Estimation of quantization noise for adaptive-prediction lifting schemes

Sara Parrilli #1, Marco Cagnazzo *2, Béatrice Pesquet-Popescu 3

DIET, Università degli Studi di Napoli Federico II
via Claudio 21, 80125 Napoli, ITALY
1 sara.parrilli@unina.it

* *Institut Télécom, Télécom ParisTech*
46 rue Barrault, F-75634 Paris Cedex 13, FRANCE
2 cagnazzo@telecom-paristech.fr
3 pesquet@telecom-paristech.fr

***Abstract*—The lifting scheme represents an easy way of implementing the wavelet transform and of constructing new content-adapted transforms. However, the adaptive version of 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 paper we focus on the problem of evaluating the reconstruction distortion (due to quantization noise) in the wavelet domain when a non-isometric adaptive-prediction lifting scheme is used. The problem arises since these transforms are nonlinear, and so common techniques for distortion evaluation cannot be used in this case. We circumvent the difficulty by computing an equivalent time-varying linear filter, for which it is possible to generalize the distortion computation technique. In addition to the theoretical formulation of the distortion estimation, in this paper we provide experimental results proving the reliability of this estimation, and the consequent improvement of RD performance, thanks to a more effective resource allocation which can be performed in the transform domain.**

I. Introduction

The lifting scheme (LS) was originally introduced by Sweldens [1] to design wavelets on complex geometrical surfaces, but at the same time it offers an efficient and simple way to understand and implement classic wavelet transforms.

The blocks composing the lifting scheme are illustrated in Fig.1. First, the input signal x is split into its even and odd samples, respectively called the approximation signal x_a and the detail signal x_d. Then, a prediction operator P is used in order to predict the current polyphase component (odd samples of x) from a linear combination of the other component samples (even samples). In practice, the prediction operator is chosen such that its output is a good 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 certain constraints, such as preserving the average of the input.

MMSP'09, October 5-7, 2009, Rio de Janeiro, Brazil.
978-1-4244-4464-9/09/$25.00 ©2009 IEEE.

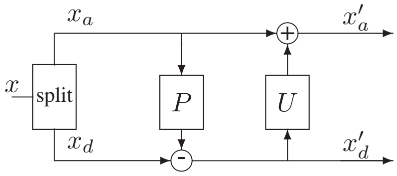

Fig. 1. Classical lifting scheme. Analysis.

The great flexibility of lifting scheme offers the possibility to replace linear filters by nonlinear ones. In particular, LS with adaptive update [2], [3], [4] or adaptive prediction [5], [6], [7] have been proposed in the literature, with the target of avoiding oversmoothing of important features such as borders, and at the same time of exploiting the correlation of homogeneous regions by using long filter on them: different filters are thus used in different parts of image, and 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, either explicitly like in the EBCOT algorithm [8], or implicitly, like in zero-tree based techniques like SPIHT [9] and EZW [10]. Therefore, in order to efficiently use the adaptive lifting scheme for image compression, we need to estimate correctly the distortion from the transform coefficients. Usevitch showed that the energy of an uncorrelated signal (such as the quantization noise is supposed to be) can be estimated for generic linear wavelet filter banks [11]. We extended this approach to adaptive update LS (AULS) in a previous paper [12], obtaining satisfying results in term of distortion estimation and of rate-distortion (RD) performance improvement. However, since the AULSs have strong constraints (to assure the perfect reconstruction without sending any side information), their absolute RD performances are not satisfying (several dB below popular non-adaptive 9/7 filters). For this reason, we consider in this work adaptive prediction LS (APLS), in particular those inspired by the paper by Claypoole *et al.* [6], because of their good RD performances. After a brief recall on APLS in Section II, we show how to estimate distortion in the transform

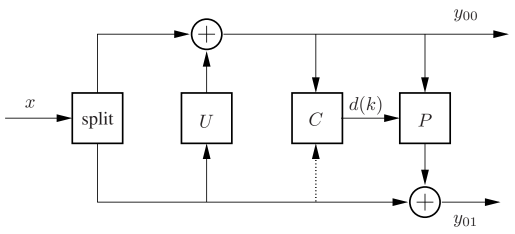

Fig. 2. Adaptive prediction lifting scheme. The result of the classification C determines the predictor.

