Vrije Universiteit Brussel

Wavelet based volumetric medical image compression

Bruylants, Tim; Munteanu, Adrian; Schelkens, Peter

Published in:
Signal Processing: Image Communication

DOI:
10.1016/j.image.2014.12.007

Publication date:
2015

License:
CC BY

Document Version:
Final published version

Link to publication

Citation for published version (APA):
Bruylants, T., Munteanu, A., & Schelkens, P. (2015). Wavelet based volumetric medical image compression. *Signal Processing: Image Communication*, *31*, 112-133. https://doi.org/10.1016/j.image.2014.12.007

Copyright
No part of this publication may be reproduced or transmitted in any form, without the prior written permission of the author(s) or other rights holders to whom publication rights have been transferred, unless permitted by a license attached to the publication (a Creative Commons license or other), or unless exceptions to copyright law apply.

Take down policy
If you believe that this document infringes your copyright or other rights, please contact openaccess@vub.be, with details of the nature of the infringement. We will investigate the claim and if justified, we will take the appropriate steps.

Download date: 25. Aug. 2022

Signal Processing: *Image Communication* 31 (2015) 112–133

Contents lists available at ScienceDirect

Signal Processing: *Image Communication*

journal homepage: www.elsevier.com/locate/image

Wavelet based volumetric medical image compression

Tim Bruylants [a,b,*], Adrian Munteanu [a,b], Peter Schelkens [a,b]

[a] *Vrije Universiteit Brussel, Department of Electronics and Informatics (ETRO), Pleinlaan 2, B-1050 Brussels, Belgium*
[b] *iMinds, Department of Multimedia Technologies (MT), Gaston Crommenlaan 8 (Box 102), B-9050 Ghent, Belgium*

ARTICLE INFO

Article history:
Received 3 June 2014
Received in revised form
15 December 2014
Accepted 17 December 2014
Available online 25 December 2014

Keywords:
JPEG 2000
Directional wavelets
JP3D
H.265/MPEG-H HEVC
JPEG-LS
Subjective evaluation

ABSTRACT

The amount of image data generated each day in health care is ever increasing, especially in combination with the improved scanning resolutions and the importance of volumetric image data sets. Handling these images raises the requirement for efficient compression, archival and transmission techniques. Currently, JPEG 2000's core coding system, defined in Part 1, is the default choice for medical images as it is the DICOM-supported compression technique offering the best available performance for this type of data. Yet, JPEG 2000 provides many options that allow for further improving compression performance for which DICOM offers no guidelines. Moreover, over the last years, various studies seem to indicate that performance improvements in wavelet-based image coding are possible when employing directional transforms. In this paper, we thoroughly investigate techniques allowing for improving the performance of JPEG 2000 for volumetric medical image compression. For this purpose, we make use of a newly developed generic codec framework that supports JPEG 2000 with its volumetric extension (JP3D), various directional wavelet transforms as well as a generic intra-band prediction mode. A thorough objective investigation of the performance-complexity trade-offs offered by these techniques on medical data is carried out. Moreover, we provide a comparison of the presented techniques to H.265/MPEG-H HEVC, which is currently the most state-of-the-art video codec available. Additionally, we present results of a first time study on the subjective visual performance when using the aforementioned techniques. This enables us to provide a set of guidelines and settings on how to optimally compress medical volumetric images at an acceptable complexity level.

© 2014 The Authors. Published by Elsevier B.V. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

1. Introduction

Today, all modern hospitals heavily rely on digital medical imaging as an important and well-established part of the full chain of patient care handling. In fact, the completely digitized medical workflow, the improved imaging scanner technologies and the importance of volumetric image data sets have all raised the requirements for more efficient compression techniques. Currently image slice resolutions of 512 × 512 are considered to be the minimum standard. However, more recent scanning systems are able to output image slices with spatial resolutions of 1024 × 1024 or more at increasing pixel bit-depths [1]. Additionally, volumetric and time-lapse capable scanning technologies are both increasing the amount of output data even more. With thin-slice Computed Tomography (CT) scanning, the number of slices in volumetric datasets exploded as the inter-slice distance decreased from typically 5 mm to 0.6 mm over the years [2,3]. As such, efficient compression and improved transmission techniques for handling medical images are of utmost importance. Moreover, the ubiquity of Internet

* Corresponding author. Tel.: +32 2 629 1672; fax: +32 2 629 2883.
E-mail address: tbruylan@etro.vub.ac.be (T. Bruylants).

http://dx.doi.org/10.1016/j.image.2014.12.007
0923-5965/© 2014 The Authors. Published by Elsevier B.V. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

usage for e-Health platforms also mandates adequate support for features such as region-of-interest (ROI) coding and progressive quality and resolution scalability more than ever before.

