Available at
www.ElsevierComputerScience.com
POWERED BY SCIENCE DIRECT®
Signal Processing 85 (2005) 1103–1124
SIGNAL PROCESSING
www.elsevier.com/locate/sigpro

Single-rate calculation of overcomplete discrete wavelet transforms for scalable coding applications

Yiannis Andreopoulos*, Adrian Munteanu, Geert Van der Auwera, Jan Cornelis, Peter Schelkens

Vrije Universiteit Brussel, Department of Electronics and Information Processing (ETRO), Pleinlaan 2, B1050, Belgium

Received 13 December 2002; received in revised form 14 November 2004

Abstract

A number of emerging resolution-scalable image and video coding algorithms have recently shown very promising performance due to the use of overcomplete wavelet representations. In these applications, the overcomplete discrete wavelet transform (ODWT) is derived starting from the critically-sampled subbands of the DWT (complete representation) of a certain decomposition (resolution) level. This process, which is a complete-to-overcomplete DWT (CODWT), is used for wavelet domain operations that require shift invariance. Specifically, both the encoder and decoder independently construct the overcomplete representation at the best accuracy possible, given the critically-sampled subbands of a certain resolution level. In contrast to the classical approach for performing the CODWT, which is a *multi-rate* calculation scheme that requires the reconstruction of the input spatial-domain signal, in this paper we propose a new, *single-rate* calculation scheme, which is formalized for the general case of an arbitrary decomposition (resolution) level. Based on derived symmetry properties, a simple implementation structure of the proposed approach provides interesting tradeoffs for the required multiplication budget in comparison to the conventional approach. This leads to a complexity-scalable solution that fits the versatile requirements of scalable coding environments. The use of the proposed single-rate calculation scheme of the CODWT is demonstrated in several image and video coding systems.

Keywords: Overcomplete discrete wavelet transform; Complexity reduction; Resolution-scalable image and video coding; Shift invariance

1. Introduction

Recently, new techniques have been proposed for wavelet-based resolution-scalable image and video coding systems that utilize the overcomplete discrete wavelet transform (ODWT) [1–11].

*Corresponding author. Tel.: +32 2 629 3951; fax: +32 2 629 2883.
E-mail address: yandreop@etro.vub.ac.be (Y. Andreopoulos).

0165-1684/$ - see front matter © 2005 Elsevier B.V. All rights reserved.
doi:10.1016/j.sigpro.2005.01.006

The competitive coding performance of these techniques stems from the use of wavelet-domain operations that require shift-invariance, such as in-band motion estimation and compensation [1–10], or phase-based prediction of wavelet coefficients [11]. The ODWT is a shift-invariant wavelet representation of the input signal, and is used in these systems because the critically sampled DWT is only periodically shift-invariant due to the downsampling operations of each decomposition level [12].

Given the input signal, the classical construction of the ODWT is trivial by using for example the "a-trous" algorithm [12]. However, in coding systems where the decoder receives the critically sampled DWT subbands in a *resolution-scalable* manner (i.e. from coarse to fine resolution [1–11]), both ends of the system have to *independently* construct the identical ODWT information at each resolution level. In these systems, the level-by-level complete-to-overcomplete DWT (CODWT) provides the best shift-invariant approximation of the ODWT, given the resolution levels that are available at both the encoder and decoder sides [1–11].

The conventional level-by-level CODWT can be performed by using the framework proposed in [1]. Hence, for each resolution level of the critically sampled transform, the input signal is reconstructed with a certain accuracy by performing the inverse DWT (IDWT). Subsequently, the subbands of the ODWT of the current level are constructed by performing a number of forward transforms that retain the even and odd polyphase components of the undecimated decomposition [1]. This conventional approach has several disadvantages:

- The reconstruction of the input signal X is required for each resolution level. This causes significant calculation overhead and delay since the input signal has the highest sampling rate.
- A multi-rate calculation scheme is used. As a result, the achievable level of parallelism is limited since the filtering of every level has to be pipelined with the production of the results of the previous level.

In this paper, we present a new level-by-level CODWT, formulated by a generic set of propositions, in which the ODWT is produced *directly* from the critically sampled DWT of a certain resolution level by using a set of prediction filters [4,7]. The proposed approach provides a *single-rate* calculation scheme that allows the *maximum* level of parallelization, while providing exactly the same results as the conventional multi-rate scheme of [1] operating in a resolution-scalable framework.

The paper is structured as follows. Section 2 introduces the notations used in the paper and presents the conventional method for the calculation of the level-by-level CODWT. In addition, the application framework for coding systems based on the CODWT is highlighted. In Section 3 the proof of our propositions for the general case of k decomposition levels is given, while Section 4 presents a complexity study that identifies the implementation benefits offered by the proposed approach. Finally, in order to demonstrate potential applications for the proposed scheme, we present in Section 5 some indicative results for resolution-scalable image and video coding that utilize the level-by-level CODWT.

2. Preliminaries-problem formulation

2.1. Definitions and notations

In this paper, the low-pass and high-pass decomposition and reconstruction filters are denoted as $H(z)$, $G(z)$ and $\widetilde{H}(z)$, $\widetilde{G}(z)$, respectively. The low and high-frequency subbands (i.e. the approximation and the details) are denoted as A and D, respectively. In the Z-domain expressions (e.g. $A_4^3(z)$, $F_j^i(z)$), all the used indices are integers and the superscripts always denote the decomposition level. We use the typical definitions for the type-I polyphase components [12], i.e. $X_0(z)=\frac{1}{2}(X(z^{1/2})+X(-z^{1/2}))$, $X_1(z)=\frac{1}{2}z^{-1/2}(X(z^{1/2})-X(-z^{1/2}))$ for signal $X(z)$, while $F_0(z)=\frac{1}{2}(F(z^{1/2})+F(-z^{1/2}))$, $F_1(z)=\frac{1}{2}z^{1/2}(F(z^{1/2})-F(-z^{1/2}))$ denote the even, odd polyphase components of filter $F(z)$.[1] For the reconstruction filters $\widetilde{H}(z)$, $\widetilde{G}(z)$, we use the type-II

[1] In the general case of filter $F_j^i(z)$, the Type-I polyphase components are denoted as $F_{j,0}^i(z)$, $F_{j,1}^i(z)$.

Table 1
The 5/3 and 9/7 filter-pairs modified so that $\Delta = H_0(z)G_1(z) - G_0(z)H_1(z) = -1$ (filter G is linear phase)

Degree in Z	$H(z)$	Degree in Z	$G(Z)$
5/3 *Filter pair*			
1, −3	−0.17677669529664	1, −1	0.70710678118655
0, −2	0.35355339059327	0	−0.35355339059327
−1	1.06066017177982		
9/7 *Filter-pair*			
3, −5	0.03782845550726	3, −3	0.06453888262870
2, −4	−0.02384946501956	2, −2	−0.04068941760916
1, −3	−0.11062440441844	1, −1	−0.41809227322162
0, −2	0.37740285561283	0	0.78848561640558
−1	0.85269867900889		

polyphase representation [12], i.e. $\widetilde{H}_0(z) = \frac{1}{2}z^{1/2}(\widetilde{H}(z^{1/2}) - \widetilde{H}(-z^{1/2}))$ and $\widetilde{H}_1(z) = \frac{1}{2}(\widetilde{H}(z^{1/2}) + \widetilde{H}(-z^{1/2}))$. Furthermore, for biorthogonal point symmetric filter-pairs, the relations $\tilde{H}_0(z) = \Delta^{-1}G_1(z)$, $\tilde{H}_1(z) = -\Delta^{-1}G_0(z)$, $\tilde{G}_0(z) = -\Delta^{-1}H_1(z)$, $\tilde{G}_1(z) = \Delta^{-1}H_0(z)$ between the polyphase components of the decomposition and reconstruction filters in a filter-bank are used, where $\Delta = H_0(z)G_1(z) - H_1(z)G_0(z)$ is the determinant of the analysis polyphase matrix. These relationships can be immediately verified based on the perfect reconstruction condition. In the following, in order to simplify the expressions we always assume that the filters $H(z)$ and $G(z)$ are properly shifted so that $\Delta = -1$. For example, for biorthogonal point-symmetric filters we can satisfy this constraint, if:

$$G(z) = G(z^{-1}), \quad zH(z) = z^{-1}H(z^{-1}), \tag{1}$$

i.e. if $G(z)$ is linear-phase and $H(z)$ is point-symmetric around z^{-1}. Two such examples of biorthogonal point-symmetric filter-pairs, which are used for the results of this paper, are given in Table 1.

2.2. Application framework for the resolution-scalable CODWT

In this subsection we define the generic coding framework in which the level-by-level CODWT is employed. For simplicity, we present the decoder design; in each application the encoder design is following the reverse order of operations performed by the decoder.

A generic architecture that indicates the core functionalities of a resolution-scalable decoding system that utilizes the CODWT is given in Fig. 1. The processing starts by decoding the compressed information of the current decomposition level l, i.e. the three high-frequency subbands and potentially the low-frequency subband, if l constitutes the coarsest decomposition level [1–11]. Then, after receiving any potential auxiliary information from the compressed bit-stream, the CODWT(l) module of Fig. 1 creates the overcomplete representation of the current level. The auxiliary information can be used in the construction of the CODWT in order to limit the calculation to certain subbands of the ODWT or to dynamically create portions of the utilized subband contents for block-based operations [8,11]. After the necessary ODWT information is created, the operation that requires shift-invariance takes place in the ODWT subbands. For example, in a video coding system this operation can be wavelet-domain motion estimation and compensation [1–10], or in wavelet-based still image coding it can be the phase-based prediction of the wavelet coefficients using information from the coarser resolution level $(l+1)$ [11]. Auxiliary information from the compressed bit-stream is used during this process as well, for example decoded motion-vector data and phase indices for the utilized ODWT subbands.

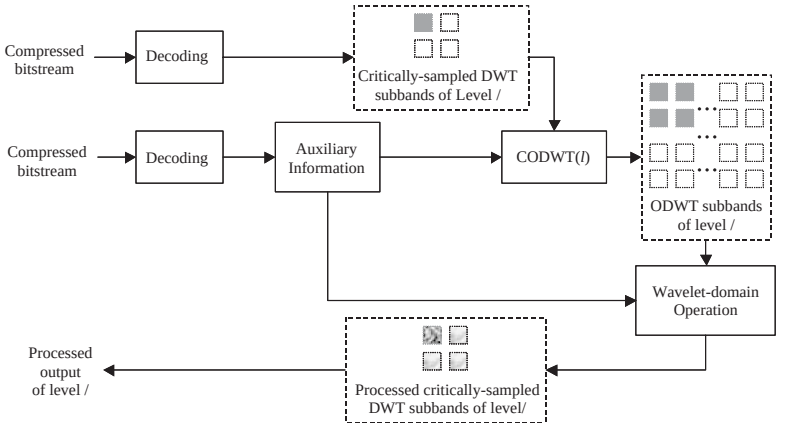

Fig. 1. The general architecture of a resolution-scalable decoder performing a wavelet-domain operation that requires shift invariance.

As seen in Fig. 1, the resulting output from the wavelet-domain operation leads to a modification of the subbands of the current resolution level (e.g. for the case of motion compensation); potentially, it can be also stored for later usage (e.g. for the phase-based prediction of the next resolution level). Concrete examples of such systems that yield good coding efficiency as well as comparisons to conventional approaches that operate in the spatial domain are demonstrated in Section 5.

2.3. Problem formulation

The CODWT module is the key part in a resolution-scalable coding system that utilizes wavelet-domain operations requiring shift-invariance. In this section, we briefly present the two methods for this module, namely the conventional LBS method of [1] with its level-by-level adaptation, and the proposed prediction-filters method.