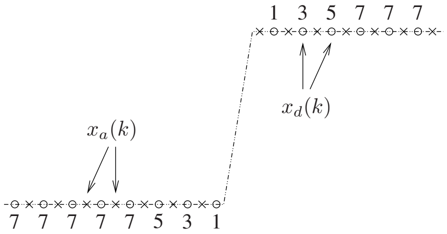

Fig. 3. Predictor selection at an ideal step edge. Numbers indicate the order of the predictors used. The closer to the edge, the lower the order of the predictor.

domain for these LS in Section III. Then, in Section IV we report experimental results proving both the reliability of the distortion estimation, and the RD performance improvement resulting from it. Finally we conclude the paper and outline future work.

II. Adaptive predict lifting schemes

Let us start by introducing some notation: x is the input signal; y_{ij} is a 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. 2.

The class of lifting scheme that we want to analyze is inspired by Claypoole *et al.* [6], because of its popularity and good performance. In this APLS the adaptivity involves only the prediction step, and in opposition to classic LS, the update step precedes the prediction one. The general APLS scheme is shown in Fig. 2. The basic idea is that higher order predictors are used where the image is locally smooth, and lower order ones near edges, to avoid prediction across discontinuities (see Fig. 3) as well as oversmoothing of boundaries. We call $d(\cdot)$ the decision (or classification) map, *i.e.* a signal indicating for each position k the predictor to use. 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 the side information $d(k)$, the update stage is applied first and the decision is based on the approximation signal only, that is, the dotted line in Fig. 2 does not operate.

According to the value of the decision map at position k, we use one out of $\mathcal{D}$ linear prediction filters, and so, since the decision map depends in 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 segments from highly variable parts. For example in AULS described in [4] the decision map is a threshold function of the local gradient seminorm, while in Claypoole *et al.* work the decision is based on the distance of current sample from the discontinuities, which are detected by a modified version of algorithm [13].

Once one has the decision map $d(\cdot)$, the APLS equations are the following:

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

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

while the synthesis is described by:

$$x(2k+1) = y_{01}(k) + \sum_{n\in\mathbb{Z}} \gamma_{d(k)}(n) y_{00}(k-n) \tag{3}$$

$$x(2k) = y_{00}(k) - \sum_{n\in\mathbb{Z}} \beta(n) x(2k+1-2n). \tag{4}$$

III. Computing distortion for APLS

When non-isometric linear analysis is used, Usevitch [11] showed that, for an uncorrelated signal (as the quantization noise is supposed to be) the energy D in the original domain is related to the energies D_{ij} of the wavelet subbands y_{ij} by the linear relation $D = \sum_{ij} w_{ij} D_{ij}$. The weight w_{ij} is computed as norm of the reconstruction polyphase matrix columns for subband y_{ij}.

However APLSs (as well as AULSs) are nonlinear systems, therefore no polyphase representation of them exist. However, once the decision map $d(k)$ is given (and in any case it is recovered from transform coefficients themselves), we can look at APLS described in Eq. (4) as a linear, time-varying system. The representation of APLSs (and AULSs) 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. In facts, the nonlinearity of the system depends only on d, and not on the whole input signal, thus the weights depend only on the decision map. In the following of this section, we will show how to compute the polyphase reconstruction matrices, which depend on the decision map. Then, we compute the analytical expression of weights w_{ij}. The energy D is finally obtained as weighted sum of the subband energies.

A. Computation of the polyphase matrices

First we consider the computation of the reconstruction polyphase matrices for the mono-dimensional case. For the decomposition level 0, the synthesis matrices associated to channels 0 and 1 are called respectively $\mathbf{G}_{00}$ and $\mathbf{G}_{01}$. According to this convention, the reconstruction of the original signal from its wavelet coefficients is described by the equation

$$\mathbf{x} = \mathbf{G}_{00}\mathbf{y}_{00} + \mathbf{G}_{01}\mathbf{y}_{01},$$

where the bold font refers to the vector form of the reconstructed signal and of the wavelet sub-bands. This equation can be interpreted as follows: the k-th row of $\mathbf{G}_{00}$ [of $\mathbf{G}_{01}$] represents the contribution (*i.e.* the coefficients in a linear combination) of the low-pass channel [of the high-pass channel] to the sample $x(k)$. It is then useful to distinguish the case where k is odd from the case where k is even. We report here for ease of reading the equation (3):

