0.710	**0.715**
	VSNR	**15.33**	15.25	18.62	**18.71**	20.37	**20.47**	21.59	**21.94**
lena	PSNR	**26.70**	26.68	**29.59**	29.56	31.25	**31.47**	32.70	**32.90**
	SSIM	**0.747**	0.734	**0.818**	0.808	**0.851**	0.850	0.871	**0.873**
	VSNR	**15.94**	15.73	**20.56**	20.18	**24.06**	23.95	26.12	**26.15**
cameraman	PSNR	**26.51**	26.43	29.81	**30.33**	31.84	**32.63**	33.61	**34.44**
	SSIM	**0.783**	0.774	**0.847**	0.842	0.887	**0.892**	0.914	**0.915**
	VSNR	**16.74**	16.34	**21.73**	21.66	24.94	**25.70**	27.75	**28.34**
boat	PSNR	**24.65**	24.55	26.82	**26.86**	28.43	**28.54**	29.52	**29.74**
	SSIM	**0.675**	0.661	**0.753**	0.746	**0.806**	0.802	**0.837**	0.836
	VSNR	**13.41**	13.03	**17.14**	16.89	**20.24**	19.76	**22.19**	21.89
peppers	PSNR	**25.75**	25.50	**29.24**	29.17	30.88	**31.16**	31.12	**32.38**
	SSIM	**0.720**	0.705	**0.789**	0.778	**0.818**	0.815	**0.834**	0.832
	VSNR	**16.00**	15.51	**21.87**	21.19	**25.18**	25.00	**27.22**	27.09
plane	PSNR	**24.19**	23.84	**30.66**	29.88	**33.99**	33.10	**36.13**	35.82
	SSIM	**0.809**	0.754	**0.890**	0.871	**0.917**	0.903	**0.931**	0.921
	VSNR	**9.48**	7.72	**17.73**	15.51	**21.28**	20.30	**24.68**	24.12
average	PSNR	**24.88**	24.72	27.67	**27.73**	29.24	**29.46**	30.45	**30.65**
	SSIM	**0.647**	0.633	**0.727**	0.720	**0.773**	0.771	**0.803**	0.802
	VSNR	**14.50**	13.98	**18.90**	18.62	**21.77**	21.71	**23.90**	23.85

The *average* evaluation was computed over 50 still images.
The values in bold have been used to identify the method achieving the best coding performance.

generalizes the notion of a projection operator onto a closed convex set C in the sense that $\text{prox}_{\iota_C} = P_C$, and it moreover possesses most of its attractive properties [49] that make it particularly well-suited for designing iterative minimization algorithms.

B The Douglas Rachford algorithm

The solution of the Problem (13) (which is the sum of the two functions f_1 and f_2) is obtained by the following iterative algorithm:

$$\begin{aligned} &\text{Set } \mathbf{t}_{j,0}^{(o)} \in \mathbb{R}^{K_j}, \gamma > 0, \lambda \in]0,2[, \text{ and,} \\ &\text{for } k = 0, 1, 2, \ldots \\ &\quad \mathbf{z}_{j,k}^{(o)} = \text{prox}_{\gamma f_2} \mathbf{t}_{j,k}^{(o)} \\ &\quad \mathbf{t}_{j,k+1}^{(o)} = \mathbf{t}_{j,k}^{(o)} + \lambda\left(\text{prox}_{\gamma f_1}\left(2\mathbf{z}_{j,k}^{(o)} - \mathbf{t}_{j,k}^{(o)}\right) - \mathbf{z}_{j,k}^{(o)}\right). \end{aligned} \tag{33}$$

An important feature of this algorithm is that it proceeds by splitting, in the sense that the functions f_1 and f_2 are dealt with in separate steps: in the first step, only function f_2 is required to obtain $\mathbf{z}_{j,k}^{(o)}$ and, in the second

Table 2 Computation time (s) of the sparse optimization methods for the design of each prediction filter

	Plane 256 × 256		Girl 256 × 256		Boat 512 × 512		Cameraman 512 × 512	
	it	Time (s)	*it*	time(s)	*it*	time(s)	*it*	time(s)
ℓ_1 criterion: $\mathbf{p}_0^{(HL)}$	22	0.09	27	0.09	30	0.38	60	0.81
ℓ_1 criterion: $\mathbf{p}_0^{(LH)}$	55	0.15	28	0.09	31	0.39	100	1.13
weighted ℓ_1 criterion: $\mathbf{p}_0^{(HH)}$	30	0.42	35	0.49	49	3.08	30	2.01