Fig. 2(a) shows an example of the 1-D ODWT construction for two decomposition levels starting from an input signal X. This figure facilitates the description of the LBS method of [1]. Initially, the input signal X is decomposed in two subband sets A_0^1, D_0^1 & A_1^1, D_1^1 by retaining separately the even and odd polyphase components of the undecimated decomposition respectively, or equivalently, by performing two wavelet decompositions: one to the zero-shifted and one to the unit-shifted input signals, respectively [1]. Each of the low-frequency subbands A_0^1 and A_1^1 is further analyzed in the same manner, while the high-frequency subbands D_0^1 and D_1^1 are the output of the first decomposition level. The process is repeated recursively, yielding the conventional approach for the ODWT calculation from the input signal X (see Fig. 2(a)). The subbands A_0^2 and D_0^l, $l=1,2$ represent the critically sampled DWT of two levels, while the subbands A_i^2, D_i^l, $1 \leqslant l \leqslant 2$, $0 \leqslant i < 2^l$ represent the calculated ODWT for two decomposition levels.

Notice that the ODWT subbands A_i^2, D_i^l shown in Fig. 2(a) stem from the classical ODWT decomposition scheme of [14], which is equivalent to the "à trous" algorithm [12]. The difference is that, at every level, the subbands of Fig. 2(a) must be interleaved in order to produce the non-decimated ODWT obtained with the algorithm of [14]. As a result, any subband D_i^l in the ODWT of Fig. 2(a) is the ith polyphase component of the non-decimated ODWT of level l [14,12].

In the two-dimensional case, the 2-D ODWT can be constructed in the same manner as in Fig. 2(a), by applying the LBS method on the input-subband rows and on the columns of the results [4,1]. Hence, to facilitate the description, we focus in the following on the one-dimensional case, with the extension in two dimensions following the row-column approach.

In a resolution-scalable coding framework, the key-difference with respect to non-scalable coding is that the subband-transmission and decoding

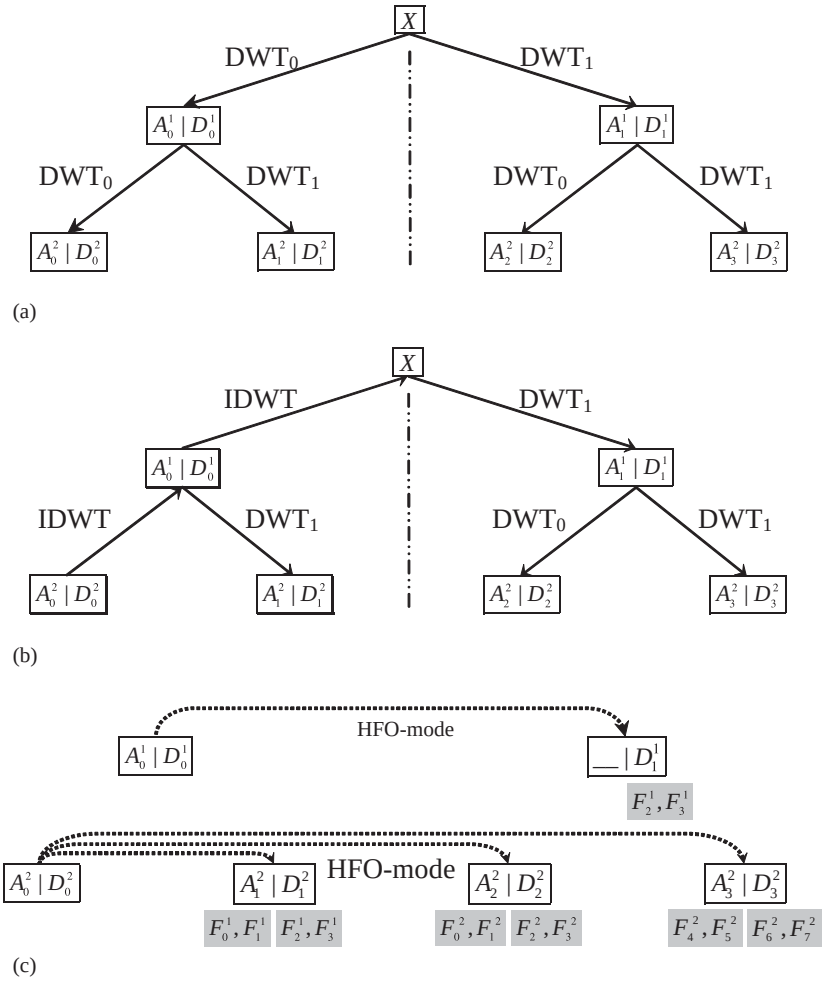

Fig. 2. (a) The construction of the ODWT of two levels starting from the input signal X. A number of one-level discrete wavelet transforms are performed that retain the even or odd samples of the undecimated transform (DWT_0 and DWT_1 respectively); (b) The level-by-level CODWT (example for level two) using the conventional multi-rate LBS approach that performs a set of inverse and forward transforms; (c) The level-by-level CODWT for both levels using the proposed single-rate approach. The prediction filters are denoted as F_i^l, $1 \leqslant l \leqslant 2$, $0 \leqslant i < 2^{l+1}$.

occur in a bottom-up manner [4,5,7–10]: the coarsest-resolution subbands of the critically sampled decomposition are received first (subbands A_0^2, D_0^2) and for every higher-resolution level l (in this example, $l = 1$), the subband D_0^1 is received. Under such a resolution-progressive decoding environment, the LBS method is readily adaptable to perform a *level-by-level* construction of the ODWT representation (denoted by LL-LBS), starting from the subbands of the critically sampled transform of each decoded level. This process is illustrated in Fig. 2(b). There, starting from subbands A_0^2, D_0^2 (coarsest resolution level), two inverse wavelet transforms are performed. Subsequently, from the reconstructed signal ($\overline{X}$), all the subbands A_i^2, D_i^2, $1 \leqslant i < 4$ are constructed by performing the forward transforms shown in Fig. 2(b). Since in this case the subband D_0^1 is not available, and due to the fact that lossy decoding may have occurred for the subbands A_0^2, D_0^2, the

reconstructed signal $\overline{X}$ and the subbands A_i^2, D_i^2 are only an approximation of X and of the original ODWT of level two, respectively, shown in Fig. 2(a). In this resolution-scalable scenario, given the information available at both the encoder and decoder sides, this ODWT representation is the best possible approximation for the current resolution level.

As soon as more resolution levels are received, the construction of the ODWT by the LL-LBS is repeated for the finer resolution levels (level one). Notice that for the CODWT of every decomposition level l, the low-frequency subbands A_j^l, $1 \leqslant j < 2^l$ consist a part of the ODWT only if level l is the coarsest-resolution of the decomposition (i.e. if $l = 2$ in the example of Fig. 2(c)). We denote this construction as the *full-overcomplete* transform-production mode (FO-mode). In all the other cases (level $l = 1$ in the example of Fig. 2(c)), the critically sampled DWT consists of the subband D_0^l and hence only the high-frequency ODWT subbands D_j^l need to be constructed [1,7]; this case is denoted as the *high-frequency overcomplete* transform-production mode (HFO-mode). The difference between the FO and HFO modes is illustrated in Fig. 2(c).

Additionally, Fig. 2(c) presents the proposed alternative approach to the LL-LBS method for the level-by-level CODWT of levels two and one. In this approach, the CODWT uses a set of prediction filters [4,7], denoted as F_j^i, $1 \leqslant i \leqslant 2$, $0 \leqslant j < 2^{i+1}$, which are convolved with the subbands A_0^i, D_0^i to calculate the ODWT representation of each level. Notice that for level one of Fig. 2(c), the ODWT construction occurs in the HFO mode, i.e. only the high-frequency subbands of the ODWT are calculated. Additionally, the dotted lines in the figure illustrate that by using the prediction-filters, the overcomplete representation is "predicted" in a level-by-level manner from the critically sampled representation. As a result, no upsampling or downsampling is performed with this algorithm and no reconstruction of the spatial-domain signal X is performed. The mathematical derivation of the prediction filters for biorthogonal filter-banks and the proposed level-by-level CODWT are presented in the next section.

3. Level-by-level CODWT

In this section we present a generic framework for the CODWT based on the prediction-filters. The mathematical formulation is derived for the one-dimensional case with the two-dimensional extension following the row-column approach. In specific, the subbands of decomposition level k of the overcomplete representation are expressed as a function of the critically sampled subbands of the current resolution level (i.e. the subbands A_0^k, D_0^k). As a result, this approach presents the CODWT of level k under the assumption that $D_0^l = 0$, $1 \leqslant l < k$, i.e. under resolution-scalable (de)coding. Finally, for the best-performing filter-bank family in image coding [15], a set of symmetry properties for the corresponding prediction filters of every decomposition level is proven.

3.1. Derivation of the ODWT subbands of decomposition level k from the critically sampled subbands of level k

This subsection presents the generalized form of the level-by-level CODWT based on the prediction filters. The level-by-level CODWT is defined by a set of propositions $\mathcal{Q}(k)$ formulated for every decomposition level k.

3.1.1. Definition of propositions $\mathcal{Q}(1)$, $\mathcal{Q}(2)$

The proposition $\mathcal{Q}(1)$ corresponding to $k = 1$ is

$$\mathcal{Q}(1): \{A_1^1(z) = F_0^1(z)A_0^1(z) + F_1^1(z)D_0^1(z),\quad D_1^1(z) = F_2^1(z)A_0^1(z) + F_3^1(z)D_0^1(z)\}. \tag{2}$$

The prediction filters of the proposition $\mathcal{Q}(1)$ have the form

$$F_0^1(z) = \Delta^{-1}(H_1(z)G_1(z) - zH_0(z)G_0(z)),\quad F_1^1(z) = \Delta^{-1}(zH_0(z)H_0(z) - H_1(z)H_1(z)), \tag{3}$$

$$F_2^1(z) = \Delta^{-1}(G_1(z)G_1(z) - zG_0(z)G_0(z)),\quad F_3^1(z) = \Delta^{-1}(zH_0(z)G_0(z) - H_1(z)G_1(z)). \tag{4}$$

Table 2 shows two examples of the filters defined by Eqs. (3), (4) for the cases of the 5/3 and 9/7

Table 2
The prediction filters of level 1 for the 5/3 and the 9/7 filter pairs shown in Table 1

Degree in Z	$F_0^1(z)$	$F_1^1(z)$	$F_2^1(z)$	$F_3^1(z)$
5/3 *Filter pair*				
2	−0.0625	0.03125	−0.1250	0.0625
1	0.5625	−0.5	0.25	−0.5625
0	0.5625	0.9375	−0.125	−0.5625
−1	−0.0625	−0.5		0.0625
−2		0.03125		
9/7 *Filter-pair*				
4	−0.00244140625001	0.00143099204607	−0.00416526737096	0.00244140625001
3	0.02392578125006	−0.00893829770284	0.05562204500420	−0.02392578125006
2	−0.11962890624961	0.09475201933935	−0.18500077367828	0.11962890624961
1	0.59814453124955	−0.32145927076690	0.26708799209208	−0.59814453124955
0	0.59814453124955	0.46842911416863	−0.18500077367828	−0.59814453124955
−1	−0.11962890624961	−0.32145927076690	0.05562204500420	0.11962890624961
−2	0.02392578125006	0.09475201933935	−0.00416526737096	−0.02392578125006
−3	−0.00244140625001	−0.00893829770284		0.00244140625001
−4		0.00143099204607		

filter-pairs. For level $k=2$, the proposition $\mathcal{Q}(2)$ is

$$\mathcal{Q}(2):\begin{cases} A_1^2(z)=F_0^1(z)A_0^2(z)+F_1^1(z)D_0^2(z),\\ D_1^2(z)=F_2^1(z)A_0^2(z)+F_3^1(z)D_0^2(z),\\ A_2^2(z)=F_0^2(z)A_0^2(z)+F_1^2(z)D_0^2(z),\\ D_2^2(z)=F_2^2(z)A_0^2(z)+F_3^2(z)D_0^2(z),\\ A_3^2(z)=F_4^2(z)A_0^2(z)+F_5^2(z)D_0^2(z),\\ D_3^2(z)=F_6^2(z)A_0^2(z)+F_7^2(z)D_0^2(z). \end{cases} \tag{5}$$

The prediction filters of (5) are defined within the generic description given in Section 3.1.2, while Table 2 shows two examples of these filters for the 5/3 and 9/7 filter pairs. The proofs of $\mathcal{Q}(1)$ and $\mathcal{Q}(2)$ are summarized in [4]; for completeness they are briefly reviewed in Appendix A. The propositions of Eqs. (2) and (5) hint that, in general, the ODWT subbands of any level $k, k>1$ are derived from the DWT subbands of level k (A_0^k, D_0^k) by the single-rate filtering of these subbands with filters $F_i^l, 1\leqslant l\leqslant k,\ 0\leqslant i<2^{l+1}$. This intuitive link is mathematically formulated in the next subsection.

