

RESEARCH

Open Access

# Two-dimensional integer wavelet transform with reduced influence of rounding operations

Tilo Strutz\* and Ines Rennert

## Abstract

If a system for lossless compression of images applies a decorrelation step, this step must map integer input values to integer output values. This can be achieved, for example, using the integer wavelet transform (IWT). The non-linearity, introduced by the obligatory rounding steps, is the main drawback of the IWT, since it deteriorates the desired filter characteristic.

This paper discusses different methods for reducing the influence of rounding in 5/3 and 9/7 filter banks. A novel combination of two-dimensional implementations of the JPEG2000 9/7 filter bank with new filter coefficients is proposed and the effects of the methods on lossless image compression are investigated. In addition, these filter banks are compared to the 9/7 Deslauriers-Dubuc filter bank (97DD).

The analysed two-dimensional implementations generally perform better than their one-dimensional counterparts in terms of compression ratio for natural images. On average, the 2D 97DD filter bank performs best. In addition, it has been found that the compression results cannot be improved by simply reducing the number of lifting steps via 2D implementations of the JPEG2000 9/7 filter bank. Only the 2D implementation with a minimum number of lifting steps, in combination with modified lifting coefficients, leads to fewer bits per pixel than the separable implementation on average for a selected set of images.

**Keywords:** integer wavelet transform; filter bank; 2D lifting; image compression; rounding

## 1 Introduction

Efficient systems for the lossless compression of image data require a decorrelation step which maps the integer input samples to integer output values. In wavelet-based compression systems (see [1] for an overview), this is achieved by using the lifting implementation of a discrete wavelet transform (DWT) [2] in combination with the rounding of intermediate computation results, which is called integer wavelet transform (IWT) [3]. Beginning with initial investigations on the IWT, which were also motivated by the standardization of the new image compression system JPEG2000 [4], and its application in the JPEG2000 framework [5], this topic has received growing attention. The idea of integer transforms relates back to the so-called S

transform [6], the improved version called S + P transform [7], and the reversible TS transform [8]. Since then, several integer wavelet transforms have been analysed in terms of their performance in image compression systems [9].

Wavelet filter banks are typically designed without taking the integer-to-integer mapping into account. The conversion into an integer wavelet transform requires rounding steps which introduce non-linear effects, which deteriorate the desired filter properties. That is one of the major reasons why JPEG2000 (Part 1) uses the simple 5/3 filter bank, which can be realised with a minimum of lifting steps, *i.e.* a minimum of rounding operators. In addition, this filter bank has favourable coefficients of  $-0.5$  in its first lifting step, leading to rounding errors in only 50% of all cases. The 9/7 filter bank, in contrast, requires irrational filter coefficients, making it per se less suitable for integer-to-integer mapping.

Besides the degradation of the decorrelation performance, the non-linearity of rounding makes the two-

\* Correspondence: strutz@hft-leipzig.de  
Deutsche Telekom, Hochschule für Telekommunikation Leipzig, Institute of Communications Engineering, Gustav-Freytag-Str. 43-45, Leipzig, 04277, Germany

dimensional application pseudo non-separable, because the result of a 2D transform is dependent on the order of row and column decomposition.

The effects of rounding have been investigated in [10] with the assumption that rounding errors are similar to quantization errors and can be modelled by additive noise. In [11], the optimizations of filters for a lossy-to-lossless framework and its implementation in hardware are discussed. Another issue of interest is to make the IWT adaptive to the statistics of the image to be processed. This can be achieved by either switching between filters of different length [12–14] or by optimising the lifting steps [15–19]. Many papers are dedicated to the approximation of the standard 9/7 filter coefficients by rational values for low-complexity hardware or software implementations [20–22]. More recent research has also addressed the problem of reducing the adverse effect of rounding operations in terms of designing new filters with more favourable lifting coefficients [23] or modifications of the signal flow, reducing the number of lifting steps [24, 25]. The construction of three-channel filter banks based on the lifting scheme has also been considered for lossless image compression [26].

This paper investigates methods for reducing the influence of rounding in terms of reduced number of lifting steps and new filter coefficients for the two-dimensional JPEG2000 9/7 filter bank. The derived processing structures are evaluated with respect to compression efficiency and complexity. In addition, the results are compared to the one- and two-dimensional implementation of the LeGall 5/3 filter bank [27] and the 9/7 Deslauriers-Dubuc filter bank [3, 28–30], which both can be implemented with only two lifting steps.

The paper is organised as follows: first, Section 2 reviews the reversible 5/3 filter bank used in JPEG2000, discusses the handling of signal boundaries, and explains how the 2D implementation reduces the number of rounding steps. Section 3 is dedicated to the JPEG2000 9/7 filter bank. It explains the separable 2D signal flow and afterwards describes how the number of lifting steps can be decreased by rearranging the processing steps. Section 4 discusses a Deslauriers-Dubuc filter bank (97DD), which can be implemented with the same 2D structure as the 5/3 filter bank, while having filter lengths of nine and seven. In Section 5, the design of ‘rounding-friendly’ filters is addressed. Section 6 investigates all processing structures by means of sub-band entropy, compression performance, and complexity. Section 7 concludes the paper.

## 2 The 5/3 filter bank

### 2.1 One-dimensional decomposition

The biorthogonal 5/3 two-channel filter bank is based on following analysis filters [27]

$$(h_1[n]) = \begin{pmatrix} -1/2 & 1 & -1/2 \end{pmatrix},$$

$$(h_0[n]) = \begin{pmatrix} -1/8 & 1/4 & 3/4 & 1/4 & -1/8 \end{pmatrix}. \quad (1)$$

$(h_1[n])$  and  $(h_0[n])$  are the impulse responses of the analysis high-pass and analysis low-pass filters, respectively. This filter bank can be easily implemented *via* the lifting scheme [2].

Both filters have two vanishing moments each, *i.e.* they satisfy

$$\begin{aligned} 0 &= \sum_n h_1[n] \cdot n^p, \\ 0 &= \sum_n h_0[n] \cdot n^p \cdot (-1)^n \end{aligned} \quad \text{for } p \in \{0; 1\}. \quad (2)$$

Let  $x_n$  ( $n = 0, 1, 2, \dots$ ) be the samples of an integer-valued input signal. The filtering according to the lifting scheme computes, in a primal lifting step, the high-pass filter output  $d_n$  (detail signal)

$$d_n = x_{2n+1} + \lfloor \alpha \cdot (x_{2n} + x_{2n+2}) + 0.5 \rfloor \quad (3)$$

using the lifting coefficient  $\alpha$ . A dual lifting step using the coefficient  $\beta$  produces the low-pass filter output  $a_n$  (approximation signal)

$$a_n = x_{2n} + \lfloor \beta \cdot (d_{n-1} + d_n) + 0.5 \rfloor \quad (4)$$

based on the detail signal  $d_n$  and the original signal values  $x_n$ . The property of integer-to-integer mapping, which is essential for lossless compression, is imposed simply by properly rounding the intermediate values to integer values [3].

The application of the lifting steps in reverse order and using subtraction instead of summation reconstructs the original signal  $x_n$

$$x_{2n} = a_n - \lfloor \beta \cdot (d_{n-1} + d_n) + 0.5 \rfloor, \quad (5)$$

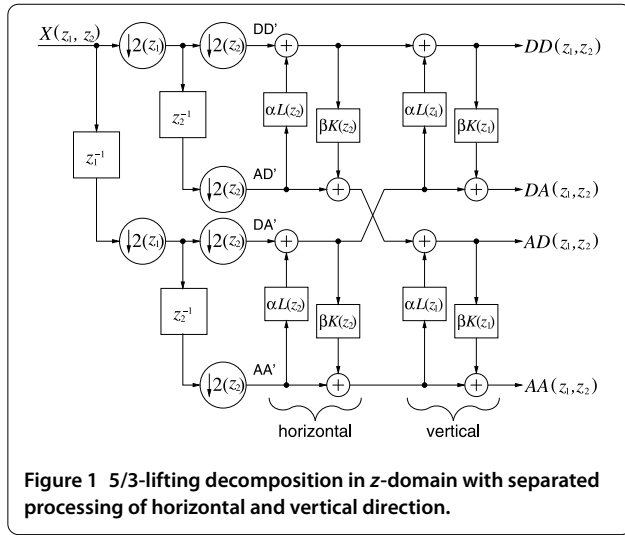
$$x_{2n+1} = d_n - \lfloor \alpha \cdot (x_{2n} + x_{2n+2}) + 0.5 \rfloor. \quad (6)$$

Setting the lifting coefficients equal to  $\alpha = -0.5$  and  $\beta = 0.25$ , the lifting scheme (without rounding) performs the same operations as the filters in equation (1) of the conventional filter bank.

### 2.2 Two-dimensional decomposition

The impulse responses of the two-dimensional filters are the products of the one-dimensional filters

$$\begin{aligned} (h_{11}[n]) &= (h_1[n])^T \cdot (h_1[n]) \\ &= \begin{pmatrix} 1/4 & -1/2 & 1/4 \\ -1/2 & 1 & -1/2 \\ 1/4 & -1/2 & 1/4 \end{pmatrix}, \end{aligned} \quad (7)$$

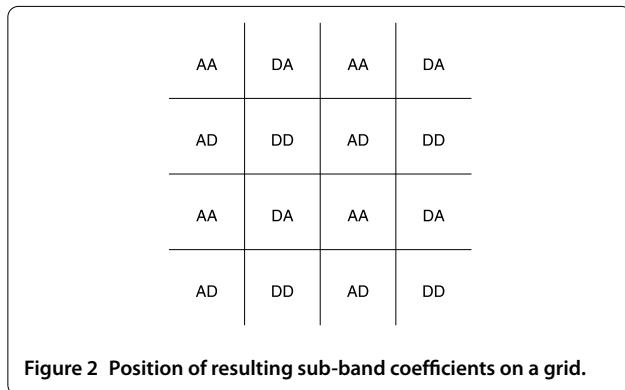


$$\begin{aligned} (h_{10}[n]) &= (h_1[n])^T \cdot (h_0[n]) \\ &= \frac{1}{16} \cdot \begin{pmatrix} 1 & -2 & -6 & -2 & 1 \\ -2 & 4 & 12 & 4 & -2 \\ 1 & -2 & -6 & -2 & 1 \end{pmatrix}, \end{aligned} \quad (8)$$

$$(h_{01}[n]) = (h_0[n])^T \cdot (h_1[n]) = (h_{10}[n])^T, \quad (9)$$

$$\begin{aligned} (h_{00}[n]) &= (h_0[n])^T \cdot (h_0[n]) \\ &= \frac{1}{64} \cdot \begin{pmatrix} 1 & -2 & -6 & -2 & 1 \\ -2 & 4 & 12 & 4 & -2 \\ -6 & 12 & 36 & 12 & -6 \\ -2 & 4 & 12 & 4 & -2 \\ 1 & -2 & -6 & -2 & 1 \end{pmatrix}. \end{aligned} \quad (10)$$

Based on the polyphase description in Figure 1, the lifting coefficients of the 2D implementation can be derived. The variable  $z_1$  corresponds to the vertical and  $z_2$  to the horizontal direction. The annotations DD, AD, DA, and AA are related to the positions in Figure 2. The letters A and D denote the 1D approximation signal (low-pass band) and the detail signal (high-pass band), respectively. In 2D,



they appear in pairs. DA, for example, denotes the positions of 2D sub-band coefficients derived from horizontal low-pass filtering and vertical high-pass filtering, *i.e.* using the two-dimensional filter  $(h_{10}[n])$ . The lifting filters are  $L(z_1) = (1 + z_1)$ ,  $L(z_2) = (1 + z_2)$ ,  $K(z_1) = (1 + z_1^{-1})$ , and  $K(z_2) = (1 + z_2^{-1})$ .

The actual number of rounding steps can be determined based on Figure 1. Rounding is required at each lifting step before the summation, *i.e.* eight times in total. The rounding steps are interdependent, however. The horizontal dual steps, for instance, process values from DD and DA positions, which have already been affected by the rounding in previous primal steps. This possible accumulation of rounding errors continues in the second dimension of the image signal.