(a): PSNR=26.68 dB, SSIM=0.734, VSNR=15.73 dB (b): PSNR=26.70 dB, SSIM=0.747, VSNR=15.94 dB

Figure 15 Zoom applied on the reconstructed "lena" image at 0.05 bpp using: (a) 9/7 transform (b) NSLS(2,2)-OPT-WL1.

step, only function f_1 is involved to obtain $\mathbf{t}_{j,k+1}^{(o)}$. Furthermore, it can be seen that the algorithm requires to compute two proximity operators $\text{prox}_{\gamma f_1}$, and $\text{prox}_{\gamma f_2}$ at each iteration. One can find in [46] closed-form expression of the proximity operator of various functions in $\Gamma_0(\mathbb{R})$. In our case, the proximity operator of γf_1 is given by the soft-thresholding rule:

$$\forall\, \mathbf{t}_{j,k}^{(o)} \in \mathbb{R}^{K_j}, \quad \text{prox}_{\gamma f_1}(\mathbf{t}_{j,k}^{(o)}) = \left(\pi_{j,k}^{(o)}(m,n)\right)_{\substack{1\le m\le M_j \\ 1\le n\le N_j}} \quad (34)$$

(a): PSNR=29.56 dB, SSIM=0.808, VSNR=20.18 dB (b): PSNR=29.59 dB, SSIM=0.818, VSNR=20.56 dB

Figure 16 Zoom applied on the reconstructed "lena" image at 0.1 bpp using: (a) 9/7 transform (b) NSLS(2,2)-OPT-WL1.

(a): PSNR=28.68 dB, SSIM=0.682, VSNR=19.27 dB

(b): PSNR=28.76 dB, SSIM=0.698, VSNR=19.63 dB

Figure 17 Zoom applied on the reconstructed target image for the "shrub" stereo images at 0.1 bpp using: (a) 9/7 transform (b) NSLS (2,2)-OPT-WL1.

where

$$\pi_{j,k}^{(o)}(m,n) = \mathrm{soft}_{[-\gamma,\gamma]}\left(t_{j,k}^{(o)}(m,n) - x_{i,j}(m,n)\right) + x_{i,j}(m,n)$$

and

$$\forall \alpha \in \mathbb{R}, \quad \mathrm{soft}_{[-\gamma,\gamma]}(\alpha) = \begin{cases} \mathrm{sign}(\alpha)(|\alpha| - \gamma) & \text{if} |\alpha| > \gamma \\ 0 & \text{otherwise.} \end{cases} \tag{35}$$

Concerning $\mathcal{G}_2$, it is easy to check that its proximity operator is expressed as:

$$\begin{aligned} \forall \mathbf{t}_{j,k}^{(o)} \in \mathbb{R}^{K_j}, \quad & \mathrm{prox}_{\gamma f_2}(\mathbf{t}_{j,k}^{(o)}) = P_V(\mathbf{t}_{j,k}^{(o)}) \\ & = \left(\hat{z}_{j,k}^{(o)}(m,n)\right)_{\substack{1\le m\le M_j \\ 1\le n\le N_j}} = \left((\mathbf{p}_{j,k}^{(o)})^T \tilde{\mathbf{x}}_j^{(o)}(m,n)\right)_{\substack{1\le m\le M_j \\ 1\le n\le N_j}} \end{aligned} \tag{36}$$

where

$$\mathbf{p}_{j,k}^{(o)} = \left(\sum\nolimits_{m,n} \tilde{\mathbf{x}}_j^{(o)}(m,n)(\tilde{\mathbf{x}}_j^{(o)}(m,n))^{\mathrm{T}}\right)^{-1} \sum\nolimits_{m,n} \tilde{\mathbf{x}}_j^{(o)}(m,n) t_{j,j}^{(o)}(m,n)..$$

Finally it is important to note that it has been shown (see [62] and references therein) that every sequence $(\mathbf{z}_{j,k}^{(o)})_{k\in\mathbb{N}}$ generated by the Douglas-Rachford algorithm (33) converges to a solution to problem (13) provided that the parameters γ and λ are fixed as indicated.