3.1.2. Definition of the set of propositions $\mathcal{Q}(k)$

The generalization of the previous propositions and the proof for an arbitrary level k is based on mathematical induction. Thus, we define the set of propositions $\mathcal{Q}(k)$ for an arbitrary level k and we derive the set of propositions $\mathcal{Q}(k+1)$ and the filters for level $k+1$. The formulas derived for level $k+1$ are true if, and only if, they can be derived from level k by replacing k with $k+1$. In addition, if they are true for level $k+1$, then they are true for any level. Let us assume that the set of propositions $\mathcal{Q}(k)$, $k\geqslant 1$ is true:

$$\mathcal{Q}(k):\begin{cases} A_x^k(z)=F_{4p}^{l+1}(z)A_0^k(z)+F_{4p+1}^{l+1}(z)D_0^k(z),\\ D_x^k(z)=F_{4p+2}^{l+1}(z)A_0^k(z)+F_{4p+3}^{l+1}(z)D_0^k(z). \end{cases} \tag{6}$$

In these equations, $1\leqslant x<2^k$ denotes the ODWT subband index at level k, and it is written as $x=2^l+p$, where l is given by $l=\lfloor\log_2 x\rfloor$ ($\lfloor a\rfloor$ denotes the integer part of a). For any x, it follows that $0\leqslant l<k$ and $0\leqslant p<2^l$. A pictorial representation of the prediction filters employed by the CODWT of two levels is given in Fig. 2(c).

The prediction filters needed to calculate the ODWT subbands of level k, $k>1$, are the filters $F_0^1, F_1^1, F_2^1, F_3^1$ of Eqs. (3), (4) and the filters $F_{8i}^{l+1}, F_{8i+1}^{l+1}, \ldots, F_{8i+7}^{l+1}$, with $0<l<k$ and $0\leqslant i<2^{l-1}$, satisfying the relationships:

$$F_{8i}^{l+1}(z)=F_{4i,0}^{l}(z)-z^{-1}F_3^1(z)F_{4i,1}^{l}(z), \tag{7}$$

$$F_{8i+1}^{l+1}(z) = z^{-1}F_1^1(z)F_{4i,1}^l(z), \tag{8}$$

$$F_{8i+2}^{l+1}(z) = z^{-1}F_2^1(z)F_{4i,1}^l(z), \tag{9}$$

$$F_{8i+3}^{l+1}(z) = F_{4i,0}^l(z) + z^{-1}F_3^1(z)F_{4i,1}^l(z), \tag{10}$$

$$F_{8i+4}^{l+1}(z) = F_{4i,1}^l(z) + F_0^1(z)F_{4i,0}^l(z), \tag{11}$$

$$F_{8i+5}^{l+1}(z) = F_1^1(z)F_{4i,0}^l(z), \tag{12}$$

$$F_{8i+6}^{l+1}(z) = F_2^1(z)F_{4i,0}^l(z), \tag{13}$$

$$F_{8i+7}^{l+1}(z) = F_{4i,1}^l(z) + F_3^1(z)F_{4i,0}^l(z), \tag{14}$$

3.1.3. Derivation of the set of propositions $\mathcal{2}(k+1)$

The set of propositions $\mathcal{2}(k+1)$ is derived in this subsection. In addition, the form of the prediction filters is proven as well. Let us start by performing an inverse wavelet transform in order to calculate the A_0^k subband in function of the A_0^{k+1} and D_0^{k+1} subbands:

$$A_0^k(z) = z[\tilde{H}(z)A_0^{k+1}(z^2) + \tilde{G}(z)D_0^{k+1}(z^2)]. \tag{15}$$

By performing a single-level forward transform retaining the odd samples (“complementary” transform), we obtain:

$$\begin{bmatrix} A_1^{k+1}(z) \\ D_1^{k+1}(z) \end{bmatrix} = \frac{1}{2} z^{1/2} \begin{bmatrix} H(z^{1/2}) & -H(-z^{1/2}) \\ G(z^{1/2}) & -G(-z^{1/2}) \end{bmatrix} \times \begin{bmatrix} A_0^k(z^{1/2}) \\ A_0^k(-z^{1/2}) \end{bmatrix}. \tag{16}$$

Eq. (16) consists the proof of the proposition of (6) for the specific case of $x = 1$: the replacement of the terms $A_0^k(z^{1/2})$ and $A_0^k(-z^{1/2})$ from (15) with the use of the relationships of Section 2.1 for the polyphase components of the filter-bank yields (6) with k replaced by $k+1$ and $l = p = 0$. In addition, the form of the derived prediction filters agrees with Eqs. (3), (4). For the remaining part of the proof, we have to account for the fact that in the resolution-scalable framework we have $D_0^k = 0$, i.e. this subband is not available at both the encoder and decoder that operate at the resolution level $k+1$. However, given the subband A_0^k now available from Eq. (15), we can apply the set of propositions of level k (involving the prediction filters of level k), since they are all true by assumption. Hence, we can calculate any subband A_x^k, $1 \leqslant x < 2^k$ with the use of the set of propositions $\mathcal{2}(k)$ of (6). By replacing (15) in Eq. (6) we obtain

$$A_x^k(z) = zF_{4p}^{l+1}(z)\tilde{H}(z)A_0^{k+1}(z^2) + zF_{4p}^{l+1}(z)\tilde{G}(z)D_0^{k+1}(z^2), \tag{17}$$

with the definitions of x, l, p given as for (6). As shown pictorially in Fig. 2(a), which can be seen as an example for the case of $k+1$ levels (with $k+1 = 2$), in order to calculate from subband A_x^k the $A_{2x}^{k+1}, D_{2x}^{k+1}$ subbands (even-numbered subbands of level $k+1$), we need to perform a single-level forward transform, retaining the even samples (“classic” transform):

$$\begin{bmatrix} A_{2x}^{k+1}(z) \\ D_{2x}^{k+1}(z) \end{bmatrix} = \frac{1}{2} \begin{bmatrix} H(z^{1/2}) & H(-z^{1/2}) \\ G(z^{1/2}) & G(-z^{1/2}) \end{bmatrix} \times \begin{bmatrix} A_x^k(z^{1/2}) \\ A_x^k(-z^{1/2}) \end{bmatrix}. \tag{18}$$

In Eq. (18), we can replace $A_x^k(z^{1/2})$ and $A_x^k(-z^{1/2})$ by using (17). The filter $F_{4p}^{l+1}(z^{1/2})$ can be written using the type-I polyphase representation, while $\tilde{H}(z^{1/2})$ and $\tilde{G}(z^{1/2})$ can be substituted by the type-II polyphase representation. As a result we obtain the expression for $A_x^k(z^{1/2})$ from (17) as

$$\begin{aligned} A_x^k(z^{1/2}) &= [(z^{1/2}G_0(z) - G_1(z))F_{4p,0}^{l+1}(z) \\ &\quad + (G_0(z) - z^{-1/2}G_1(z))F_{4p,1}^{l+1}(z)]A_0^{k+1}(z) \\ &\quad + [(H_1(z) - z^{1/2}H_0(z))F_{4p,0}^{l+1}(z) + (z^{-1/2}H_1(z) \\ &\quad - H_0(z))F_{4p,1}^{l+1}(z)]D_0^{k+1}(z). \end{aligned} \tag{19}$$

We denote the factors multiplying the subbands A_0^{k+1}, D_0^{k+1} in (19) as $L_{\text{even\_}A}^{k+1}$ and $L_{\text{even\_}D}^{k+1}$, respectively.

Similarly as above, we can obtain the expression of $A_x^k(-z^{1/2})$ from Eq. (17) by using the type-I polyphase representation of filter $F_{4p}^{l+1}(-z^{1/2})$ and the type-II polyphase representation of filters

$\widetilde{H}(-z^{1/2}), \widetilde{G}(-z^{1/2})$. The result is

$$\begin{aligned} A_x^k(-z^{1/2}) &= [(-z^{1/2}G_0(z) - G_1(z))F_{4p,0}^{l+1}(z) + (G_0(z) \\ &\quad + z^{-1/2}G_1(z))F_{4p,1}^{l+1}(z)]A_0^{k+1}(z) + [(H_1(z) \\ &\quad + z^{1/2}H_0(z))F_{4p,0}^{l+1}(z) + (-z^{-1/2}H_1(z) \\ &\quad - H_0(z))F_{4p,1}^{l+1}(z)]D_0^{k+1}(z). \end{aligned} \tag{20}$$

We denote the factors multiplying the two subbands A_0^{k+1}, D_0^{k+1} as $L_{\text{odd\_}A}^{k+1}$ and $L_{\text{odd\_}D}^{k+1}$, respectively. The final expressions of $A_{2x}^{k+1}, D_{2x}^{k+1}$ are obtained by replacing Eqs. (19) and (20) in Eq. (18):

$$\begin{bmatrix} A_{2x}^{k+1}(z) \\ D_{2x}^{k+1}(z) \end{bmatrix} = \frac{1}{2}\begin{bmatrix} H(z^{1/2}) & H(-z^{1/2}) \\ G(z^{1/2}) & G(-z^{1/2}) \end{bmatrix} \times \begin{bmatrix} L_{\text{even\_}A}^{k+1}(z)A_0^{k+1}(z) + L_{\text{even\_}D}^{k+1}(z)D_0^{k+1}(z) \\ L_{\text{odd\_}A}^{k+1}(z)A_0^{k+1}(z) + L_{\text{odd\_}D}^{k+1}(z)D_0^{k+1}(z) \end{bmatrix}. \tag{21}$$

Eq. (21) shows that the calculation of the subbands $A_{2x}^{k+1}, D_{2x}^{k+1}$ consists of calculations of factors like

$$\tfrac{1}{2}[H(z^{1/2})L_{\text{even\_}A}^{k+1}(z) + H(-z^{1/2})L_{\text{odd\_}A}^{k+1}(z)]$$

and

$$\tfrac{1}{2}[H(z^{1/2})L_{\text{even\_}D}^{k+1}(z) + H(-z^{1/2})L_{\text{odd\_}D}^{k+1}(z)]$$

that are multiplied with A_0^{k+1} and D_0^{k+1}, respectively. These factors are the prediction filters of the even-numbered subbands of level $k+1$: replacing $L_{\text{even\_}A}^{k+1}$, $L_{\text{even\_}D}^{k+1}$, $L_{\text{odd\_}A}^{k+1}$ and $L_{\text{odd\_}D}^{k+1}$ in these factors yields Eqs. (7)–(10) with l replaced by $l' = l+1$ and hence $1 < l' < k+1$, while the ODWT subband index $2x$ of (21) is bounded by: $2 \leqslant 2x < 2^{k+1}$. As a result, Eq. (21) corresponds to Eq. (6) for the even-numbered subbands, with the replacement of k with $k+1$. Hence, in this case the proposition $\mathcal{Q}(k+1)$ is true.