Yet, a large variety of image coding techniques exist, ranging from transform based techniques utilizing Discrete Cosine Transform (DCT) [4,5], Discrete Wavelet Transforms (DWT) [6] or Karhunen-Loève Transforms (KLT) [7–9] to prediction-based techniques, such as CALIC [10] or LOCO-I [11] (as used in JPEG-LS [12]). Some proposals investigated using video coding methodologies such as H.264/MPEG-4 AVC [13] or the very recent H.265/MPEG-H HEVC [14,15] to compress 3D and 4D medical image datasets [16–20]. All of the aforementioned techniques have their strengths and weaknesses. For the compression of volumetric medical datasets it is shown that 3D wavelet based codecs outperform the DCT-based solutions while providing required functionalities such as quality and resolution scalability, random access and ROI coding [21]. In contrast, prediction-based techniques (like CALIC [10] and LOCO-I [11]), typically deliver competitive (near-)lossless compression performance, at the cost of not supporting these functionality constraints.

Most, if not all, medical informatics systems rely on the Digital Imaging and Communications in Medicine (DICOM) standard [22], which in turn relies on major ISO/IEC and ITU-T standards such as JPEG [5], JPEG-LS [12] and JPEG 2000 [6] for encoding of medical image datasets. From this set of adopted coding standards, the wavelet based JPEG 2000 is still the best-suited coding technique. It provides excellent rate-distortion performance for volumetric medical datasets [21,23–25], it supports lossy-to-lossless coding, resolution scalability, region of interest access at varying degrees of granularity, flexible file formats and resilience against transmission errors [6].

The JPEG 2000 [26,27] standard is subdivided in multiple parts of which Part 1 [6] defines the core image coding technology. The other parts specify optional extensions that add extra functionality to JPEG 2000. As such, Part 2 [28] – providing the Multi-Component Transform (MCT) extension – and Part 10 [29] – known as JP3D, providing volumetric extensions – are also relevant for this work.

To compress an image, the JPEG 2000 encoder decomposes the input into wavelet sub-bands (see Section 2) and consecutively quantizes and encodes the resulting wavelet coefficients by use of the Embedded Block Coding by Optimized Truncation (EBCOT) [30] paradigm. EBCOT is a two-tiered entropy coder. Tier-1 represents a context-based adaptive binary arithmetic coding unit that processes the various sub-bands as smaller independent units or so-called code-blocks. Tier-2 then packetizes the resulting code-block bit-streams to generate the final output JPEG 2000 code-stream that optionally meets given rate and/or distortion requirements. The decoding process, on the other side, can be coarsely considered as the inverse of the encoding process without an EBCOT Tier-2 module.

Currently, two methodologies exist for the compression of volumetric medical image data using JPEG 2000. The first is used by DICOM and employs the Multi Component Transform (MCT) feature, as defined in JPEG 2000 Part 2, to perform an axial DWT step with the image slices interpreted by the codec as separate image components. However, this has the minor drawback of causing an inherent ambiguity between actual component information and slice information. The second methodology involves JP3D to handle compression of volumetric image data by properly extending the required portions of JPEG 2000 Part 1. JP3D redefines the DWT to enable support for three dimensions (3D-DWT) [27]. Additionally, JP3D allows the number of decomposition steps and the applied wavelet kernels along the X, Y and Z dimensions to mutually differ. This offers extra flexibility to easily adapt the wavelet transform to specific characteristics of the data at hand. For the purpose of this work, our presented framework will support both of these methodologies. We would like to point out that both the MCT and JP3D extensions allow fall-back to a 2D-DWT setting when opportune, guaranteeing that the worst-case performance of these extensions is equal to the best-case performance of JPEG 2000 Part 1.