Let  $X$  and  $Y$  be the polyphase input and output vectors

$$X = \begin{pmatrix} DD' \\ AD' \\ DA' \\ AA' \end{pmatrix}, \quad Y = \begin{pmatrix} DD \\ AD \\ DA \\ AA \end{pmatrix}. \quad (11)$$

Then, the first two lifting steps can be denoted as

$$\begin{aligned} L_1 &= \begin{pmatrix} 1 & a_2 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & a_2 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \\ L_2 &= \begin{pmatrix} 1 & 0 & 0 & 0 \\ b_2 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & b_2 & 1 \end{pmatrix} \end{aligned} \quad (12)$$

with  $a_2 = \alpha \cdot L(z_2)$  and  $b_2 = \beta \cdot K(z_2)$ . After exchanging the orientation using the permutation matrix

$$P = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad (13)$$

another two lifting steps follow

$$\begin{aligned} L_3 &= \begin{pmatrix} 1 & a_1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & a_1 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \\ L_4 &= \begin{pmatrix} 1 & 0 & 0 & 0 \\ b_1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & b_1 & 1 \end{pmatrix} \end{aligned} \quad (14)$$

$$\mathbf{Y} = \begin{pmatrix} 1 & a_2 & a_1 & a_2 \cdot a_1 \\ b_2 & a_2 \cdot b_2 + 1 & a_1 \cdot b_2 & a_1 \cdot (a_2 \cdot b_2 + 1) \\ b_1 & a_2 \cdot b_1 & a_1 \cdot b_1 + 1 & a_2 \cdot (a_1 \cdot b_1 + 1) \\ b_1 b_2 & b_1 \cdot (a_2 \cdot b_2 + 1) & b_2 \cdot (a_1 \cdot b_1 + 1) & (a_1 \cdot b_1 + 1) \cdot (a_2 \cdot b_2 + 1) \end{pmatrix} \cdot \mathbf{X} \quad (16)$$

**Figure 3 Processing of 5/3-lifting decomposition.** See text for details.

with  $a_1 = \alpha \cdot L(z_1)$  and  $b_1 = \beta \cdot K(z_1)$ . The entire processing is

$$\mathbf{Y} = (\mathbf{P} \cdot \mathbf{L}_4 \cdot \mathbf{L}_3) \cdot (\mathbf{P} \cdot \mathbf{L}_2 \cdot \mathbf{L}_1) \cdot \mathbf{X}, \quad (15)$$

leading to equation (16) in Figure 3.

In Figure 1, the single lifting steps are annotated with either  $\alpha$  or  $\beta$ , expressing which of the two equations (3) or (4) is used (in one dimension). The 2D-lifting computation is derived from this signal flow as follows. If  $X(z_1, z_2)$  corresponds to  $x_{n,m}$ , then we get the following equivalences:

$$\begin{aligned} \text{AD}' &\bullet \text{---} \circ x_{2n,2m-1}, & \text{DD}' &\bullet \text{---} \circ x_{2n,2m}, \\ \text{DA}' &\bullet \text{---} \circ x_{2n-1,2m}, & \text{AA}' &\bullet \text{---} \circ x_{2n-1,2m-1}. \end{aligned}$$

With respect to Figure 2, the sub-band signals

$$\begin{aligned} \text{DD}(z_1, z_2) &\bullet \text{---} \circ dd_{2n,2m}, & \text{AD}(z_1, z_2) &\bullet \text{---} \circ ad_{2n,2m-1}, \\ \text{DA}(z_1, z_2) &\bullet \text{---} \circ da_{2n-1,2m}, & \text{AA}(z_1, z_2) &\bullet \text{---} \circ aa_{2n-1,2m-1} \end{aligned}$$

are accordingly computed as follows. The two-dimensional detail signal is computed with

$$\begin{aligned} dd_{2n,2m} &= x_{2n,2m} + \alpha \cdot (x_{2n,2m-1} + x_{2n,2m+1}) \\ &+ \alpha \cdot (x_{2n-1,2m} + x_{2n+1,2m}) \\ &+ \alpha^2 \cdot (x_{2n-1,2m-1} + x_{2n+1,2m-1} \\ &+ x_{2n-1,2m+1} + x_{2n+1,2m+1}). \end{aligned} \quad (17)$$

According to the processing structure of the lifting scheme, the signal values  $x_{2n,2m}$  are now overwritten by  $dd_{2n,2m}$  and are not available anymore. Therefore, the next lifting step is

$$\begin{aligned} da_{2n-1,2m} &= x_{2n-1,2m} + \alpha \cdot (x_{2n-1,2m-1} + x_{2n-1,2m+1}) \\ &+ \beta \cdot (dd_{2n-2,2m} + dd_{2n,2m}). \end{aligned} \quad (18)$$

Since we know that the filter ( $h_{01}[n]$ ) is the transpose of ( $h_{10}[n]$ ), the computation of the AD path must analogously

be equal to

$$\begin{aligned} ad_{2n,2m-1} &= x_{2n,2m-1} + \alpha \cdot (x_{2n-1,2m-1} + x_{2n+1,2m-1}) \\ &+ \beta \cdot (dd_{2n,2m-2} + dd_{2n,2m}). \end{aligned} \quad (19)$$

Based on these three relations we get

$$\begin{aligned} aa_{2n-1,2m-1} &= x_{2n-1,2m-1} \\ &+ \beta \cdot (ad_{2n-2,2m-1} + ad_{2n,2m-1}) \\ &+ \beta \cdot (da_{2n-1,2m-2} + da_{2n-1,2m}) \\ &- \beta^2 \cdot (dd_{2n-2,2m-2} + dd_{2n-2,2m} \\ &+ dd_{2n,2m-2} + dd_{2n,2m}). \end{aligned} \quad (20)$$

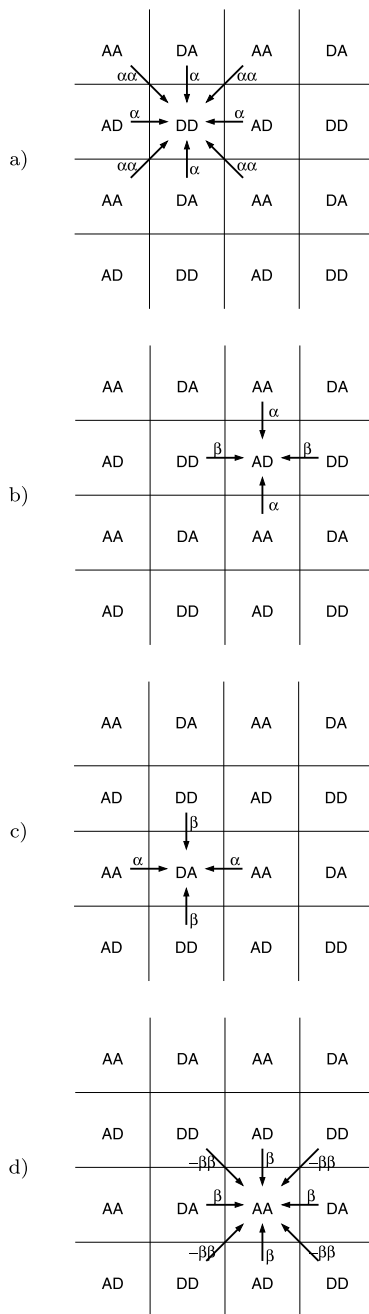
Consequently, the two-dimensional filters  $h_{11} \cdots h_{00}$  can be implemented directly using a 2D lifting structure. Figure 4 illustrates the processing steps.

Based on equations (17)-(20), it becomes obvious that only one single rounding step is required for each sub-band value in the 2D implementation. And, in fact, the scheme in Figure 1 can be modified accordingly, reducing the total number of lifting steps from eight to four (Figure 5). This corresponds to a factorization of equation (16) in merely three matrices, since signals AD and DA can be computed concurrently

$$\mathbf{Y} = \mathbf{L}_\beta \cdot \mathbf{L}_{\alpha\beta} \cdot \mathbf{L}_\alpha \cdot \mathbf{X} \quad (21)$$

with

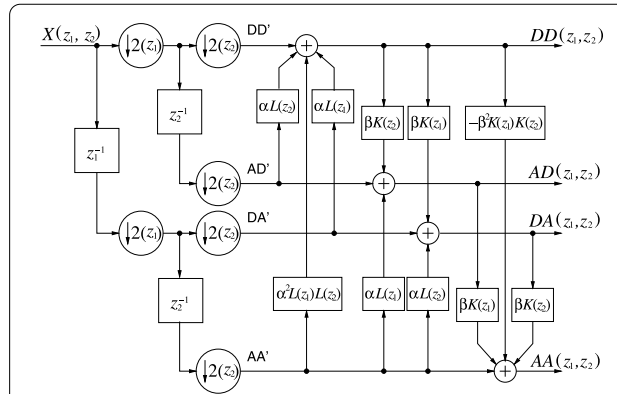
$$\begin{aligned} \mathbf{L}_\alpha &= \begin{pmatrix} 1 & a_2 & a_1 & a_1 a_2 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \\ \mathbf{L}_{\alpha\beta} &= \begin{pmatrix} 1 & 0 & 0 & 0 \\ b_2 & 1 & 0 & a_1 \\ b_1 & 0 & 1 & a_2 \\ 0 & 0 & 0 & 1 \end{pmatrix} \end{aligned} \quad (22)$$



**Figure 4 General two-dimensional signal flow for 5/3 lifting: (a)-(d) first, second, third, and fourth lifting step.**

and

$$\mathbf{L}_\beta = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -b_1b_2 & b_1 & b_2 & 1 \end{pmatrix}. \quad (23)$$



**Figure 5 Two-dimensional 5/3-lifting decomposition in z-domain with reduced number of lifting steps.**

Section 6 investigates the influence of the reduced number of rounding steps on the performance of an IWT-based compression system.

### 3 JPEG2000 9/7 filter bank

#### 3.1 One-dimensional decomposition

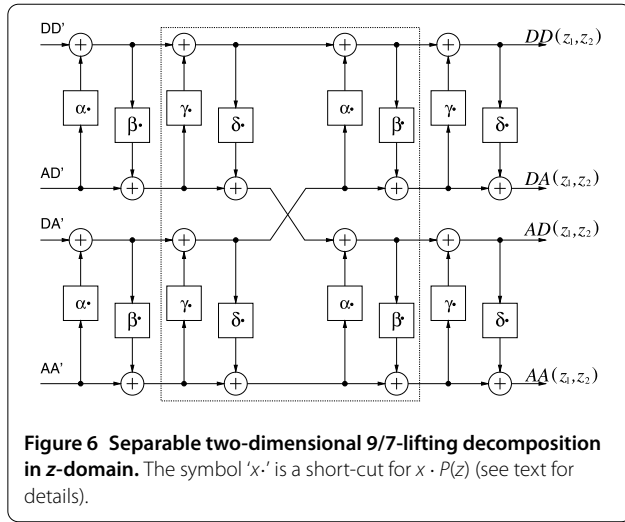
The standard 9/7 filter bank can be implemented based on the lifting scheme in the same manner as the 5/3 filter bank shown in the previous section. The only difference is that, in total, four lifting steps are needed in each direction ([23], Figure 2).

In literature, the factorization of a scaling factor is also discussed. We would like to remark that in the case of lossless compression, this scaling is not necessary and would merely introduce more rounding operations degrading the performance of the wavelet transform. In lossy compression, the scaling can be shifted into the quantization step, thus, there is no reason to treat the scaling within the transformation stage.

Further, it must be pointed out that for the practical implementation and in order to avoid an exception handling at the signal boundaries, it is sufficient to extend the original or intermediate signals by a single value at both boundaries.