C The Douglas-Rachford algorithm in a product space

The solution of the problem (26) (which is the sum of the two functions f_3 and f_4) is obtained by the following iterative algorithm:

$$\begin{aligned} &\text{Set } \mathbf{t}_{j,0} \in \mathbb{H}_j, \gamma > 0, \lambda \in]0,2[, \text{ and,} \\ &\quad \text{for } k = 0, 1, 2, \ldots \\ &\qquad \mathbf{z}_{j,k} = \mathrm{prox}_{\gamma f_4} \mathbf{t}_{j,k} \\ &\qquad \mathbf{t}_{j,k+1} = \mathbf{t}_{j,k} + \lambda\left(\mathrm{prox}_{\gamma f_3}(2\mathbf{z}_{j,k} - \mathbf{t}_{j,k}) - \mathbf{z}_{j,k}\right). \end{aligned} \tag{37}$$

Note that the above algorithm requires to compute the proximity operators of 2 new functions $\mathcal{G}_3$ and $\mathcal{G}_4$. Concerning the proximity operator of $\mathcal{G}_3$, we have

$$\forall \mathbf{t}_j = \begin{pmatrix} \mathbf{t}_j^{(HH,1)} \\ \mathbf{t}_j^{(LH,1)} \\ \mathbf{t}_j^{(HL,1)} \end{pmatrix} \in \mathbb{H}_j, \mathrm{prox}_{\gamma f_3}(\mathbf{t}_{j,k}) = \begin{pmatrix} \mathrm{soft}_{\left[-\gamma\kappa_{j+1}^{(HH)}, \gamma\kappa_{j+1}^{(HH)}\right]}(\mathbf{t}_{j,k}^{(HH,1)}) \\ \mathrm{soft}_{\left[-\gamma\kappa_{j+1}^{(LH)}, \gamma\kappa_{j+1}^{(LH)}\right]}(\mathbf{t}_{j,k}^{(LH,1)}) \\ \mathrm{soft}_{\left[-\gamma\kappa_{j+1}^{(HL)}, \gamma\kappa_{j+1}^{(HL)}\right]}(\mathbf{t}_{j,k}^{(HL,1)}) \end{pmatrix} \tag{38}$$

Where

$$\forall o \in \{HH, LH, HL\},$$

$$\mathrm{soft}_{\left[-\gamma\kappa_{j+1}^{(o)}, \gamma\kappa_{j+1}^{(o)}\right]}(\mathbf{t}_{j,k}^{(o,1)}) = \left(\mathrm{soft}_{\left[-\gamma\kappa_{j+1}^{(o)}, \gamma\kappa_{j+1}^{(o)}\right]}(t_{j,k}^{(o,1)}(m,n))\right)_{\substack{1\le m\le M_j \\ 1\le n\le N_j}}.$$

Concerning $\mathcal{G}_4$, its proximity operator is given by:

$$\begin{aligned} \mathrm{prox}_{\gamma f_4}(\mathbf{t}_{j,k}) &= P_U(\mathbf{t}_{j,k}) \\ &= (\hat{\mathbf{Z}}_{j,k}(m,n))_{\substack{1\le m\le M_j \\ 1\le n\le N_j}} = (\mathbf{X}_j(m,n)^{\mathrm{T}} \mathbf{p}_{j,k}^{(HH)})_{\substack{1\le m\le M_j \\ 1\le n\le N_j}} \end{aligned} \tag{39}$$

where

$$\mathbf{p}_{j,k}^{(HH)} = \left(\sum_m \sum_n \mathbf{X}_j(m,n)\mathbf{X}_j(m,n)^{\mathrm{T}}\right)^{-1} \sum_{m,n} \mathbf{X}_j(m,n)\mathbf{t}_{j,k}(m,n).$$

(a): PSNR=27.22 dB, SSIM=0.678, VSNR=13.41 dB (b): PSNR=27.25 dB, SSIM=0.680, VSNR=13.44 dB

Figure 18 Zoom applied on the reconstructed target image for a "spot5" stereo images at 0.1 bpp using: (a) 9/7 transform (b) NSLS(2,2)-OPT-WL1.

Endnotes

[a]The z-transform of a signal x will be denoted in capital letters by X. [b]http://sipi.usc.edu/database. [c]http://vasc.ri.cmu.edu/idb/html/stereo/index.html, http://vasc.ri.cmu.edu/idb/html/jisct/index.html and http://cat.middlebury.edu/stereo/data.html.

Acknowledgements
Part of this study has been presented in [63].

