AN ADAPTIVE UPDATE LIFTING SCHEME WITH PERFECT RECONSTRUCTION

Gemma Piella and Henk J.A.M. Heijmans

Center for Mathematics and Computer Science
Kruislaan 413, 1098 SJ Amsterdam
The Netherlands

ABSTRACT

The lifting scheme provides a general and flexible tool for the construction of wavelet decompositions and perfect reconstruction filter banks. In this paper we propose an adaptive version of this scheme which has the intriguing property that it allows perfect reconstruction without any overhead cost. We restrict ourselves to the update lifting step which affects the approximation signal only. The update lifting filter is assumed to depend pointwise on the norm of the associated gradient vector, in such a way that a large gradient induces a weak update filter. Thus, sharp transitions in a signal (e.g., edges in an image) will not be smoothed to the same extent as regions which are more homogeneous.

1. INTRODUCTION

The past fifteen years have shown a steadily growing interest in wavelet transforms and perfect reconstruction filter banks (PRFB) for application in numerous areas in signal and image processing. Today we know various construction techniques for PRFB's [1]. One which is particularly interesting, not only because of its generality and flexibility, but also since it allows efficient implementations, is the *lifting scheme* due to Sweldens [2]. In the filter bank literature, this scheme is also sometimes referred to as *ladder structure* [3].

A severe limitation of most existing PRFB's is that the filter structure is fixed over the entire signal. In many applications it is desirable to have a filter bank that somehow determines how to shape itself for the data it analyzes. The idea of locally adapting the filters is not new and, recently, it has been investigated by several independent groups [4, 5, 6, 7]. In this paper we aim to provide a generalized lifting scheme for building adaptive PRFB's which (and this is crucial) do not require any additional bookkeeping to enable inversion. To the best of our knowledge, no such systematic construction yet exists.

In [8] we present a more comprehensive discussion on adaptive update lifting. We refer also to that paper for technical proofs of the results presented here, as well as for some additional experimental results.

2. LIFTING SCHEME

In this section we briefly recall the basic idea behind the original update lifting scheme illustrated in Fig. 1.

This scheme can be applied after making a decomposition of the original signal x_0 into two other signals, the *approximation signal* $x \in \mathcal{X}$, and the *detail signal* $y \in \mathcal{Y}$. Such a decomposition may be the result of a particular wavelet transform, but also of a subdivision of the signal's domain into two disjoint subsets, in general the even and odd samples. This latter decomposition is sometimes referred to as the *lazy wavelet transform*. We distinguish two basic lifting steps, prediction and update lifting. A general lifting scheme may comprise any sequence of basic lifting steps being alternatively of prediction and update type. In this paper we restrict ourselves to the update lifting step. The idea behind update lifting is to apply an update operator U to y and to combine it with x in order to get a "better" approximation signal $x' = x \oplus U(y)$. Here, the specific meaning of the adjective "better" depends on the particular application. For example, we may want to apply update lifting to obtain more vanishing moments for the corresponding low-pass wavelet filter.

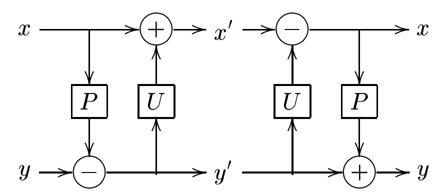

Fig. 1. Classical lifting scheme.

3. ADAPTIVE LIFTING

Whereas in the standard lifting scheme described in the previous section, the operator U and the addition $\oplus$ are fixed, in the adaptive case the choice of these operations may be governed by the local properties of the approximation signal x and the detail signal y. In fact, in our approach this choice will be triggered by a so-called *decision map* $D : \mathcal{X} \times \mathcal{Y} \to \mathcal{D}^{\mathbb{Z}}$, where $\mathcal{D}$ is the *decision set*. In Fig. 2 we give a schematic representation of our lifting scheme.

For every possible outcome $d \in \mathcal{D}$ of the decision map, we have a different update operator U_d and addition $\oplus_d$. Thus, the analysis step of our adaptive update lifting scheme looks as follows:

$$x'(n) = x(n) \oplus_{d_n} U_{d_n}(y)(n)\,, \tag{1}$$

where $d_n = D(x, y)(n)$ is the decision at location n.

We denote the subtraction which inverts $\oplus_d$ by $\ominus_d$. At synthesis we can invert (1) by

$$x(n) = x'(n) \ominus_{d_n} U_{d_n}(y)(n)\,. \tag{2}$$

The research of the first author is sponsored by the Dutch Technology Foundation STW.