Finally, JP3D also introduces volumetric ROI and volumetric code-block support, which are both unavailable in Part 1 or Part 2. The impact of volumetric ROI coding on the lossless compression performance is known and largely depends on the size of the ROI [31,27]. The max-shift ROI method is further enhanced by JPEG 2000 Part 2 to support variable shifts, allowing to balance the ROI overhead and the coding performance more precise. It was observed that with the max-shift method the impact on the lossless bit-rate for small ROIs - spanning less than 1% of the volume – is negligible, though for larger ROIs can increase up to 10%. Alternatively, JPEG 2000 also supports another type of ROI coding, that is block-based and involves rearranging the code-block data in the code-stream into quality layers, and can be done, unlike the shift-based ROI, after encoding. The impact for this type of ROI on the lossless compression performance, caused by the overhead to signal the extra layers and packets within the code-stream, is negligible (typically less than 0.5% overhead) and depends on the precinct sizes. These overhead figures were obtained for the most likely case that the code-blocks are smaller or equal to their containing precinct. When precincts sizes are chosen to be smaller than the code-block size to support very small ROIs, the overhead will significantly increase. Nonetheless, practice demonstrates that this type of ROI encoding is rarely required.

The classical wavelet transform employed in JPEG 2000 has the drawback of not being able to optimally represent curvilinear discontinuities in images. The origin of this problem stems from the fact that the n-dimensional DWT is assumed to be a separable transform, given by the tensor product of individual 1D-DWTs along their respective orthogonal dimensions. As such, the classical DWT is limited to efficiently representing point-singularities, and it is unable to sparsely capture more complex, higher-order discontinuities such as lines and curves. Directional transforms on the other hand can efficiently adapt and sparsely represent such geometric structures. Significant research regarding different types of directional wavelet transforms has been proposed in the past, for 2D image data [32–35]. To our knowledge a directional discrete wavelet has never been deployed in the context of volumetric image coding. This paper extends directional wavelet transforms to allow support for volumetric image data sets. We note that finding optimized directional wavelets in the axial direction relies on similar concepts as those of motion-compensated temporal filtering, which was very popular in wavelet-based coding of video [36–40]. The state-of-the-art in video compression, which is the recently developed High Efficiency Video Coding (H.265/MPEG-H HEVC) standard [41] is used as benchmark in the experimental section. We note that employing and optimizing video coding techniques for the compression of volumetric medical data calls for thorough investigations and possibly algorithmic modifications, which goes beyond the scope of this paper. This paper shows how the JPEG 2000 standard can be further extended while also maintaining backward compatibility with its current specification, and provides a thorough evaluation of the resulting performance for volumetric medical data. With this respect, our paper analyses the impact on the compression efficiency by examining various compression settings, such as the employed wavelet kernel, the applied decomposition structure, the directional transform and the related entropy-coding settings. Subsequently, it investigates the possibility of exploiting symmetries that are typically present in medical images by additionally decorrelating the wavelet coefficients before the entropy-coding step. This is achieved based upon the work of [42], through the use of a generic block-based intra-band prediction scheme.

Alternatively, it is also possible to replace the axial DWT with an axial KLT for the compression of volumetric medical image data, which is supported through the Multi-Component Transform (MCT) extension of JPEG 2000 Part 2. The KLT is able to optimally remove existing correlations from the image data, but it has two serious drawbacks that make it less suitable for the compression of volumetric medical image data. First of all, the KLT has very high computational complexity and memory usage. Secondly, it comes with the inability to access individual slices, without decoding all slices of the volumetric image. Promising work was done in the context of compression of hyper-spectral image data to (1) provide a reversible KLT implementation [43] that allows scalable lossy-to-lossless compression, and (2) reduce the computational and memory complexities of the KLT by applying a divide-and-conquer strategy [8,9]. However, for the compression of volumetric medical image data with a slice-based 2D-DWT, it was shown that the axial KLT does not maximize the coding gain and consequently performs less or equal to an axial DWT [7]. Combined with the fact that the axial DWT does not suffer from the drawbacks that the KLT brings, we decided to not include it in this study.

Summarizing, this paper provides a comprehensive and unified study on the efficient compression of volumetric medical image data, in both lossy – with rate-distortion optimization – and lossless modes. We present a novel, JPEG 2000 based volumetric image codec, enhanced with additional coding modes based on the aforementioned state-of-the-art techniques, that is, directional wavelet transforms, block-based intra-band prediction and arbitrary decomposition structures. Though these individual coding techniques have been presented in literature before, it is the first time that they are jointly integrated in a volumetric coding system and that directional wavelet transforms are deployed in a volumetric context. Subsequently, the paper provides thorough experimental results, comparing existing coding techniques, such as JPEG 2000 Part 1, JPEG 2000 Part 2 MCT (as supported by DICOM), JPEG 2000 Part 10 (JP3D), JPEG-LS and H.265/MPEG-H HEVC, and the proposed volumetric coding system. The results are obtained using both objective and subjective metrics.