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

It follows that the odd rows of $\mathbf{G}_{00}$ and $\mathbf{G}_{01}$ are:

$$\mathbf{G}_{00}(2k+1,n) = \gamma_{d(k)}(k-n) \tag{5}$$
$$\mathbf{G}_{01}(2k+1,n) = \delta_{k-n}, \tag{6}$$

As for the even rows, we develop the expression of $x(2k)$ from Eq. (4), and with a little computation we find that:

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

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

$$\mathbf{G}_{00}(2k,n) = \delta_{n-k} - \Sigma_m \beta(k-n-m)\gamma_{d(m+n)}(m) \tag{7}$$
$$\mathbf{G}_{01}(2k,n) = -\beta(k-n). \tag{8}$$

We note that the expressions for $\mathbf{G}_{00}$ and $\mathbf{G}_{01}$ rows are more complex than those one can obtain for AULS [12]. 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 on different values of d (see Eq. 7). This aspect does not prevent us to compute the equivalent polyphase matrix but impedes the possibility to find a simple interpretation of the matrices structure. Equations from (5) to (8) allow us to compute the polyphase matrices for any decision map and any APLS. However, from now on, we consider the particular class of APLS used by Claypoole *et al.* since it is simple and very efficient. In this case $\beta(n) = \delta_n$, so we have:

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

The structure of the reconstruction polyphase matrices for the low-pass channel in this simplified hypothesis, is summarized in Fig. 4, while for the high-pass channel the structure is even simpler, since the n-th column has just -1 in position $2n$ and 1 in position $2n+1$.

B. Weight computation for one level of decomposition

Once polyphase matrices has been computed, we find the expression for corrective weights simply by following the approach of Usevitch. It is interesting to develop the formulas in the hypothesis of a one-dimensional signal and one level of decomposition, because they provide a simple and intuitive relation between weights in the adaptive and in the non-adaptive case. In our previous work [12] we showed that, in the case of AULS, the weight of each subband is a linear combination of weights associated to each of the $\mathcal{D}$ filters used in the adaptive scheme. The coefficients of the linear combination are the relative frequency of the filters in the decision map.

This simple yet interesting result holds for the APLS as well, and the proof is formally identical to the one of the AULS case. We briefly show it here. Let us introduce the matrices $\mathbf{G}^{(h)}$, where $h \in \{0, 1, \ldots, \mathcal{D}-1\}$. This is the polyphase synthesis matrix associated to the filter corresponding to the value h of the decision map. One can see it as a non-adaptive case, where the h-th filter is always used. In [11] Usevitch showed that for a generic (non-adaptive) filter with an associated synthesis matrix $\mathbf{G}^{(h)}$, the weight is:

$$w^{(h)} = \frac{2}{N} \sum_{n,m} \mathbf{G}^{(h)}(n,m)^2.$$

We develop it as:

$$w^{(h)} = \frac{2}{N} \sum_n \left[\sum_m \mathbf{G}^{(h)}(2n,m)^2 + \sum_m \mathbf{G}^{(h)}(2n+1,m)^2 \right] \tag{9}$$

Now, in a polyphase matrix we have $\sum_m \mathbf{G}^{(h)}(2n,m)^2 = \sum_m \mathbf{G}^{(h)}(0,m)^2$ since all even rows have the same elements of row 0. Likewise, we have $\sum_m \mathbf{G}^{(h)}(2n+1,m)^2 = \sum_m \mathbf{G}^{(h)}(1,m)^2$. In conclusion, the weight $w^{(h)}$ is expressed as:

$$w^{(h)} = \sum_m \mathbf{G}^{(h)}(0,m)^2 + \sum_m \mathbf{G}^{(h)}(1,m)^2. \tag{10}$$

Let us now consider the adaptive case, where we call w the weight for the subband whose reconstruction matrix is $\mathbf{G}$. Then, Eq. (9) is written as:

$$w = \frac{2}{N} \sum_n \left[\sum_m \mathbf{G}(2n,m)^2 + \sum_m \mathbf{G}(2n+1,m)^2 \right]$$

However, the $2n$-th and $2n+1$-th rows depend only on the decision map in n, and in particular coincide with the rows of the non-adaptive matrix $\mathbf{G}^{(d(n))}$. So we have:

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

where we used Eq. (10). If we denote by N_h the number of occurrences of the value h in the decision map, we can write:

$$w = \sum_{h=0}^{\mathcal{D}-1} \frac{2N_h}{N} w^{(h)} = \sum_{h=0}^{\mathcal{D}-1} \rho_h w^{(h)}, \tag{11}$$

where $\rho_h = \frac{2N_h}{N}$ is the relative frequency of filter h in the decision map for the current subband. Unfortunately, this simple interpretation of subband weights does not hold any longer when more than one decomposition level is performed.