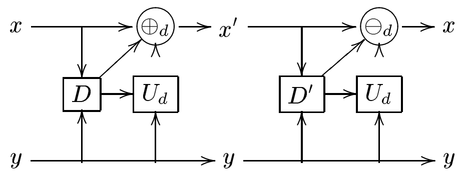

Fig. 2. Adaptive update lifting scheme.

Since $d_n = D(x, y)(n)$, this expression depends on the original signal x. However, at synthesis, we do not know x but 'only' its update x'. In general, this prohibits the computation of d_n and in such cases perfect reconstruction is out of reach. But, as we show below, there exist a number of situations in which it is still possible to recover d from a posteriori decision map D' which uses x' and y as input. Obviously, D' needs to satisfy

$$D'(x', y) = D(x, y)\,, \tag{3}$$

for $(x, y) \in \mathcal{X} \times \mathcal{Y}$ and with x' given by (1). We refer to this property as the *decision conservation condition*.

As we said before the decision criterion depends on local properties of the approximation and detail signal. Throughout most of this paper we restrict ourselves to the case that d depends on s, the l^1-norm of the gradient. In Section 7, however, where we consider quincunx sampling for 2D images, s will be the l^2-norm for technical reasons. The assumption that the decision map is entirely based on local gradient information enables us to build decomposition schemes which behave different in the proximity of edges than in more or less homogeneous regions.

Define the gradient vector at location n as $(v(n), w(n)) = (x(n) - y(n-1), y(n) - x(n))$. Our previous assumptions yield that

$$D(x, y)(n) = d(s(n)) \quad \text{where } s(n) = |v(n)| + |w(n)|\,, \tag{4}$$

and d is a function which maps different gradient values to possibly different decisions, i.e., $d : \mathbb{R}_+ \to \mathcal{D}$. In the next two sections we assume that $d(s) = s$, but for convenience we will write d rather than s. Later on we consider the case where d can take different values depending on a straightforward thresholding criterion.

4. UPDATE FILTER

We assume that the update operator U_d is a 2-tap filter and that $\oplus_d$ is the standard addition followed by some scale factor. Now, the analysis step in (1) is of the form

$$x'(n) = \alpha_{d_n} x(n) + \beta_{d_n} y(n-1) + \gamma_{d_n} y(n)\,, \tag{5}$$

and the synthesis step (presumed that d_n is known and $\alpha_d \neq 0$) is given by

$$x(n) = \frac{1}{\alpha_{d_n}} \big(x'(n) - \beta_{d_n} y(n-1) - \gamma_{d_n} y(n) \big)\,. \tag{6}$$

4.1. Lemma. *In order to have perfect reconstruction it is necessary that $\alpha_d + \beta_d + \gamma_d$ is constant for all $d \in \mathcal{D}$.*

Henceforth we assume that

$$\alpha_d + \beta_d + \gamma_d = 1\,, \text{ for all } d \in \mathcal{D}\,. \tag{7}$$

In general, the condition in (7) is not sufficient. But we have established the following result.

4.2. Proposition. *Perfect reconstruction is guaranteed in each of the following two cases:*

(a) $\alpha_d > 0$ *for all* $d \geq 0$ *and* β_d, γ_d *are non-increasing with respect to* d.

(b) $\alpha_d < 0$ *for all* $d \geq 0$ *and* β_d, γ_d *are non-decreasing with respect to* d.

So far, we have only derived conditions which guarantee that perfect reconstruction is possible, but we have not yet given the corresponding algorithm. In the next section, we propose a general reconstruction algorithm.

5. RECONSTRUCTION ALGORITHM

Let us simplify the notation by replacing the vector $(x(n), y(n-1), y(n))$ by (x, y, z). For each index n, we can retrieve x as follows:

1. Compute $\Delta = |z - y|$.
2. Compute coefficients $\alpha_\Delta, \beta_\Delta, \gamma_\Delta$.
3. Compute the lower and upper limits Y and Z:

$$Y = \min(y + \gamma_\Delta(z - y), z - \beta_\Delta(z - y))$$
$$Z = \max(y + \gamma_\Delta(z - y), z - \beta_\Delta(z - y))$$

4. If $x' \in [Y, Z]$ put

$$\gamma = \gamma_\Delta \text{ and } \beta = \beta_\Delta$$

else

- compute d by solving

$$d \cdot \alpha_d = |y + z - 2x' + (\beta_d - \gamma_d((y - z)|$$

- put

$$\gamma = \gamma_d \text{ and } \beta = \beta_d\,.$$

5. Compute x from

$$x = \frac{x' - \beta y - \gamma z}{1 - \beta - \gamma}.$$

Note that the reconstruction algorithm implicitely computes the decision map at every location.