The paper is structured as follows. Section 2 provides a formal definition of the lifting-based Discrete Wavelet Transform as used in this work, along with the various tested extensions to make it directional. Section 3.1 explains how these directional wavelet transforms can be applied on volumetric images, and Section 3.2 describes the tested intra-band prediction methodology. Finally, Section 4 gives a description of the implemented codec that was used for this work, with the results in Section 5 and conclusions in Section 6.

2. Discrete wavelet transform (DWT) and beyond

2.1. Classical or non-directional DWT

This section introduces a general formulation of the discrete wavelet transform (DWT) based on the lifting scheme as applied for JPEG 2000. It is similar to the formulations used in [32–34], but extended for volumetric data. This notation is used as the basis to present extensions to the DWT in subsequent sections.

A forward 1D-DWT separates a given discrete signal into a low-pass L and a high-pass H signal [44] by means of a dyadic wavelet filter bank and down-sampling operations. The resulting low-pass output signal is a scaled version of the original signal with half the number of samples. The high-pass signal contains the missing high-frequency information needed to allow reconstruction with an inverse 1D-DWT. It is important to note that the DWT is critically sampled.

Let $\mathbf{S} = s[\mathbf{l}] | \mathbf{l} \in \mathbf{\Pi}$, with $s[\mathbf{l}] = s[l_x, l_y, l_z]$ and $\mathbf{l} = (l_x, l_y, l_z)$, denote a signal defined on a 3D orthogonal sampling grid $\mathbf{\Pi} = (l_x, l_y, l_z) \in \mathbb{Z}^3$. Let the grid $\mathbf{\Pi}$ be divided into eight distinct sub-grids, where

$$\mathbf{\Pi}_{pqr} = (l_x, l_y, l_z) \in \mathbf{\Pi} | p = l_x \bmod 2, \quad q = l_y \bmod 2, \quad r = l_z \bmod 2 \tag{1}$$

The division of the sampling grid $\mathbf{\Pi}$ into the eight sub-grids $\mathbf{\Pi}_{pqr}$ facilitates defining three instantiations of a 1D-DWT. Each instance filters samples along one of the three possible dimensions of $\mathbf{\Pi}$, labeled as 1D-DWT$_H$, 1D-DWT$_V$ and 1D-DWT$_A$, for the horizontal (along the X-axis), vertical (along the Y-axis) and axial (along the Z-axis) dimensions respectively. Then, based on the just created sub-grids $\mathbf{\Pi}_{pqr}$, we introduce the following six grid-unions:

$$\mathbf{\Pi}^H_\phi = \bigcup_{q,r \in \mathbb{B}} \mathbf{\Pi}_{\phi qr}, \quad \mathbf{\Pi}^V_\phi = \bigcup_{p,r \in \mathbb{B}} \mathbf{\Pi}_{p\phi r}, \quad \mathbf{\Pi}^A_\phi = \bigcup_{p,q \in \mathbb{B}} \mathbf{\Pi}_{pq\phi} | \phi \in \mathbb{B}, \mathbb{B} = \{0;1\} \tag{2}$$

More precisely, the six sub-grid-mergers divide $\mathbf{\Pi}$ into three pairs of odd and even poly-phase components, with respect to a given dimension. Introducing these odd and even poly-phase sub-grids, avoids using ambiguous terminology like rows or columns in the volumetric context.

In [45] it is shown that any biorthogonal 1D-DWT can be expressed as a finite sequence of prediction and update steps, jointly called lifting steps. Given that D indicates either H, V or A, the input signal $\mathbf{S}$ is first decomposed into even $(\mathbf{S}_0^{(0)} = \{\mathbf{S}[\mathbf{l}_0] | \mathbf{l}_0 \in \mathbf{\Pi}_0^D\})$ and odd $(\mathbf{S}_1^{(0)} = \{\mathbf{S}[\mathbf{l}_1] | \mathbf{l}_1 \in \mathbf{\Pi}_1^D\})$ components respectively. Then, in the subsequent stage, a finite sequence of successive prediction and update steps take place, grouped as pairs of lifting steps. Let M be the total number of required lifting steps, then the i^{th} predict and update steps are respectively defined as

$$\mathbf{S}_1^{(i)}[\mathbf{l}_1] = \mathbf{S}_1^{(i-1)}[\mathbf{l}_1] - P_{\mathbf{l}_1}^{d,(i)}(\mathbf{S}_0^{(i-1)}), \quad \forall \mathbf{l}_1 \in \mathbf{\Pi}_1^D \tag{3a}$$

$$\mathbf{S}_0^{(i)}[\mathbf{l}_0] = \mathbf{S}_0^{(i-1)}[\mathbf{l}_0] + U_{\mathbf{l}_0}^{d,(i)}(\mathbf{S}_1^{(i)}), \quad \forall \mathbf{l}_0 \in \mathbf{\Pi}_0^D \tag{3b}$$

