Centrum voor Wiskunde en Informatica

REPORT*RAPPORT*

PNA

Probability, Networks and Algorithms

Probability, Networks and Algorithms

Adaptive Wavelets for Image Compression Using Update Lifting: Quantisation and Error Analysis

Henk J.A.M. Heijmans, Gemma Piella, Béatrice Pesquet-Popescu

REPORT PNA-E0402 MARCH 1, 2004

CWI is the National Research Institute for Mathematics and Computer Science. It is sponsored by the Netherlands Organization for Scientific Research (NWO).
CWI is a founding member of ERCIM, the European Research Consortium for Informatics and Mathematics.

CWI's research has a theme-oriented structure and is grouped into four clusters. Listed below are the names of the clusters and in parentheses their acronyms.

Probability, Networks and Algorithms (PNA)

Software Engineering (SEN)

Modelling, Analysis and Simulation (MAS)

Information Systems (INS)

P.O. Box 94079, 1090 GB Amsterdam (NL)
Kruislaan 413, 1098 SJ Amsterdam (NL)
Telephone +31 20 592 9333
Telefax +31 20 592 4199

ISSN 1386-3711

Adaptive Wavelets for Image Compression Using Update Lifting: Quantisation and Error Analysis

Henk J.A.M. Heijmans

CWI

P.O. Box 94079, 1090 GB Amsterdam, The Netherlands

Gemma Piella

UPC

Theory of Signal and Communication Department

Jordi Girona 1-3, Barcelona 08034, Spain

Béatrice Pesquet-Popescu

Signal and Image Processing Department

Ecole Nationale Supérieure des Télécommunications

46 rue Barrault, 75013 Paris, France

ABSTRACT

Classical linear wavelet representations of images have the drawback that they are not optimally suited to represent edge information. To overcome this problem, nonlinear multiresolution decompositions have been designed that can take into account the characteristics of the input signal/image. In our previous work [20,22] we have introduced an adaptive lifting framework, that does not require bookkeeping but has the property that it processes edges and homogeneous image regions in a different fashion. The current paper discusses the effects of quantisation in such an adaptive wavelet decomposition, as such an analysis is essential for the application of these adaptive decompositions in image compression. We provide conditions for recovering the original decisions at the synthesis and show how to estimate the reconstruction error in terms of the quantisation error.

2000 Mathematics Subject Classification: 68U10, 94A12.

Keywords and Phrases: Wavelets, lifting scheme, adaptive wavelets, quantisation, error estimation, image compression

Note: The work was of the first and the second author was carried out under project PNA4.2 "Wavelets and Morphology".

1. Introduction

Multiscale analyses deriving from classical linear wavelet transforms lead to a uniform smoothing of the information contents in images when going to low resolutions. There is however a strong demand stemming from various applications in image and video processing (compression, denoising, feature extraction, etc) for a more 'semantic' analysis and thereby for

multiresolution (MR) decompositions that do leave intact or even enhance certain important image characteristics such as sharp transitions, edges, singularities or other regions of interest over scales. To a certain extent, this can be achieved by non-linear wavelets [9, 17, 19, 21], or by adaptive subband structures [16].

Some of these decompositions exploit the flexibility of the *lifting* or *ladder* scheme [1], [23] to involve a non-linearity into the MR decomposition. The general ingredients of the lifting scheme, as illustrated in Figure 1, are an existing wavelet transform WT, an update map U, and a prediction map P. A decomposition of an input signal x_0 into bands x', y' is obtained

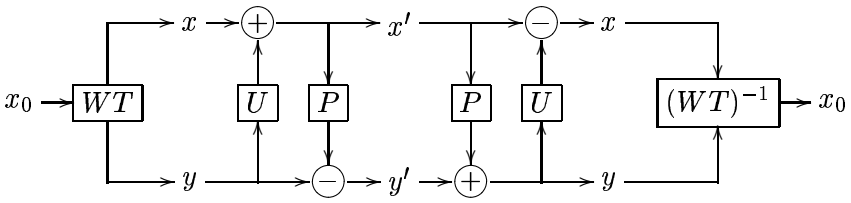

Figure 1: *General lifting scheme.*

in the following way. The original signal x_0 is first split into an approximation signal x and a detail signal y by a given wavelet transform WT (which may be a simple polyphase decomposition, also called 'lazy wavelet transform'). The update map U acting on y is used to modify x, resulting in a new approximation signal $x' = x + U(y)$. Subsequently, the prediction map P acting on x' is used to modify y, yielding a new detail signal $y' = y - P(x')$. At synthesis, the original signal x_0 is reconstructed by reversing the lifting steps and applying the inverse of WT.

It is important to observe that the invertibility of the scheme is guaranteed and does not require any condition on the lifting steps P and U. This flexibility has challenged researchers to develop various nonlinear wavelet transforms [6–8,10,18], including morphological ones [3, 11, 14, 15, 19]. The importance of such 'intelligent', 'adaptive' or 'data-driven' representations in signal and image analysis, compression, denoising, or feature extraction, has been recognised by various researchers and has led to a wealth of new approaches in wavelet theory, such as bandelets [21], ridgelets [12], curvelets [2], wedgelets [13], etc.

In [20,22] we have presented a new method for the construction of adaptive wavelets using update lifting. In the literature, one can find several other approaches for building adaptive wavelets [4, 5, 16, 24]. In [22], some of these approaches, and their drawbacks, have been discussed in more detail. The adaptive update lifting scheme introduced in [20] is general in the sense that it is neither *causal*[1], nor does require any bookkeeping to enable perfect reconstruction. Our method exploits the properties of seminorms to build lifting structures able to choose between different update filters, the choice being triggered by the local gradient-type features of the input. Moreover, we have established conditions under which these decisions can be recovered at synthesis, without the need for transmitting additional overhead information.

[1]Causality means that the computation of the detail signal at a given location depends 'only' on previously computed detail samples.

However, in order to use such adaptive wavelet schemes in lossy image compression, it is important to fully comprehend the effects of quantisation. A major difference with the non-adaptive case is that quantisation errors can dramatically affect the recovery of the switching decisions at synthesis giving rise to an 'inversion' of the update step using the wrong filter. It is evident that in such a case the reconstruction errors can be expected to be much larger than in cases where the update step is inverted in the correct way. In this paper, we derive conditions on the quantisation error that guarantee that we can still recover the original decisions. We also analyse the relations between the different parameters of the system allowing to calculate upper bounds for the reconstruction error. Rate-distortion curves for linear and adaptive schemes are compared and the reduction of visual artifacts is investigated when these adaptive structure are used for image compression.

The paper is organised as follows: in Section 2 and Section 3 we outline the framework of adaptive update lifting based on seminorms and recall the main results for perfect reconstruction. In Section 4 we extend the scheme with quantisation modules for all channels and provide conditions for recovering the original decisions at the synthesis and for relating the reconstruction error to the quantisation error. We present a case study for a particular seminorm in Section 5. Section 6 presents simulation results, and concluding remarks are made in Section 7.

2. The adaptive update lifting scheme

In [20] we have introduced an adaptive wavelet decomposition based on an adaptive update lifting step. In this section we will briefly describe this decomposition and in the next section we recall the most important results obtained in [20]. The scheme is depicted in Fig. 2 and

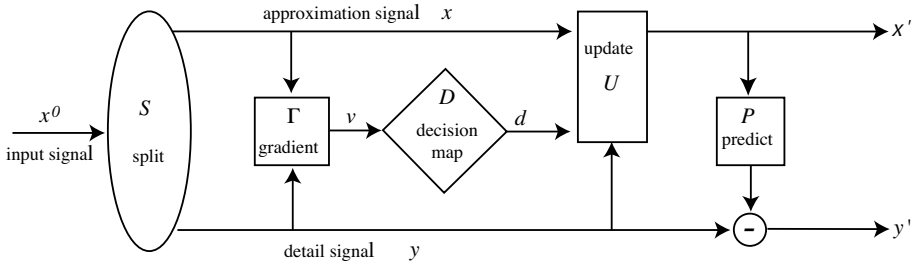

Figure 2: *Analysis part of the adaptive update lifting scheme.*

comprises four ingredients:

- a splitting S of the input signal into different bands (in this paper we assume that we have one approximation band x and M detail bands $y_1, y_2, \ldots, y_M$; the illustration in Fig. 2 shows only one detail band, however);
- computation of a (binary) decision map D based on a gradient vector (computed in the box labeled Γ);

- an adaptive update step U, where the adaptivity comes from the dependence of U on the output d of the decision map D;
- a (fixed) prediction step P.

These different steps will be discussed in more detail below. In this paper, a signal is a function from $\mathbb{Z}^d$ (where d is the dimension, not to be confused with the decision map) into the reals $\mathbb{R}$. Points in $\mathbb{Z}^d$ will be denoted by the vector notation $\boldsymbol{n}$.

The splitting S. An input signal x^0 is split into $x, \boldsymbol{y}$, where, possibly, $\boldsymbol{y}$ comprises more than one channel, say $y_1, y_2, \ldots, y_M$. The splitting S has an inverse S^{-1} which we call *merging*.