Author details
[1]Télécom ParisTech, 37-39 rue Dareau 75014 Paris, France [2]Ecole Supérieure des Communications de Tunis (SUP'COM-Tunis), Université de Carthage, Tunis 2083, Tunisia [3]Université Paris-Est, Laboratoire d'Informatique Gaspard Monge and CNRS UMR 8049, Marne-la-Vallée 77454, France

Authors' contributions
All authors read and approved the final manuscript.

Competing interests
The authors declare that they have no competing interests.

Received: 30 June 2011 Accepted: 13 January 2012
Published: 13 January 2012

References

1. DL Donoho, IM Johnstone, Ideal spatial adaptation by wavelet shrinkage. Biometrika. **81**(3), 425–455 (1994). doi:10.1093/biomet/81.3.425
2. M Antonini, M Barlaud, P Mathieu, I Daubechies, Image coding using wavelet transform. IEEE Trans Image Process. **1**(2), 205–220 (1992). doi:10.1109/83.136597
3. JW Woods, Subband Image Coding, (Kluwer Academic Publishers, Norwell, MA, USA, 1990)
4. S Mallat, A Wavelet Tour of Signal Processing, (Academic Press, San Diego, 1998)
5. W Sweldens, The lifting scheme: a custom-design construction of biorthogonal wavelets, (Appl Comput Harmonic Anal, 1996)**3**(2), , pp. 186–200
6. K Arai, Preliminary study on information lossy and lossless coding data compression for the archiving of ADEOS data. IEEE Trans Geosci Remote Sens. **28**, 732–734 (1990). doi:10.1109/TGRS.1990.573001
7. AR Calderbank, I Daubechies, W Sweldens, BL Yeo, Wavelet transforms that map integers to integers. Appl Comput Harmonic Anal. **5**(3), 332–369 (1998). doi:10.1006/acha.1997.0238
8. D Taubman, M Marcellin, JPEG2000: Image Compression Fundamentals, Standards and Practice, (Kluwer Academic Publishers, Norwell, MA, USA, 2001)
9. ON Gerek, AE Çetin, Adaptive polyphase subband decomposition structures for image compression. IEEE Trans Image Process. **9**(10), 1649–1660 (2000). doi:10.1109/83.869176
10. A Gouze, M Antonini, M Barlaud, B Macq, Optimized lifting scheme for two-dimensional quincunx sampling images, in *IEEE International Conference on Image Processing*, vol. 2. (Thessa-loniki, Greece, 2001), pp. 253–258
11. A Benazza-Benyahia, JC Pesquet, J Hattay, H Masmoudi, Block-based adaptive vector lifting schemes for multichannel image coding. EURASIP Int J Image Video Process 10 (2007). (2007)
12. H Heijmans, G Piella, B Pesquet-Popescu, Building adaptive 2D wavelet decompositions by update lifting, in *IEEE International Conference on Image Processing*, vol. 1. (Rochester, New York, USA, 2002), pp. 397–400
13. S Chokchaitam, A non-separable two-dimensional LWT for an image compression and its theoretical analysis. Thammasat Internat J Sci Technol. **9**, 35–43 (2004)
14. YK Sun, A two-dimensional lifting scheme of integer wavelet transform for lossless image compression, in *International Conference on Image Processing*, vol. 1. (Singapore, 2004), pp. 497–500
15. V Chappelier, C Guillemot, Oriented wavelet transform for image compression and denoising. IEEE Trans Image Process. **15**(10), 2892–2903 (2006)
16. ON Gerek, AE Çetin, A 2D orientation-adaptive prediction filter in lifting structures for image coding. IEEE Trans Image Process. **15**, 106–111 (2006)
17. W Ding, F Wu, X Wu, S Li, H Li, Adaptive directional lifting-based wavelet transform for image coding. IEEE Trans Image Process. **10**(2), 416–427 (2007)