Similarly, in order to calculate the subbands $A_{2x+1}^{k+1}, D_{2x+1}^{k+1}$ (odd-numbered subbands of level $k+1$), we perform a single-level forward transform retaining the odd samples ("complementary" transform):

$$\begin{bmatrix} A_{2x+1}^{k+1}(z) \\ D_{2x+1}^{k+1}(z) \end{bmatrix} = \frac{1}{2}z^{1/2}\begin{bmatrix} H(z^{1/2}) & -H(-z^{1/2}) \\ G(z^{1/2}) & -G(-z^{1/2}) \end{bmatrix} \times \begin{bmatrix} A_x^k(z^{1/2}) \\ A_x^k(-z^{1/2}) \end{bmatrix}. \tag{22}$$

The terms $A_x^k(z^{1/2}), A_x^k(-z^{1/2})$ are given by (19) and (20); replacing them in Eq. (22) yields

$$\begin{bmatrix} A_{2x+1}^{k+1}(z) \\ D_{2x+1}^{k+1}(z) \end{bmatrix} = \frac{1}{2}z^{1/2}\begin{bmatrix} H(z^{1/2}) & -H(-z^{1/2}) \\ G(z^{1/2}) & -G(-z^{1/2}) \end{bmatrix} \times \begin{bmatrix} L_{\text{even\_}A}^{k+1}(z)A_0^{k+1}(z) + L_{\text{even\_}D}^{k+1}(z)D_0^{k+1}(z) \\ L_{\text{odd\_}A}^{k+1}(z)A_0^{k+1}(z) + L_{\text{odd\_}D}^{k+1}(z)D_0^{k+1}(z) \end{bmatrix}. \tag{23}$$

Eq. (23) shows that the calculation of the subbands $A_{2x+1}^{k+1}, D_{2x+1}^{k+1}$ consists of factors like

$$\tfrac{1}{2}[z^{1/2}H(z^{1/2})L_{\text{even\_}A}^{k+1}(z) - z^{1/2}H(-z^{1/2})L_{\text{odd\_}A}^{k+1}(z)]$$

and

$$\tfrac{1}{2}[z^{1/2}H(z^{1/2})L_{\text{even\_}D}^{k+1}(z) - z^{1/2}H(-z^{1/2})L_{\text{odd\_}D}^{k+1}(z)]$$

that multiply the subbands A_0^{k+1} and D_0^{k+1}, respectively. These factors are the prediction filters of the odd-numbered subbands of level $k+1$: replacing $L_{\text{even\_}A}^{k+1}$, $L_{\text{even\_}D}^{k+1}$, $L_{\text{odd\_}A}^{k+1}$ and $L_{\text{odd\_}D}^{k+1}$ in these factors yields Eqs. (11)–(14) with l replaced by $l' = l+1$ and hence $1 < l' < k+1$, while the ODWT subband index $2x+1$ of (23) is bounded by $1 \leqslant 2x+1 < 2^{k+1}$. As a result, (23) corresponds to Eq. (6) for the odd-numbered subbands, with the replacement of k with $k+1$. Hence, $\mathcal{Q}(k+1)$ is again true in this case. We conclude by induction that $\mathcal{Q}(k)$ is true for every decomposition level k.

Finally, we remark that the set of prediction filters of level k consists of the set of filters of level $k-1$ augmented by the set of filters F_i^k, $0 \leqslant i < 2^{k+1}$. In the next subsection we demonstrate a set of symmetry properties for these filters.

3.2. Properties of the prediction filters for biorthogonal point-symmetric filter-pairs

In this subsection we discuss certain properties of the previously derived prediction filters. We focus on the case of point-symmetric biorthogonal filter-pairs, since they construct some of the best-performing wavelets in still-image coding [15]. For this category, a set of symmetry properties between the prediction filters F_i^k, $0 \leqslant i < 2^{k+1}$ exists for any decomposition level k, $k>1$. The mathematical formalism describing these symmetries is:

$$\mathscr{P}_{\mathscr{S}}(k): \begin{cases} F_{4m}^k(z) = zF_{4(2^{k-1}-m-1)}^k(z^{-1}), \\ F_{4m+1}^k(z) = F_{4(2^{k-1}-m-1)+1}^k(z^{-1}), \\ F_{4m+2}^k(z) = z^2F_{4(2^{k-1}-m-1)+2}^k(z^{-1}), \\ F_{4m+3}^k(z) = zF_{4(2^{k-1}-m-1)+3}^k(z^{-1}). \end{cases} \quad (24)$$

with $0 \leqslant m < 2^{k-2}$ and $k>1$. The interested reader can find the proof of these properties in Appendix B.

Eq. (24) indicates that for the calculation of the subbands $A_i^k, D_i^k, 1<i<2^k, k>1$, we can derive half of the corresponding F-filters as the time-inverses of the other half of the filters under some shifts. Specifically, the filters are time-inversed in groups of four filters that lay in a "mirror" fashion in the group of prediction filters. Thus, by looking at the indices of (24), the first four F-filters are related with the last four, the second four F-filters with the penultimate four, and so on. This is also numerically shown in Table 3 where we display the prediction filter taps for the biorthogonal point-symmetric 5/3 and 9/7 filter-pairs for $k=2$.

4. Benefits of the proposed single-rate CODWT

In this section, in order to evaluate the benefit of the single-rate characteristics of the proposed method in comparison to the multi-rate LL-LBS approach, we estimate the multiplication requirements for the production of the results of every level. The focus on the multiplication requirements makes the theoretical calculations of this section more applicable to custom-hardware (VLSI) architectures rather than processor-based realizations. We analyze the one-dimensional case of an N-sample signal, with the extension to two dimensions following immediately. Some state-of-the-art, custom-hardware designs for the high-parallel implementation of the DWT can be found in [16,17]; for the theoretical analysis of this section we assume that the reader is familiar with the concepts of efficient DWT implementations.

4.1. Required multiplications for the proposed, single-rate calculation scheme

The symmetry properties expressed in Section 3.2 indicate that for each level $k, k>1$, the multiplications performed for half of the ODWT subbands of level k can be reused for the production of the other half (excluding subbands A_1^k, D_1^k) by using a set of lattice structures, such as the one shown in Fig. 3. For instance, for the example of the second resolution level of Fig. 2(c), the multiplications performed for the calculation of the subbands A_2^2, D_2^2 are reused for the production of A_3^2, D_3^2 (following Eq. (24) for $k=2$). Using the symmetry properties (24), we derive the necessary multiplications for the proposed single-rate approach. In addition, the symmetry properties of the filters of the first level, as expressed by Eqs. (41)–(44) of Appendix B, are used in order to reduce the required multiplications for the production of subbands A_1^k, D_1^k. In particular, for the FO and HFO modes defined in Section 2.3, the number of multiplications required to produce the ODWT subbands of level k with the proposed prediction-filters approach is

$$M_{\text{P-F,FO}}(k) = \frac{N}{2^k}\left(\sum_{i=0}^{3}\left\lfloor\frac{T_{F_i^1}+1}{2}\right\rfloor + \sum_{l=2}^{k}\sum_{i=0}^{2^l-1} T_{F_i^l}\right), \quad (25)$$

$$M_{\text{P-F,HFO}}(k) = \frac{N}{2^k}\left[\left\lfloor\frac{T_{F_2^1}+1}{2}\right\rfloor + \left\lfloor\frac{T_{F_3^1}+1}{2}\right\rfloor + \sum_{l=2}^{k}\sum_{i=0}^{2^{l-2}-1}(T_{F_{4i+2}^l} + T_{F_{4i+3}^l})\right], \quad (26)$$

Table 3
The prediction filters of level 2 for the 5/3 and the 9/7 filter pairs

5/3 *Filter pair* Degree in Z	$F_0^2(z)$	$F_1^2(z)$	$F_2^2(z)$	$F_3^2(z)$
2	−0.03515625	0.017578125	−0.0703125	0.03515625
1	0.2578125	−0.283203125	0.1484375	−0.3828125
0	0.843750	0.55859375	−0.0859375	0.28125
−1	−0.0703125	−0.33984375	0.0078125	0.0703125
−2	0.00390625	0.048828125		−0.00390625
−3		−0.001953125		
	$F_4^2(z)$	$F_5^2(z)$	$F_6^2(z)$	$F_7^2(z)$
3	0.00390625	−0.001953125	0.0078125	−0.00390625
2	−0.0703125	0.048828125	−0.0859375	0.0703125
1	0.843750	−0.33984375	0.1484375	0.28125
0	0.2578125	0.55859375	−0.0703125	−0.3828125
−1	−0.03515625	−0.283203125		0.03515625
−2		0.017578125		

9/7 *Filter pair* Degree in Z	$F_0^2(z)$	$F_1^2(z)$	$F_2^2(z)$	$F_3^2(z)$
5	−0.00005841255188	0.00003423760266	−0.00009965727597	0.00005841255188
4	−0.00088787078858	0.00064208431103	−0.00116063101768	0.00088787078858
3	0.01174092292788	−0.00325056581524	0.02934202037389	−0.01174092292788
2	−0.06254196166961	0.05005002314451	−0.11091074747671	0.05765914916960
1	0.26671171188334	−0.19238483073456	0.17732657799233	−0.50596952438255
0	0.88179588317802	0.31072164151829	−0.14082618249241	0.31449317932109
−1	−0.12007284164365	−0.24526493865167	0.05464973467748	0.16792440414377
−2	0.02710342407219	0.09377374163572	−0.00869377426124	−0.02710342407219
−3	−0.00403046607971	−0.01586242405270	0.00036249037151	0.00403046607971
−4	0.00023365020752	0.00169389067232	0.00001016910979	−0.00023365020752
−5	0.00000596046448	−0.00014936599745		−0.00000596046448
−6		−0.00000349363292		
	$F_4^2(z)$	$F_5^2(z)$	$F_6^2(z)$	$F_7^2(z)$
6	0.00000596046448	−0.00000349363292	0.00001016910979	−0.00000596046448
5	0.00023365020752	−0.00014936599745	0.00036249037151	−0.00023365020752
4	−0.00403046607971	0.00169389067232	−0.00869377426124	0.00403046607971
3	0.02710342407219	−0.01586242405270	0.05464973467748	−0.02710342407219
2	−0.12007284164365	0.09377374163572	−0.14082618249241	0.16792440414377
1	0.88179588317802	−0.24526493865167	0.17732657799233	0.31449317932109
0	0.26671171188334	0.31072164151829	−0.11091074747671	−0.50596952438255
−1	−0.06254196166961	−0.19238483073456	0.02934202037389	0.05765914916960
−2	0.01174092292788	0.05005002314451	−0.00116063101768	−0.01174092292788
−3	−0.00088787078858	−0.00325056581524	−0.00009965727597	0.00088787078858
−4	−0.00005841255188	0.00064208431103		0.00005841255188
−5		0.00003423760266		

where $M_{\text{P-F,FO}}(k)$ and $M_{\text{P-F,HFO}}(k)$ are the number of multiplications performed in decomposition level k of an N-sample input signal when the CODWT is calculated in FO and HFO modes, respectively, and $T_{F_i^l}$ denotes the number of non-zero taps of filter F_i^l.