The gradient vector. Assume that the samples

$$y_{p_j}(\boldsymbol{n} + \boldsymbol{l}_j), \quad \boldsymbol{l}_j \in L\,, \quad j = 1, \ldots, N\,,$$

are used by the update step (including the the gradient vector used in the computation of the decision map). Here $p_j \in \{1, \ldots, M\}$ and L is a window in $\mathbb{Z}^d$ centered around the origin. Note that the p_j are not necessarily different. In our previous work [20] one can find some specific examples that may help the reader to understand our notation.

The gradient vector $\boldsymbol{v}$ has components v_j given by

$$v_j(\boldsymbol{n}) = x(\boldsymbol{n}) - y_{p_j}(\boldsymbol{n} + \boldsymbol{l}_j), \quad j = 1, \ldots, N\,, \tag{2.1}$$

and we write

$$\boldsymbol{v}(\boldsymbol{n}) = \Gamma(x, \boldsymbol{y})(\boldsymbol{n})\,.$$

Decision map. The decision map D maps a gradient vector $\boldsymbol{v} \in \mathbb{R}^N$ onto a binary output $D(\boldsymbol{v}) \in \{0, 1\}$. We assume that

$$D(\boldsymbol{v}) = [p(\boldsymbol{v}) > T]\,,$$

where p is a seminorm. Here the expression $[P]$ equals 1 if the predicate P is true and 0 if it is false.

Update step. The update step at location $\boldsymbol{n}$ is triggered by the outcome $d_{\boldsymbol{n}} = D(\boldsymbol{v}(\boldsymbol{n})) \in \{0, 1\}$ of the decision map. It uses both $d_{\boldsymbol{n}}$ and $\boldsymbol{y}$ to update the approximation signal x:

$$x'(\boldsymbol{n}) = \alpha_{d_{\boldsymbol{n}}} x(\boldsymbol{n}) + \sum_{j=1}^{N} \beta_{d_{\boldsymbol{n}}, j} y_{p_j}(\boldsymbol{n} + \boldsymbol{l}_j)\,, \tag{2.2}$$

which we will write as

$$x'(\boldsymbol{n}) = U_d(x \mid \boldsymbol{y})(\boldsymbol{n})\,. \tag{2.3}$$

For the filter coefficients in (2.2) we assume that

$$\alpha_d \neq 0 \;\text{ and }\; \alpha_d + \sum_{j=1}^{N} \beta_{d,j} = 1, \quad \text{for } d = 0, 1\,. \tag{2.4}$$

The update step is invertible in the sense that we can recover x if $x', \boldsymbol{y}, d$ are given:

$$x(\boldsymbol{n}) = U_d^{-1}(x' \mid \boldsymbol{y})(\boldsymbol{n})\,. \tag{2.5}$$

In fact, we have

$$x(\boldsymbol{n}) = \frac{1}{\alpha_{d_{\boldsymbol{n}}}}\Big(x'(\boldsymbol{n}) - \sum_{j=1}^{N} \beta_{d_{\boldsymbol{n}},j} y_{p_j}(\boldsymbol{n}+\boldsymbol{l}_j)\Big)\,. \tag{2.6}$$

Prediction step. In a similar spirit, the prediction step uses x (and possibly some bands of $\boldsymbol{y}$ if $\boldsymbol{y}$ is a multi-channel signal) to modify $\boldsymbol{y}$:

$$\boldsymbol{y}'(\boldsymbol{n}) = P(\boldsymbol{y} \mid x)(\boldsymbol{n})\,, \tag{2.7}$$

where P is called *prediction operator*. The prediction P is invertible, i.e., we can recover $\boldsymbol{y}$ from $x, \boldsymbol{y}'$:

$$\boldsymbol{y}(\boldsymbol{n}) = P^{-1}(\boldsymbol{y}' \mid x)(\boldsymbol{n})\,. \tag{2.8}$$

3. Perfect reconstruction for the adaptive update lifting scheme

The classical update lifting scheme does not involve the decision map D and, hence, the update filter U (i.e., the coefficients α and β in (2.2)) is fixed. It is obvious that in that case the update lifting step is invertible. This follows from the very structure of the scheme. As a matter of fact, the formula in (2.6) provides the inverse of the update lifting step. One says in this case that *perfect reconstruction* is possible. For the adaptive scheme, perfect reconstruction is less obvious since the inversion in (2.6) requires $d_{\boldsymbol{n}}$. It is not a priori true that $d_{\boldsymbol{n}}$ is known at synthesis.

In [20] it has been explained that in order to have a truly adaptive scheme one needs that

- the seminorm p satisfies $p(\boldsymbol{u}) > 0$, where $\boldsymbol{u} = (1, \ldots, 1)^T$ is a vector of length N;
- the update filters for $d = 0$ and $d = 1$ do not coincide, that is, $\beta_{0,j} \neq \beta_{1,j}$ for at least one j.

Throughout the remainder of this paper we assume that these two conditions hold along with (2.4). Subtraction of $y_{p_i}(\boldsymbol{n}+\boldsymbol{l}_i)$ at both sides of (2.2) yields

$$v_i'(\boldsymbol{n}) = (1-\beta_{d,i})v_i(\boldsymbol{n}) - \sum_{j\neq i} \beta_{d,j} v_j(\boldsymbol{n})\,, \tag{3.1}$$

where $\boldsymbol{v}'(\boldsymbol{n}) = \Gamma(x', \boldsymbol{y})(\boldsymbol{n})$, i.e.,

$$v_i'(\boldsymbol{n}) = x'(\boldsymbol{n}) - y_{p_i}(\boldsymbol{n}+\boldsymbol{l}_i)\,, \qquad i = 1, \ldots, N\,. \tag{3.2}$$

Define the $N \times N$-matrix A_d by the right hand-side expression in (3.1), i.e.,

$$A_d = \begin{pmatrix} 1-\beta_{d,1} & -\beta_{d,2} & -\beta_{d,3} & \ldots & -\beta_{d,N} \\ -\beta_{d,1} & 1-\beta_{d,2} & -\beta_{d,3} & \ldots & \vdots \\ -\beta_{d,1} & -\beta_{d,2} & \ddots & & \vdots \\ \vdots & \vdots & & \ddots & \vdots \\ -\beta_{d,1} & -\beta_{d,2} & -\beta_{d,3} & \ldots & 1-\beta_{d,N} \end{pmatrix}. \tag{3.3}$$

Thus the adaptive update lifting step is described by:

$$\begin{cases} \boldsymbol{v}' = A_d \boldsymbol{v} \\ d = [p(\boldsymbol{v}) > T]\,, \end{cases} \tag{3.4}$$

where we have suppressed the argument '$(\boldsymbol{n})$' from our notation.

In vector notation, the matrix A_d is given by

$$A_d = I - \boldsymbol{u}\boldsymbol{\beta}_d^T, \tag{3.5}$$

where $\boldsymbol{\beta}_d = (\beta_{d,1}, \ldots, \beta_{d,N})^T$ and I is the identity matrix. The determinant is given by

$$\det(A_d) = 1 - \boldsymbol{u}^T\boldsymbol{\beta}_d = 1 - \sum_{j=1}^{N} \beta_{d,j} = \alpha_d\,.$$

Since $\alpha_d \neq 0$ (see (2.4)) we have that A_d is invertible and it is not difficult to show that

$$A_d^{-1} = I - \boldsymbol{u}\boldsymbol{\beta}_d'^T\,,$$

where

$$\boldsymbol{\beta}_d' = -\boldsymbol{\beta}_d/\alpha_d\,.$$

Given a matrix A and a seminorm p, we define the *matrix seminorm* $p(A)$ and the *inverse matrix seminorm* $p^{-1}(A)$ as

$$\begin{aligned} p(A) &= \sup\{p(A\boldsymbol{v}) \mid \boldsymbol{v} \in \mathbb{R}^N \text{ and } p(\boldsymbol{v}) = 1\} \\ p^{-1}(A) &= \sup\{p(\boldsymbol{v}) \mid \boldsymbol{v} \in \mathbb{R}^N \text{ and } p(A\boldsymbol{v}) = 1\}\,. \end{aligned}$$

In the last expression we use the convention that $p^{-1}(A) = \infty$ if $p(A\boldsymbol{v}) = 0$ for all $\boldsymbol{v} \in \mathbb{R}^N$, unless p is identically zero, in which case both $p(A)$ and $p^{-1}(A)$ are zero. It is easy to verify that $p^{-1}(A) = p(A^{-1})$ if A is invertible.

Consider the adaptive update lifting step in (3.4). If $p(\boldsymbol{v}) \leq T$ at analysis, then the decision equals $d = 0$ and $\boldsymbol{v}' = A_0\boldsymbol{v}$. If, on the other hand, $p(\boldsymbol{v}) > T$, then $d = 1$ and $\boldsymbol{v}' = A_1\boldsymbol{v}$. For perfect reconstruction it must be possible to recover the decision d from the transformed gradient vector $\boldsymbol{v}'$. Here we shall restrict ourselves to the case where d can be recovered by thresholding the seminorm $p(\boldsymbol{v}')$, i.e., the case that

$$d = [p(\boldsymbol{v}) > T] = [p(\boldsymbol{v}') > T']\,,$$

for some $T' > 0$. We say that the *Threshold Criterion* holds, when, given a threshold $T > 0$, there exists a threshold $T' > 0$ such that

(*i*) if $p(\boldsymbol{v}) \leq T$ then $p(A_0\boldsymbol{v}) \leq T'$;

(*ii*) if $p(\boldsymbol{v}) > T$ then $p(A_1\boldsymbol{v}) > T'$.