The prediction function $P_{\mathbf{l}_1}^{d,(i)}(\cdot)$ and the update function $U_{\mathbf{l}_0}^{d,(i)}(\cdot)$ are functions operating on the sample values of the previous and current lifting steps respectively and returning a scalar output. The variable d represents a constant vector depending on D being H, V or A as respectively $\mathbf{d} = \mathbf{d}_H = (1,0,0)$, $\mathbf{d} = \mathbf{d}_V = (0,1,0)$ or $\mathbf{d} = \mathbf{d}_A = (0,0,1)$. Finally, after M pairs of lifting steps, the even and odd poly-phase values represent the result as respective low-pass $(\mathbf{L}_D)$ and high-pass $(\mathbf{H}_D)$ coefficients, up to scaling factors G_L and G_H:

$$\mathbf{H}_D[\mathbf{l}_1] = G_H \mathbf{S}_1^{(M)}[\mathbf{l}_1], \quad \forall \mathbf{l}_1 \in \mathbf{\Pi}_1^D \tag{4a}$$

$$\mathbf{L}_D[\mathbf{l}_0] = G_L \mathbf{S}_0^{(M)}[\mathbf{l}_0], \quad \forall \mathbf{l}_0 \in \mathbf{\Pi}_0^D \tag{4b}$$

The applied prediction and update functions are defined by:

$$P_{\mathbf{l}_1}^{\mathbf{d},(i)}(\mathbf{S}_0^{(i-1)}) = \sum_{k=-K_P}^{K_P-1} c_{P,i,k} \mathbf{S}_0^{(i-1)}[\mathbf{l}_1 - (2k+1)\mathbf{d}] \tag{5a}$$

$$U_{\mathbf{l}_0}^{\mathbf{d},(i)}(\mathbf{S}_1^{(i)}) = \sum_{k=-K_U}^{K_U-1} c_{U,i,k} \mathbf{S}_1^{(i)}[\mathbf{l}_0 - (2k+1)\mathbf{d}] \tag{5b}$$

where K_P, $c_{P,i,k}$, K_U and $C_{U,i,k}$, as well as G_L, G_H and M, are constants determined by the applied DWT kernel. The scaling factors G_L and G_H handle the normalization of the transform to make it unitary, which is important in order to achieve good lossy coding performances. The normalization factors are computed from the synthesis wavelet filters such that for one decomposition level, the total noise energy is preserved in the reconstructed image, when quantizing the coefficients with very high-rate scalar uniform quantizers and assuming that the quantization noise on the wavelet coefficients is white. Wavelet coefficients in JPEG 2000 can be quantized due to either an explicit but optional uniform with dead-zone quantization step, or implicitly as the consequence of bit-stream truncation during the Embedded Block Coding by Optimized Truncation (EBCOT) process. In both cases, quantization is considered to be uniform.

Typically, the kernel's predict- and update filter coefficients and the scaling factors are rational, or even irrational, numbers, approximated as floating point numbers. Hence, even if the input is given by integer numbers, the output of the filtering operations is no longer guaranteed to be integer. Moreover, due to its limited representation precision, floating point arithmetic is not exact and thus inherently introduces approximation errors, meaning that both the prediction and the update functions are irreversible in practice. In order to ensure perfect reversibility of the transform, one needs to define specialized prediction and update functions that rely on integer calculus alone to prevent rounding errors. The integer-based prediction and update functions are defined by

$$\hat{P}_{\mathbf{l}_1}^{\mathbf{d},(i)}(\mathbf{S}_0^{(i-1)}) = \left\lfloor \sum_{k=-K_P}^{K_P-1} c_{P,i,k} \mathbf{S}_0^{(i-1)}[\mathbf{l}_1 - (2k+1)\mathbf{d}] \right\rfloor \tag{6a}$$

$$\hat{U}_{\mathbf{l}_0}^{\mathbf{d},(i)}(\mathbf{S}_1^{(i)}) = \left\lfloor \sum_{k=-K_U}^{K_U-1} c_{U,i,k} \mathbf{S}_1^{(i)}[\mathbf{l}_0 - (2k+1)\mathbf{d}] + 0.5 \right\rfloor \tag{6b}$$