In application to image compression, the typical filter design aims to achieve maximal flat magnitude responses at frequency  $f = 0$  (high-pass filter) and at half of the sampling frequency (low-pass filter), i.e. a maximum number of vanishing moments is desired [31]. The 9/7 filters allow for 4 vanishing moments each, thus the equations (2) hold true for  $p \in \{0; 1; 2; 3\}$ . These constraints lead to the following lifting coefficients, either *via* factorisation of the two-channel filter bank [32] or *via* direct filter design based on the lifting structure [23]

$$\begin{aligned} \alpha &\approx -1.58613434206, & \gamma &\approx 0.88291107553, \\ \beta &\approx -0.05298011857, & \delta &\approx 0.44350685204. \end{aligned} \quad (24)$$



### 3.2 Two-dimensional decomposition

The two-dimensional signal flow can be described in similar manner as the 2D 5/3 filter bank. Figure 6 depicts the separable transformation without rounding steps. The annotation of primal and dual lifting steps with  $\alpha$ ,  $\gamma$ ,  $\beta$ , and  $\delta$ , relates to the subsequent lifting steps shown in [23], Figure 2. For the sake of visualisation, the annotation with  $L(z_1)$ ,  $L(z_2)$ ,  $K(z_1)$ , and  $K(z_2)$  is omitted. Its influence should be clear by now. The coefficients  $\alpha$  and  $\gamma$  are associated with  $L(z_k)$ ,  $\beta$  and  $\delta$  with  $K(z_k)$ . The processing can be denoted as

$$Y = (P \cdot L_8 \cdot L_7 \cdot L_4 \cdot L_3) \cdot (P \cdot L_6 \cdot L_5 \cdot L_2 \cdot L_1) \cdot X. \quad (25)$$

The four additional lifting  $L_5 \dots L_8$  steps are identical to  $L_1 \dots L_4$  in their structure, only the lifting coefficients are exchanged from  $\alpha$  to  $\gamma$  and  $\beta$  to  $\delta$ .

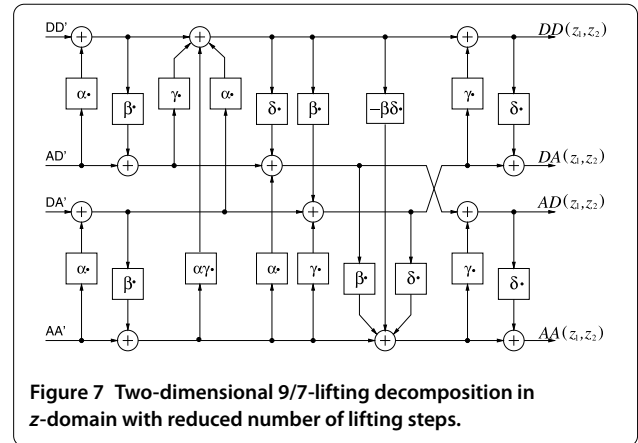
The inner eight lifting steps encircled by the dotted line have the same structure as the 5/3 decomposition in Figure 1

$$Y = (PL_8L_7P) \cdot (PL_4L_3PL_6L_5) \cdot (L_2L_1) \cdot X. \quad (26)$$

In [24], it is suggested to substitute them, using the lifting structure of Figure 5, facilitating the reduction of rounding steps. Figure 7 shows the result. As can be seen, the first two lifting steps in the horizontal direction, as well as the last two steps in the vertical direction, remained untouched. The number of rounding operations has been reduced from four (Figure 6) to only three per sub-band sample.

It is, however, possible to reduce the number of rounding steps further by reordering the matrices

$$Y = (PL_6L_5PL_8L_7) \cdot (PL_4L_3PL_2L_1) \cdot X. \quad (27)$$



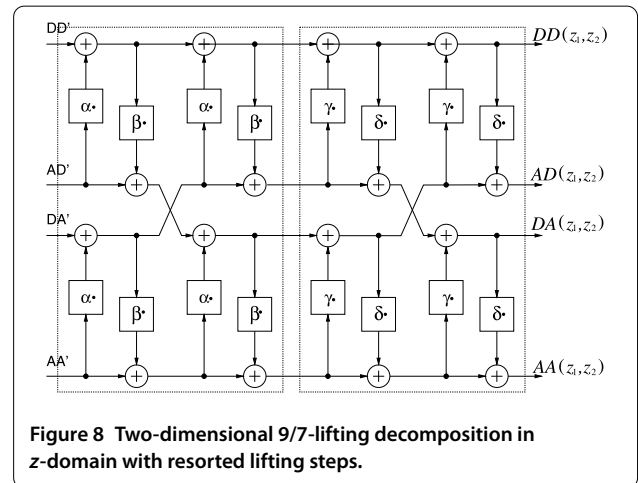
The third and fourth lifting step of the horizontal decomposition of Figure 7 can be moved to the end of the processing pipeline (Figure 8). Now, both parts encircled by the dotted lines can be substituted using the lifting structure of Figure 5. The corresponding processing is

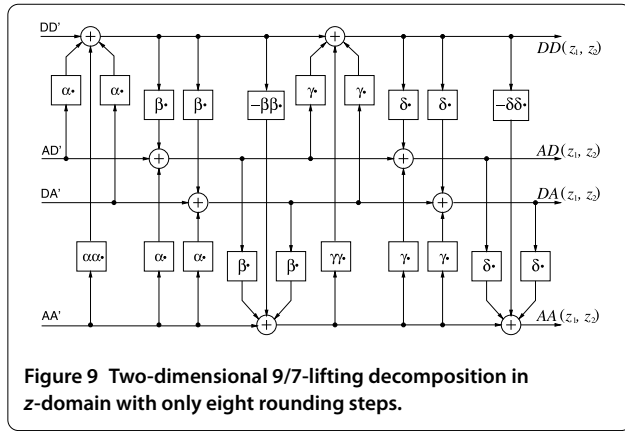
$$Y = (L_\delta L_\gamma L_\delta L_\gamma) \cdot (L_\beta L_\alpha L_\beta L_\alpha) \cdot X \quad (28)$$

with

$$L_\gamma = \begin{pmatrix} 1 & g_2 & g_1 & g_1 g_2 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad (29)$$

$$L_{\gamma\delta} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ d_2 & 1 & 0 & g_1 \\ d_1 & 0 & 1 & g_2 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$





and

$$\mathbf{L}_\delta = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -d_1 d_2 & d_1 & d_2 & 1 \end{pmatrix}, \quad (30)$$

with  $g_1 = \gamma \cdot L(z_1)$ ,  $g_2 = \gamma \cdot L(z_2)$ ,  $d_1 = \delta \cdot K(z_1)$  and  $d_2 = \delta \cdot K(z_2)$ .

The new decomposition in Figure 9 requires even fewer lifting steps and is still fully compatible with the conventional 9/7 filter bank (Figure 6), when the rounding steps are left out. Now, only two steps per sub-band sample are required. The number of consecutive processing steps is even reduced from eight (Figure 6) to six, since the AD and DA channels can be computed at the same time. Iwahashi and Kiya developed this structure independently with the focus on the reduced latency of this 2D decomposition [25].

#### 4 Deslauriers-Dubuc 9/7 filter bank

In addition to the filter banks defined in the JPEG2000 standard part I, the Deslauriers-Dubuc 9/7 filter bank (97DD) is of interest in this context [3, 28, 29]. Its high-pass filter has four vanishing moments and the low-pass filter two. As the name suggests, the filters have 9 and 7 taps, respectively

$$\begin{aligned} (h_1[n]) &= \left( \frac{1}{16} \quad 0 \quad -\frac{9}{16} \quad 1 \quad -\frac{9}{16} \quad 0 \quad \frac{1}{16} \right), \\ (h_0[n]) &= \left( \frac{1}{64} \quad 0 \quad -\frac{1}{8} \quad \frac{1}{4} \quad \frac{23}{32} \right. \\ &\quad \left. \frac{1}{4} \quad -\frac{1}{8} \quad 0 \quad \frac{1}{64} \right). \end{aligned} \quad (31)$$

The filter bank, however, can be implemented with only two lifting steps, similar to the 5/3 filter bank. In integer

arithmetic, the lifting steps are

$$\begin{aligned} d_n &= x_{2n+1} \\ &+ \left( (x_{2n-2} - 9 \cdot (x_{2n} + x_{2n+2}) \right. \\ &\quad \left. + x_{2n+4} + 8) \gg 4 \right), \end{aligned} \quad (32)$$

$$a_n = x_{2n} + (x_{2n} + x_{2n+2} + 2) \gg 2). \quad (33)$$

This makes the structure in Figure 5 applicable for the 97DD filter bank as well. The only difference concerns the following lifting polynomials:  $L(z_1) = (z_1^{-1} - 9 - 9 \cdot z_1 + z_1^2)$  and  $L(z_2) = (z_2^{-1} - 9 - 9 \cdot z_2 + z_2^2)$  both in combination with  $\alpha = 1/16$ . The other lifting steps remain  $K(z_1) = (1 + z_1^{-1})$  and  $K(z_2) = (1 + z_2^{-1})$  with  $\beta = 1/4$ .

The 2D lifting steps of the 97DD filter bank are implemented as

$$\begin{aligned} dd_{2n,2m} &= x_{2n,2m} \\ &+ \left\{ \left[ (x_{2n,2m-3} + x_{2n,2m+3} \right. \right. \\ &\quad \left. \left. + x_{2n-3,2m} + x_{2n+3,2m}) \ll 4 \right) \right. \\ &\quad \left. + (x_{2n-1,2m-1} + x_{2n-1,2m+1} \right. \\ &\quad \left. + x_{2n+1,2m-1} + x_{2n+1,2m+1}) \cdot 81 \right. \\ &\quad \left. - \left( (x_{2n-1,2m} + x_{2n+1,2m} \right. \right. \\ &\quad \left. \left. + x_{2n,2m-1} + x_{2n,2m+1}) \ll 4 \right) \right. \\ &\quad \left. + x_{2n-3,2m-1} + x_{2n-3,2m+1} \right. \\ &\quad \left. + x_{2n-1,2m-3} + x_{2n-1,2m+3} \right. \\ &\quad \left. + x_{2n+1,2m-3} + x_{2n+1,2m+3} \right. \\ &\quad \left. + x_{2n+3,2m-1} + x_{2n+3,2m+1}) \cdot 9 \right. \\ &\quad \left. + x_{2n-3,2m-3} + x_{2n-3,2m+3} \right. \\ &\quad \left. + x_{2n+3,2m-3} + x_{2n+3,2m+3} \right. \\ &\quad \left. + 128 \right] \gg 8 \}, \end{aligned} \quad (34)$$

$$\begin{aligned} ad_{2n,2m-1} &= x_{2n,2m-1} \\ &+ \left\{ \left[ (dd_{2n,2m-2} + dd_{2n,2m}) \ll 2 \right) \right. \\ &\quad \left. + x_{2n-3,2m-1} \right. \\ &\quad \left. - 9 \cdot (x_{2n-1,2m-1} + x_{2n+1,2m-1}) \right. \\ &\quad \left. + x_{2n+3,2m-1} + 8 \right] \gg 4 \}, \end{aligned} \quad (35)$$

$$\begin{aligned} da_{2n-1,2m} &= x_{2n-1,2m} \\ &+ \left\{ \left[ (dd_{2n-2,2m} + dd_{2n,2m}) \ll 2 \right) \right. \\ &\quad \left. + x_{2n-1,2m-3} \right. \\ &\quad \left. - 9 \cdot (x_{2n-1,2m-1} + x_{2n-1,2m+1}) \right. \\ &\quad \left. + x_{2n-1,2m+3} + 8 \right] \gg 4 \}, \end{aligned} \quad (36)$$

$$\begin{aligned}
 aa_{2n-1,2m-1} &= x_{2n-1,2m-1} \\
 &+ \{ [ ((ad_{2n-2,2m-1} + ad_{2n,2m-1} \\
 &+ da_{2n-1,2m-2} + da_{2n-1,2m}) \ll 2) \\
 &- (dd_{2n-2,2m-2} + dd_{2n-2,2m} \\
 &+ dd_{2n,2m-2} + dd_{2n,2m}) + 8 ] \gg 4 \}. \quad (37)
 \end{aligned}$$

## 5 'Rounding-friendly' lifting coefficients

The superiority of the separable 5/3 filters over the separable standard 9/7 IWT filter bank, documented in literature [3, 9, 23], has its reason in the minimal accumulated influence of rounding at each lifting step, leading to a minimal change in the filter characteristics. Not only is the number of steps lower (only two instead of four in each direction), but the rounding error in its first lifting step is also smaller, since the factor of  $\alpha = -1/2$  only results in errors when the sum of  $x_{2m}$  and  $x_{2m+2}$  is odd (see eq. (3)). The degradation of the magnitude response of the standard 9/7 filter is distinctly higher.