It is obvious that the Threshold Criterion guarantees perfect reconstruction. In [20] we have proved the following result.

3.1 Proposition. *The Threshold Criterion holds if and only if the following three conditions are satisfied:*

$$p(A_0) < \infty \text{ and } p^{-1}(A_1) < \infty \tag{3.6}$$

$$p(A_0)p^{-1}(A_1) \leq 1\,. \tag{3.7}$$

We have also shown in [20] that a necessary condition for the Threshold Criterion to be valid is $|\alpha_0| \leq |\alpha_1|$. In [20] we have investigated the validity of the Threshold Criterion for various seminorms, among which the quadratic seminorms and the weighted gradient seminorms. Here we consider only this latter family. Thus assume that p is given by

$$p(\boldsymbol{v}) = \left|\boldsymbol{a}^T\boldsymbol{v}\right|\,, \tag{3.8}$$

where $\boldsymbol{a} \in \mathbb{R}^N$ satisfies the assumption

$$\boldsymbol{a}^T\boldsymbol{u} \neq 0\,.$$

Note that this assumption guarantees that $p(\boldsymbol{u}) > 0$. In [20] we have established the following result.

3.2 Proposition. *Let $p(\boldsymbol{v}) = |\boldsymbol{a}^T\boldsymbol{v}|$, with $\boldsymbol{a}^T\boldsymbol{u} \neq 0$. If $\boldsymbol{\beta}_d$ and $\boldsymbol{a}$ are not collinear, then $p(A_d) = p^{-1}(A_d) = \infty$. If $\boldsymbol{\beta}_d$ and $\boldsymbol{a}$ are collinear, then $p(A_d) = |\alpha_d|$ and $p^{-1}(A_d) = |\alpha_d|^{-1}$. In particular, the Threshold Criterion holds if and only if $\boldsymbol{\beta}_d$ and $\boldsymbol{a}$ are collinear and $|\alpha_0| \leq |\alpha_1|$.*

4. Compression and quantisation

Given a wavelet decomposition W which maps an input signal x into an output approximation signal x' and an (possibly multi-channel) output detail signal $\boldsymbol{y}'$, one can build a *multiresolution decomposition* of x by concatenation of several wavelet steps W. Using the output x' as the input for another wavelet decomposition step one obtains $x'', \boldsymbol{y}''$. Now one can repeat this for x'', etc. Thus concatenation of K steps yields a so-called K-level wavelet decomposition of x^0 into $\boldsymbol{y}^1, \boldsymbol{y}^2, \ldots, \boldsymbol{y}^K, x^K$, where we have written $\boldsymbol{y}^1, \boldsymbol{y}^2$, etc, instead of $\boldsymbol{y}', \boldsymbol{y}''$, etc. We write

$$(x^k, \boldsymbol{y}^k) = W(x^{k-1}), \quad k = 1, 2, \ldots, K\,.$$

Note that we have assumed for simplicity that the wavelet decomposition is identical for all levels. If, however, the wavelet differs per level, then we should write

$$(x^k, \boldsymbol{y}^k) = W^{k-1}(x^{k-1}), \quad k = 1, 2, \ldots, K\,.$$

In this paper we are interested in the case where W^k is the decomposition shown in Fig. 2 with a threshold depending on k.

A major field of application of wavelet transforms is image compression. A typical lossy compression scheme comprises three steps as shown in Fig. 3. For the transformation one often uses a discrete cosine transform (DCT) like in JPEG, or a wavelet transform like in JPEG2000. In this paper we are exclusively interested in the latter case, and more particularly in the wavelet transform shown in Fig. 2, constructed by means of an adaptive update lifting

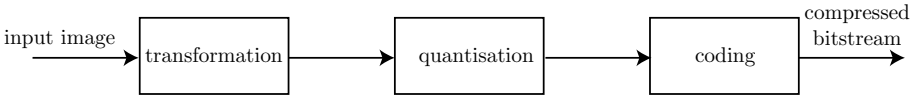

Figure 3: *A typical lossy compression scheme.*

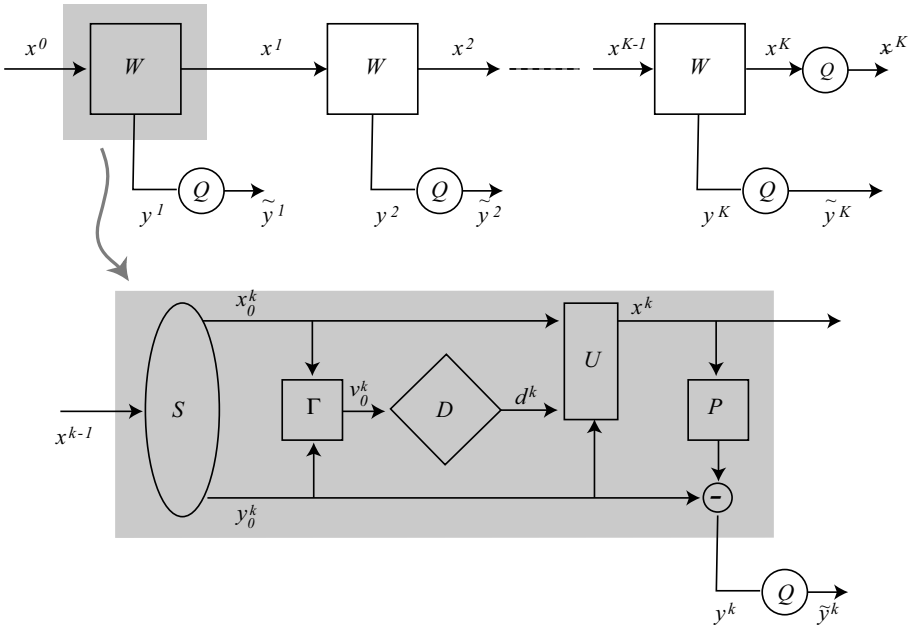

Figure 4: *Multiresolution wavelet analysis with quantisation. The wavelet transform W (grey box) is obtained by an adaptive update step followed by a fixed prediction step.*

step followed by a fixed prediction step. Here we briefly discuss quantisation. The coding step shown in Fig. 3 falls outside the scope of this paper.

Consider the multiresolution decomposition of an input signal x^0 as shown in Fig. 4. The output x^1 of the wavelet transform W^1 is used as an input for W^2, and, subsequently, the output x^2 of W^2 is used as input for W^3, up to level K. The final output approximation signal x^K is rounded by a quantisation operator Q^x and the output is denoted by $\tilde{x}^K$:

$$\tilde{x}^K(\boldsymbol{n}) = Q^x(x^K(\boldsymbol{n}))\,. \tag{4.1}$$

The quantisation operator Q^x will be described in more detail below.

All bands of the output detail signals $\boldsymbol{y}^1, \boldsymbol{y}^2, \ldots, \boldsymbol{y}^K$ are rounded by quantisation operators $Q_1^y, Q_2^y, \ldots, Q_K^y$, respectively. The quantised detail signal at level K is denoted by $\tilde{\boldsymbol{y}}^k$, that is

$$\tilde{y}_j^k(\boldsymbol{n}) = Q_k^y(y_j^k(\boldsymbol{n}))\,, \qquad j = 1, \ldots, M\,. \tag{4.2}$$

In this paper we will deal exclusively with uniform quantisation. The corresponding operator Q maps $\mathbb{R}$ to $\mathbb{Z}$ and is given by

$$Q(z) = \{z/c\}\,, \tag{4.3}$$

where $\{\cdot\}$ means rounding to the closest integer (with the convention that $Q(n+1/2) = n$ for every integer n) and c is the quantisation step. It is easy to verify that Q has a right inverse Q^{-1}, mapping $\mathbb{Z}$ into $\mathbb{R}$, given by

$$Q^{-1}(z) = c \cdot z\,.$$

The fact that Q^{-1} is a right inverse means that $QQ^{-1}(t) = t$ for $t \in \mathrm{Ran}(Q)$. If Q is given by (4.3), then $|Q^{-1}Q(z) - z| \le c/2$, for every $z \in \mathbb{R}$.