6. THRESHOLD CRITERION

For some special cases we can have a much simpler reconstruction algorithm. In particular, we now concentrate on the case where the decision map is based on a simple threshold criterion. To be precise, we assume that $D = \{0, 1\}$ and that the function d in (4) has the form

$$d(s) = \begin{cases} 1, & \text{if } s > T \\ 0, & \text{if } s \leq T\,, \end{cases} \tag{8}$$

where T is the gradient threshold. Instead of (8) we sometimes use the shorthand notation

$$d(s) = [s > T] \tag{9}$$

where $[P]$ returns 1 if the predicate P is true and 0 if it is false. Note that now it is not necessary to recover s at the synthesis step but rather if s was smaller or equal than the threshold T, or if s was larger than T. In fact, we are interested in finding a second threshold value T' such that the decision at synthesis $d'(s')$, which must equal the decision at analysis, i.e.,

$$d'(s') = d(s)\,,$$

can also be written as a threshold criterion, i.e.,

$$d'(s') = [s' > T']\,, \tag{10}$$

where s' is the l^1-norm of the gradient at synthesis:

$$s' = |x' - y| + |z - x'|\,.$$

If we restrict ourselves to $T' = T$, we can state the following result

6.1. Proposition. *We have $d'(s') = d(s)$ with $T' = T$ if and only if*

$$0 \le \beta_0, \gamma_0 \le 1 \;\; \textit{and either} \;\; \beta_1, \gamma_1 \le 0 \;\textit{or}\; \beta_1, \gamma_1 \ge 1\,.$$

If the conditions of Proposition 6.1 hold, then the reconstruction algorithm consists of the following steps:

1. Compute $s' = |x' - y| + |z - x'|$.
2. Let $d = [s' > T]$ and put

$$\gamma = \gamma_d \;\text{ and }\; \beta = \beta_d\,.$$

3. Compute x from

$$x = \frac{x' - \beta y - \gamma z}{1 - \beta - \gamma}.$$

Additional results can be found in [8].

7. QUINCUNX LIFTING FOR IMAGES

In this section we propose a 2-dimensional quincunx version of the threshold-based adaptive scheme. Now, a sample $x(n)$ is updated with its four neighbors $y(n_i)$,where n_i, $i = 1, \ldots, 4$, are the indexes of the north, south, east and west nearest samples. For simplicity, we assume they equally contribute to the update, and hence the analysis step looks as follows:

$$x'(n) = \alpha_{d_n} x(n) + \frac{(1 - \alpha_{d_n})}{4} \sum_{i=1}^{4} y(n_i). \tag{11}$$

By means of example, take $\alpha_0 = \frac{1}{5}$ and $\alpha_1 = 1$, and consider s to be the l^2-norm of the gradient, that is,

$$s = (\sum_{i=1}^{4} |x(n) - y(n_i)|^2)^{1/2}.$$

The reason for choosing the l^2-norm is a technical one. It guarantees that the gradient s' after the update lifting step satisfies $s' \le s$ if $s \le T$, i.e., $d = 0$.

8. RESULTS AND DISCUSSION

We first apply the threshold-based adaptive scheme to the signal in Fig. 3(a). We choose $\beta_0 = \gamma_0 = \frac{1}{3}$ and $\beta_1 = \gamma_1 = 0$. Thus the resulting low-pass filters are the average filter for $d = 0$, and the identity for $d = 1$. The update step is followed by a fixed prediction step of the form $y'(n) = y(n) - \frac{1}{2}(x'(n) + x'(n+1))$. The approximation and detail signals, x' and y', are depicted in Fig. 3(c)-(d), where we have taken a threshold $T = 0.2$. The corresponding decision map is shown in Fig. 3(b). For comparison we show in Fig. 3(e)-(h) the approximation and detail signals obtained for both non-adaptive cases corresponding with $d = 0$ and $d = 1$. Fig. 3(e)-(f) correspond with $d = 0$, that is, an average filter, and Fig. 3(g)-(h) correspond with $d = 1$, i.e., the lazy wavelet. One can see that the adaptive scheme performs as an average filter except for those locations where the gradient is large (i.e., where it exceeds the threshold T); in these cases it 'recognizes' that there is an edge and does not apply any smoothing. As a result, the detail signal is not double-peaked in the adaptive case depicted in Fig. 3(d), unlike the detail signal in Fig. 3(f) where averaging takes place everywhere. By varying the threshold T, the resulting system can be tuned to one or another behavior.