where $\lfloor \cdots \rfloor$ represents the mathematical floor operator. The loss of accuracy by applying rounding in $\hat{P}_{\mathbf{l}_1}^{\mathbf{d},(i)}(\cdot)$ and $\hat{U}_{\mathbf{l}_0}^{\mathbf{d},(i)}(\cdot)$ depends on the kernel coefficient values and causes a loss in the energy compaction efficiency of the transform. This means that certain kernels are inherently more usable in a lossless compression fashion than other kernels, depending on their respective kernel constants and the ability to maintain a minimal reduction of accuracy due to rounding.

As stated before, the lifting-based forward 1D-DWT$_D$ described above is able to filter a discrete sampled signal into a low-pass (L_D) and a high-pass (H_D) sub-band, along dimension D. Assuming that the two-dimensional DWT (2D-DWT) is separable, it can be written as the tensor product of two separated one-dimensional DWTs (1D-DWT). As such, applying the forward 1D-DWT$_V$ on an input signal $\mathbf{S}$, followed by application of a 1D-DWT $_H$ on the resulting L_V and H_V is equivalent to the forward 2D-DWT as used by JPEG 2000 Part 1 to perform a single level of decomposition, yielding four sub-bands $L_H L_V$, $L_H H_V$, $H_H L_V$ and $H_H H_H$ (the H and V sub-indices will be omitted from now on). As specified in JPEG 2000 Part 1, the complete forward DWT step performs a multi-resolution wavelet analysis, as introduced by Mallat in [46]. The full decomposition then represents an iteration of consecutive 2D-DWT operations on the LL sub-bands generated at different resolution levels (see Fig. 1(a)).

The 3D-DWT (see Fig. 1(b) and (c)) is similarly defined as the tensor product of the independent horizontal, vertical and axial 1D-DWTs. That is, for volumetric instances of a given signal $\mathbf{S}$, the 3D-DWT$_{HVA}$ can be defined as the application of the 1D-DWT$_A$, followed by the 1D-DWT$_V$ and finally the 1D-DWT$_H$, effectively decomposing $\mathbf{S}$ into eight wavelet subbands. With volumetric Mallat (see Fig. 1(c)), only the resulting low-pass (LLL) subband is further analyzed in order to generate a new decomposition level.

2.2. Wavelet filters in JPEG 2000

JPEG 2000 supports two built-in wavelet filter banks, labeled 5 × 3 and 9 × 7, both originating from the same

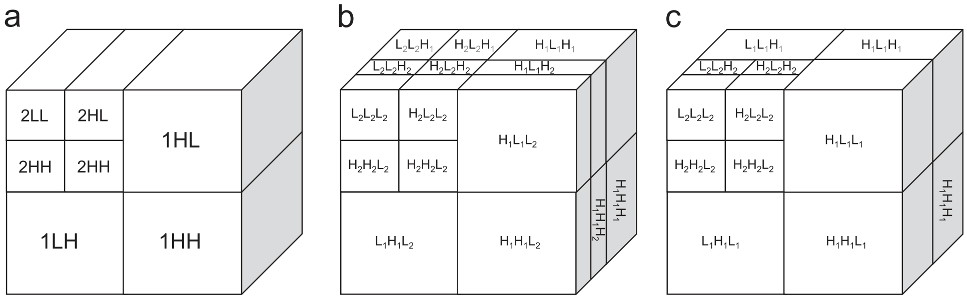

Fig. 1. Examples of possible decomposition structures for volumetric images. (a) 2-level 2D slice-base Mallat decomposition (i.e. without axial DWT). (b) 2-level 2D slice decomposition, combined with a 2-level axial DWT. (c) 2-level 3D Mallat decomposition.

family of biorthogonal Cohen-Daubechies-Feauveau (CDF) wavelets [47]. With the minimum support requirement, both of these wavelets can be constructed by factorizing a maximally flat Daubechies or Dubuc–Deslaurier half-band filter [48]. This means that the 5 × 3 kernel is in fact constructed from the CDF 5/3, which is the shortest symmetrical biorthogonal CDF wavelet with two vanishing moments. Its synthesis scaling function or low-pass filter is a linear B-spline. The 5 × 3 kernel is extremely useful for supporting lossless compression because all of its filters have rational coefficients with the dividends being powers of two. Due to this property, achieving perfect reversibility is possible without loss of performance, as rounding errors can be perfectly controlled.