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

Fig. 4. Structure of the matrix $\mathbf{G}_{00}$. APLS. Highlighted cell is in position $(2k, k)$.

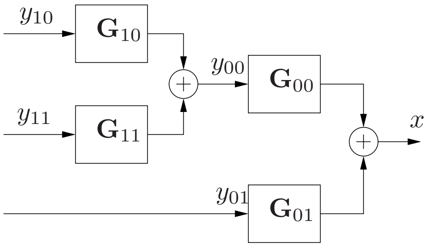

Fig. 5. Equivalent filter bank for synthesis ALS, two decomposition levels.

In fact in this case the polyphase reconstruction matrices have a more complex structure, as we show in the following.

In the case of multiple decomposition levels, we denote with $\mathbf{G}_{ij}$ the reconstruction matrix for the i-th decomposition level and for the j-th channel (see Fig. 5). For example, in the case of diadic decomposition, the first level low-pass subband is obtained from the second level subbands as:

$$\mathbf{y}_{00} = \mathbf{G}_{10}\mathbf{y}_{10} + \mathbf{G}_{11}\mathbf{y}_{11} \tag{12}$$

The matrix $\mathbf{G}_{ij}$ is obtained just as $\mathbf{G}_{0j}$, except that we have to use the appropriate decision map at level i. Now we observe that the reconstructed signal can be expressed using recursively Eq. (12) (with straightforward modification in the case the decomposition was not diadic). The reconstructed signal is of course a linear combination of the transform subbands, each weighted with a suitable matrix $\mathbf{A}_{ij}$:

$$\mathbf{x} = \sum_{ij} \mathbf{A}_{ij}\mathbf{y}_{ij}, \tag{13}$$

where, in the diadic case $(i, j) \in \{(0, 1), (1, 1), (2, 1), \ldots, (N-1, 1), (N-1, 0)\}$. The weight w_{ij} it just the average norm of the columns of $\mathbf{A}_{ij}$. These matrices are computed just as product of one-level matrices $\mathbf{G}_{ij}$ corresponding to the filters between the subband y_{ij} and the reconstructed signal x. For the diadic case we have:

$$\mathbf{A}_{01} = \mathbf{G}_{01}, \tag{14}$$

$$\mathbf{A}_{i1} = \mathbf{G}_{i1}\prod_{\ell=0}^{i-1}\mathbf{G}_{\ell 0}, \qquad \forall i \in \{1, ..., I-1\} \tag{15}$$

$$\mathbf{A}_{I-1,0} = \prod_{\ell=0}^{I-1}\mathbf{G}_{\ell 0}. \tag{16}$$

These equations are easily extended to non-diadic decompositions. Unfortunately, the simple interpretation obtained for the one-level decomposition does not hold anymore when more levels are used, because of the matrix product in Eq. (15) or Eq. (16). Moreover, there is no simple, closed form expression for $\mathbf{A}_{ij}$ matrices, even though they can always be computed by Eqs. (14)-(16), once we have the decision maps for all levels.

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

Fig. 6. The bi-dimensional signal x represented via four channels; x has $2M$ columns, and $k = Mn + m$.

C. Extension to multi-dimensional case

The extension of the proposed technique to multi-dimensional adaptive LS is straightforward once one recognizes that a linear time-varying relationship between subbands and reconstructed signal can be established in this case as well. It is intuitive that one can find an equation in the same form of Eq. (13). In the remainder of this section we show how to compute the matrices $\mathbf{A}_{ij}$ in the 2D case.