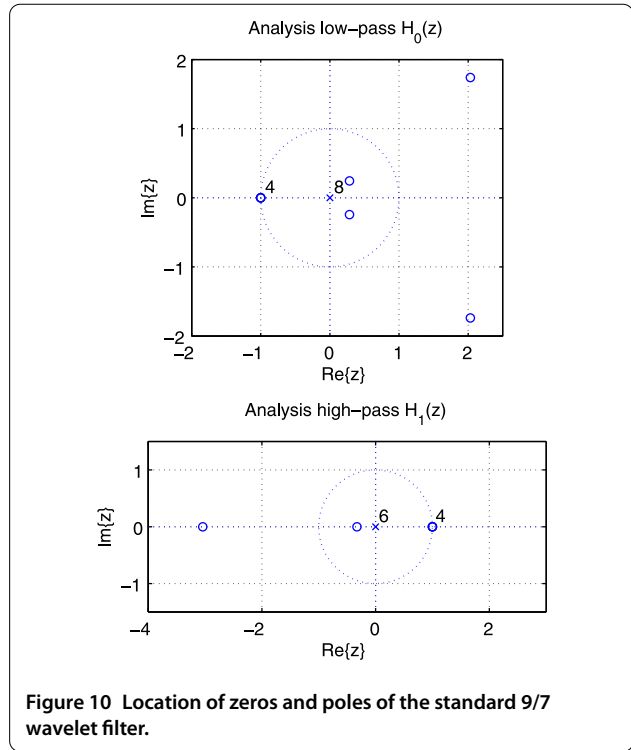
With this degradation in mind, we investigated whether some of the filter-design constraints leading to a maximum number of vanishing moments can be released, to obtain lifting coefficients which introduce fewer rounding errors, while keeping the essential characteristic of the magnitude responses.

Introducing a maximal number of vanishing moments as realized with equation (2) corresponds to a maximum number of zeros at  $z = -1$  for the analysis low-pass filter  $H_0(z)$  and at  $z = 1$  for the analysis high-pass filter  $H_1(z)$  (Figure 10). The corresponding lifting coefficients have already been shown in equation (24).

In general, the relaxation of these filter constraints leads to poorer decorrelation of the input image. If, however, the adverse effects of rounding decrease due to the modification of the lifting coefficients, then there might be an optimal compromise. Reducing the number of vanishing moments from four to two increases the degree of freedom in selecting lifting coefficients with suitable properties.

Inspecting the flow chart in Figure 6, it becomes obvious that the first lifting step has the greatest influence on the non-linear filter characteristic. All rounding errors introduced at this point are likely to propagate through the entire lifting cascade. Consequently, it is desirable to find a set of lifting coefficients leading to similar magnitude responses as the original coefficients in eq. (24), but with  $\alpha$  equal to  $-1$  avoiding any rounding error in the first lifting step. A promising set of lifting coefficients was found in [23] with

$$\begin{aligned}
 \alpha &= -1, & \beta &= \frac{2 - \sqrt{7}}{6}, \\
 \gamma &= \frac{\sqrt{7} - 1}{4}, & \delta &= \frac{1}{2}.
 \end{aligned} \quad (38)$$



**Figure 10** Location of zeros and poles of the standard 9/7 wavelet filter.

The approximation to

$$\alpha = -1, \quad \beta = -\frac{7}{64}, \quad \gamma = \frac{105}{256}, \quad \delta = \frac{1}{2} \quad (39)$$

allows a division-free implementation in integer arithmetic as follows

$$\begin{aligned}
 dd'_{2n,2m} &= x_{2n,2m} \\
 &- (x_{2n-1,2m} + x_{2n+1,2m} + x_{2n,2m-1} + x_{2n,2m+1}) \\
 &+ (x_{2n-1,2m-1} + x_{2n-1,2m+1} \\
 &+ x_{2n+1,2m-1} + x_{2n+1,2m+1}) \quad (40)
 \end{aligned}$$

$$\begin{aligned}
 ad'_{2n,2m-1} &= x_{2n,2m-1} \\
 &- (x_{2n-1,2m-1} + x_{2n+1,2m-1}) \\
 &- \{ [7 \cdot (dd'_{2n,2m-2} + dd'_{2n,2m}) + 32] \gg 6 \} \quad (41)
 \end{aligned}$$

$$\begin{aligned}
 da'_{2n-1,2m} &= x_{2n-1,2m} \\
 &- (x_{2n-1,2m-1} + x_{2n-1,2m+1}) \\
 &- \{ [7 \cdot (dd'_{2n-2,2m} + dd'_{2n,2m}) + 32] \gg 6 \} \quad (42)
 \end{aligned}$$



$$\begin{aligned}
 aa'_{2n-1,2m-1} &= x_{2n-1,2m-1} \\
 &- \{ [448 \cdot (ad'_{2n-2,2m-1} + ad'_{2n,2m-1} \\
 &+ da'_{2n-1,2m-2} + da'_{2n-1,2m}) \\
 &+ 49 \cdot (dd'_{2n-2,2m-2} + dd'_{2n-2,2m} \\
 &+ dd'_{2n,2m-2} + dd'_{2n,2m}) + 2,048] \gg 12 \}, \quad (43)
 \end{aligned}$$

$$\begin{aligned}
 dd_{2n,2m} &= dd'_{2n,2m} \\
 &+ \{ [26,880 \cdot (da'_{2n-1,2m} + da'_{2n+1,2m} \\
 &+ ad'_{2n,2m-1} + ad'_{2n,2m+1}) \\
 &+ 11,025 \cdot (aa'_{2n-1,2m-1} + aa'_{2n-1,2m+1} \\
 &+ aa'_{2n+1,2m-1} + aa'_{2n+1,2m+1}) \\
 &+ 32,768] \gg 16 \} \quad (44)
 \end{aligned}$$

$$\begin{aligned}
 ad_{2n,2m-1} &= ad'_{2n,2m-1} \\
 &+ \{ [105 \cdot (aa'_{2n-1,2m-1} + aa'_{2n+1,2m-1}) \\
 &+ ((dd_{2n,2m-2} + dd_{2n,2m}) \ll 7) + 128] \gg 8 \} \quad (45)
 \end{aligned}$$

$$\begin{aligned}
 da_{2n-1,2m} &= da'_{2n-1,2m} \\
 &+ \{ [105 \cdot (aa'_{2n-1,2m-1} + aa'_{2n-1,2m+1}) \\
 &+ ((dd_{2n-2,2m} + dd_{2n,2m}) \ll 7) + 128] \gg 8 \} \quad (46)
 \end{aligned}$$

$$\begin{aligned}
 aa_{2n-1,2m-1} &= aa'_{2n-1,2m-1} \\
 &+ \{ [((ad_{2n-2,2m-1} + ad_{2n,2m-1} \\
 &+ da_{2n-1,2m-2} + da_{2n-1,2m}) \ll 1) \\
 &- (dd_{2n-2,2m-2} + dd_{2n-2,2m} \\
 &+ dd_{2n,2m-2} + dd_{2n,2m}) + 2] \gg 2 \}. \quad (47)
 \end{aligned}$$

The corresponding filters have no real vanishing moments. Figure 11 shows that in comparison to Figure 10, zeros have moved from their original location on the unit circle to other places.

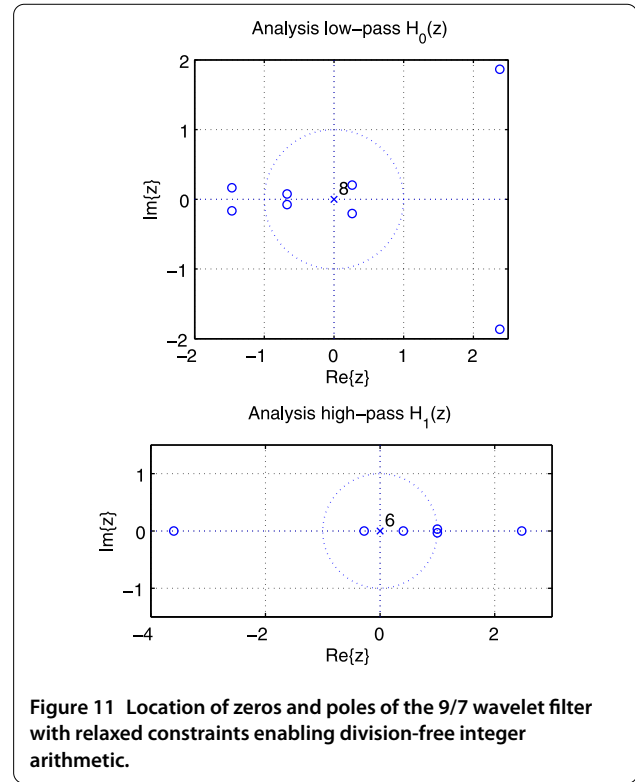
## 6 Investigations

### 6.1 Effects of rounding in 1D

The effects of rounding on the impulse responses of the high and low-pass filters have been investigated in the following manner.

The convolution

$$y[n] = h[n] * (\nu \cdot x[n])$$



was computed with  $(x[n])$  equal to the Kronecker delta  $\delta[k]$  as input signal. The convolution result is therefore equal to

$$\nu \cdot h[n-k] = h[n] * (\nu \cdot x[n]).$$

If, however, the rounding to integer values is additionally performed, then the result is not  $\nu \cdot h[n-k]$ , but  $\nu \cdot h'[n-k]$ . When using an impulse magnitude of  $\nu = 9$  and setting  $k = 0$ , for example, the designed impulse responses of the 5/3 decomposition change from eq. (1) to

$$(h'_1[n]) = (-4 \quad 9 \quad -4) / 9,$$

$$(h'_0[n]) = (-1 \quad 2 \quad 7 \quad 2 \quad -1) / 9.$$

In case of the standard 9/7 filters, the impulse responses change from

$$(h_1[n]) = (0.074194 \quad -0.046777 \quad -0.480641$$

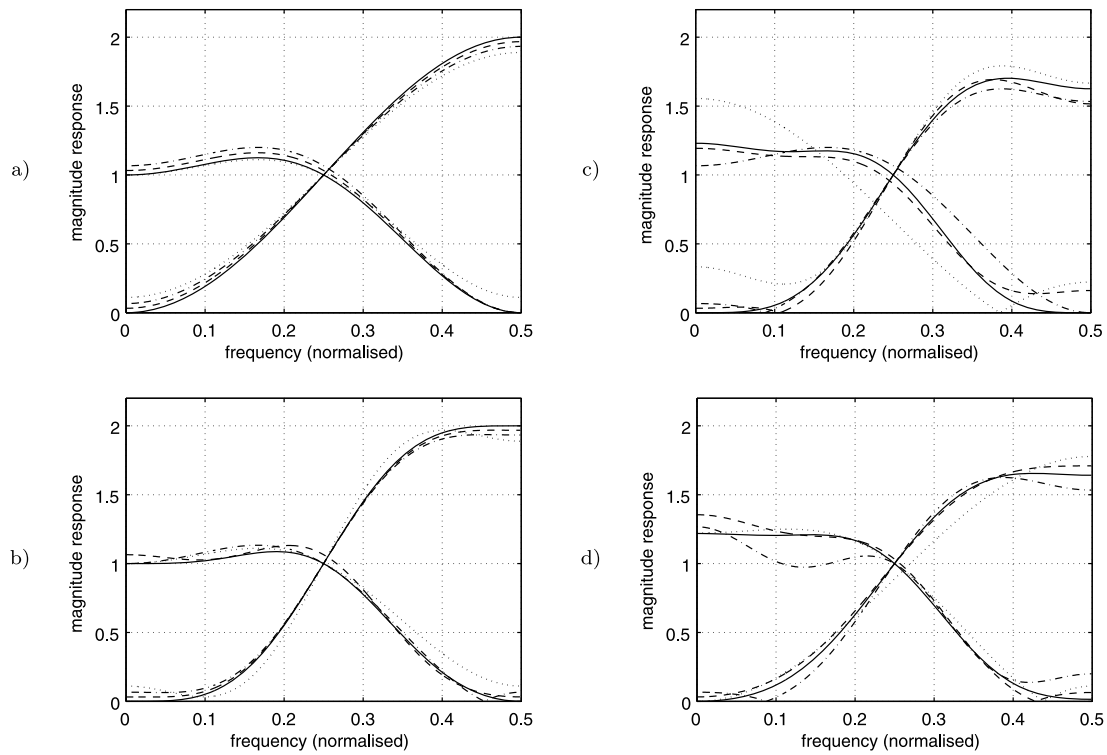
$$0.906447 \quad -0.480641 \quad \dots),$$

$$(h_0[n]) = (0.032906 \quad -0.020746 \quad -0.096228$$

$$0.328289 \quad 0.741732 \quad 0.328289 \quad \dots)$$

to

$$(h'_1[n]) = (1 \quad 0 \quad -4 \quad 9 \quad -4 \quad 0 \quad 1) / 9,$$



**Figure 12** Change of magnitude response caused by rounding: (a) 5/3 filter bank; (b) 9/7 Deslauriers-Dubuc filter bank; (c) 9/7 filter bank; (d) 9/7 filter bank with lifting coefficients from eq. (39) (solid: transfer function without rounding; dash, dash-dot, and dot indicate signal values  $v$  of 31, 15, and 9 used for determination of the impulse responses with rounding).

$$(h'_0[n]) = (0 \ 0 \ 0 \ 4 \ 6 \ 4 \ 0 \ 0 \ 0)/9.$$

Figure 12 shows the corresponding magnitude responses depending on the value  $v$  of the single impulse functioning as input signal.