We assume that Q^x and Q^y_k have quantisation step $2q^x$ and $2q^y_k$, respectively. Thus we have

$$|(Q^x)^{-1}Q^x(x) - x| \le q^x\,, \tag{4.4}$$

for all $x \in \mathbb{R}$, and

$$|(Q^y_k)^{-1}Q^y_k(y) - y| \le q^y_k\,, \tag{4.5}$$

for all $y \in \mathbb{R}$ and all $k = 1, \ldots, K$. In the sequel we assume that the Threshold Criterion

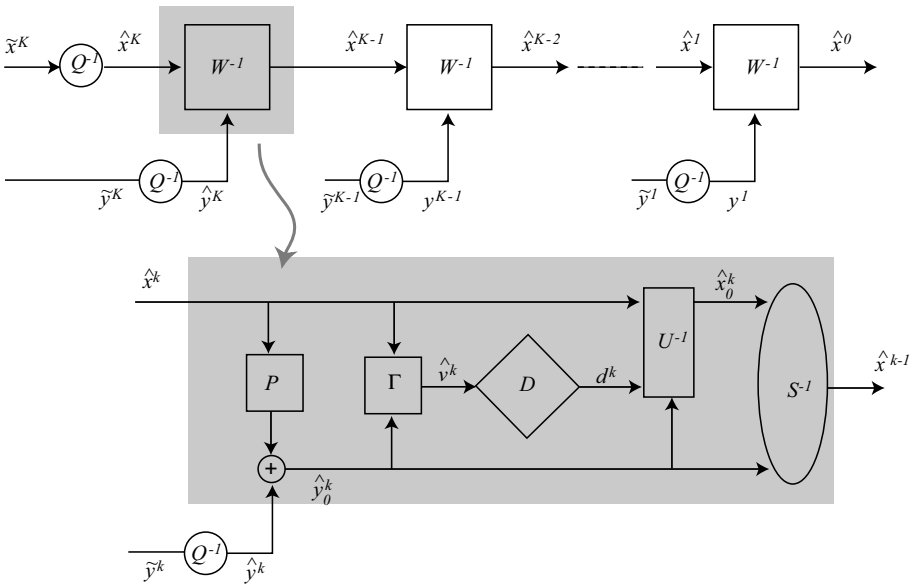

Figure 5: *Synthesis scheme associated with the wavelet decomposition in Fig. 4.*

holds. In the scheme of Fig. 4 this means that we will be able to recover $d^k_{\boldsymbol{n}} = D(\boldsymbol{v}^k_0(\boldsymbol{n}))$, where $\boldsymbol{v}^k_0(\boldsymbol{n}) = \Gamma(x^k_0, \boldsymbol{y}^k_0)(\boldsymbol{n})$, from $\boldsymbol{v}^k(\boldsymbol{n}) = \Gamma(x^k, \boldsymbol{y}^k_0)(\boldsymbol{n})$. Here $x^k(\boldsymbol{n}) = U_{d^k}(x^k_0 \mid \boldsymbol{y}^k_0)(\boldsymbol{n})$. However, as becomes clear from the scheme in Fig. 5, we can only dispose of $\hat{\boldsymbol{v}}^k(\boldsymbol{n}) = \Gamma(\hat{x}^k, \hat{\boldsymbol{y}}^k_0)(\boldsymbol{n})$, where $\hat{x}^k(\boldsymbol{n})$, $\hat{\boldsymbol{y}}^k_0(\boldsymbol{n})$ are 'approximations' of $x^k(\boldsymbol{n}), \boldsymbol{y}^k_0(\boldsymbol{n})$ due to errors resulting from the various quantisation steps.

For a vector $\boldsymbol{y} \in \mathbb{R}^M$, the notation $|\boldsymbol{y}|$ denotes the ℓ^∞-norm, i.e., $|\boldsymbol{y}| = \max\{|y_1|, \ldots, |y_M|\}$. If $x : \mathbb{Z}^d \to \mathbb{R}$ and $\boldsymbol{y} : \mathbb{Z}^d \to \mathbb{R}^M$, then both $\|x\|$ and $\|\boldsymbol{y}\|$ denote the sup-norm, i.e.,

$$\|x\| = \sup\{|x(\boldsymbol{n})| : \boldsymbol{n} \in \mathbb{Z}^d\},\ \|\boldsymbol{y}\| = \sup\{|\boldsymbol{y}(\boldsymbol{n})| : \boldsymbol{n} \in \mathbb{Z}^d\}.$$

For the mappings Γ, P in the scheme above, we make the following assumptions:

$$p\big(\Gamma(x',\boldsymbol{y}')(\boldsymbol{n})-\Gamma(x,\boldsymbol{y})(\boldsymbol{n})\big)\le \pi_x\|x-x'\|+\pi_y\|\boldsymbol{y}'-\boldsymbol{y}\|\,, \tag{4.6}$$

$$|P^{-1}(\boldsymbol{y}'\mid x')(\boldsymbol{n})-P^{-1}(\boldsymbol{y}\mid x)(\boldsymbol{n})|\le \rho_x\|x-x'\|+\rho_y\|\boldsymbol{y}'-\boldsymbol{y}\|\,, \tag{4.7}$$

for all $x,x':\mathbb{Z}^d\to\mathbb{R}$, $\boldsymbol{y},\boldsymbol{y}':\mathbb{Z}^d\to\mathbb{R}^M$ and all $\boldsymbol{n}\in\mathbb{Z}^d$. The estimate in (4.6) can easily be obtained from the observation that

$$\begin{aligned}
&p\big((v_1'(\boldsymbol{n}),\dots,v_N'(\boldsymbol{n}))-(v_1(\boldsymbol{n}),\dots,v_N(\boldsymbol{n}))\big)\\
&=p\Big(\big(x'(\boldsymbol{n})-y_{p_1}'(\boldsymbol{n}+\boldsymbol{l}_1),\dots,x'(\boldsymbol{n})-y_{p_N}'(\boldsymbol{n}+\boldsymbol{l}_N)\big)\\
&\quad-\big(x(\boldsymbol{n})-y_{p_1}(\boldsymbol{n}+\boldsymbol{l}_1),\dots,x(\boldsymbol{n})-y_{p_N}(\boldsymbol{n}+\boldsymbol{l}_N)\big)\Big)\\
&\le p\Big(\big(x'(\boldsymbol{n})-x(\boldsymbol{n})\big)\cdot\boldsymbol{u}\Big)+p\Big(y_{p_1}'(\boldsymbol{n}+\boldsymbol{l}_1)-y_{p_1}(\boldsymbol{n}+\boldsymbol{l}_1),\dots,y_{p_N}'(\boldsymbol{n}+\boldsymbol{l}_N)-y_{p_N}(\boldsymbol{n}+\boldsymbol{l}_N)\Big)\,.
\end{aligned}$$

Note that the first part of the last expression can be estimated by

$$p\Big(\big(x'(\boldsymbol{n})-x(\boldsymbol{n})\big)\cdot\boldsymbol{u}\Big)\le|x(\boldsymbol{n})-x'(\boldsymbol{n})|\cdot p(\boldsymbol{u})\,.$$

For the update operator we can show that

$$\|U_d^{-1}(x'\mid\boldsymbol{y}')-U_d^{-1}(x\mid\boldsymbol{y})\|\le\theta_x\|x-x'\|+\theta_y\|\boldsymbol{y}'-\boldsymbol{y}\|\,, \tag{4.8}$$

for all $x,x',\boldsymbol{y},\boldsymbol{y}'$ and $d=0,1$, where

$$\theta_x=\max\{\frac{1}{|\alpha_0|},\frac{1}{|\alpha_1|}\}\text{ and }\theta_y=\max\{\frac{\sum_{j=1}^N|\beta_{0,j}|}{|\alpha_0|},\frac{\sum_{j=1}^N|\beta_{1,j}|}{|\alpha_1|}\}\,. \tag{4.9}$$

Now we have all ingredients to formulate and prove the main result of this paper. This result shows that it may be possible to reconstruct the decision map at all levels without errors (i.e., $D(\boldsymbol{v}_0^k)=D(\hat{\boldsymbol{v}}^k)$ for all k). This, however, requires a substantial strengthening of the condition $p(A_0)p^{-1}(A_1)\le 1$ in (3.7), namely the condition (4.12) below. In that case we are also able to derive an upper bound of the reconstruction error $\|\hat{x}^k-x^k\|$; see (4.15).

4.1 Theorem. *Define Δ_k recursively as*

$$\Delta_{k-1}=\max\{\rho_x\Delta_k+\rho_y q_k^y,\theta_x\Delta_k+\theta_y(\rho_x\Delta_k+\rho_y q_k^y)\}\,, \tag{4.10}$$

with $\Delta_K=q^x$. Define also

$$\tau_k=\pi_x\Delta_k+\pi_y(\rho_x\Delta_k+\rho_y q_k^y)\,. \tag{4.11}$$

Assume that the threshold T_k is such that

$$\Big(\big(p^{-1}(A_1)\big)^{-1}-p(A_0)\Big)T_k\ge 2\tau_k\,, \tag{4.12}$$

and choose a threshold $\hat{T}_k$ such that

$$p(A_0)T_k+\tau_k\le\hat{T}_k\le\Big(\big(p^{-1}(A_1)\big)^{-1}\Big)T_k-\tau_k\,. \tag{4.13}$$

Then we have

$$d^k=[p(\hat{\boldsymbol{v}}^k)>\hat{T}_k]\ \textit{for}\ k=1,\dots,K\,, \tag{4.14}$$

and

$$\|\hat{x}^k-x^k\|\le\Delta_k\ \textit{for}\ k=0,\dots,K\,. \tag{4.15}$$

Proof. We will prove this theorem by induction. First observe that (4.15) holds for $k = K$, i.e.,

$$\|\hat{x}^K - x^K\| \leq \Delta_K = q^x \,.$$

Now assume that (4.15) holds for some k with $1 \leq k \leq K$. We show that (4.14) holds for k and that we can also establish (4.15) for $k-1$. Thus, by induction, we conclude that, indeed, (4.14) holds for $k = 1, \ldots, K$ and that (4.15) holds for $k = 0, \ldots, K$.

Since $\hat{y}_j^k(\boldsymbol{n}) = (Q_k^y)^{-1} Q_k^y(y_j^k(\boldsymbol{n}))$, we get immediately that

$$\|\hat{\boldsymbol{y}}^k - \boldsymbol{y}^k\| \leq q_k^y, \;\; k = 1, \ldots, K \,. \tag{4.16}$$

In the remainder of the proof, we write q_k instead of q_k^y.