18. NV Boulgouris, MG Strintzis, Reversible multiresolution image coding based on adaptive lifting, in *IEEE International Conference on Image Processing*, vol. 3. (Kobe, Japan, 1999), pp. 546–550
19. RL Claypoole, G Davis, W Sweldens, RG Baraniuk, Nonlinear wavelet transforms for image coding. the 31st Asilomar Conference on Signals, Systems and Computers. **1**, 662–667 (1997)
20. A Gouze, M Antonini, M Barlaud, B Macq, Design of signal-adapted multidimensional lifting schemes for lossy coding. IEEE Trans Image Process. **13**(12), 1589–1603 (2004). doi:10.1109/TIP.2004.837556
21. J Solé, P Salembier, Generalized lifting prediction optimization applied to lossless image compression. IEEE Signal Process Lett. **14**(10), 695–698 (2007)
22. CL Chang, B Girod, Direction Adaptive discrete wavelet transform for image compression. IEEE Trans Image Process. **16**(5), 1289–1302 (2007)
23. JC Rolon, P Salembier, Generalized lifting for sparse image representation and coding, Picture Coding Symposium, (Lisbon, Portugal, 2007)
24. Y Liu, KN Ngan, Weighted adaptive lifting-based wavelet transform for image coding. IEEE Trans Image Process. **17**(4), 500–511 (2008)
25. S Mallat, Geometrical grouplets. Appl Comput Harmonic Anal. **26**(2), 161–180 (2009). doi:10.1016/j.acha.2008.03.004
26. JE Candes, LD Donoho, New tight frames of curvelets and optimal representations of objects with piecewise C^2 singularities. Commun Pure Appl Math. **57**(2), 219–266 (2004). doi:10.1002/cpa.10116
27. MN Do, M Vetterli, The contourlet transform: an efficient directional multiresolution image representation. IEEE Trans Image Process. **14**(12), 2091–2106 (2005)
28. V Chappelier, C Guillemot, S Marinkovic, Image coding with iterated contourlet and wavelet transforms, in *International Conference on Image Processing*, vol. 5. (Singapore, 2004), pp. 3157–3160
29. EL Pennec, S Mallat, Sparse geometric image representations with bandelets. IEEE Trans Image Process. **14**(4), 423–438 (2005)
30. NG Kingsbury, Complex wavelets for shift invariant analysis and filtering of signals. J Appl Comput Harmonic Anal. **10**, 234–253 (2001). doi:10.1006/acha.2000.0343
31. JE Fowler, JB Boettcher, B Pesquet-Popescu, Image coding using a complex dual-tree wavelet transform, the European Signal Processing Conference, (Poznan, Poland, 2007), pp. 994–998
32. JB Boettcher, JE Fowler, Video coding using a complex wavelet transform and set partitioning. IEEE Signal Process Lett. **14**(9), 633–636 (2007)
33. TH Reeves, NG Kingsbury, Overcomplete image coding using iterative projection-based noise shaping, in *International Conference on Image Processing*, vol. 3. (Rochester, NY, 2007), pp. 597–600
34. BK Natarajan, Sparse approximate solutions to linear systems. SIAM J Comput. **24**(2), 227–234 (1995). doi:10.1137/S0097539792240406
35. D Donoho, Compressed Sensing. IEEE Trans Inf Theory. **52**(4), 1289–1306 (2006)
36. M Kaaniche, JC Pesquet, A Benazza-Benyahia, B Pesquet-Popescu, Two-dimensional non separable adaptive lifting scheme for still and stereo image coding, IEEE International Conference on Acoustics, Speech and Signal Processing, (Dallas, Texas, USA, 2010), pp. 1298–1301
37. M Kaaniche, A Benazza-Benyahia, B Pesquet-Popescu, JC Pesquet, Non separable lifting scheme with adaptive update step for still and stereo image coding. Elsevier Signal Processing: Special issue on Advances in Multirate Filter Bank Structures and Multiscale Representations. **91**(12), 2767–2782 (2011)
38. B Pesquet-Popescu, Two-stage adaptive filter bank. first filling date 1999/07/27, official filling number 99401919.8, European patent number EP1119911. (1999)
39. SM LoPresto, K Ramchandran, MT Orchard, Image coding based on mixture modeling of wavelet coefficients and a fast estimation quantization framework, Data Compression Conference, (Snowbird, USA, 1997), pp. 221–230
40. S Mallat, A theory for multiresolution signal decomposition. IEEE Trans Pattern Anal Mach Intell. **11**, 674–693 (1989). doi:10.1109/34.192463