The smaller the input value  $v$ , the higher the effect of rounding.

## 6.2 Effects of rounding in 2D

In order to illustrate the rounding effects in two dimensions, the same procedure as in the previous section was performed, but with the distinction that 2D filters were used.

### 6.2.1 Effects in the 5/3 filter bank

Figure 13(a) depicts the original magnitude response of the 5/3 filter ( $h_{11}[n]$ ) from eq. (7). The frequency axes are normalised by the sampling frequency. If rounding is applied, the magnitude response changes (Figure 13(b)). In this and subsequent investigations, the impulse magnitude used as input signal was always set equal to  $v = 9$ .

In general, the resulting impulse responses depend on the processing structure. In the particular case of  $v = 9$  however, the  $h_{11}$ ,  $h_{10}$ , and  $h_{01}$  filters are the same for both

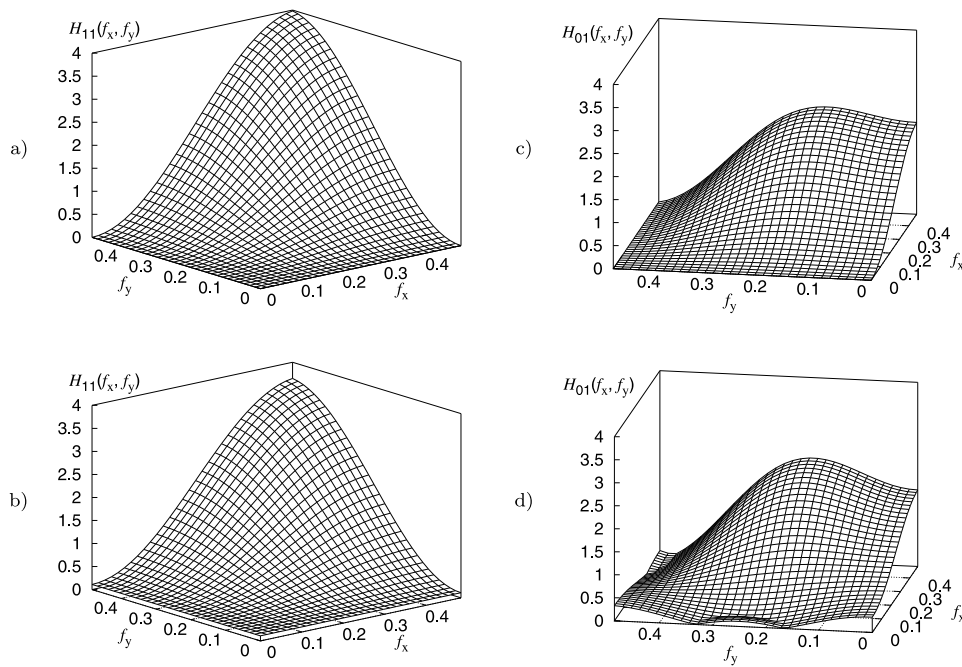
(Figure 1 or from Figure 5). The impulse responses from eqs. (7)-(9) change in both cases to

$$(h'_{11}[n]) = \frac{1}{9} \cdot \begin{pmatrix} 2 & -4 & 2 \\ -4 & 9 & -4 \\ 2 & -4 & 2 \end{pmatrix},$$

$$(h'_{01}[n]) = \frac{1}{9} \cdot \begin{pmatrix} 1 & -1 & 1 \\ -1 & 2 & -1 \\ -3 & 7 & -3 \\ -1 & 2 & -1 \\ 1 & -1 & 1 \end{pmatrix} = (h'_{10}[n])^T.$$

Figures 13(c) and (d) compare the original with the deformed magnitude responses for the filter  $h_{01}$ .

The different impact of rounding in the discussed signal-flow structures becomes visible when inspecting the magnitude responses of the 5/3 filter  $h_{00}$  from eq. (10). While in the separable implementation, *i.e.* the consecutive processing of horizontal and vertical direction, the deformation is different in  $f_x$  and  $f_y$ , the changes are symmetric in the modified implementation (Figure 14).



**Figure 13** 5/3 filter bank: distorted magnitude response caused by rounding to integer values; (a) original magnitude response  $H_{11}(f_x, f_y)$ ; (b) separable/2D implementation  $H'_{11}(f_x, f_y)$  (Figures 1 and 5); (c) original magnitude response  $H_{01}(f_x, f_y)$ ; (d) separable/2D implementation  $H'_{01}(f_x, f_y)$ .

The two-dimensional impulse responses from (10) deteriorate to

$$(h'_{00}[n])_{\text{sep}} = \frac{1}{9} \cdot \begin{pmatrix} 0 & 0 & -1 & 0 & 0 \\ 0 & 1 & 2 & 1 & 0 \\ 0 & 2 & 6 & 2 & 0 \\ 0 & 1 & 2 & 1 & 0 \\ 0 & 0 & -1 & 0 & 0 \end{pmatrix}.$$

in case of the separable implementation and to

$$(h'_{00}[n])_{2D} = \frac{1}{9} \cdot \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 \\ 0 & 2 & 6 & 2 & 0 \\ 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}.$$

in case of the 2D implementation.

### 6.2.2 Effects in the 9/7 filter bank

For the case of the standard 9/7 filter bank, we have to compare three different implementations, which are called '97v1', '97v2', and '97v3' in the following, according to the flow charts in Figures 6, 7, and 9, respectively.

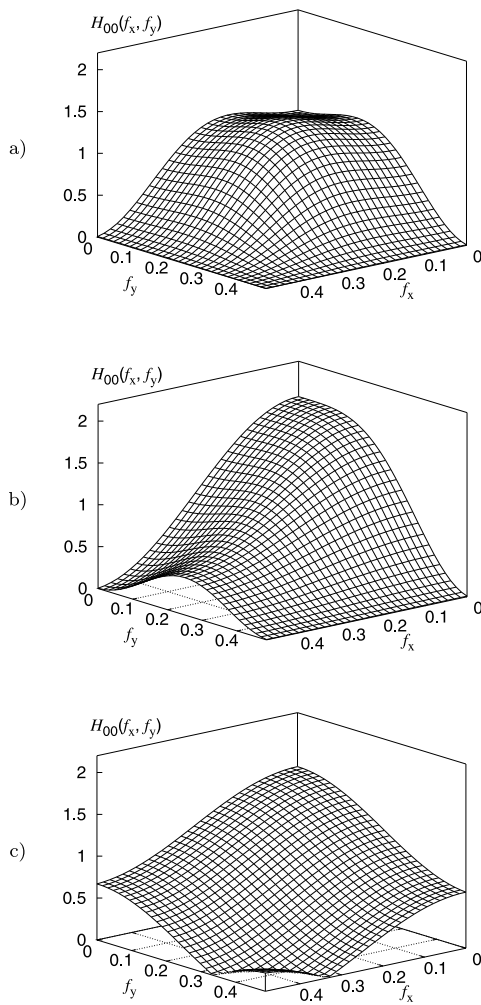
The  $h_{11}$  impulse response changes from the matrix shown in Figure 15 to following

$$(h'_{11}[n])_{97v1} = \frac{1}{9} \cdot \begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 2 & -4 & 2 & 0 & -1 \\ 1 & 0 & -4 & 9 & -4 & 0 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{pmatrix},$$

$$(h'_{11}[n])_{97v2} = \frac{1}{9} \cdot \begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 3 & -4 & 3 & 0 & 0 \\ 1 & 0 & -4 & 9 & -4 & 0 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{pmatrix},$$

$$(h'_{11}[n])_{97v3} = \frac{1}{9} \cdot \begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & -4 & 2 & 0 & 0 \\ 1 & 0 & -4 & 9 & -4 & 0 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{pmatrix}.$$

The magnitude of the input impulse was again equal to  $\nu = 9$ . In order to comply with at least the basic requirements of a real high-pass filter, the sums in all rows and columns of the impulse response should be zero. However, this condition is no longer fulfilled, as can be concluded



**Figure 14** 5/3 filter bank: distorted magnitude response of  $H_{00}(f_x, f_y)$  caused by rounding to integer values; (a) original magnitude response; (b) separable implementation (Figure 1); (c) 2D implementation (Figure 5).

from the matrices above. The graph of the corresponding magnitude responses underlines this observation (Figure 16).

Similar alterations of the filter coefficients can be observed for the  $h_{01}$ ,  $h_{10}$ , and  $h_{00}$  filters. Another effect of non-linearity is that the  $h_{10}$  filters are no longer the transposes of the  $h_{01}$  filters.

Comparing the similarities of the two-dimensional impulse responses between the different implementations, the deviation is higher for the  $h_{10}$  and  $h_{01}$  filters than for the  $h_{11}$  filter. This continues for the  $h_{00}$  filter, which shows the highest variation from one implementation to another.

Using the new lifting coefficients, the degradation of the filters is generally reduced (Figure 17 in comparison with Figure 16).

Note that the degradation is dependent on the magnitude of the impulse functioning as input signal. The shown magnitude responses merely give an impression of the effects, but do not allow a general conclusion or even a theoretical analysis.

### 6.3 Efficiency of decorrelation

The efficiency of the different filter banks was first analysed based on the zero-order entropies of single sub-bands.

Table 1-Table 4 contain the zero-order entropies of the sub-bands DD, AD, DA and AA after one 2D decomposition step is applied to different grey-scale images. Barbara-Zelda are taken from [33], cats\_g-educ from [34]. kodim07-kodim09 are the green components of true colour images found at [35]. The meaning of the column titles is

- 53v1: 5/3 filter bank, implementation of Figure 1,
- 53v2: 5/3 filter bank, implementation of Figure 5,
- 97D1: 9/7 Deslauriers-Dubuc, implementation of Figure 1,
- 97D2: 9/7 Deslauriers-Dubuc, implementation of Figure 5,
- 97v1: 9/7 filter bank, implementation of Figure 6,
- 97v2: 9/7 filter bank, implementation of Figure 7,
- 97v3: 9/7 filter bank, implementation of Figure 9,
- 97v1a - 97v3a: same as before, but with lifting coefficients from equation (39).

The numbers in bold denote the best (smallest) value in each entire row. In each category (5/3, Deslauriers-Dubuc, 9/7, 9/7 with modified coefficients), the best value is underlined.