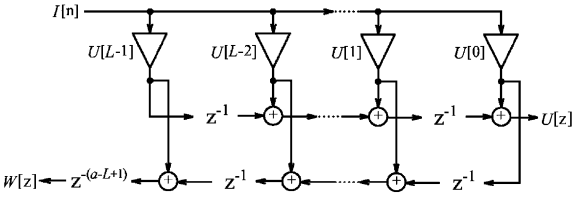

Fig. 3. A lattice structure that implements the two convolutions $U[n] * I[n]$ and $W[n] * I[n]$ with L multipliers, where: $W(z) = z^{-a}U(z^{-1})$.

4.2. Required multiplications for the LL-LBS method

Concerning the LL-LBS method, for an one-level wavelet decomposition or reconstruction of an N-sample signal, the required number of multiplications is $\Lambda_C(N) = \Upsilon_C(N) = M_{\text{imp}}N/2$, where the symbols $\Lambda_C(N), \Upsilon_C(N)$ refer to multiplications for decomposition and reconstruction, respectively, and M_{imp} is a factor that depends on the implementation technique (convolution or lifting [18]). For example, by using the classical lifting factorization proposed in [13,18] for the 9/7 filter-pair, we have $M_{\text{imp}} = 6$ instead of $M_{\text{imp}} = 9$ that is achieved with convolution. However, the multiplication gain of the lifting implementation comes only when both the low and high-frequency subbands are used for the reconstruction, or conversely when both are produced from an one-level decomposition. As a result, for a decomposition where only the low-frequency (average) subband or the high-frequency (detail) is produced, the required number of multiplications for an N-sample signal is $\Lambda_A(N) = \lfloor (T_H + 1)/2 \rfloor N/2$ or $\Lambda_D(N) = \lfloor (T_G + 1)/2 \rfloor N/2$ respectively, where T_H, T_G denote the number of taps for filters H, G. Notice that this is invariant to which implementation (convolution or lifting) is used [13,18]. Similarly, for a reconstruction from the low-frequency (average) subband only, the multiplications for an N-sample output signal are $\Upsilon_A(N) = \lfloor (T_G + 1)/2 \rfloor N/2$ under both convolution and lifting.

As described in Section 2.3, for a decomposition in k levels, k inverse transforms should be performed to reconstruct the input signal X in the case of the LL-LBS method (see Fig. 2(b) for the example of $k = 2$). Then, by performing a number of forward transforms that retain the even or odd polyphase components of the undecimated transform, all the ODWT subbands of level k are produced. Thus, for the construction in the FO-mode, where both the low and high-frequency ODWT subbands of level k are produced, the necessary number of multiplications is

$$\begin{aligned} M_{\text{LL-LBS,FO}}(k) &= \left[\Upsilon_C\left(\frac{N}{2^{k-1}}\right) + \Upsilon_A\left(\frac{N}{2^{k-2}}\right) + \Upsilon_A\left(\frac{N}{2^{k-3}}\right) + \cdots + \Upsilon_A(N)\right] \\ &\quad + \left[(2-1)\Lambda_A(N) + (4-1)\Lambda_A\left(\frac{N}{2}\right) + \cdots + (2^{k-1}-1)\Lambda_A\left(\frac{N}{2^{k-2}}\right) + (2^k-1)\Lambda_C\left(\frac{N}{2^{k-1}}\right)\right] \\ &= N\left[M_{\text{imp}} + \left\lfloor\frac{T_G+1}{2}\right\rfloor(1-2^{1-k}) + \left\lfloor\frac{T_H+1}{2}\right\rfloor \cdot (k-2+2^{1-k})\right]. \end{aligned} \tag{27}$$

When operating in the HFO-mode, where only the high-frequency subbands of level k are produced, the number of multiplications is

$$\begin{aligned} M_{\text{LL-LBS,HFO}}(k) &= \left[\Upsilon_C\left(\frac{N}{2^{k-1}}\right) + \Upsilon_A\left(\frac{N}{2^{k-2}}\right) + \Upsilon_A\left(\frac{N}{2^{k-3}}\right) + \cdots + \Upsilon_A(N)\right] \\ &\quad + \left[(2-1)\Lambda_A(N) + (4-1)\Lambda_A\left(\frac{N}{2}\right) + \cdots + (2^{k-1}-1)\Lambda_A\left(\frac{N}{2^{k-2}}\right) + (2^k-1)\Lambda_D\left(\frac{N}{2^{k-1}}\right)\right] \\ &= N\left[M_{\text{imp}}2^{-k} + (2-3\cdot 2^{-k})\left\lfloor\frac{T_G+1}{2}\right\rfloor + (k-2+2^{1-k})\left\lfloor\frac{T_H+1}{2}\right\rfloor\right]. \end{aligned} \tag{28}$$

4.3. The multiplication budget of each method under the application framework

For the application of the CODWT in a (de)coding environment, such as the one presented in Section 2.2, a system that supports a total of k decomposition (resolution levels) performs the CODWT in FO mode for level k and in HFO mode for levels $k-1, k-2, \ldots, l$, with l indicating the output resolution of a certain (de)coder [5,8,9,11]. The *total* multiplication budget for a (de)coder that ceases the processing at level l $(1 \leqslant l \leqslant k)$ is denoted as $M^k_{\text{LL-LBS}}(l), M^k_{\text{P-F}}(l)$ for the LL-LBS and the prediction-filters method, respectively. Based on (25)–(28), $M^k_{\text{LL-LBS}}(l)$, $M^k_{\text{P-F}}(l)$ are calculated as

$$M^k_{\text{LL-LBS}}(l) = \begin{cases} M_{\text{LL-LBS,FO}}(k) & \text{if } l = k, \\ M_{\text{LL-LBS,FO}}(k) + \sum_{j=l}^{k-1} M_{\text{LL-LBS,HFO}}(j) & \\ \quad \text{if } 1 \leqslant l < k, & \end{cases} \quad (29)$$

$$M^k_{\text{P-F}}(l) = \begin{cases} M_{\text{P-F,FO}}(k) & \text{if } l = k, \\ M_{\text{P-F,FO}}(k) + \sum_{j=l}^{k-1} M_{\text{P-F,HFO}}(j) & \\ \quad \text{if } 1 \leqslant l < k. & \end{cases} \quad (30)$$

In this paper we also explore the additional approach of approximating the prediction filters by setting to zero all taps that are smaller than a threshold. In this way, the size of the filters of each level is reduced, while a good approximation of the final result is obtained. This technique cannot be applied in the LL-LBS approach since the lifting coefficients and the taps of the biorthogonal filter-pairs do not have magnitudes below the chosen thresholds.

Table 4 presents a typical comparison between the two approaches with respect to the multiplication budget for the case of a system supporting a maximum of $k = 4$ decomposition levels and (de)coding at any level l, $1 \leqslant l \leqslant 4$. The results were obtained from Eqs. (29), (30), and the multiplications for the LL-LBS approach are derived both under lifting and convolution. The thresholds for the prediction-filters method shown in the last column of Table 4 were chosen so that adequate precision for very high-bit-rate coding is obtained in the ODWT domain: a PSNR above 50 dB was imposed in every ODWT subband constructed with the prediction-filters method in comparison to the construction with the LL-LBS. The results of Table 4 demonstrate that the proposed approach achieves comparable or superior results to the multi-rate LL-LBS with respect to the required multiplication budget. In addition, if thresholding is used for the prediction filters, significant gains in the multiplication budget are achieved.

We investigate also the relative multiplication complexity of the proposed CODWT in comparison to the DWT. The number of multiplications for the production of the k-level DWT of N input samples is $M_{\text{DWT}}(k) = M_{\text{imp}} \cdot N \sum_{l=1}^{k} 2^{-l}$. With the settings used in Table 4, the proposed CODWT requires an average of 2.5–5 times more multiplications than the DWT for decoding at resolutions $k = 4$–1, respectively. In fact, we find that the multiplication overhead of the proposed CODWT scales down linearly with the decoded image dimensions, an effect that is expected since the ODWT construction occurs separately for each resolution level. This ensures complexity scalability for the proposed method, a feature that is very important for efficient scalable coding [4,5,7,10].

The results of this subsection are applicable also for the 2-D case of a $R \times C$ input image since both techniques can be applied in a separable manner along the rows and columns of the input.

5. Application results

Recent developments in image and video coding systems have focused on the use of the level-by-level CODWT in order to offer resolution scalability with competitive coding performance to conventional, non-scalable systems. The proposed framework can naturally satisfy the needs of such systems and offer at the same time significant implementation benefits. In the following subsections we highlight these application areas.

Table 4
The reduction in the multiplication budget of the proposed single-rate calculation scheme

5/3 *Filter-pair*, *total decomposition levels* $k = 4$					
(De)coding stops at resolution level	Reduction under convolution ($M_{\text{imp}} = 5$)		Reduction under lifting ($M_{\text{imp}} = 3$)		Thresholds used per level l^*: $[T_l, T_{l+1}, \ldots, T_k]$
	Non-thresholded P-F vs. LL-LBS (%)	Thresholded P-F vs. LL-LBS (%)	Non-thresholded P-F vs. LL-LBS (%)	Thresholded P-F vs. LL-LBS (%)	
$l = 1$	33.26	46.82	23.54	39.08	$[4 \cdot 10^{-2}, 2 \cdot 10^{-2}, 10^{-2}, 5 \cdot 10^{-3}]$
$l = 2$	31.97	47.36	23.92	41.13	$[2 \cdot 10^{-2}, 10^{-2}, 5 \cdot 10^{-3}]$
$l = 3$	30.42	47.29	21.96	40.88	$[10^{-2}, 5 \cdot 10^{-3}]$
$l = 4$	21.43	40.48	7.3	29.78	$[5 \cdot 10^{-3}]$
9/7 *Filter-pair, total decomposition levels* $k = 4$					
(De)coding stops at resolution level	Reduction under convolution ($M_{\text{imp}} = 9$)		Reduction under lifting ($M_{\text{imp}} = 6$)		Thresholds used per level l^* : $[T_l, T_{l+1}, \ldots, T_k]$
	Non-thresholded P-F vs. LL-LBS (%)	Thresholded P–F vs. LL-LBS (%)	Non-thresholded P-F vs. LL-LBS (%)	Thresholded P-F vs. LL-LBS (%)	
$l = 1$	17.02	53.40	7.22	47.90	$[10^{-2}, 10^{-2}, 5 \cdot 10^{-3}, 2.5 \cdot 10^{-3}]$
$l = 2$	14.04	53.34	5.72	48.83	$[10^{-2}, 5 \cdot 10^{-3}, 2.5 \cdot 10^{-3}]$
$l = 3$	10.98	52.53	2.04	47.77	$[5 \cdot 10^{-3}, 2.5 \cdot 10^{-3}]$
$l = 4$	−0.81	46.76	−15.84	38.82	$[2.5 \cdot 10^{-3}]$

Both the original and the thresholded prediction-filters methods are compared with the conventional LL-LBS method (implemented with convolution and lifting) for the case of a (de)coding system that reconstructs any resolution out of the four available decomposition levels ($l = 4$: coarsest resolution, and 1: original resolution).

*The thresholds $T_i, l \leqslant i \leqslant k$ were chosen so that the minimum PSNR between the ODWT subbands produced by the thresholded prediction filters and the LL −LBS was always above 50 dB, for each level i. The JPEG-2000 test-set of images was used for this purpose.