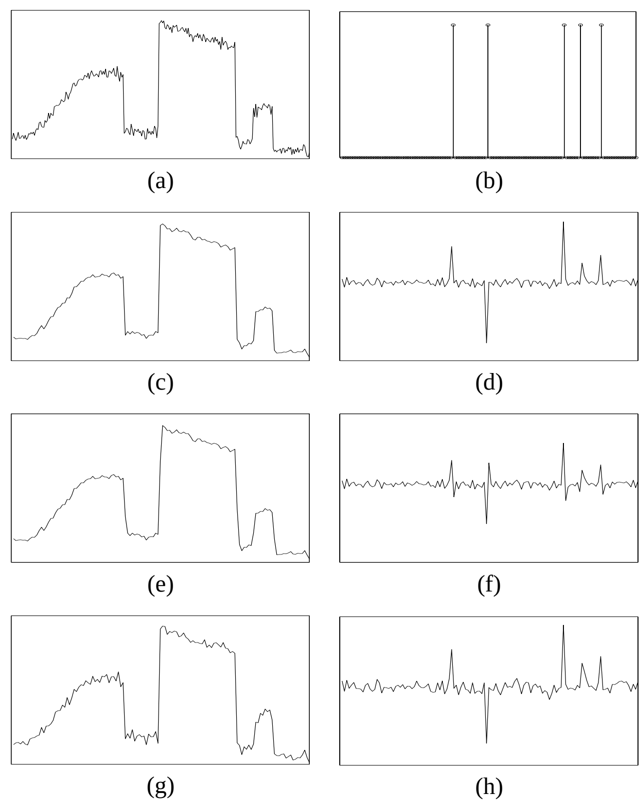

Fig. 3. (a) Original signal; (b) Decision map for T=0.2; (c) Approximation for T=0.2; (d) Detail for T=0.2; (e) Approximation for fixed d=0 (average); (f) Detail for fixed d=0 (average); (g) Approximation for fixed d=1 (lazy); (h) Detail for fixed d=1 (lazy).

In Fig. 4 we show some experiments for quincunx lifting of images as proposed in Section 7. The original image is depicted in Fig. 4(a). The decisions maps in Fig. 4(b), (g), and (h), show

how the amount of smoothing can be tuned by changing the threshold T. The approximation and detail signals for $T = 50$ are shown in Fig. 4(c)-(d). We can compare them with the ones obtained for the fixed averaging filter in Fig. 4(e)-(f). The figure shows that the edges are better preserved in the adaptive case.

In this preliminary investigation, we have introduced a new framework for an adaptive lifting scheme. Our scheme is non-redundant in the sense that no bookkeeping is required. Obviously, this scheme needs further investigation to see if the idea is fruitful. Preliminary experiments seem to indicate that these adaptive decompositions yield detail signals with lower entropies than in the non-adaptive case. The results obtained so far suggest that our non-redundant adaptive schemes may be useful for compression, and we intend to investigate this more thoroughly in the future. In the literature there exist several wavelet decomposition schemes which, in various ways, try to take into account discontinuities (e.g., singularities in signals, edges in images) [5, 6, 9]. It is important to understand the similarities and differences between all these techniques. Finally, we want to examine cases where the linear update (and prediction) filters can be replaced by morphological ones [10].

9. REFERENCES

[1] G. Strang and T. Nguyen, *Wavelets and Filter Banks*, Wellesley-Cambridge Press, 1996.

[2] W. Sweldens, "The lifting scheme: A new philosophy in biorthogonal wavelet constructions," in *Wavelet Applications in Signal and Image Processing III*. SPIE 2569, 1995, pp. 68–79.

[3] F. A. M. L. Bruekers and A. W. M. van den Enden, "New networks for perfect inversion and perfect reconstruction," *IEEE Journal on Selected Areas in Communications*, vol. 10, no. 1, pp. 130–137, 1992.

[4] W. Trappe and K. J. R. Liu, "Adaptivity in the lifting scheme," in *33th Conference on Information Sciences and Systems*, Baltimore, March 1999, pp. 950–955.