While the explanations in Section 2 were given with implicit floating-point arithmetic plus rounding operation (eqs. (3), (4), and (17)–(20)), the practical implementation of the 5/3 filter banks (53v1, 53v2) is based on plain integer arithmetic as follows

$$d_n = x_{2n+1} - ((x_{2n} + x_{2n+2}) \gg 1), \quad (3*)$$

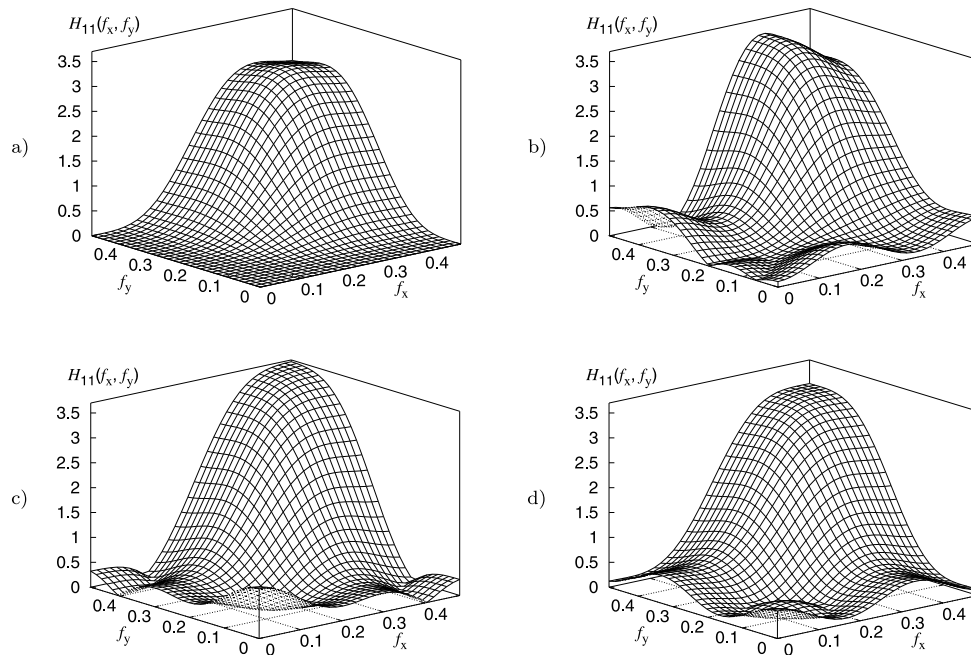
$$a_n = x_{2n} + ((x_{2n} + x_{2n+2} + 2) \gg 2), \quad (4*)$$

$$\begin{aligned} dd_{2n,2m} &= x_{2n,2m} \\ &+ \{ [x_{2n-1,2m-1} + x_{2n+1,2m-1} \\ &+ x_{2n-1,2m+1} + x_{2n+1,2m+1} \\ &- ((x_{2n,2m-1} + x_{2n,2m+1} + x_{2n-1,2m} \\ &+ x_{2n+1,2m}) \ll 1) + 2] \gg 2 \}, \end{aligned} \quad (17*)$$

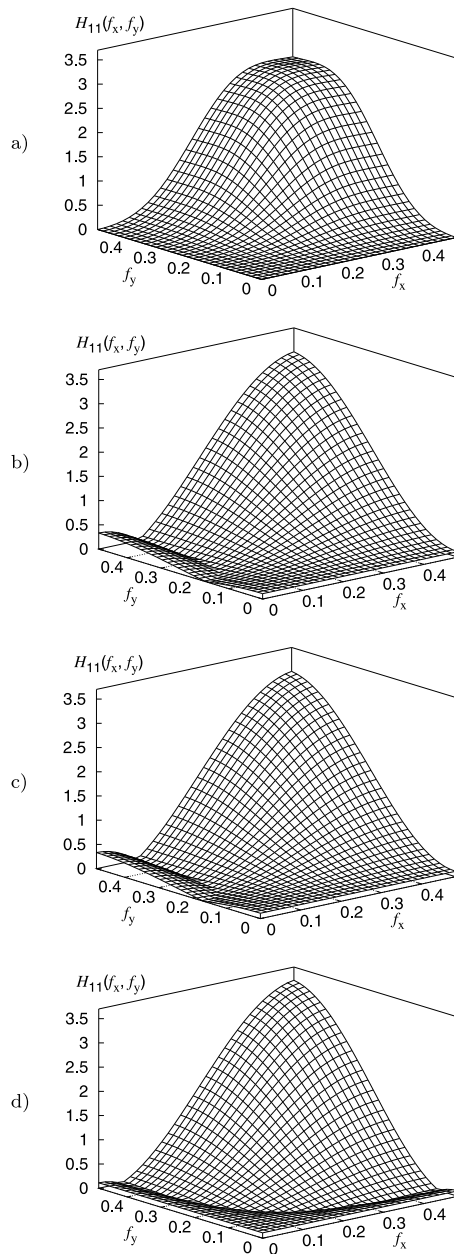
$$\begin{aligned} da_{2n-1,2m} &= x_{2n-1,2m} \\ &+ \{ [dd_{2n-2,2m} + dd_{2n,2m} \\ &- ((x_{2n-1,2m-1} + x_{2n-1,2m+1}) \ll 1) \\ &+ 2] \gg 2 \}, \end{aligned} \quad (18*)$$

$$\begin{aligned}
 (h_{11}[n]) &\approx \frac{1}{9} \cdot \begin{pmatrix} 0.0495 & -0.0312 & -0.3209 & 0.6053 & -0.3209 & -0.0312 & 0.0495 \\ -0.0312 & 0.0197 & 0.2023 & -0.3816 & 0.2023 & 0.0197 & -0.0312 \\ -0.3209 & 0.2023 & 2.0791 & -3.9211 & 2.0791 & 0.2023 & -0.3209 \\ 0.6053 & -0.3816 & -3.9211 & 7.3948 & -3.9211 & -0.3816 & 0.6053 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{pmatrix} \\
 (h'_{01}[n]) &\approx \frac{1}{9} \cdot \begin{pmatrix} 0.0220 & -0.0139 & -0.1423 & 0.2684 & -0.1423 & -0.0139 & 0.0220 \\ -0.0139 & 0.0087 & 0.0897 & -0.1692 & 0.0897 & 0.0087 & -0.0139 \\ -0.0643 & 0.0405 & 0.4163 & -0.7850 & 0.4163 & 0.0405 & -0.0643 \\ 0.2192 & -0.1382 & -1.4201 & 2.6782 & -1.4201 & -0.1382 & 0.2192 \\ 0.4953 & -0.3123 & -3.2086 & 6.0511 & -3.2086 & -0.3123 & 0.4953 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{pmatrix} \\
 (h_{10}[n]) &= (h_{01}[n])^T \\
 (h_{00}[n]) &\approx \frac{1}{9} \cdot \begin{pmatrix} 0.0097 & -0.0061 & -0.0285 & 0.0972 & 0.2197 & 0.0972 & -0.0285 & -0.0061 & 0.0097 \\ -0.0061 & 0.0039 & 0.0180 & -0.0613 & -0.1385 & -0.0613 & 0.0180 & 0.0039 & -0.0061 \\ -0.0285 & 0.0180 & 0.0833 & -0.2843 & -0.6424 & -0.2843 & 0.0833 & 0.0180 & -0.0285 \\ 0.0972 & -0.0613 & -0.2843 & 0.9700 & 2.1915 & 0.9700 & -0.2843 & -0.0613 & 0.0972 \\ 0.2197 & -0.1385 & -0.6424 & 2.1915 & 4.9515 & 2.1915 & -0.6424 & -0.1385 & 0.2197 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{pmatrix}
 \end{aligned}$$

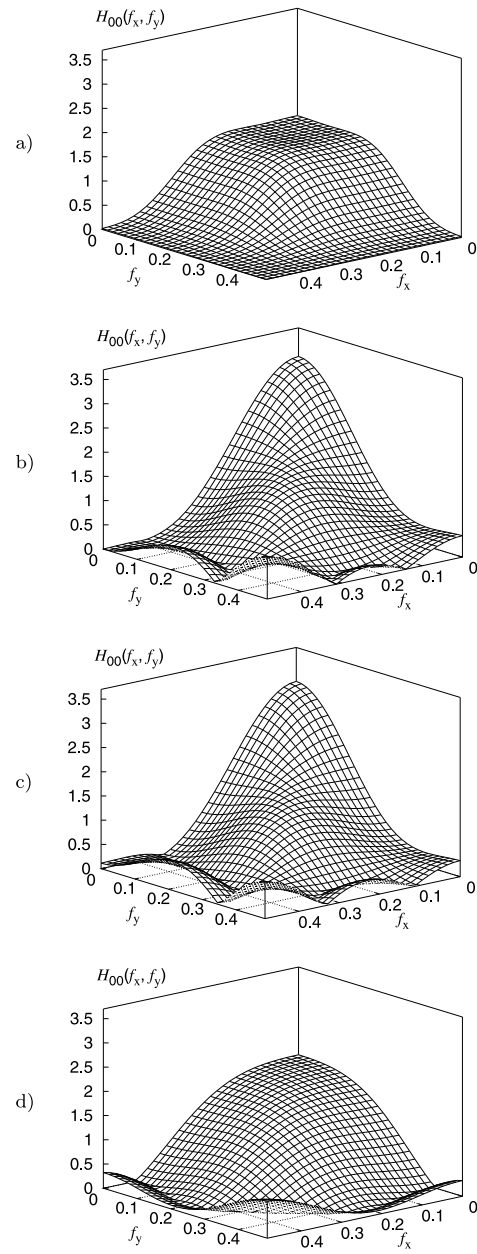
**Figure 15 Two-dimensional impulse responses of the original 9/7 filter bank.** The matrices are symmetric in the horizontal and vertical directions. The factor 1/9 is extracted for better comparison with numbers in the text.



**Figure 16 9/7 filter bank: distorted magnitude response of  $H_{11}(f_x, f_y)$  caused by rounding; (a) original magnitude response; (b) 97v1 implementation; (c) 97v2 implementation; (d) 97v3 implementation.**



**Figure 17** 9/7 filter bank with modified lifting coefficients (eq. (39)): distorted magnitude response of  $H_{11}(f_x, f_y)$  caused by rounding; (a) original magnitude response; (b) 97v1a implementation; (c) 97v2a implementation; (d) 97v3a implementation.



**Figure 18** 9/7 filter bank with modified lifting coefficients (eq. (39)): distorted magnitude response of  $H_{00}(f_x, f_y)$  caused by rounding; (a) original magnitude response; (b) 97v1a implementation; (c) 97v2a implementation; (d) 97v3a implementation.

$$\begin{aligned}
 ad_{2n,2m-1} &= x_{2n,2m-1} \\
 &+ \{ [dd_{2n,2m-2} + dd_{2n,2m} \\
 &- ((x_{2n-1,2m-1} + x_{2n+1,2m-1}) \ll 1) \\
 &+ 2] \gg 2 \}, \quad (19*)
 \end{aligned}$$

$$\begin{aligned}
 aa_{2n-1,2m-1} &= x_{2n-1,2m-1} \\
 &+ \{ [((ad_{2n-2,2m-1} + ad_{2n,2m-1} \\
 &+ da_{2n-1,2m-2} + da_{2n-1,2m}) \ll 2) \\
 &- (dd_{2n-2,2m-2} + dd_{2n-2,2m} \\
 &+ dd_{2n,2m-2} + dd_{2n,2m}) + 8] \gg 4 \}. \quad (20*)
 \end{aligned}$$

**Table 1 DD-band entropies in bit per pixel after first decomposition**

Image	53v1	53v2	97D1	97D2	97v1	97v2	97v3	97v1a	97v2a	97v3a
barbara.y	4.503	<u>4.494</u>	4.427	<u>4.415</u>	4.218	<b>4.202</b>	4.206	4.235	4.230	<u>4.223</u>
barbara2.y	4.765	<u>4.746</u>	4.834	<u>4.819</u>	4.595	4.578	<u>4.577</u>	4.553	4.536	<b>4.532</b>
black.y	3.716	<u>3.708</u>	3.828	<u>3.814</u>	3.550	<u>3.529</u>	3.537	3.470	3.466	<b>3.450</b>
boats.y	3.795	<u>3.791</u>	3.873	<u>3.865</u>	3.611	<u>3.598</u>	3.605	3.548	3.548	<b>3.536</b>
goldhill.y	4.361	<u>4.348</u>	4.469	<u>4.461</u>	4.203	<u>4.190</u>	4.192	4.137	4.130	<b>4.125</b>
zelda.y	<u>3.961</u>	<u>3.961</u>	4.094	<u>4.091</u>	3.785	<u>3.775</u>	3.778	3.697	3.702	<b>3.696</b>
cats_g	3.098	<u>3.083</u>	3.100	<u>3.086</u>	2.967	<u>2.952</u>	2.970	2.958	2.942	<b>2.937</b>
bike	4.098	<u>4.087</u>	4.163	<u>4.147</u>	4.000	<u>3.967</u>	3.979	3.933	3.918	<b>3.905</b>
educ	3.717	<u>3.707</u>	3.610	<u>3.588</u>	3.537	<u>3.525</u>	3.531	3.529	3.523	<b>3.507</b>
kodim07	3.478	<u>3.463</u>	3.582	<u>3.551</u>	3.451	<u>3.377</u>	3.425	3.350	3.323	<b>3.292</b>
kodim08	4.933	<u>4.930</u>	5.009	<u>5.008</u>	4.794	<u>4.792</u>	4.792	4.743	4.742	<b>4.740</b>
kodim09	3.842	<u>3.837</u>	3.938	<u>3.931</u>	3.728	<u>3.711</u>	3.722	3.655	3.654	<b>3.642</b>
average	4.022	<u>4.013</u>	4.077	<u>4.065</u>	3.870	<u>3.850</u>	3.860	3.817	3.810	<b>3.800</b>