We know that

$$\hat{\boldsymbol{y}}_0^k(\boldsymbol{n}) = P^{-1}(\hat{\boldsymbol{y}}^k \mid \hat{x}^k)(\boldsymbol{n}) \text{ and } \boldsymbol{y}_0^k(\boldsymbol{n}) = P^{-1}(\boldsymbol{y}^k \mid x^k)(\boldsymbol{n}) \,,$$

and from the estimate in (4.7) we get that

$$\|\hat{\boldsymbol{y}}_0^k - \boldsymbol{y}_0^k\| \leq \rho_x \|\hat{x}^k - x^k\| + \rho_y q_k \leq \rho_x \Delta_k + \rho_y q_k \,. \tag{4.17}$$

We now investigate the two possibilities for the decision map, namely $d^k = 0$ and $d^k = 1$. To simplify notation we will suppress the argument $\boldsymbol{n}$.

Case $d^k = 0$: this means that $p(\boldsymbol{v}_0^k) \leq T$ and we know from (3.4) that

$$\boldsymbol{v}^k = A_0 \boldsymbol{v}_0^k$$

in this case. We derive an upper estimate for $p(\hat{\boldsymbol{v}}^k)$. Here we use that p satisfies the triangle inequality:

$$\begin{aligned} p(\hat{\boldsymbol{v}}^k) &\leq p(\boldsymbol{v}^k) + p(\hat{\boldsymbol{v}}^k - \boldsymbol{v}^k) = p(A_0 \boldsymbol{v}_0^k) + p(\hat{\boldsymbol{v}}^k - \boldsymbol{v}^k) \\ &\leq p(A_0) p(\boldsymbol{v}_0^k) + p(\hat{\boldsymbol{v}}^k - \boldsymbol{v}^k) \\ &\leq p(A_0) T_k + \pi_x \|x^k - \hat{x}^k\| + \pi_y \|\hat{\boldsymbol{y}}_0^k - \boldsymbol{y}_0^k\| \,, \end{aligned}$$

where we have used estimate (4.6). Thus we get that

$$p(\hat{\boldsymbol{v}}^k) \leq p(A_0) T_k + \pi_x \Delta_k + \pi_y (\rho_x \Delta_k + \rho_y q_k) \leq \hat{T}_k \,.$$

Case $d^k = 1$: thus $p(\boldsymbol{v}_0^k) > T_k$ and we know from (3.4) that

$$\boldsymbol{v}^k = A_1 \boldsymbol{v}_0^k$$

in this case. We derive a lower estimate for $p(\hat{\boldsymbol{v}}^k)$.

$$\begin{aligned} p(\hat{\boldsymbol{v}}^k) &\geq p(\boldsymbol{v}^k) - p(\hat{\boldsymbol{v}}^k - \boldsymbol{v}^k) = p(A_1 \boldsymbol{v}_0^k) - p(\hat{\boldsymbol{v}}^k - \boldsymbol{v}^k) \\ &\geq \big(p^{-1}(A_1)\big)^{-1} p(\boldsymbol{v}_0^k) - p(\hat{\boldsymbol{v}}^k - \boldsymbol{v}^k) \\ &> \big(p^{-1}(A_1)\big)^{-1} T_k - \pi_x \|x^k - \hat{x}^k\| - \pi_y \|\hat{\boldsymbol{y}}_0^k - \boldsymbol{y}_0^k\| \,, \end{aligned}$$

and we arrive at the estimate

$$p(\hat{\boldsymbol{v}}^k) > \big(p^{-1}(A_1)\big)^{-1} T_k - \pi_x \Delta_k - \pi_y (\rho_x \Delta_k + \rho_y q_k) \geq \hat{T}_k \,.$$

Thus we conclude that

$$d^k = [p(\hat{\boldsymbol{v}}^k) > \hat{T}_k] \,.$$

From $\hat{x}_0^k(\boldsymbol{n}) = U_d^{-1}(\hat{x}^k \mid \hat{\boldsymbol{y}}_0^k)(\boldsymbol{n})$ and $x_0^k(\boldsymbol{n}) = U_d^{-1}(x^k \mid \boldsymbol{y}_0^k)(\boldsymbol{n})$ in combination with (4.8) we derive that

$$\|\hat{x}_0^k - x_0^k\| \leq \theta_x \|\hat{x}^k - x^k\| + \theta_y \|\hat{\boldsymbol{y}}_0^k - \boldsymbol{y}_0^k\| \,.$$

Using (4.17) we arrive at

$$\|\hat{x}_0^k - x_0^k\| \le \theta_x \Delta_k + \theta_y(\rho_x \Delta_k + \rho_y q_k)\,. \tag{4.18}$$

Now we need one final step to compute $\hat{x}^{k-1}$, namely the merging of $\hat{x}_0^k$ and $\hat{\boldsymbol{y}}_0^k$ by means of S^{-1}. It is obvious that

$$\|\hat{x}^{k-1} - x^{k-1}\| \le \max\{\|\hat{x}_0^k - x_0^k\|, \|\hat{\boldsymbol{y}}_0^k - \boldsymbol{y}_0^k\|\}\,.$$

With (4.17) and (4.18) this yields

$$\begin{aligned} \|\hat{x}^{k-1} - x^{k-1}\| &\le \max\{\rho_x \Delta_k + \rho_y q_k, \theta_x \Delta_k + \theta_y(\rho_x \Delta_k + \rho_y q_k)\} \\ &= \Delta_{k-1}\,. \end{aligned}$$

This proves the result. □

In many cases one has

$$\rho_x = \rho_y = 1\,.$$

Put

$$R = \theta_x + \theta_y \quad \text{and} \quad r = \theta_y\,.$$

From (4.10) we get that

$$\Delta_{k-1} = \max\{\Delta_k + q_k^y, R\Delta_k + r q_k^y\}\,. \tag{4.19}$$

Define

$$\lambda_k = \frac{\Delta_k}{\Delta_K} \quad \text{and} \quad \mu_k = \frac{q_k^y}{\Delta_K}\,.$$

Thus in particular,

$$\lambda_K = 1 \quad \text{and} \quad \mu_K = \mu = \frac{q_K^y}{q_K^x}\,.$$

Put $q = \Delta_K = q_K^x$, hence $q_K^y = \mu q$. From (4.19) we get that

$$\lambda_{k-1} = \max\{\lambda_k + \mu_k, R\lambda_k + r\mu_k\}\,, \tag{4.20}$$

and from (4.11) we get

$$\frac{\tau_k}{q} = (\pi_x + \pi_y)\lambda_k + \pi_y \mu_k\,. \tag{4.21}$$

We can find a particular solution of (4.20) in the following case. Put

$$\omega = \max\{1 + \mu, R + \mu r\}\,,$$

and assume that

$$\mu_k = \omega^{K-k}\mu\,.$$

Then we have the following solution of (4.20):

$$\lambda_k = \omega^{K-k}\,.$$

From (4.21) we get that in this case

$$\frac{\tau_k}{q} = \big(\pi_x + (1+\mu)\pi_y\big)\omega^{K-k}\,. \tag{4.22}$$

We derive another particular solution of (4.20). Assume that both R and r are larger or equal than 1. Then (4.20) reduces to

$$\lambda_{k-1} = R\lambda_k + r\mu_k \,. \tag{4.23}$$

Assume also that

$$\mu_k = \delta^{K-k}\mu \,.$$

We have the following solution of (4.23):

$$\lambda_k = \begin{cases} R^{K-k}\big(1 + \frac{r\mu}{R-\delta}\big) - \dfrac{r\mu}{R-\delta}\delta^{K-k} & \text{if } R \neq \delta \\ \\ R^{K-k}\big(1 + (K-k)\dfrac{r\mu}{R}\big) & \text{if } R = \delta. \end{cases} \tag{4.24}$$

Now we get that if $R \neq \delta$, then

$$\frac{\tau_k}{q} = aR^{K-k} + b\delta^{K-k} \,, \tag{4.25}$$

where

$$a = (\pi_x + \pi_y) \cdot (1 + \frac{r\mu}{R-\delta}) \quad \text{and} \quad b = \pi_y\mu - (\pi_x + \pi_y) \cdot \frac{r\mu}{R-\delta} \,. \tag{4.26}$$

For the particular case where $R = \delta$, we obtain

$$\frac{\tau_k}{q} = \Big((\pi_x + \pi_y) \cdot \big(1 + (K-k)\frac{r\mu}{R}\big) + \mu\pi_y\Big) R^{K-k} \,. \tag{4.27}$$

5. Case study: $p(\boldsymbol{v}) = |a^T \boldsymbol{v}|$

Throughout this section we assume that we are using a seminorm of the form

$$p(\boldsymbol{v}) = |\boldsymbol{a}^T \boldsymbol{v}| \,, \tag{5.1}$$

where $\boldsymbol{a} \in \mathbb{R}^N$ with $\boldsymbol{a}^T \boldsymbol{u} \neq 0$.