All that we need at this end is to observe that the 2D analysis filter bank can be seen as a 1D filter bank with four channels instead of two. The relationship between the four channels and the subband coefficient positions is shown in Fig. 6. With this notation, we can write the equation of the bi-dimensional APLS shown in Fig.7:

$$\begin{aligned} y_{00}(k) = {} & x_0(k) + \sum_{n\in\mathbb{Z}} \beta_1(n)x_1(k-n) \\ & + \sum_{n\in\mathbb{Z}} \beta_2(n)x_2(k-n) + \sum_{n\in\mathbb{Z}} \beta_3(n)x_3(k-n), \end{aligned} \tag{17}$$

$$\begin{aligned} y_{01}(k) = {} & x_1(k) - \sum_{n\in\mathbb{Z}} \gamma_{1,0,d(k)}(n)y_{00}(k-n) \\ & + \sum_{n\in\mathbb{Z}} \sigma_{13}(n)x_3(k-n), \end{aligned} \tag{18}$$

$$\begin{aligned} y_{02}(k) = {} & x_2(k) - \sum_{n\in\mathbb{Z}} \gamma_{2,0,d(k)}(n)y_{00}(k-n) + \\ & - \sum_{n\in\mathbb{Z}} \gamma_{2,1,d(k)}(n)y_{01}(k-n) + \sum_{n\in\mathbb{Z}} \sigma_{23}(n)x_3(k-n), \end{aligned} \tag{19}$$

$$\begin{aligned} y_{03}(k) = {} & x_3(k) - \sum_{n\in\mathbb{Z}} \gamma_{3,0,d(k)}(n)y_{00}(k-n) + \\ & - \sum_{n\in\mathbb{Z}} \gamma_{3,1,d(k)}(n)y_{01}(k-n) - \sum_{n\in\mathbb{Z}} \gamma_{3,2,d(k)}(n)y_{02}(k-n). \end{aligned} \tag{20}$$

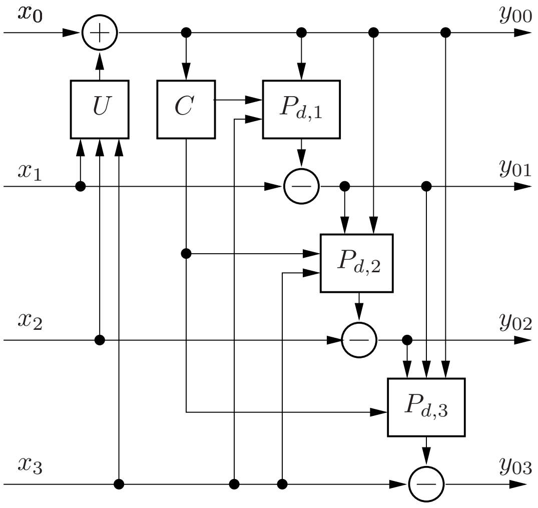

Fig. 7. Claypoole inspired 2D APLS.

We observe that these equation actually describe a non-separable 2D APLS where the $\gamma_{i,j}$ (with $j < i$) represents the contribution to y_{0i} of other output signal y_{0j}; while $\sigma_{i,j}$ (with $j > i$) represents the contribution to y_{0i} of the input signal x_j. This LS is slightly different from the one described in [6]. In particular, the original LS is separable and switches between four 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 our non-separable case, we 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.

Equations from (17) to (20) correspond to (1) and (2); in this case as well they are simplified because we use a very simple update step:

$$y_{00}(k) = \frac{1}{4}\left[x_0(k) + x_1(k) + x_2(k) + x_3(k)\right]$$

Once derived the expression of 2D APLS, the synthesis equations are easily obtained from the analysis ones. Then, the equivalent polyphase matrix for reconstruction, $\mathbf{G}_{0j}$ is 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 for the 1D case, and so we do not report the computation of the reconstruction matrices for the sake of simplicity. However, we note that in this case the $\mathbf{G}_{0j}$ matrices are composed of blocks of 4 rows that depend on the k-th value of the decision map. As in the 1-D case, the weight for one level of analysis is obtained as linear combination of non-adapted weights, using relative frequencies as coefficients. The extension to multiple decomposition levels is achieved by computing the average norms of matrices $\mathbf{A}_{ij}$.

Levels	1	2	3	4	5
No weights	76.41%	87.18%	91.31%	93.38%	94.59%
Weighted	0.14%	0.15%	0.35%	0.25%	0.83%

TABLE I
RELATIVE ERROR OF THE ENERGY ESTIMATION.