5.1. Resolution-scalable video coding

Research efforts on predictive and open-loop video-coding have recently focused on ODWT representations for the efficient performance of wavelet-domain motion estimation and compensation (ME/MC) in a resolution-scalable framework. In the following subsection we demonstrate the use of the level-by-level CODWT for ME/MC in the predictive coding framework [8,5] while, as a second example, the open-loop coding framework with wavelet-domain motion compensated temporal filtering (MCTF) [9] is presented next.

5.1.1. Predictive video coding with resolution-scalable in-band ME/MC

Motion-compensated wavelet video coders with in-band prediction have been proposed in [1–3,5,6,8]. All these examples utilize the CODWT and hence they can be modified to operate in a resolution-scalable framework by using the resolution-scalable codec structure of Fig. 1. This was already demonstrated in some detail in [5,8]. There, coding schemes that provide resolution, quality and frame-rate scalability with the classical predictive (closed-loop) coding structure have been proposed. A generic architecture of these systems is presented in Fig. 4. As shown in this figure, the encoder performs the ME and MC in the wavelet domain (in-band) following a level-by-level refinement of the compressed information. However, since the critically-sampled wavelet decomposition is only periodically shift-invariant, the ME/MC procedures are performed in the ODWT domain [1–3], which is constructed per resolution level by the CODWT module of Fig. 4. The produced error frames from the motion compensation are coded using an embedded intra-band coding technique (SBC module). The embedded nature of the employed wavelet-based compression algorithm and the level-by-level operation of the ME/MC guarantee that the produced bit-stream can be decoded at a variety of resolutions and quality levels without drifting problems, as long as every decoder receives a certain portion (i.e. a base-quality layer) of the encoded bit-stream [5,8]. Additionally, depending on the type of successive prediction used in the coding scheme, temporal scalability is supported as well, by simply skipping the frames that are not used as references. In this way, fine-grain scalable video decoding in resolution, quality and frame-rate is achieved.

From this description it can be seen that in this case the wavelet-domain operation that uses the ODWT is the in-band ME/MC while the auxiliary data used in the CODWT and in the in-band ME/MC are the decoded motion vectors. This description highlights the correspondence between the components of Fig. 1 and the architecture of Fig. 4.

In Fig. 5 we present typical coding results obtained with the scalable video coder of [8] equipped with half-pixel accurate in-band ME and employing a two-level decomposition ($k = 2$) with the 9/7 filter-pair. Decoding to level $l = 1$ (full-resolution) and $l = 2$ (half-resolution) is performed. For comparison purposes, Fig. 5 presents the results obtained with two *non-scalable*

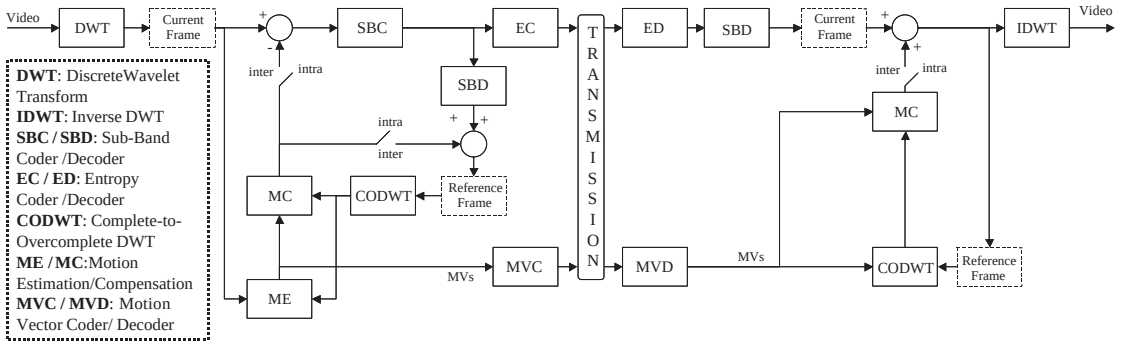

Fig. 4. The architecture of a predictive coder with wavelet-domain (in-band) ME/MC that uses the resolution-scalable CODWT.

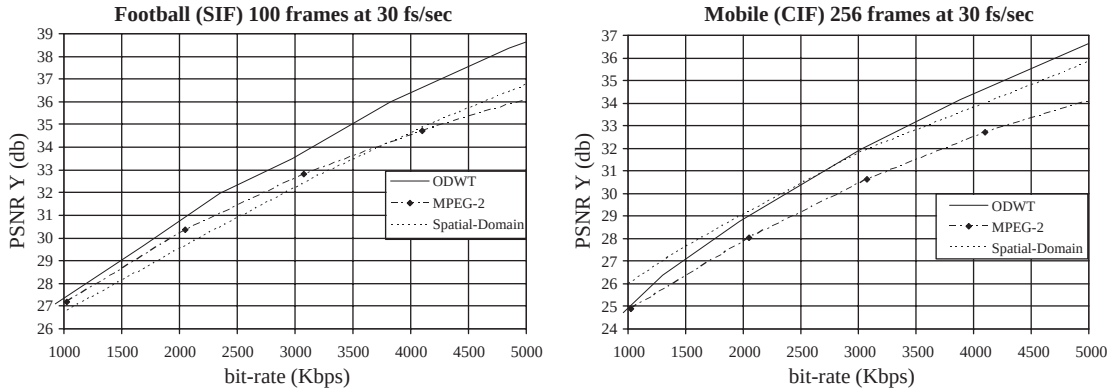

Fig. 5. Comparison of different techniques for predictive coding of "Football" and "Mobile" sequences.

solutions: the equivalent coder that performs conventional spatial-domain ME/MC [1,8] followed by DWT and error-frame coding, and the fixed-rate, DCT-based MPEG-2 coder using half-pixel ME/MC. It is demonstrated that the presented system typically outperforms the equivalent system that uses spatial-domain ME/MC, as well as the standardized MPEG-2 coder. At the same time, scalable decoding to two resolution levels is provided from the same compressed bit-stream.

In this application, the proposed single-rate CODWT offers implementation benefits in comparison to the LL-LBS approach. With the settings used for the results of Fig. 5, by using Eqs. (29), (30) with $k = 2$, $1 \leqslant l \leqslant 2$ and for the 9/7 filter-pair, it follows that the proposed prediction-filters scheme implemented with thresholding offers 24.19% and 16.67% reduction in the multiplication budget in comparison to the multi-rate lifting-based LL-LBS alternative for full-resolution and half-resolution (de)coding respectively. In addition, we found that the thresholding did not have any effect in the coding results of Fig. 5, even at high bit-rates.

5.1.2. *Open-loop, in-band motion-compensated temporal filtering*

Although the predictive schemes with in-band ME/MC can achieve efficient spatial-scalability, there are certain disadvantages concerning quality scalability [8,9] due to the fact that the reference frame must be decoded at a certain accuracy (base-layer) in the prediction loop of the codec of Fig. 4. To alleviate this inefficiency, the system proposed in [10,20] utilizes wavelet-domain motion estimation within open-loop video coding systems performing motion compensated temporal filtering (MCTF) [19,20]. In contrast to the conventional approaches for open-loop video coding in which the MCTF is applied on the input-frames and the DWT is applied on the resulting motion-compensated residual frames [19], in [20] the video frames are first spatially decomposed into multiple subbands using the DWT, and then the temporal correlations for each subband are removed using MCTF (see Fig. 6). The residual signal after the MCTF can be coded subband-by-subband using any desired embedded texture coding technique (DCT-based, wavelet-based, matching pursuit, etc.). In this way decoding to arbitrary resolution, quality or frame-rate is achieved from one compressed bit-stream.

For efficient wavelet-domain ME, the ODWT is constructed from the critically sampled decomposition of the reference frame(s) assuming resolution scalability, i.e. the codec of Fig. 6 that performs k spatial decomposition levels can decode up to k dyadically-reduced resolution levels from the same compressed bit-stream. As seen in Fig. 6, the level-by-level construction of the ODWT from the DWT (a procedure that can use the proposed calculation framework of Section 3) occurs at both the encoder and decoder side to reconstruct the reference wavelet subbands. In fact, Fig. 1 can be directly matched to the system

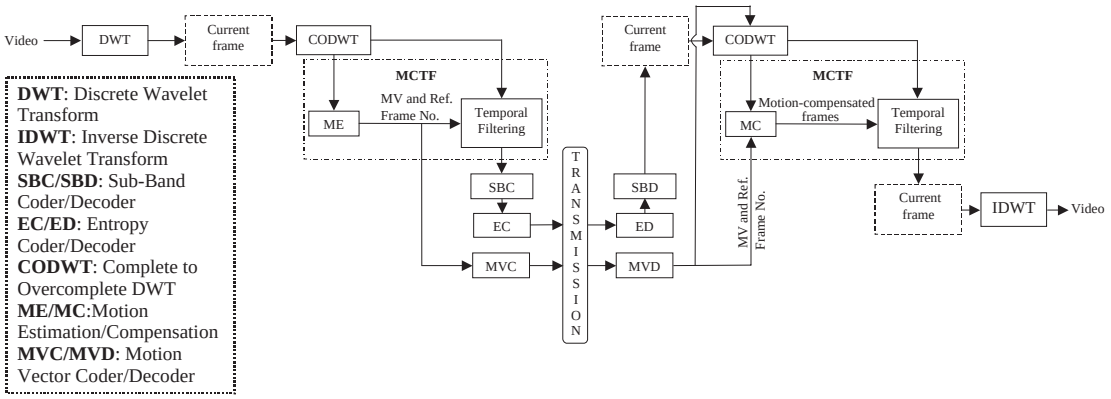

Fig. 6. The architecture of an open-loop coder with MCTF.

Table 5
Experimental comparison between in-band MCTF, spatial-domain MCTF and the MPEG-4 AVC coder for 576 frames of sequences "Harbour" and "City"

Harbour	3000 Kbps, 704 × 576, 60 Hz	1500 Kbps, 704 × 576, 30 Hz	750 Kbps, 352 × 288, 30 Hz
SDMCTF	37.16	35.33	33.71
IBMCTF	37.35	35.38	34.89
MPEG-4 AVC	36.46	34.64	35.17
City	2000 Kbps, 704 × 576, 60 Hz	1024 Kbps, 704 × 576, 30 Hz	512 Kbps, 352 × 288, 30 Hz
SDMCTF	38.10	36.41	35.80
IBMCTF	37.53	35.85	37.40
MPEG-4 AVC	38.98	36.84	39.14

The mean PSNR values (averages over all frames) are presented, with mean_PSNR = (4·PSNR_Y + PSNR_U + PSNR_V)/6. The two MCTF produce all resolutions/frame-rates/bit-rates from a single compressed bit-stream.

of Fig. 6 by identifying the MCTF operation in Fig. 6 as the wavelet-domain operation performed in the ODWT of the reference frames.

The experimental performance of the system that performs in-band motion compensated temporal filtering (IBMCTF) is compared against: (a) the conventional solution that performs spatial-domain MCTF (SDMCTF) [20] and, (b) the current state-of-the-art in video coding (MPEG-4 AVC coder, with the settings defined in [22]). The results are shown in Table 5. Two typical test sequences are compressed with the experimental conditions defined in the recent MPEG core experiment for scalable video coding [22]. For a fair comparison of the two different architectures—SDMCTF and IBMCTF—the multihypothesis variable block-size ME/MC scheme of [20] was chosen for both schemes. The block size and search range for decomposition level one in IBMCTF are the in-band equivalent of the SDMCTF selections, as explained in detail in [9]. In addition, both schemes use the same embedded intra-band wavelet coder [21]. To ensure resolution scalability, the coder is used separately per resolution level. For both systems, the same rate control was used during the bitstream extraction and both used four temporal decomposition levels and four levels for the spatial transform (9/7 filter-pair). In fact, apart from the different processing order (spatial transform followed by temporal filtering), the CODWT of the IBMCTF and the different representation in the

ME/MC approach (wavelet coefficients instead of input spatial-domain samples), both systems were implemented with the same software.