5.1 General

Define

$$\sigma_a = |\sum_{j=1}^{N} a_j| \text{ and } \Sigma_a = \sum_{j=1}^{N} |a_j| \,. \tag{5.2}$$

We compute the constants π_x, π_y in (4.6) for this case. Obviously,

$$\begin{aligned} p\big(\Gamma(x', \boldsymbol{y}')(\boldsymbol{n}) - \Gamma(x, \boldsymbol{y})(\boldsymbol{n})\big) &= |\sum_{j=1}^{N} a_j \big(x'(\boldsymbol{n}) - y'_j(\boldsymbol{n}) - x(\boldsymbol{n}) + y_j(\boldsymbol{n})\big)| \\ &\leq |\sum_{j=1}^{N} a_j| \cdot \|x' - x\| + |\sum_{j=1}^{N} a_j \big(y'_j(\boldsymbol{n}) - y_j(\boldsymbol{n})\big)| \\ &\leq |\sum_{j=1}^{N} a_j| \cdot \|x' - x\| + \sum_{j=1}^{N} |a_j| \cdot \|y' - y\| \,. \end{aligned}$$

This yields

$$\pi_x = \sigma_a \text{ and } \pi_y = \Sigma_a \,. \tag{5.3}$$

We know from Proposition 3.2 that, for the Threshold Criterion to hold, we must have

$$\boldsymbol{\beta}_d = \gamma_d \boldsymbol{a} \text{ and } |\alpha_0| \le |\alpha_1| \,, \tag{5.4}$$

where γ_d is a constant. If the Threshold Criterion holds, then

$$p(A_0) = |\alpha_0| \quad \text{and} \quad p^{-1}(A_1) = |\alpha_1|^{-1} \,.$$

From (4.9) and (5.2) we get

$$\theta_x = \frac{1}{|\alpha_0|} \text{ and } \theta_y = \max\Big\{ |\frac{\gamma_0}{\alpha_0}|, |\frac{\gamma_1}{\alpha_1}| \Big\} \cdot \Sigma_a \,, \tag{5.5}$$

and from the fact that $\alpha_d + \gamma_d \sum a_j = 1$ we get

$$|\gamma_d| \sigma_a = |1 - \alpha_d| \,. \tag{5.6}$$

This yields

$$\theta_y = \frac{\Sigma_a}{\sigma_a} \cdot \max\Big\{ |\frac{1-\alpha_0}{\alpha_0}|, |\frac{1-\alpha_1}{\alpha_1}| \Big\} \,. \tag{5.7}$$

5.2 The 1D case

Consider the 1D case with $\boldsymbol{v} = (v_{-K}, \ldots, v_0, \ldots, v_L)^T$ where $v_j(n) = x(n) - y(n+j)$. Assume that prediction is given by $y'(n) = y(n) - \frac{1}{2}\big(x(n) + x(n+1)\big)$. Since $y_j(n) = y(n+j)$, we get that the prediction operator P in (2.7) is given by

$$y'_j(n) = y_j(n) - \frac{1}{2}\big(x(n+j) + x(n+j+1)\big) \,,$$

for $j = -K, \ldots, L$. Then P^{-1} in (2.8) is given by

$$y_j(n) = y'_j(n) + \frac{1}{2}\big(x(n+j) + x(n+j+1)\big) \,,$$

and we find immediately that ρ_x, ρ_y in (4.7) are given by

$$\rho_x = \rho_y = 1 \,.$$

Consider henceforth the case $\boldsymbol{a} = (1,1)^T$. For the Threshold Criterion to hold we must satisfy (5.4). Thus we obtain

$$\alpha_d + 2\gamma_d = 1, \quad d = 0, 1 \,.$$

Furthermore

$$\pi_x = \pi_y = \sigma_a = \Sigma_a = 2 \,.$$

Also

$$\theta_x = \frac{1}{|\alpha_0|} \quad \text{and} \quad \theta_y = \max\Big\{ |\frac{1-\alpha_0}{\alpha_0}|, |\frac{1-\alpha_1}{\alpha_1}| \Big\} \,.$$

Assume that $0 < \alpha_0 < \alpha_1 \le 1$, then

$$r = \theta_y = \frac{1-\alpha_0}{\alpha_0} \quad \text{and} \quad R = \frac{2-\alpha_0}{\alpha_0} \,.$$

Choosing $\alpha_0 = \frac{1}{2}$, we get that $r = 1$, $R = 3$, and

$$\omega = \max\{1 + \mu, 3 + \mu\} = 3 + \mu\,.$$

From (4.22) we derive that

$$\frac{\tau_k}{q} = (4 + 2\mu)\omega^{K-k}\,.$$

If we take $\mu = 1$ and $q = \frac{1}{2}$ (the latter choice corresponds with quantisation step equal to 1), we get that

$$\tau_k = 3 \cdot 4^{K-k}\,.$$

The condition on T_k in (4.12) amounts to

$$(\alpha_1 - \frac{1}{2}) \cdot T_k \geq 6 \cdot 4^{K-k}\,.$$

Choosing $\alpha_1 = 1$ this gives

$$T_k \geq 12 \cdot 4^{K-k}\,.$$

Thus $T_K \geq 12,\ T_{K-1} \geq 48$, etc.

With the same parameters, if we use the solution in (4.24) with $\delta = 2$, we get

$$a = 8 \quad \text{and} \quad b = -2$$

and hence

$$T_k \geq 16 \cdot 3^{K-k} - 4 \cdot 2^{K-k}\,.$$

Thus, $T_K \geq 12,\ T_{K-1} \geq 40$, etc.

5.3 The 2D case

Consider a 2D decomposition with 4 bands x, y_1, y_2, y_3 as depicted in Fig. 6. The sampling

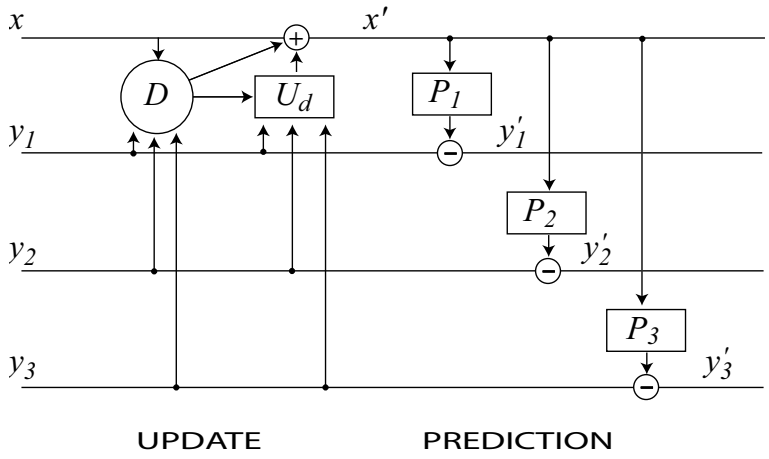

Figure 6: *2D wavelet decomposition comprising an adaptive update lifting step (left) and three consecutive prediction lifting steps (right).*

scheme is depicted in Fig. 7. After prediction, the bands y_1', y_2', y_3' capture, respectively, the

horizontal, vertical, and diagonal details. Assume that the gradient vector $v_j(m,n)$, $j = 1,2,\ldots,8$ is based on $x(m,n)$ and the 8 neighbouring samples of $x(m,n)$ shown in in Fig. 7, with the labeling indicated inside the small disks. For example $v_1(m,n) = x(m,n) - y_1(m,n)$ and $v_6(m,n) = x(m,n) - y_3(m-1,n)$.

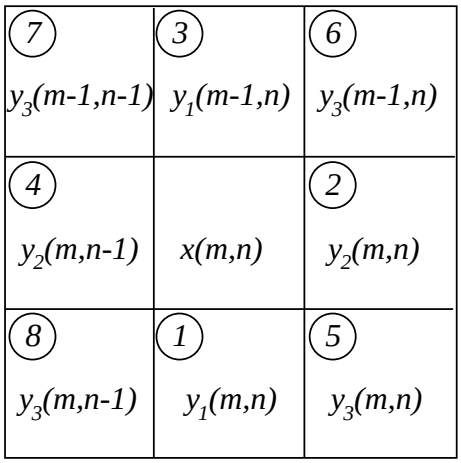

Figure 7: *Labeling of samples in* 3×3 *window centered around* $x(m,n)$.

We use the seminorm given by

$$p(\boldsymbol{v}) = |\boldsymbol{a}^T \boldsymbol{v}| = |a_1 v_1 + \cdots + a_8 v_8| \,.$$

The prediction steps in Fig. 6 are chosen to be

$$P_1(x) = P_2(x) = P_3(x) = x \,,$$

which amounts to

$$y'_j(m,n) = y_j(m,n) - x(m,n) \,, \;\; j = 1,2,3 \,.$$

We investigate two different choices for $\boldsymbol{a}$, labeled Case 1 and Case 2, respectively. In these cases the filter $\boldsymbol{a}$ corresponds, respectively, with a 2nd order derivative filter and a Laplacian derivative filter.