[5] R. Claypoole, G. Davis, W. Sweldens, and R. Baraniuk, "Nonlinear wavelet transforms for image coding via lifting," submitted to IEEE Transactions on Image Processing, 1999.

[6] T. F. Chan and H. M. Zhou, "Adaptive ENO-wavelet transforms for discontinuous functions," Technical Report 21, Computational and Applied Mathematics, UCLA, Los Angeles, 1999.

[7] Ö. N. Gerek and A. E. Çetin, "Adaptive polyphase subband decomposition structures for image compression," *IEEE Transactions on Image Processing*, vol. 9, pp. 1649–1659, October 2000.

[8] G. Piella and H. J. A. M. Heijmans, "Adaptive lifting schemes with perfect reconstruction," Research Report PNA-R0104, CWI, Amsterdam, February 2001.

[9] E. Le Pennec and S. Mallat, "Image compression with geometrical wavelets," in *IEEE Conference on Image Processing (ICIP)*, Vancouver, September 2000.

[10] H. J. A. M. Heijmans and J. Goutsias, "Nonlinear multiresolution signal decomposition schemes: Part II: morphological wavelets," *IEEE Transactions on Image Processing*, vol. 9, no. 11, pp. 1897–1913, 2000.

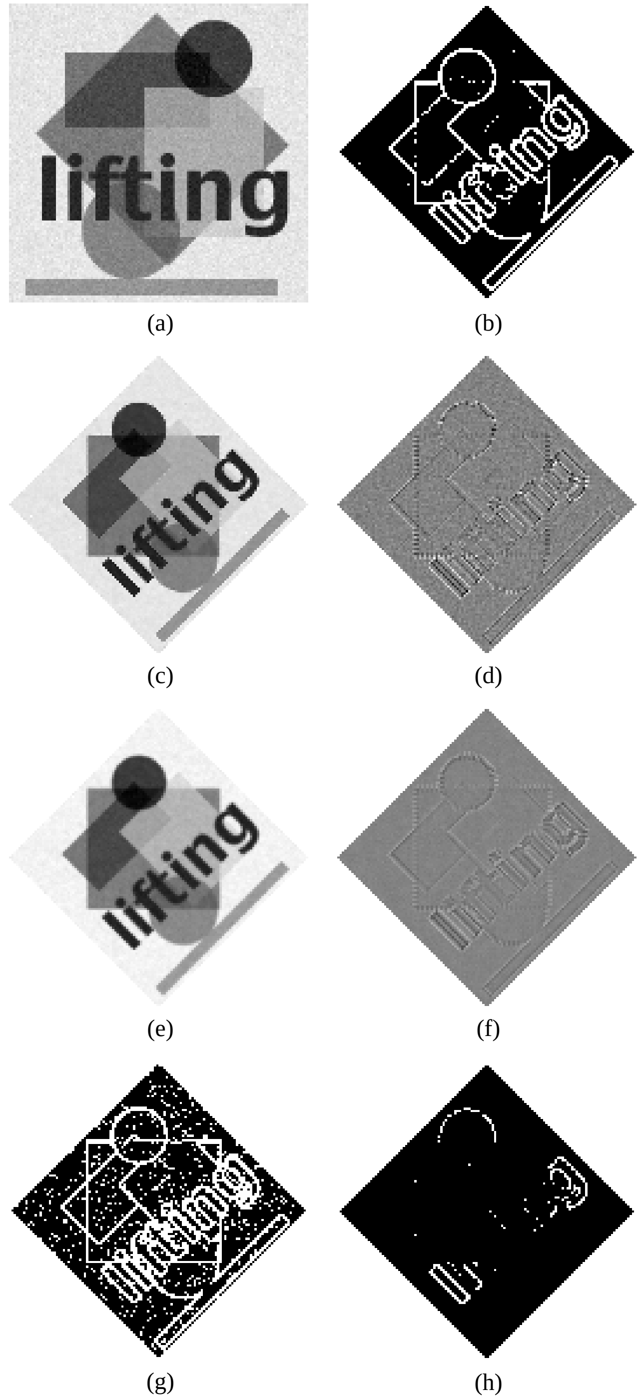

Fig. 4. (a) Original image; (b) Decision map for T=50; (c) Approximation for T=50; (d) Detail for T=50; (e) Approximation for fixed d=0 (average); (f) Detail for fixed d=0 (average); (g) Decision map for T=30; (h) Decision map for T=150.