The 9 × 7 kernel, on the other hand, is a variation on the CDF biorthogonal cubic B-spline construction, using the shortest scaling of order four. It is a variation, because the vanishing moments (6 and 2) are divided up equally on both analysis and synthesis sides in a way that makes the resulting basis functions almost orthogonal [48]. As such, the 9 × 7 kernel has four vanishing moments per wavelet filter. Moreover, this kernel is nearly orthogonal and has a higher factorization-order that the 5 × 3 kernel. As such, it also offers an improved energy compaction performance for lossy compression compared to the 5 × 3 kernel.

An exhaustive study on the performance of various types of wavelet kernels [49] drove the JPEG committee to select the 5 × 3 kernel for lossless and the 9 × 7 kernel for lossy compression for JPEG 2000. One notes also that in literature, most new advancements utilize also the 5 × 3 and 9 × 7 kernels [32,35,42]. Consequently, we will also investigate these kernels in the context of volumetric image coding.

2.3. Alternative wavelet filters

Some proposals in the literature focus on directional transforms [33,34], making use of the so-called (6, 6) interpolation wavelet kernel [45] as an alternative to the 9 × 7 wavelet kernel. For some data sets, the (6, 6) kernel is found to offer higher energy compaction efficiency than the 9 × 7 kernel. Complexity-wise, the 9 × 7 and the (6, 6) kernels are notably different, both with benefits and drawbacks depending on the implementation architecture. On one hand, the (6, 6) filter-bank can be implemented through a single lifting step per filter, unlike the 9 × 7 kernel that requires two lifting steps per filter, which might complicate efficient memory access implementations. On the other hand, the (6, 6) kernel requires more operations per calculated coefficient than the 9 × 7 kernel due to it's large support length of 21 samples. Still, a complexity study of the kernels is beyond the scope of this work and given the good overall performance of the (6, 6) kernel, we decided to also investigate this kernel.

Finally, and in light of the fact that the study is being performed in a volumetric imaging context, we also opted to include results using the Haar filter. This wavelet kernel has the shortest possible support length of one and basically performs a linear prediction to generate the high-pass coefficients. Therefore, it might prove beneficial for the compression of volumetric medical image data sets for which the axial sampling pitch (i.e. intra slice distance) is much larger than the intra slice sampling pitch, as it is often the case with modalities such as CT or Magnetic Resonance Imaging (MRI).

2.4. Non-interpolated directional DWT

As shown in literature [32–35], it is possible to improve transform efficiency by making use of directional wavelet transforms. Modifying the previously defined classic or non-directional 1D-DWT$_D$ to make it directional requires the prediction function (5a) and the update function (5b) to accept more generic direction vectors than $\mathbf{d} = \mathbf{d}_D$ (recall that D can be H, V or A). As such, d now represents the direction vector and is confined only by

$$\mathbf{l}_1 - (2k+1)\mathbf{d} \in \mathbf{\Pi}_0^D, \quad \forall \mathbf{l}_1 \in \mathbf{\Pi}_1^D \wedge k \in [-K_P; K_P[\tag{7a}$$

$$\mathbf{l}_0 - (2k+1)\mathbf{d} \in \mathbf{\Pi}_1^D, \quad \forall \mathbf{l}_0 \in \mathbf{\Pi}_0^D \wedge k \in [-K_U; K_U[\tag{7b}$$

This restriction implies that $d \in \mathbb{Z}^3$ (i.e. no interpolation will be required) and that depending on D one of its coordinates is always odd. Moreover, $\mathbf{d}$ is further restricted so that the line segment between the grid origin $(0,0,0)$ and $\mathbf{d}$ does not intersect with any other point in $\mathbf{\Pi}$ to avoid using linear dependent vectors. An encoder could be made to try out all possible direction vectors within a well-defined range of angles, in order to identify the best possible direction in some rate-distortion sense. However, it is more convenient in practice to predefine a discrete set of direction vectors. This significantly limits the number of possibilities that have to be evaluated by the transform while encoding.

2.5. Interpolated directional DWT

Subsequently, it is possible to modify the prediction and update functions once more to allow direction vectors with fractional coordinates (i.e. $\mathbf{d} \in \mathbb{R}^3$). This implies that the actual prediction and update values to be used by the two functions need to be sub-sampled on the grid $\mathbf{\Pi}$. However, in order to respect the causality constraint for the inverse directional DWT, it is required that the applied interpolation function uses only samples from $\mathbf{\Pi}_0$ during the predict step or samples from $\mathbf{\Pi}_1$ during the update step. The implementation used in this work makes use of a 1D Lanczos [50] interpolation filter $L(X)$ given by

$$L(X) = \begin{cases} \text{sinc}(x)\text{sinc}(x/a) & \text{if } -a < x < a \\ 0 & \text{otherwise} \end{cases} \tag{8}$$

where $\text{sinc}(x) = \sin(x)/x$ and with $a=2$ in order to perform the interpolation as in [32].