41. F Payan, M Antonini, An efficient bit allocation for compression normal meshes whith an error driven quantization. Comput Aid Geometr Design (Special Issue On Geometric Mesh Processing). **22**(5), 466–486 (2005)
42. MN Do, M Vetterli, Wavelet-based texture retrieval using generalized Gaussian density and Kullback-Leibler distance. IEEE Trans Image Process. **11**(2), 146–158 (2002). doi:10.1109/83.982822
43. H Gish, JN Pierce, Asymptotically efficient quantizing. IEEE Trans Inf Theory. **14**(5), 676–683 (1969)
44. T Petrisor, B Pesquet-Popescu, JC Pesquet, A Compressed Sensing Approach to Frame-Based Multiple Description Coding, in *IEEE International Conference on Acoustics, Speech and Signal Processing*, (Honolulu, HI, 2007), pp. 709–712
45. S Chen, D Doniho, M Saunders, Atomic decomposition by basis pursuit. SIAM Rev. **43**, 129–159 (2001). doi:10.1137/S003614450037906X
46. PL Combettes, JC Pesquet, Proximal splitting methods in signal processing, in *Fixed-Point Algorithms for Inverse Problems in Science and Engineering*, ed. by Bauschke HH, Burachik, R, Combettes, PL, Elser, V, Luke, DR, Wolkowicz, H (Springer-Verlag, New York, 2010)
47. M Afonso, JM Bioucas-Dias, MAT Figueiredo, An augmented Lagrangian approach to the constrained optimization formulation of imaging inverse problems. IEEE Trans Image Process. **20**(3), 681–695 (2011)
48. J Eckstein, DP Bertsekas, On the Douglas-Rachford splitting methods and the proximal point algorithm for maximal monotone operators. Math Programm. **55**, 293–318 (1992). doi:10.1007/BF01581204
49. JJ Moreau, Proximité et dualité dans un espace hilbertien. Bulletin de la Societé Mathématique de France. **93**, 273–288 (1965)
50. C Chaux, P Combettes, JC Pesquet, V Wajs, A variational formulation for frame based inverse problems. Inverse Probl. **23**(4), 1495–1518 (2007). doi:10.1088/0266-5611/23/4/008
51. PL Combettes, VR Wajs, Signal Recovery by Proximal Forward-Backward Splitting. Multiscale Model Simul. **4**(4), 1168–1200 (2005). doi:10.1137/050626090
52. JB Hiriart-Urruty, C Lemaréchal, Convex Analysis and Minimization Algorithms, (Springer-Verlag, Berlin, London, 1993)
53. RT Rockafellar, Convex Analysis, (Princeton University Press, N.J, 1970)**28**
54. LM Briceno-Arias, PL Combettes, JC Pesquet, N Pustelnik, Proximal algorithms for multicomponent image recovery problems. J Math Imag Vision. **41**(1-2), 3–22 (2010)
55. B Usevitch, Optimal bit allocation for biorthogonal wavelet coding, in *Data Compression Conference*, (Snowbird, USA, 1996), pp. 387–395
56. S Parrilli, M Cagnazzo, B Pesquet-Popescu, Distortion evaluation in transform domain for adaptive lifting schemes, in *International Workshop on Multimedia Signal Processing*, (Cairns, Queensland, Australia, 2008), pp. 200–205
57. DM Chandler, SS Hemami, VSNR: A wavelet-based Visual Signal-to-Noise Ratio for natural images. IEEE Trans Image Process. **16**(9), 2284–2298 (2007)
58. NV Boulgouris, MG Strintzis, A family of wavelet-based stereo image coders. IEEE Trans. Circuits and Syst. Video Technol. **12**(10), 898–903 (2002)
59. M Kaaniche, A Benazza-Benyahia, B Pesquet-Popescu, JC Pesquet, Vector lifting schemes for stereo image coding. IEEE Trans Image Process. **18**(11), 2463–2475 (2009)
60. T Frajka, K Zeger, Residual image coding for stereo image compression. Opt Eng. **42**, 182–189 (2003). doi:10.1117/1.1526492
61. MS Moellenhoff, MW Maier, Characteristics of disparity-compensated stereo image pair residuals. Signal Process: Image Commun. **14**, 49–55 (1998)
62. PL Combettes, JC Pesquet, A Douglas-Rachford splitting approach to nonsmooth convex variational signal recovery. IEEE J Sel Top Signal Process. **1**, 564–574 (2007)
63. M Kaaniche, JC Pesquet, A Benazza-Benyahia, B Pesquet-Popescu, Schémas de lifting adaptatifs via des critères parcimonieux, Colloque GRETSI, (Bordeaux, France, 2011), p. 4

doi:10.1186/1687-6180-2012-10
Cite this article as: Kaaniche *et al.*: **Adaptive lifting scheme with sparse criteria for image coding.** *EURASIP Journal on Advances in Signal Processing* 2012 **2012**:10.