**Table 2 AD-band entropies in bit per pixel after first decomposition**

Image	53v1	53v2	97D1	97D2	97v1	97v2	97v3	97v1a	97v2a	97v3a
barbara.y	4.202	<u>4.198</u>	3.853	<b>3.846</b>	<u>3.960</u>	3.961	3.971	4.087	4.093	<u>4.081</u>
barbara2.y	<u>4.231</u>	4.239	<b>4.108</b>	4.113	<u>4.169</u>	4.172	4.178	4.190	4.197	<u>4.188</u>
black.y	3.438	<u>3.423</u>	3.280	<b>3.273</b>	3.472	<u>3.471</u>	3.487	3.468	3.468	<u>3.451</u>
boats.y	3.733	<u>3.718</u>	3.547	<b>3.515</b>	3.734	<u>3.716</u>	3.736	3.753	3.749	<u>3.732</u>
goldhill.y	<u>4.492</u>	4.528	<b>4.443</b>	4.470	<u>4.553</u>	4.581	4.582	<u>4.554</u>	4.587	4.580
zelda.y	3.147	<u>3.120</u>	3.024	<b>3.001</b>	3.132	<u>3.117</u>	3.132	3.114	3.103	<u>3.084</u>
cats_g	3.348	<u>3.341</u>	3.255	<b>3.245</b>	3.332	<u>3.329</u>	3.338	<u>3.360</u>	3.371	3.364
bike	4.613	<u>4.598</u>	4.611	<b>4.596</b>	4.737	<u>4.730</u>	4.740	4.710	4.705	<u>4.698</u>
educ	4.989	<u>4.983</u>	4.733	<b>4.721</b>	4.939	<u>4.935</u>	4.939	5.031	5.028	<u>5.026</u>
kodim07	<u>4.031</u>	4.045	<b>3.801</b>	3.823	<u>3.982</u>	4.018	4.031	<u>4.044</u>	4.071	4.057
kodim08	5.672	<b>5.644</b>	5.696	<u>5.679</u>	5.806	<u>5.786</u>	5.789	5.780	5.754	<u>5.753</u>
kodim09	4.169	<u>4.150</u>	4.146	<b>4.124</b>	4.255	<u>4.237</u>	4.250	4.234	4.220	<u>4.212</u>
average	4.172	<u>4.166</u>	4.041	<b>4.034</b>	4.173	<u>4.171</u>	4.181	4.194	4.196	<u>4.186</u>

The results for the DD band (high-pass filtering in both directions) show a clear trend with decreasing entropy from left to right (Table 1), when excluding 97D1 and 97D2. There is only one exception: the implementation 97v2 leads to lower entropy on average than 97v3 when the standard lifting coefficients (eq. (24)) are used. The Deslauriers-Dubuc filter bank (97D1) performs worst despite the four vanishing moments of its high-pass filter. One reason lies in the gain of  $H_1(z)|_{z=-1} = 2$ , which is significantly higher than the gain of the JPEG2000 9/7 high pass. In comparison to the 5/3 filter banks (also having  $H_1(-1) = 2$ ), the filter characteristic seems to be more influenced by the rounding operations deteriorating the advantage of two additional vanishing moments.

The 2D implementations of the 5/3 filter bank (53v2) and of the Deslauriers-Dubuc filter bank (97D2) are superior to the separable implementations. All other 9/7 implementations yield better results, despite the higher number of lifting steps. It also can be seen that the modified lifting coefficients truly improve the decorrelation on average. However, the numbers in this table also reveal that the number of rounding steps is not a unique measure to estimate

the influence of rounding steps. The DD values depend on three steps in 97v2 and only on two steps in 97v3; nevertheless, the result is better for 97v2.

Table 2 draws another picture. The implementations 53v2, 97D2, 97v2, and 97v3a are again the winner within their categories. The filter banks based on only two lifting steps, however, lead to the smaller entropies in the AD band on average. Surprisingly, the results for the DA band in Table 3 are not very similar to the values for the AD band. Very interesting, although not in the primary scope of this paper, is the fact that all images taken from [33] have a higher entropy in the DA band compared to the AD band. The vertical edges are obviously sharper than the horizontal ones. Using rotated versions of the images reverses the results. It might be that the point spread function of the camera system used for taking these pictures was not symmetric. This can also cause the 5/3 filter bank to perform better for the DA band than for the AD band. Short filters are more suitable at pronounced edges.

The entropies in the AA band are in principle independent of whether a special 2D implementation is used or not (Table 4). Within each category, the average values are al-

**Table 3 DA-band entropies in bit per pixel after first decomposition**

Image	53v1	53v2	97D1	97D2	97v1	97v2	97v3	97v1a	97v2a	97v3a
barbara.y	<u>5.431</u>	5.449	<b>5.383</b>	5.389	<u>5.460</u>	5.473	5.480	<u>5.475</u>	5.485	5.490
barbara2.y	5.537	<b>5.531</b>	<u>5.539</u>	5.541	<u>5.617</u>	5.623	5.626	5.607	<u>5.600</u>	5.602
black.y	<b>4.005</b>	4.070	<u>4.055</u>	4.077	<u>4.114</u>	4.158	4.173	<u>4.063</u>	4.115	4.126
boats.y	<b>4.471</b>	4.490	4.492	4.492	<u>4.574</u>	4.593	4.603	<u>4.552</u>	4.559	4.564
goldhill.y	<b>4.582</b>	4.585	4.584	4.584	<u>4.647</u>	4.657	4.660	4.639	<u>4.637</u>	4.638
zelda.y	<b>4.062</b>	4.087	<u>4.072</u>	4.076	<u>4.124</u>	4.139	4.151	<u>4.109</u>	4.123	4.125
cats_g	3.445	<u>3.423</u>	3.350	<b>3.339</b>	<u>3.428</u>	3.460	3.432	<u>3.444</u>	3.456	3.454
bike	4.507	<u>4.503</u>	4.501	<b>4.496</b>	<u>4.577</u>	4.609	4.618	<u>4.567</u>	4.569	4.577
educ	4.734	<u>4.708</u>	4.450	<b>4.445</b>	<u>4.609</u>	4.633	4.648	<u>4.724</u>	4.725	4.730
kodim07	3.627	<u>3.614</u>	3.557	<b>3.548</b>	<u>3.660</u>	3.733	3.755	3.665	<u>3.664</u>	3.683
kodim08	5.814	<b>5.812</b>	5.837	<u>5.831</u>	5.960	<u>5.955</u>	5.957	5.936	<u>5.926</u>	5.928
kodim09	<u>4.046</u>	4.061	<b>4.039</b>	4.047	<u>4.115</u>	4.143	4.162	<u>4.093</u>	4.103	4.112
average	<u>4.522</u>	4.528	<b>4.488</b>	4.489	<u>4.574</u>	4.598	4.605	<u>4.573</u>	4.580	4.586

**Table 4 AA-band entropies in bit per pixel after first decomposition**

Image	53v1	53v2	97D1	97D2	97v1	97v2	97v3	97v1a	97v2a	97v3a
barbara.y	7.563	<u>7.558</u>	7.553	<b>7.538</b>	8.119	8.113	<u>8.109</u>	8.100	8.097	<u>8.090</u>
barbara2.y	7.504	<u>7.495</u>	7.490	<b>7.483</b>	8.047	8.055	<u>8.045</u>	8.031	<u>8.025</u>	8.027
black.y	6.759	<b>6.744</b>	6.773	<b>6.744</b>	7.372	<u>7.342</u>	7.360	7.330	7.327	<u>7.316</u>
boats.y	7.111	<u>7.097</u>	7.103	<b>7.082</b>	7.669	<u>7.659</u>	7.672	7.646	<u>7.645</u>	7.647
goldhill.y	7.556	<u>7.548</u>	7.551	<b>7.537</b>	<u>8.127</u>	8.134	<u>8.127</u>	8.103	<u>8.099</u>	<u>8.099</u>
zelda.y	<u>7.332</u>	7.334	7.330	<b>7.319</b>	<u>7.912</u>	7.916	7.915	<u>7.884</u>	7.897	7.888
cats_g	<u>4.738</u>	4.740	<b>4.732</b>	4.734	<u>5.008</u>	5.020	5.023	<u>5.006</u>	5.007	5.010
bike	<u>7.429</u>	7.430	<b>7.396</b>	<b>7.396</b>	<u>7.918</u>	7.925	7.929	<u>7.919</u>	7.920	7.920
educ	7.449	<u>7.448</u>	7.446	<b>7.445</b>	<u>8.041</u>	8.044	8.047	<u>8.017</u>	8.019	8.017
kodim07	<u>7.139</u>	7.147	<b>7.114</b>	7.121	<u>7.672</u>	7.677	7.680	<u>7.662</u>	7.669	7.667
kodim08	<u>7.822</u>	7.828	<b>7.794</b>	7.800	8.333	8.333	<u>8.332</u>	<u>8.320</u>	<u>8.320</u>	8.321
kodim09	<u>7.237</u>	7.246	<b>7.223</b>	7.232	<u>7.793</u>	7.801	7.802	<u>7.776</u>	7.782	7.784
average	7.137	<u>7.135</u>	7.125	<b>7.119</b>	<u>7.668</u>	<u>7.668</u>	7.670	7.650	7.651	<u>7.649</u>

most the same. The superior results for the 5/3 and 97DD filter banks are also caused by the lower amplification of the low-pass filter (see Figure 12). While their low-pass filter transfer function  $H_0(z)$  is equal to 1 at  $z = 1$ , the amplification for the JPEG2000 9/7 low pass is about 1.25 affecting also the 2D characteristics (compare Figures 14a and 18a).

In addition, we would like to point out that the modified lifting coefficients again have a positive influence on the decorrelation.

#### 6.4 Compression results

The overall performance of the different filter banks was tested in combination with a compression system, which applies the basic coding algorithm of JPEG2000 without using the header/marker structure. The number of decomposition steps was dependent on the image size, for example, five decompositions for images with  $720 \times 576$  pixels and seven decompositions for images with  $2048 \times 2560$  pixels. The focus was exclusively on lossless image compression. A rate-distortion analysis exploiting the scalability of the compression scheme is not considered in this paper.

When comparing the lossless compression results in bits per pixel (Table 5), the superiority of the 2D implementation of the 9/7 Deslauriers-Dubuc filter bank becomes apparent. There is also only a single case (goldhill.y) where the separable 5/3 implementation is better than any 2D implementation. Only in cases of special image texture (barbara.y, cats\_g, and educ) can a standard 9/7 implementation compete with the 5/3 filter banks.

The 97v2(a) implementation does not result in any advantage over the separable 97v1(a) implementation. The new non-separable implementation 97v3a in combination with the lifting coefficients according to eq. (39) performs best within all 9/7 filter banks based on the JPEG2000 9/7 filter-bank structure.