Case 1: 2nd order derivative filter

We choose $\boldsymbol{a} = (1,1,1,1,-\frac{1}{2},-\frac{1}{2},-\frac{1}{2},-\frac{1}{2})^T$. In [20] we have explained that this corresponds with a 2nd order derivative filter. The Threshold Criterion holds only if we assume that $\boldsymbol{\beta}_d = \gamma_d \boldsymbol{a}$ with $|\alpha_0| \leq |\alpha_1|$, that is $|1 - 2\gamma_0| \leq |1 - 2\gamma_1|$. The parameters that have been defined before can easily be computed in this case. They are:

$$\rho_x = \rho_y = 1, \;\; \pi_x = \sigma_a = 2, \;\; \pi_y = \Sigma_a = 6 \,,$$

and

$$\theta_x = \frac{1}{|\alpha_0|} \;\; \text{and} \;\; \theta_y = 3 \cdot \max\Big\{ |\frac{1-\alpha_0}{\alpha_0}|, |\frac{1-\alpha_1}{\alpha_1}| \Big\} \,.$$

Assume again that $0 < \alpha_0 < \alpha_1 \leq 1$, then

$$r = \theta_y = \frac{3 - 3\alpha_0}{\alpha_0} \;\; \text{and} \;\; R = \frac{4 - 3\alpha_0}{\alpha_0} \,.$$

Choosing $\alpha_0 = \frac{2}{3}$, we get that $r = \frac{3}{2}$ and $R = 3$, and hence

$$\omega = 3 + \frac{3}{2}\mu\,.$$

From (4.22) we derive that

$$\frac{\tau_k}{q} = (8 + 6\mu)\omega^{K-k}\,.$$

If we take $\mu = 1$ and $q = \frac{1}{2}$ (the latter choice corresponds with quantisation step equal to 1), we get that

$$\tau_k = 7 \cdot \left(\frac{9}{2}\right)^{K-k}\,.$$

Thus the condition on T_k in (4.12) amounts to

$$(\alpha_1 - \frac{2}{3}) \cdot T_k \geq 14 \cdot \left(\frac{9}{2}\right)^{K-k}\,.$$

Choosing $\alpha_1 = 1$ this gives

$$T_k \geq 42 \cdot \left(\frac{9}{2}\right)^{K-k}\,.$$

Thus, $T_K \geq 42,\ T_{K-1} \geq 189$, etc.

Using the same parameters and the solution given in (4.24) with $\delta = 2$, we get

$$a = 20 \quad \text{and} \quad b = -6\,,$$

which yields

$$T_k \geq 60 \cdot 3^{K-k} - 18 \cdot 2^{K-k}\,.$$

Thus, $T_K \geq 42,\ T_{K-1} \geq 144$, etc.

Case 2: Laplacian derivative

Here we consider the case $\boldsymbol{a} = (1, 1, 1, 1, 0, 0, 0, 0)^T$, and in this case the seminorm p models the Laplacian derivative. The parameters that have been defined before are:

$$\rho_x = \rho_y = 1,\ \pi_x = \pi_y = \sigma_a = \Sigma_a = 4\,,$$

and

$$\theta_x = \frac{1}{|\alpha_0|} \quad \text{and} \quad \theta_y = \max\left\{|\frac{1-\alpha_0}{\alpha_0}|, |\frac{1-\alpha_1}{\alpha_1}|\right\}.$$

Assume again that $0 < \alpha_0 < \alpha_1 \leq 1$, then

$$r = \theta_y = \frac{1-\alpha_0}{\alpha_0} \quad \text{and} \quad R = \frac{2-\alpha_0}{\alpha_0}\,.$$

Choosing $\alpha_0 = \frac{1}{2}$, we get that $r = 1$ and $R = 3$, and hence

$$\omega = 3 + \mu\,.$$

Let us consider that $\mu = 1$. Then, from (4.22) we derive that

$$\frac{\tau_k}{q} = 12 \cdot 4^{K-k}\,.$$

Choosing $q = \frac{1}{2}$ and $\alpha_1 = 1$, the condition on T_k in (4.12) amounts to

$$T_k \geq 24 \cdot 4^{K-k}.$$

Thus, $T_K \geq 24$, $T_{K-1} \geq 96$, etc.

Using the same parameters and the solution given in (4.24) with $\delta = 2$, we get

$$T_k \geq 32 \cdot 3^{K-k} - 8 \cdot 2^{K-k}.$$

Thus, $T_K \geq 24$, $T_{K-1} \geq 80$, etc.

6. Simulations

For the simulations we use the 'Camera' image and the synthetic 'Rectangles' image shown in Fig. 8. The images are decomposed with the adaptive wavelet scheme described in Section 4 for the particular cases (Case 1 and Case 2) analysed above. We consider three levels of decomposition (i.e., $K = 3$) and take $\mu = 1$ and $q = 1/2$. Recall that this choice means that the quantisation step is 1 for level $k = K$. We do not use any coding scheme. At synthesis the quantised transform coefficients are de-quantised (e.g., $\hat{y}^k(\boldsymbol{n}) = q_k^y \cdot \tilde{y}^k(\boldsymbol{n})$ where $q_k^y = \mu_k q$) and the inverse wavelet transform is applied. We repeat this compression/decompression process varying the quantisation steps (for all $k < K$) in order to compute the rate-distortion curves (bitrate versus PSNR).

The bitrate is computed from the weighted entropy

$$h = 2^{-2K} H(\tilde{x}^K) + \sum_{k=1}^{K} 2^{-2k} \sum_{j=1}^{3} H\left(\tilde{y}_j^k\right)$$

where $H(x)$ denotes the first order entropy of image x. It is measured in bits per pixel (bpp). The peak-signal-to-noise-ratio (PSNR) is defined as

$$\text{PSNR} = 10 \log_{10} \frac{255^2}{MSE}$$

where MSE is the mean squared error between the original image and the reconstructed one.

For comparison purposes, we perform the same experiments using the non-adaptive wavelet decomposition with fixed update step associated with $d = 0$. Fig. 9 shows the rate-distortion curves of the adaptive wavelet decompositions against the non-adaptive ones for the 'Camera' image (left) and the 'Rectangles' image (right). The top row corresponds to Case 1 (i.e., $\boldsymbol{a} = (1, 1, 1, 1, -1/2, -1/2, -1/2, -1/2)^T$ with $\alpha_1 = 1$, $\alpha_0 = 2/3$) and the bottom row to Case 2 (i.e., $\boldsymbol{a} = (1, 1, 1, 1, 0, 0, 0, 0)^T$ with $\alpha_1 = 1$, $\alpha_0 = 1/2$). One can see that the adaptive scheme has in general better PSNR performance. Fig. 10 shows a zoom of the reconstructed synthetic 'Rectangles' image (in Case 2) at bitrate 0.5 bpp for both the adaptive (left) and the non-adaptive (right) wavelet decomposition. We note that in the non-adaptive case, the images suffer from blurring and ringing around the edges. In the adaptive case, ringing is reduced and edges are better preserved.

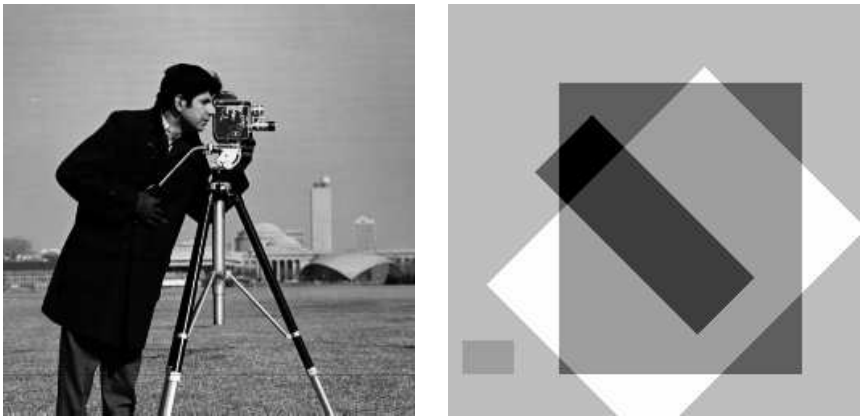

Figure 8: *Test images: 'Camera' and 'Rectangles'. Their size is of* 256×256 *pixels and have a bit-depth of 8 (i.e.,* 256 *possible gray levels).*

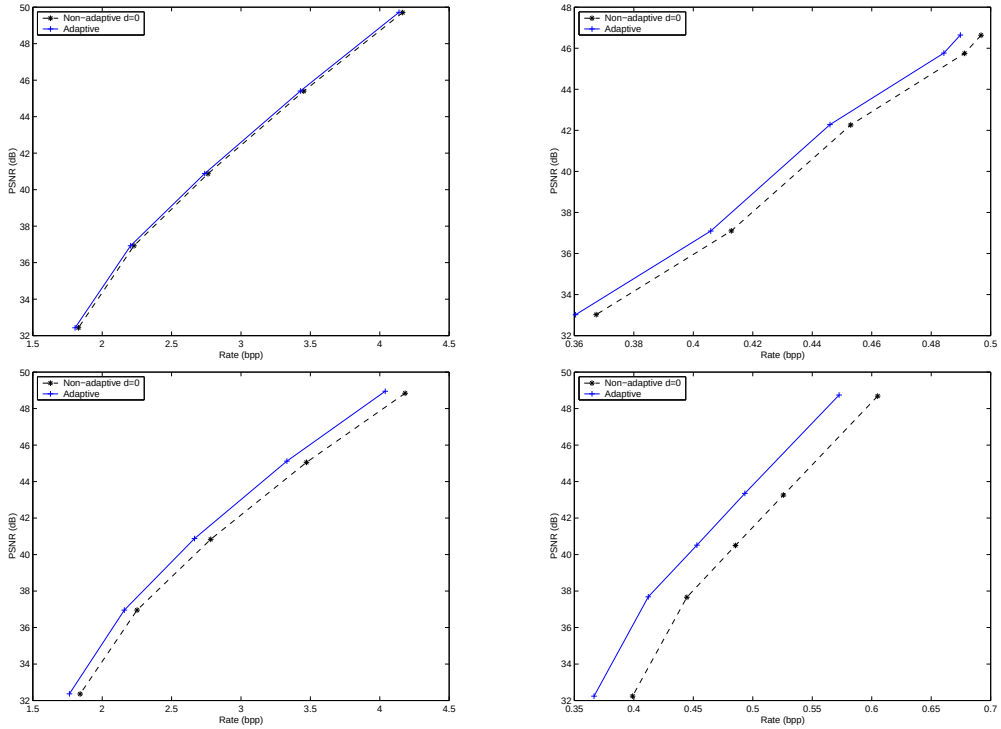

Figure 9: *Rate-distortion curves for 'Camera' (left) and 'Rectangles' (right) images in Fig. 8. Top: Case 1, the 2nd order derivative filter. Bottom: Case 2, the Laplacian derivative filter.*