For the PSNR results of the SDMCTF and IBMCTF-based schemes, the input sequence was compressed once with a very high-quality setting. The reference for the lower-resolution is the reconstructed low-frequency subband of the input frames at a very high bitrate. For low-resolution, this can be said to represent the decoder-side "original" sequence. Similarly, for the lower frame rate, the decoder-side "original" sequence for SDMCTF and IBMCTF is extracted by decoding the low frame-rate result to a very high bitrate [9]. Concerning the non-scalable MPEG-4 AVC, separate encoding and decoding was performed for *each* experimental point and the original sequences for low-resolution/low frame-rate were generated using conventional MPEG-4 spatial downsampling filters [22] and frame-skipping.

From the results of Table 5, one notices that for full-resolution decoding, the IBMCTF scheme achieves comparable performance to the conventional SDMCTF-based system and the highly-optimized MPEG-4 AVC. Concerning low-resolution decoding, a significant performance-difference exists between the IBMCTF and SDMCTF [9]. Moreover, the AVC coder appears to outperform both MCTF codecs in terms of PSNR. However, we remark that significant PSNR differences exist between the AVC decoder-side generated "original" and the MCTF-generated "original" sequences for lower-resolution/lower frame-rate decoding. Furthermore, visual evaluation indicates that, for the sequences and experimental points of Table 5, the IBMCTF-based coder appears to produce at least comparable if not superior visual quality to MPEG-4 AVC.

5.2. Resolution-scalable image coding

In [11], the author proposes a system that uses the level-by-level CODWT for the realization of an efficient system that performs phase-based prediction of wavelet coefficients for still-image coding. A potential architecture for such a system is given in Fig. 7. There, the encoding occurs with a level-by-level phase-based prediction of the subbands of each resolution level by using the ODWT representation of the previous resolution level [11]. This representation is created independently at both the encoder and the decoder side with the CODWT module. Hence, instead of simply coding with an intra-band coder the original subband content, a prediction of the subband coefficients of each resolution level is performed using the interpolated ODWT of the previous-level subbands, which, in a resolution-scalable framework, are available at both the encoder and decoder sides. The prediction error is coded with the SBC module of Fig. 7. The decoder receives the compressed information and performs the decoding with the equivalent set of operations. Hence, in this system, the level-by-level

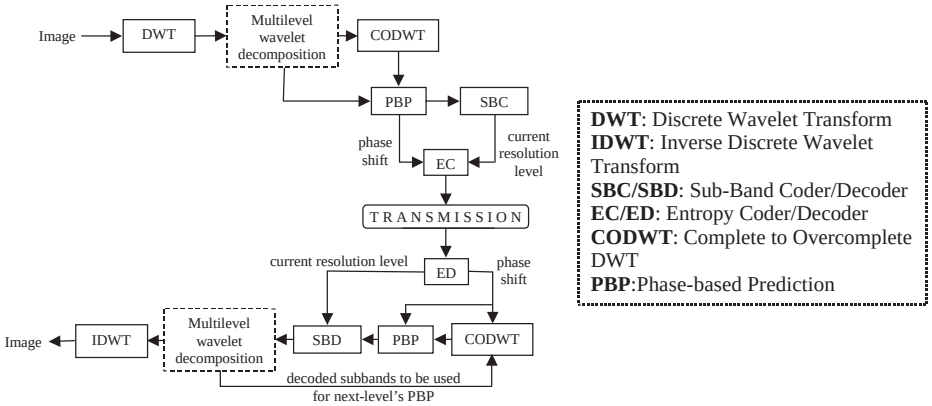

Fig. 7. Phase-based prediction for resolution-scalable still-image coding [11].

CODWT is employed at both the encoder and decoder sides [11], as seen in Fig. 7.

In [11], significant coding gains are reported in comparison to the coding of the original wavelet coefficients with the same subband and entropy coding techniques. As an example, these gains range in the "Lena" image from 0.54 dB for the 9/7 filter-pair to 2.82 dB for the Haar filter [11]. If four decomposition levels are used in this image codec, the multiplication-gains obtained by the proposed prediction-filters method are shown in Table 4 for the 9/7 and 5/3 filter-pairs.

6. Conclusions

A new theoretical framework is presented for the level-by-level construction of the overcomplete DWT starting from the subbands of the critically sampled decomposition of a certain resolution level. The proposed prediction-filters scheme is more efficient than the conventional method when applied in a resolution-scalable coding system, as it enables the single-rate calculation of the ODWT with trade-offs for computational savings. This feature leads to inherent computational scalability for the proposed scheme in comparison to the conventional approach, since the hardware requirements can be reduced in the low-resolution codecs without surpassing the delay-constraint of the full-resolution codec. Several scalable coding examples with very promising prospects for compression efficiency have been identified where the proposed approach can be incorporated in order to achieve complexity scalability.

Acknowledgements

This work was supported in part by the Federal Office for Scientific, Technical and Cultural Affairs (IAP Phase V - Mobile Multimedia), the European Community under the IST Programme (Mascot, IST-2000-26467) and the Fund for Scientific Research - Flanders (FWO) (G.0053.03 & G.0021.03). P. Schelkens and A. Munteanu have post-doctoral fellowships with the Fund for Scientific Research - Flanders (FWO), Egmontstraat 5, B-1000 Brussels, Belgium.

Appendix A

For the proposition $\mathcal{Q}(1)$ (Eq. (2)), the calculated subbands of the first decomposition level, namely subbands A_0^1, D_0^1 and A_1^1, D_1^1, are shown pictorially in Fig. 2(a). These subbands can be written using X_0, X_1 (the polyphase components of X) and the polyphase components of H, G [4]:

$$A_0^1(z) = H_0(z)X_0(z) + H_1(z)X_1(z), \tag{31}$$

$$D_0^1(z) = G_0(z)X_0(z) + G_1(z)X_1(z), \tag{32}$$

$$A_1^1(z) = H_1(z)X_0(z) + zH_0(z)X_1(z), \tag{33}$$

$$D_1^1(z) = G_1(z)X_0(z) + zG_0(z)X_1(z). \tag{34}$$

By solving the system of (31), (32) for the polyphase components X_0, X_1 and replacing the outcome in Eqs. (33) and (34), we derive the proposition $\mathcal{Q}(1)$ of (2) with the filters defined in (3), (4).

For proposition $\mathcal{Q}(2)$ (Eq. (5)), by starting from subbands A_1^2, D_1^2, and assuming that subband A_0^1 is the input signal, the subbands A_0^2, D_0^2 and A_1^2, D_1^2 can be considered as the one-level overcomplete DWT of A_0^1 (see Fig. 2(a)). For this case, proposition $\mathcal{Q}(1)$ is applicable:

$$A_1^2(z) = F_0^1(z)A_0^2(z) + F_1^1 D_0^2(z), \tag{35}$$

$$D_1^2(z) = F_2^1(z)A_0^2(z) + F_3^1(z)D_0^2(z). \tag{36}$$

Eqs. (29), (36) express the first part of proposition $\mathcal{Q}(2)$. For the remaining parts of the proposition, we perform an inverse wavelet transform in order to calculate A_0^1 from subbands A_0^2, D_0^2:

$$A_0^1(z) = z[\tilde{H}(z)A_0^2(z^2) + \tilde{G}(z)D_0^2(z^2)]. \tag{37}$$

In (37) the factor z ensures that the delay of the system of forward and inverse transform is unitary. Now that the subband A_0^1 is known, we can apply proposition $\mathcal{Q}(1)$ to calculate A_1^1:

$$A_1^1(z) = zF_0^1(z)\tilde{H}(z)A_0^2(z^2) + zF_0^1(z)\tilde{G}(z)D_0^2(z^2). \tag{38}$$

Note that Eq. (38) is obtained with $D_0^1 = 0$ because in a resolution-scalable coding system this subband is not available at both the encoder and decoder of level 2. To calculate the second part of (5), i.e. subbands A_2^2, D_2^2, we perform a forward wavelet transform retaining the even samples (''classic'' transform):

$$\begin{bmatrix} A_2^2(z) \\ D_2^2(z) \end{bmatrix} = \frac{1}{2} \begin{bmatrix} H(z^{1/2}) & H(-z^{1/2}) \\ G(z^{1/2}) & G(-z^{1/2}) \end{bmatrix} \times \begin{bmatrix} A_1^1(z^{1/2}) \\ A_1^1(-z^{1/2}) \end{bmatrix}. \tag{39}$$

The proof of the second part follows by replacing $A_1^1(z^{1/2})$ and $A_1^1(-z^{1/2})$ from (38), with the use of the type-I polyphase components of filters F_0^1, F_1^1 and the type-II polyphase components of $\widetilde{H}, \widetilde{G}$. Similarly, to calculate the third part of (5), i.e. subbands A_3^2, D_3^2, one needs to perform a forward wavelet transform retaining the odd samples (''complementary'' transform):

$$\begin{bmatrix} A_3^2(z) \\ D_3^2(z) \end{bmatrix} = \frac{1}{2} z^{1/2} \begin{bmatrix} H(z^{1/2}) & -H(-z^{1/2}) \\ G(z^{1/2}) & -G(-z^{1/2}) \end{bmatrix} \times \begin{bmatrix} A_1^1(z^{1/2}) \\ A_1^1(-z^{1/2}) \end{bmatrix}. \tag{40}$$

The proof of the third part follows as for the second part by replacing $A_1^1(z^{1/2})$ and $A_1^1(-z^{1/2})$ from (38).

Appendix B

The properties shown in (24) will be demonstrated by mathematical induction. Starting from the first decomposition level, the following relationships are used for the prediction filters:

$$F_0^1(z^{-1}) = z^{-1} F_0^1(z), \tag{41}$$

$$F_1^1(z^{-1}) = F_1^1(z), \tag{42}$$

$$F_2^1(z^{-1}) = z^{-2} F_2^1(z), \tag{43}$$

$$F_3^1(z^{-1}) = z^{-1} F_3^1(z), \tag{44}$$

$$F_0^1(z) = -F_3^1(z), \tag{45}$$

$$F_3^1(z) = -z F_0^1(z^{-1}). \tag{46}$$

The proof of Eqs. (41)–(46) follows immediately by deriving the relations between the polyphase components of the filter-bank based on (1) and using them in conjunction with (3) and (4).

The symmetry propositions $\mathscr{P}_{\mathscr{S}}(2)$ that correspond to the case $k = 2$ are:

$$\mathscr{P}_{\mathscr{S}}(2): \begin{cases} F_0^2(z) = zF_4^2(z^{-1}), \\ F_1^2(z) = F_5^2(z^{-1}), \\ F_2^2(z) = z^2 F_6^2(z^{-1}), \\ F_3^2(z) = zF_7^2(z^{-1}). \end{cases} \tag{47}$$

To prove this proposition, we use Eq. (41) with the type-I polyphase components of filter $F_0^1(z)$, yielding $F_{0,0}^1(z^{-1}) = z^{-1}F_{0,1}^1(z)$ and $F_{0,1}^1(z^{-1}) = z^{-1}F_{0,0}^1(z)$. The proof of (47) is concluded by replacing filters $F_{\{4,5,6,7\}}^2(z^{-1})$ from Eqs. (11)–(14), respectively (with $l = 1$), and by replacing $F_{0,0}^1(z^{-1})$, $F_{0,1}^1(z^{-1})$ and $F_{\{0,1,2,3\}}^1(z^{-1})$ from the last two equations and (41)–(44).