#### 6.5 Complexity of implementations

The advantage of separable transformations results from the reduction of the required number of operations compared to their non-separable counterparts. The reduction of the number of rounding steps, however, implies a transfer from separable to non-separable filter banks. Table 6 contains the number of operations needed for the compu-



**Table 5 Compression results in bits per pixel [bpp]**

Image	53v1	53v2	97D1	97D2	97v1	97v2	97v3	97v1a	97v2a	97v3a
barbara.y	4.594	<u>4.584</u>	4.478	<b>4.469</b>	4.538	<u>4.535</u>	4.549	4.558	4.563	<u>4.556</u>
barbara2.y	4.778	<u>4.776</u>	<b>4.737</b>	4.744	<u>4.777</u>	4.793	4.802	<u>4.769</u>	4.772	<u>4.772</u>
black.y	3.765	<b>3.750</b>	3.774	<b>3.750</b>	3.854	<u>3.844</u>	3.864	3.796	3.796	<u>3.786</u>
boats.y	4.057	<u>4.033</u>	4.024	<b>3.991</b>	4.104	<u>4.095</u>	4.117	4.075	4.070	<u>4.059</u>
goldhill.y	<b>4.593</b>	4.595	<u>4.594</u>	4.596	<u>4.633</u>	4.650	4.656	<u>4.618</u>	4.620	4.616
zelda.y	3.870	<u>3.850</u>	3.853	<b>3.834</b>	3.912	<u>3.901</u>	3.920	3.876	3.872	<u>3.864</u>
cats_g	2.542	<u>2.534</u>	2.500	<b>2.493</b>	<u>2.527</u>	2.532	2.541	<u>2.537</u>	2.539	<u>2.537</u>
bike	4.364	<b>4.342</b>	4.343	<b>4.324</b>	4.412	<u>4.409</u>	4.431	4.387	4.383	<u>4.373</u>
educ	4.534	<u>4.513</u>	4.342	<b>4.315</b>	4.493	<u>4.490</u>	4.512	4.534	4.530	<u>4.515</u>
kodim07	3.777	<u>3.741</u>	3.715	<b>3.687</b>	<u>3.846</u>	3.850	3.888	3.814	3.809	<u>3.794</u>
kodim08	5.531	<b>5.516</b>	5.539	<u>5.533</u>	5.572	<u>5.568</u>	5.574	5.553	5.544	<u>5.545</u>
kodim09	4.027	<u>4.013</u>	4.027	<b>4.012</b>	4.092	<u>4.090</u>	4.116	4.054	4.053	<u>4.046</u>
average	4.203	<u>4.187</u>	4.161	<b>4.146</b>	<u>4.230</u>	<u>4.230</u>	4.248	4.214	4.213	<u>4.205</u>

**Table 6 Complexity estimation for different filter-bank implementations**

Filter bank	Adds	Shifts	Mults
53v1	20	8	0
53v2	28	8	0
97D1	32	8	4
97D2	48	9	4
97v1a	40	12	8
97v2a	47	11	10
97v3a	55	10	8

tation of four sub-band values for the investigated filter-bank structures. The numbers have been determined as follows. The implementation of 53v1 is based on equations (3\*) and (4\*). There are five adds and two shifts. To compute four sub-band values, both equations have to be applied two times horizontally and two times vertically, leading to a total number of twenty adds and eight shifts. Equations (17\*)-(20\*) compute four sub-band values according to the 2D structure 53v2. There are  $9 + 5 + 5 + 9 = 28$  adds and  $2 + 2 + 2 + 2 = 8$  shift operations. Filter bank 97D1 is based on equations (32) and (33), requiring eight adds, two shifts and one integer multiplication. Again, these numbers must be multiplied by four. Equations (34)-(37) reflect the complexity of filter bank 97D2. There are  $25 + 7 + 7 + 9 = 48$  adds, nine shift operations and four integer multiplications.

The numbers for filter banks 97v1a, 97v2a and 97v3a have been derived from their implementations (see also eqs. (40)-(47)).

It can be noticed that the increase of complexity is rather moderate when switching from a one-dimensional to a two-dimensional implementation. This is largely due to the properties of the lifting scheme and the utilisation of the symmetry of filters. With respect to the memory access, the 2D implementation is even somewhat advantageous, as the algorithm runs only once through the data

and not twice, separately in horizontal and vertical directions. Practical implementations, however, also have to consider the efforts of signal extension at the signal boundaries, which are highest for the 97Dx filter banks.

## 7 Summary and conclusions

The paper has discussed and analysed different attempts to decrease the effects of rounding in implementations of the integer wavelet transform. The number of rounding steps could be reduced practically by special two-dimensional implementations of the 5/3 and 9/7 filter banks and virtually by using a first lifting coefficient of  $\alpha = -1$  in the JPEG2000 9/7 filter bank. Dependent on the implementation, the rounding of intermediate values to integers has different effects on the impulse responses and, consequently, on the decomposition of the signal.

The set of test images contained twelve natural images. It can be shown that the various processing schemes affect the single sub-bands differently in terms of entropy. As soon as the low-pass filter comes into play, the 5/3 filter bank and the 97DD filter bank tend to decorrelate the image data better than the standard 9/7 filter bank and its derivatives. The gain of the low-pass filter at  $z = 1$  has a high impact on lossless compression performance.

The compression results of the 5/3 and 97DD filter bank have been improved on average by substituting the separable implementation with a special 2D processing, which would be compatible in the absence of rounding. The mere reduction of rounding steps by 2D implementations of the JPEG2000 9/7 filter bank does not lead to increased compression ratios. Only when the modified lifting coefficients enabling division-free integer arithmetic are combined with the new proposed 2D implementation does the average bitrate of the compressed images decrease from 4.230 bpp to 4.205 bpp. This is about the same bitrate as for the standard 5/3 filter bank while having distinctly higher complexity.

The results of the different implementations of the 9/7 filter bank reveal that the influence of rounding is not

solely determined by the number of lifting steps, which means the rounding errors (strictly: their variances) do not simply sum up. The degradation of the filter characteristics under the presence of rounding also depends on the values of the lifting coefficients and on the structure of processing. The best compression result could be obtained with the new 2D implementation of the 9/7 Deslauriers-Dubuc filter bank (97D2). It requires only two sequential lifting steps as the 5/3 filter bank, while having more suitable filter characteristics. The improvement is 1.36% compared to the standard 5/3 implementation (53v1) for the set of images used.

#### Competing interests

The authors declare that they have no competing interests.

#### Acknowledgements

The authors would like to thank the unknown reviewers who gave valuable comments on an earlier version of this manuscript.

Received: 26 May 2011 Accepted: 11 March 2012 Published: 4 April 2012

#### References

- JE Fowler, B Pesquet-Popescu, An overview on wavelets in source coding, communications, and networks. *EURASIP J. Image Video Coding* **2007**, Article ID 60539 (2007)
- W Sweldens, The lifting scheme: a new philosophy in biorthogonal wavelet construction, in *Proc. of SPIE*, vol. 2569, San Diego, USA, July 1995, pp. 68–79
- AR Calderbank, I Daubechies, W Sweldens, Y Boon-Lock, Lossless image compression using integer to integer wavelet transforms, in *Proc. of ICIP*, vol. 1, 26–29 Oct. 1997, pp. 596–599
- ISO/IEC JTC1/SC29/WG11 N1890, Information technology - JPEG 2000 Image Coding System. JPEG 2000 Part I, Final Draft Intern. Standard 15444, 25 Sep. 2000
- F Sheng, A Bilgin, PJ Sementilli, MW Marcellin, Lossy and lossless image compression using reversible integer wavelet transforms, in *Proc. of ICIP*, vol. 3, Los Alamitos, CA, USA (1998), pp. 876–880
- VK Heer, H-E Reinfelder, A comparison of reversible methods for data compression, in *Proc. of SPIE - Medical Imaging IV*, vol. 1233 (1990), pp. 354–365
- A Said, WA Pearlman, An image multiresolution representation for lossless and lossy image compression. *IEEE Trans. Image Process.* **5**(9), 1303–1310 (1996)
- A Zandi, JD Allen, EL Schwartz, M Boliek, CREW: compression with reversible embedded wavelets, in *Proc. of DCC*, Snowbird, Utah, USA, 28–30 Mar 1995, pp. 212–221
- MD Adams, F Kossentini, Reversible integer-to-integer wavelet transforms for image compression: performance evaluation and analysis. *IEEE Trans. Image Process.* **9**(6), 1010–1024 (2000)
- J Reichel, G Menegaz, MJ Nadenau, M Kunt, Integer wavelet transform for embedded lossy to lossless image compression. *IEEE Trans. Image Process.* **10**(3), 383–392 (2001)
- M Granelto, E Magli, M Martina, G Olmo, Optimization and implementation of the integer wavelet transform for image coding. *IEEE Trans. Image Process.* **11**(6), 596–604 (2002)
- D Sersic, Integer to integer mapping wavelet filter bank with adaptive number of zero moments, in *Proc. of ICASSP*, vol. 1, Istanbul, Turkey, 9 June 2000, pp. 480–483
- AT Deever, SS Hemami, Lossless image compression with projection-based and adaptive reversible integer wavelet transforms. *IEEE Trans. Image Process.* **12**(5), 489–499 (2003)
- H Li, G Liu, Z Zhang, Optimization of integer wavelet transforms based on difference correlation structures. *IEEE Trans. Image Process.* **14**(11), 1831–1847 (2005)
- GCK Abhayaratne, G Piella, B Pesquet-Popescu, H Heijmans, Adaptive integer-to-integer wavelet transforms using update lifting, in *Proc. of SPIE*, vol. 5207 (2003), pp. 813–824
- GCK Abhayaratne, Spatially adaptive integer lifting with no side information for lossless video coding, in *Proc. Picture Coding Symposium (PCS)*, St. Malo, France, April 2003, pp. 495–500
- ÖN Gerek, AE Cetin, An edge-sensing predictor in wavelet lifting structures for lossless image coding. *EURASIP J. Image Video Coding* **2007**, Article ID 19313 (2007)
- J Solé, P Salembier, Generalized lifting prediction optimization applied to lossless image compression. *IEEE Signal Process. Lett.* **14**(10), 695–698 (2007)
- V Kitanovski, M Kseneman, D Gleich, D Taskovski, Adaptive lifting integer wavelet transform for lossless image compression, in *Proc. of IWSSIP*, Bratislava, Slovakia, 25–28 June 2008
- DBH Tay, A class of lifting based integer wavelet transform, in *IEEE Int. Conf. on Image Processing*, vol. 1, Thessaloniki, Greece, 07–10 Oct. 2001, pp. 602–605
- Z Guangjun, C Lizhi, C Huowang, A simple 9/7-tap wavelet filter based on lifting scheme. *IEEE Int. Conf. Image Process.* **2**, 249–252 (2001)
- S Barua, KA Kotteri, AE Bell, JE Carletta, Optimal quantized lifting coefficients for the 9/7 wavelet, in *ICASSP'04*, vol. 5, 17–21 May 2004, pp. 193–196
- T Strutz, Wavelet filter design based on the lifting scheme and its application in lossless image compression. *WSEAS Trans. Signal Process.* **5**(2), 53–62 (2009)
- M Iwahashi, H Kiya, Non separable 2D factorization of separable 2D DWT for lossless image coding, in *Proc. of ICIP*, Cairo, Egypt, 7–11 Nov 2009, pp. 17–20
- M Iwahashi, H Kiya, A new lifting structure of non separable 2D DWT with compatibility to JPEG 2000, in *Proc. of ICASSP*, Dallas, Texas, USA, 14–19 March 2010, pp. 1306–1309
- T Strutz, Design of three-channel filter banks for lossless image compression, in *Proc. of ICIP*, Cairo, Egypt, 7–11 Nov 2009, pp. 2841–2844
- D Le Gall, A Tabatabai, Sub-band coding of digital images using symmetric short kernel filters and arithmetic coding techniques, in *Proc. of ICASSP*, New York, NY, USA, 11–14 Apr 1988, pp. 761–764
- G Deslauriers, S Dubuc, Symmetric iterative interpolation process. *Constr. Approx.* **5**, 49–68 (1989)
- G Strang, T Nguyen, *Wavelets and Filter Banks* (Wellesley-Cambridge Press, Wellesley, 1996)
- L Cheng, DL Liang, ZH Zhang, Popular biorthogonal wavelet filters via a lifting scheme and its application in image compression. *IEE Proc., Vis. Image Signal Process.* **150**(4), 227–232 (2003)
- M Antonini, M Barlaud, P Mathieu, I Daubechies, Image coding using wavelet transform. *IEEE Trans. Image Process.* **1**(2), 205–220 (1992)
- I Daubechies, W Sweldens, Factoring wavelet transform into lifting steps. *J. Fourier Anal. Appl.* **4**(3), 247–269 (1998)
- www1.hft-leipzig.de/strutz/Papers/Testimages/Bath-AC-UK/bath.html, Dezember 2010 (copied from the former web page www.bath.ac.uk/elec-eng/research/sipg/resource/images.htm)
- ITU-T Recommendation T.24: Standardized digitized image set, Version 3, 1998
- http://r0k.us/graphics/kodak, Dec. 2009

doi:10.1186/1687-6180-2012-75

**Cite this article as:** Strutz and Rennert: Two-dimensional integer wavelet transform with reduced influence of rounding operations. *EURASIP Journal on Advances in Signal Processing* 2012 **2012**:75.