7. Conclusions

In this paper, we have analysed the effect of a scalar uniform quantisation in an adaptive wavelet scheme based on a lifting implementation. The update lifting filter in this scheme is trigerred by the gradient vector, while the prediction step is fixed. After reviewing the

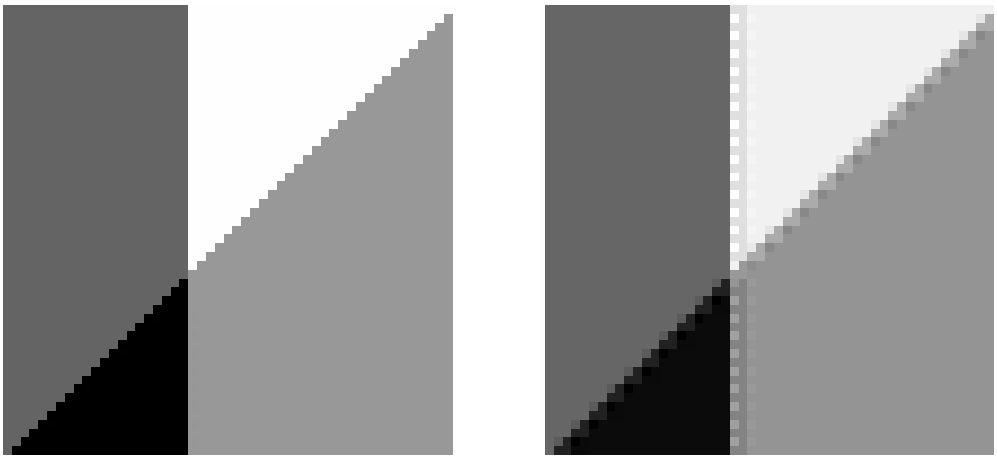

Figure 10: *Reconstructed images at* 0.5 *bpp for adaptive (left) and non-adaptive (right) schemes.*

theoretical framework involving operator seminorms that allowed us to provide the Threshold Criterion for perfect reconstruction in the scheme without quantisation, we derived sufficient conditions for recovering the decision map without errors. These conditions are similar in form to the original Threshold Criterion, and involve the quantisation step and the decision threshold at all decomposition levels. Furthermore, we have computed upper bounds for the reconstruction error at all levels in cases where the aformentioned conditions hold.

We have made explicit calculations in the particular case of a weighted gradient seminorm, and applied them to one-dimensional as well as two-dimensional signals. For this latter case, we have provided several examples of usual update operators and we have shown by simulation results the superiority of the proposed adaptive scheme compared with a non-adaptive one. Indeed, visual inspection of the reconstructed images reveals that ringing artefacts have been reduced and that edges have been reconstructed more sharply. Moreover, the rate-distortion curves exhibit 0.5 - 2 dB improvement in compression performance. This improvement depends on the update filter and is more significant for synthetic images.

As an additional benefit of our adaptive lifting scheme in the context of image compression we may mention the fact that it suggests a strong potential in relation to spatial scalability, compared with linear non-adaptive decompositions. Indeed, reduction of ringing artefacts and reconstruction of sharper edges automatically results in higher quality of the approximation subband at reduced resolution.

Future work aims at refining and optimising the error analysis presented in this paper and providing solutions for other seminorms of interest, such as the weighted quadratic seminorm.

References

1. Bruekers, F. A. M. L., and van den Enden, A. W. M. New networks for perfect inversion and perfect reconstruction. *IEEE Journal on Selected Areas in Communications 10* (1992), 130–137.

2. Candès, E. J. The curvelet transform for image denoising. In *Proceedings of the IEEE Conference on Image Processing* (Thessaloniki,Greece, October 2001).

3. Cha, H., and Chaparro, L. F. Adaptive morphological representation of signals: Polynomial and wavelet methods. *Multidimensional Systems and Signal Processing 8* (1997), 249–271.

4. Chan, T. F., and Zhou, H. M. ENO-Wavelet transforms for piecewise smooth functions. *SIAM Journal on Numerical Analysis 40*, 4 (2002), 1369–1404.

5. Claypoole, R. L., Baraniuk, R. G., and Nowak, R. D. Adaptive wavelet transforms via lifting. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing* (Seattle, Washington, May 12-15, 1998), pp. 1513–1516.

6. Claypoole, R. L., Baraniuk, R. G., and Nowak, R. D. Lifting construction of non-linear wavelet transforms. In *Proceedings of the IEEE-SP International Symposium on Time-Frequency and Time-Scale Analysis* (Pittsburgh, Pennsylvania, October 6-9, 1998), pp. 49–52.

7. Claypoole, R. L., Davis, G., Sweldens, W., and Baraniuk, R. G. Lifting for non-linear wavelet processing. In *Wavelet Applications in Signal and Image Processing VII* (Denver, Colorado, July 19-23 1999), pp. 372–383.

8. Claypoole, R. L., Davis, G. M., Sweldens, W., and Baraniuk, R. G. Nonlinear wavelet transforms for image coding via lifting. *IEEE Transactions on Image Processing 12*, 12 (December 2003), 1449–1459.

9. Cohen, A., and Matei, B. Compact representations of images by edge adapted multiscale transforms. In *Proceedings of the IEEE Conference on Image Processing* (Thessaloniki, Greece, 2001).

10. Combettes, P. L., and Pesquet, J.-C. Convex multiresolution analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence 20* (1998), 1308–1318.

11. de Queiroz, R. L., Florêncio, D. A. F., and Schafer, R. W. Nonexpansive pyramid for image coding using a nonlinear filterbank. *IEEE Transactions on Image Processing 7* (1998), 246–252.

12. Donoho, D. L. Orthonormal ridgelets and linear singularities. Technical report, Statis-

tics Department, Stanford University, California, 1998.

13. Donoho, D. L. Wedgelets: nearly minimax estimation of edges. *Annals of Statistics 27* (1999), 859–897.

14. Egger, O., Li, W., and Kunt, M. High compression image coding using an adaptive morphological subband decomposition. *Proceedings of the IEEE 83* (1995), 272–287.

15. Florêncio, D. A. F., and Schafer, R. W. A non-expansive pyramidal morphological image coder. In *Proceedings of the IEEE Conference on Image Processing* (Austin, Texas, 1994), pp. 331–335.

16. Gerek, O. N., and Çetin, A. E. Adaptive polyphase subband decomposition structures for image compression. *IEEE Transactions on Image Processing 9* (October 2000), 1649–1659.

17. Hampson, F. J., and Pesquet, J.-C. A nonlinear subband decomposition with perfect reconstruction. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing* (Atlanta, Georgia, May 7-10, 1996), pp. 1523–1526.

18. Hampson, F. J., and Pesquet, J.-C. M-band nonlinear subband decompositions with perfect reconstruction. *IEEE Transactions on Image Processing 7* (1998), 1547–1560.

19. Heijmans, H. J. A. M., and Goutsias, J. Nonlinear multiresolution signal decomposition schemes: Part II: Morphological wavelets. *IEEE Transactions on Image Processing 9*, 11 (2000), 1897–1913.

20. Heijmans, H. J. A. M., Pesquet-Popescu, B., and Piella, G. Building nonredundant adaptive wavelets by update lifting. Submitted to Applied and Computational Harmonic Analysis, 2003.

21. Le Pennec, E., and Mallat, S. Sparse geometric image representations with bandelets. Submitted to *IEEE-T-IP*, 2003.

22. Piella, G., and Heijmans, H. J. A. M. Adaptive lifting schemes with perfect reconstruction. *IEEE Transactions on Signal Processing 50*, 7 (2002), 1620–1630.

23. Sweldens, W. The lifting scheme: A new philosophy in biorthogonal wavelet constructions. In *Wavelet Applications in Signal and Image Processing III* (1995), A. F. Lain and M. Unser, Eds., Proceedings of SPIE, vol. 2569, pp. 68–79.

24. Trappe, W., and Liu, K. Adaptivity in the lifting scheme. In *33rd Conference on Information Sciences and Systems* (Baltimore, March 1999), pp. 950–955.