IV. EXPERIMENTAL RESULTS

A. Distortion evaluation in transformed domain

The first aim of our experimental analysis is to verify the correctness of the weights computed. At this end we reproduce exactly the hypothesis of the Usevitch paper [11] where it is stated that the energy of an uncorrelated signal can be computed as linear combination of its subband energies, according to the formula $D = \sum_{ij} w_{ij} D_{ij}$. Therefore we use a white Gaussian source to produce fictive wavelet coefficients of the noise signal and we estimate the energy of this noise signal in the transform domain. We consider two cases: in the first one we use the proposed weights; in the second one we set $w_{ij} = 1$ for all subbands, *i.e.* we estimate the distortion in the wavelet domain without using weights. This would be exact if an orthogonal LS had been employed. Finally, the two estimations are compared to the actual distortion, obtained as energy of the noise signal after the inverse transform. The per cent relative errors of the two estimations are reported in Tab. I. We observe that when the APLS is used, a naïve energy estimation in the transform domain as sum of subband energies would be very unreliable. This affects the effectiveness of any resource allocation algorithm and finally, can undermine the compression capabilities of any coding algorithm using APLS. However, we also note that if the weights are employed, the energy estimation becomes much more accurate: we expect therefore a better rate allocation, resulting in improved RD performances for compression. In summary, results in Tab. I tell us that adaptive filter are far from being orthogonal and that weighting can help in this case. These intuitions are confirmed by the results shown in the next section.

B. Bit-rate allocation

We consider a very simple compression scheme: the input image is transformed with our adaptive prediction lifting scheme (five decomposition levels) and quantized with a dead-zone quantizer. We use an optimal bit-rate allocation algorithm [14] to choose the quantization step for each subband. We impose that the spatial domain distortion (estimated using our weights) is minimized for the assigned target rate. In order to evaluate the quality of compressed images, the inverse transform is applied on quantized coefficients, and the resulting distortion is computed. In addition, we perform the same procedure using unitary weights for all subbands. Finally, we compare the rate/distortion curves for the two schemes.

We considered five different images: Lena, House, Peppers, Cameraman, and Barbara. The PSNR improvements compared with the case when no weight is used are reported in Tab. II at 0.25, at 0.5 and at 1.0 bpp. The weights allow a remarkable gain for APLS, even if it is smaller than that what we found

	0.25 bpp	0.5 bpp B	1 bpp
Lena	0.9dB	0.6dB	0.5dB
House	0.3dB	0.4dB	0.5dB
Peppers	0.8dB	0.4dB	0.4dB
Cameraman	0.2dB	0.2dB	0.4dB
Barbara	0.7dB	0.6dB	1.0dB

TABLE II
PSNR IMPROVEMENTS FOR CLAYPOOLE *et al.* LIFTING SCHEME COMPARED WITH NO WEIGHTS.

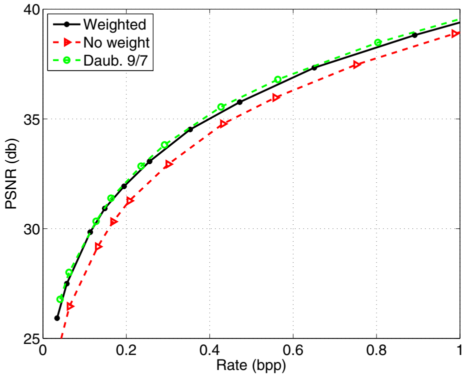

Fig. 8. RD curves for Lena with and without weights.

for AULS [12]. The RD curves for Lena are reported in Fig. 8, where we added even the curve relative to the classic 9/7 wavelet. We notice that the using weights makes the performance of the APLS very close to those of the 9/7 wavelet, while without weights they are inferior (as already noted in [6]). However, the main interest of the proposed technique is that it improves the RD performances of any non-orthogonal adaptive LS. In this way we take advantage of other beneficial characteristics of ALS, such as a better perceptive quality of decoded images as already underlined in [6] and shown in Fig. 9. For example, the APLS with weights gains up to 0.5 dB in WPSNR [15] over the 9/7 wavelet and up to 1.5 dB over the non-weighted APLS. However, a less favorable comparison can be observed w.r.t. non-standard filters such those presented in [16].

In conclusion, this paper focuses on the problem of distortion estimation in the transform domain, and it extends the approach proposed in our previous work [12] to a different class of adaptive lifting scheme, characterized by better RD performance. Our study shows on one hand that the original lifting scheme is quite far from being orthogonal, but also that the proposed technique allows a reliable estimation of the distortion, and, as a consequence, a better resource allocation for compression. We also report improved visual aspect of decoded images. Future work will focus on implementation of the proposed distortion estimation technique within more effective coding schemes, like EBCOT [8].