We assume that the symmetry propositions $\mathscr{P}_{\mathscr{S}}(k)$ are true and based on them we derive the symmetry relationships for level $k+1$. In the type-I polyphase components of F_{4m}^k, with $0 \leqslant m < 2^{k-2}$ we can replace the terms $F_{4m}^k(z^{1/2}), F_{4m}^k(-z^{1/2})$ by using the first equation of (24), yielding:

$$\begin{aligned} F_{4m,0}^k(z) &= zF_{4(2^{k-1}-m-1),1}^k(z^{-1}), \\ F_{4m,1}^k(z) &= zF_{4(2^{k-1}-m-1),0}^k(z^{-1}) \end{aligned} \tag{48}$$

with $m = 0, 1, \ldots, 2^{k-2} - 1$ and $k > 1$.

B.1. Filters F_{4m}^{k+1}

Let us start by proving the symmetry properties for the filters of the form F_{4m}^{k+1}, with $0 \leqslant m < 2^{k-1}$; these properties are separately derived for m even and m odd. For the case of m even, we define $m = 2j$, with $0 \leqslant j < 2^{k-2}$. Hence, from (7) with $l = k$ and from Eq. (46) we obtain: $F_{4m}^{k+1}(z) = F_{4j,0}^k(z) + F_0^1(z^{-1})F_{4j,1}^k(z)$. Since $0 \leqslant j < 2^{k-2}$, we can substitute (48) in the last equation, obtaining $F_{4m}^{k+1}(z) = z(F_{4(2^{k-1}-j-1),1}^k(z^{-1}) + F_0^1(z^{-1}) \cdot F_{4(2^{k-1}-j-1),0}^k(z^{-1}))$.

From (11) with $l=k$ and from the definition of m, the last equation becomes:

$$F_{4m}^{k+1}(z) = zF_{8(2^{k-1}-j-1)+4}^{k+1}(z^{-1}) \Leftrightarrow F_{4m}^{k+1}(z) = zF_{4(2^k-m-1)}^{k+1}(z^{-1}), \quad \text{for } m=0,2,4,\ldots,2^{k-1}-2. \tag{49}$$

For the case of m odd, we define $m=2j+1$ with $0\leqslant j<2^{k-2}$. Thus, from (11) with $l=k$ and from Eq. (46) we obtain

$$F_{4m}^{k+1}(z) = F_{8j+4}^{k+1}(z) = F_{4j,1}^{k}(z) + F_0^1(z)F_{4j,0}^{k}(z) \Leftrightarrow F_{4m}^{k+1}(z) = F_{4j,1}^{k}(z) - zF_3^1(z^{-1})F_{4j,0}^{k}(z). \tag{50}$$

Since $0\leqslant j<2^{k-2}$, we can substitute (48) in Eq. (50) and we get

$$F_{4m}^{k+1}(z) = z(F_{4(2^{k-1}-j-1),0}^{k}(z^{-1}) - zF_3^1(z^{-1})F_{4(2^{k-1}-j-1),1}^{k}(z^{-1})). \tag{51}$$

From (7) with $l=k$ and from the definition of m, Eq. (51) becomes

$$F_{4m}^{k+1}(z) = zF_{8(2^{k-1}-j-1)}^{k+1}(z^{-1}) \Leftrightarrow F_{4m}^{k+1}(z) = zF_{4(2^k-m-1)}^{k+1}(z^{-1}), \quad \text{for } m=1,3,5,\ldots,2^{k-1}-1. \tag{52}$$

Thus, the combination of (49) and (52) yields the first equation of (24) with the k replaced by $k+1$.

B.2. Filters F_{4m+1}^{k+1}

To prove the symmetry properties for the prediction filters of the form F_{4m+1}^{k+1}, with $0\leqslant m<2^{k-1}$, we follow the same rationale as before. Hence, these properties are separately derived for m even and m odd. For the case of m even, we denote $m=2j$ with $0\leqslant j<2^{k-2}$. By using (8) with $l=k$ and from (42) we obtain $F_{4m+1}^{k+1}(z) = z^{-1}F_1^1(z^{-1})F_{4j,1}^{k}(z)$. Since $0\leqslant j<2^{k-2}$, we can substitute (48) in the last equation, obtaining: $F_{4m+1}^{k+1}(z) = F_1^1(z^{-1})F_{4(2^{k-1}-j-1),0}^{k}(z^{-1})$.

From Eq. (12) with $l=k$ and from the definition of m we obtain

$$F_{4m+1}^{k+1}(z) = F_{8(2^{k-1}-j-1)+5}^{k+1}(z^{-1}) \Leftrightarrow F_{4m+1}^{k+1}(z) = F_{4(2^k-m-1)+1}^{k+1}(z^{-1}), \quad \text{for } m=0,2,4,\ldots,2^{k-1}-2. \tag{53}$$

For the case of m odd, we denote $m=2j+1$ with $0\leqslant j<2^{k-2}$.

Thus, from (13) with $l=k$ and from (42) we obtain

$$F_{4m+1}^{k+1}(z) = F_{8j+5}^{k+1}(z) = F_1^1(z)F_{4j,0}^{k}(z) \Leftrightarrow F_{4m+1}^{k+1}(z) = F_1^1(z^{-1})F_{4j,0}^{k}(z). \tag{54}$$

Since $0\leqslant j<2^{k-2}$, we can substitute (48) in (54), obtaining: $F_{4m+1}^{k+1}(z) = zF_1^1(z^{-1})F_{4(2^{k-1}-j-1),1}^{k}(z)$.

From Eq. (8) with $l=k$ and from the definition of m we obtain

$$F_{4m+1}^{k+1}(z) = F_{8(2^{k-1}-j-1)+1}^{k+1}(z^{-1}) \Leftrightarrow F_{4m+1}^{k+1}(z) = F_{4(2^k-m-1)+1}^{k+1}(z^{-1}), \quad \text{for } m=1,3,5,\ldots,2^{k-1}-1. \tag{55}$$

Thus, the combination of (53) and (55) yields the second equation of (24) with k replaced by $k+1$.

B.3. Filters F_{4m+2}^{k+1}, F_{4m+3}^{k+1}

For the prediction filters of the form F_{4m+2}^{k+1}, with $0\leqslant m<2^{k-1}$, we follow exactly the same reasoning as for filter F_{4m+1}^{k+1}, but now Eqs. (9), (13) with $l=k$ are used in combination with (43) for the even and odd values of m. Equivalently, for the filters F_{4m+3}^{k+1}, $0\leqslant m<2^{k-1}$, we follow similar steps as for the proof given for the filters 1 F_{4m}^{k+1}; in this case Eqs. (10), (14) with $l=k$ are used in combination with (44) for the even and odd values of m. As a result we reach the third and fourth equation of (24) with k replaced by $k+1$.

References

[1] H.W. Park, H.S. Kim, Motion estimation using low-band-shift method for wavelet-based moving-picture coding, IEEE Trans. Image Process. 9 (4) (April 2000) 577–587.

[2] J. Skowronski, Pel recursive motion estimation and comprensation in subbands, Signal Processing: Image Communication 14 (5) (April 1999) 389–396.
[3] R. Zaciu, C. Lamba, C. Burlacu, G. Nicula, Image compression using an overcomplete discrete wavelet transform, IEEE Trans. Consumer Electron. 42 (3) (August 1996) 800–807.
[4] G. Van der Auwera, A. Munteanu, P. Schelkens, J. Cornelis, Bottom-up motion compensated prediction in the wavelet domain for spatially-scalable video, IEE Electron. Lett. 38 (21) (October 2002) 1251–1253.
[5] X. Li, L. Kerofski, S. Lei, All-phase motion compensated prediction in the wavelet domain for high performance video coding, Proceedings of the IEEE International Conference on Image Processing, vol. 3, Thessaloniki, GR, October 2001, pp. 538–541.
[6] S. Cui, Y. Wang, J.E. Fowler, Mesh-based motion estimation and compensation in the wavelet domain using a redundant transform, Proceeding of the IEEE International Conference on Image Processing, vol. 1, Rochester, USA, September 2002, pp. 693–696.
[7] Y. Andreopoulos, A. Munteanu, G. Van der Auwera, P. Schelkens, J. Cornelis, A new method for complete-to-overcomplete discrete wavelet transforms, in: Proceedings of the International Conference on Digital Signal Processing, Santorini, GR, July 2002, pp. 501–504.
[8] Y. Andreopoulos, A. Munteanu, G. Van der Auwera, P. Schelkens, J. Cornelis, Scalable wavelet video-coding with in-band prediction—implementation and experimental results, Proceedings of the IEEE International Conference on Image Processing, vol. 3, Rochester, USA, September 2002, pp. 729-732.
[9] Y. Andreopoulos, A. Munteanu, M. van der Schaar, J. Cornelis, P. Schelkens, Comparison between "t + 2D" and "2D + t" architectures with advanced motion compensated temporal filtering, ISO/IEC JTC1/SC29/WG11, m11045, MPEG 69th Meeting, Redmond, US, July 2004.
[10] Y. Andreopoulos, M. van der Schaar, A. Munteanu, J. Barbarien, P. Schelkens, J. Cornelis, Open-loop, in-band, motion-compensated temporal filtering for objective full-scalability in wavelet video coding, ISO/IEC JTC1/SC29/WG11, m9026, MPEG 62nd Meeting, Shanghai, China, October 2002.
[11] X. Li, On exploiting the phase constraint with image wavelet coefficients, Proceedings of the IEEE International Conference on Image Processing, vol. 3, Rochester, USA, September 2002, pp. 221–224.
[12] S. Mallat, A Wavelet Tour of Signal Processing, Academic Press, San Diego CA, 1998.
[13] M. Boliek, C. Christopoulos, E. Majani (Eds.), JPEG 2000 Part I Final Draft International Standard, ISO/IEC JTC1/SC29/WG1 (JPEG) FCD 15444-1, December 2000.
[14] H. Sari-Sarraf, D. Brzakovic, A shift-invariant discrete wavelet transform, IEEE Trans. Signal Process. 45 (10) (October 1997) 2621–2626.
[15] J.D. Villasenor, B. Belzer, J. Liao, Wavelet filter evaluation for image compression, IEEE Trans. Image Process. 4 (8) (August 1995) 1053–1060.
[16] C. Chakrabarti, C. Mumford, Efficient realizations of encoders and decoders based on the 2-D discrete wavelet transforms, IEEE Trans. VLSI Systems 7 (3) (September 1999) 289–298.
[17] M. Vishwanath, The recursive pyramid algorithm for the discrete wavelet transform, IEEE Trans. Signal Process. 42 (3) (March 1994) 673–676.
[18] I. Daubechies, W. Sweldens, Factoring wavelet transforms into lifting steps, J. Fourier Anal. Appl. 4 (3) (March 1998) 247–269.
[19] J.R. Ohm, Three-dimensional subband coding with motion compensation, IEEE Trans. Image Process. 3 (5) (September 1994) 559–571.
[20] Y. Andreopoulos, A. Munteanu, J. Barbarien, M. van der Schaar, J. Cornelis, P. Schelkens, In-band motion compensated temporal filtering, Signal Processing: Image Communication 19 (7) (August 2004) 653–673 (special issue on "Subband/Wavelet Interframe Video Coding").
[21] A. Munteanu, J. Cornelis, G. Van der Auwera, P. Cristea, Wavelet-based lossless compression scheme with progressive transmission capability, Int. J. Imaging System Technol. 10 (1) (January 1999) 76–85.
[22] M. van der Schaar, M. Domanski (Eds.), Description of core experiments in SVC, ISO/IEC JTC1/SC29/WG11, n6373, MPEG, Munich, De, March 2004.