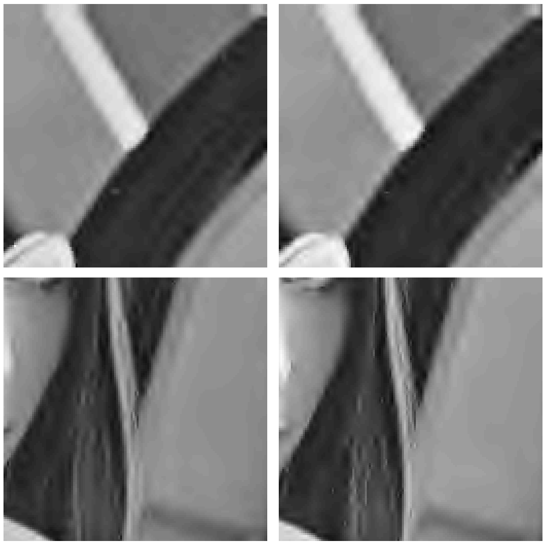

Fig. 9. Lena details decoded at 0.25 bpp. APLS (on the left) shows less artifacts than non-adaptive 9/7 wavelet (right): in the first row the mirror border is more regular; in the second less spurious structures appear in the hairs and near the nose.

REFERENCES

[1] W. Sweldens, “The lifting scheme: A custom-design construction of biorthogonal wavelets,” *Appl. Comput. Harmon. Anal.*, vol. 3, no. 2, pp. 186–200, 1996.

[2] 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.

[3] 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 Lett.*, vol. 9, pp. 329–332, Oct. 2002.

[4] ——, “Gradient-driven update lifting for adaptive wavelets,” *Signal Proc.: Image Comm. (Elsevier Science)*, vol. 20, no. 9-10, pp. 813–831, Oct.-Nov. 2005.

[5] O. N. Gerek and A. E. Çetin, “Adaptive polyphase subband decomposition structures for image compression,” *IEEE Trans. Image Processing*, vol. 9, no. 10, pp. 1649–1659, Oct. 2000.

[6] R. L. Claypoole, G. M. Davis, W. Sweldens, and R. G. Baraniuk, “Nonlinear wavelet transforms for image coding via lifting,” *IEEE Trans. Image Processing*, vol. 12, no. 12, pp. 1449–1459, Dec. 2003.

[7] N. Mehrseresht and D. Taubman, “Spatially continuous orientation adaptive discrete packet wavelet decomposition for image compression,” in *Proceed. of IEEE Intern. Conf. Image Proc.*, Atlanta, GA (USA), Oct. 2006, pp. 1593–1596.

[8] D. Taubman, “High performance scalable image compression with EBCOT,” *IEEE Trans. Image Processing*, vol. 9, no. 7, pp. 1158–1170, Jul. 2000.

[9] A. Said and W. Pearlman, “A new, fast and efficient image codec based on set partitioning in hierarchical trees,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 6, no. 3, pp. 243–250, Jun. 1996.

[10] J. M. Shapiro, “Embedded image coding using zerotrees of wavelets coefficients,” *IEEE Trans. Signal Processing*, vol. 41, pp. 3445–3462, Dec. 1993.

[11] B. Usevitch, “Optimal bit allocation for biorthogonal wavelet coding,” in *Proceed. of Data Comp. Conf*, Snowbird, USA, Mar. 1996, pp. 387–395.

[12] 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, 2008.

[13] K. Jensen and D. Anastassiou, “Subpixel edge localization and the interpolation of still images,” *IEEE Trans. Image Processing*, vol. 4, no. 3, pp. 288–295, Mar. 1995.

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

[15] M. Miyahara, K. Kotani, and V. Algazi, “Objective picture quality scale (pqs) for image coding,” *IEEE Trans. Commun.*, vol. 46, no. 9, pp. 1215–1226, Sep 1998.

[16] J. Villasenor, B. Belzer, and J. Liao, “Wavelet filter evaluation for image compression,” *IEEE Trans. Image Processing*, vol. 4, no. 8, pp. 1053–1060, Aug. 1995.