3. Investigated extensions for JPEG 2000

3.1. Volumetric coding based on direction-adaptive DWT (DA-DWT)

3.1.1. Introduction

In practice, the directional DWT of Section 2.4 is always employed in an adaptive way, such that the direction vector d dynamically changes depending on spatial location. This adaptability allows the DA-DWT to optimize energy compaction by adjusting itself to distinct localized directionality features in the image data. Doing so, requires the inverse DA-DWT operation of the decoder to know how the forward DA-DWT was applied. Hence, the encoder must also signal the directional information (i.e. the applied direction per spatial location) in the code-stream. It is important to stress that, for a direction-adaptive DWT (DA-DWT) to be profitable, the associated rate cost for encoding this directional information should be small enough so as to not inhibit the overall rate-distortion gain brought by the directional transform itself. Thus, it is practically not possible to allow per pixel direction selection. Segmenting the image data and allowing the directional DWT to employ a suited direction per segment can, however, achieve the trade-off between adaptability and overhead. For this reason, various segmentation strategies were proposed in the past, ranging from simple block-based or tree-based segmentations [32,33,35] to fully arbitrary and content dependent segmentation [34].

3.1.2. Block- and tree-based segmentation

The most straightforward approach is the application of a block-based segmentation, where the transform segments the input into fixed-sized $N_X * N_Y * N_Z$ blocks. Such blocks are subsequently referred to as DA-blocks (distinguishing them from EBCOT code-blocks). This requires only the signaling of the globally chosen DA-block dimensions along with a selected direction vector for each DA-block. Large DA-blocks result in fewer vectors to signal, but at the same time also cause the transform adaptability to be less granular, affecting its energy compaction performance. Making this trade-off between the adaptability of the directional transform and the associated overhead is content dependent and can be effectively solved by introducing a tree-based segmentation to generate the blocks. Using the tree enables measuring the splitting cost versus the coding gain in a rate-distortion optimal way, up to a predefined minimum DA-block size.

3.1.3. Generic segmentation

An alternative to the block-based and tree-based approaches for segmenting the input image was presented in the work of [34] by introducing a generic segmentation driven DA-DWT (SD-DA-DWT). This work proposes an image segmentation scheme at the sample resolution, allowing for maximal directional adaptability of the transform. The so-called Edgementation [51,52] algorithm is used which performs a gradient-based segmentation followed by rate-distortion-driven segment merging with contour simplification to significantly limit the generated overhead. For the contour simplification and contour coding, the proposed algorithm uses the sophisticated algorithm from [53]. However, for the compression of volumetric medical data sets, this last approach has two hurdles. First of all, determining the optimal segmentation that captures directional features in a volumetric dataset is complex. Secondly, the actually applied segmentation description needs to be stored, which severely influences the overall performance gains due to the incurred overhead. For this reason, and given the typical slice-oriented representation of medical volumetric images, our generic segmentation implementation uses the 2D SD-DA-DWT of [34].

3.1.4. Direction vector selection

Similar to [33,34], we use the L_1-norm of the prediction (i.e. high-pass) coefficients as the metric to compare all direction vector candidates, with $L_1(X) = \sum_i |x_i|$. In practice, this means giving preference to those direction vectors that minimize the sum of absolute values of the generated prediction coefficients. Minimizing those coefficients tends to also minimize the bit-rate contributions of the high-pass sub-bands after quantization and entropy coding.

3.1.5. Direction vector coding

To encode the selected direction vector information, we employ a rather conventional signaling methodology that is independent of the employed segmentation technique. More precisely, direction vectors are coded as indexes to the actual vectors in modulus N_D for each of 1D-DWT$_D$ steps independently, where N_D represents the number of possible direction vectors in the respective set for a given D. The direction vectors each represent an angle and as such their respective indexes are ordered in a circular way to support predictive coding. The implementation predicts each direction index from its respective causal neighborhood and the resulting prediction error value is simply sent to a N_D-symbol arithmetic coder.

More advanced coding strategies, such as a tree-based coding approach, can be useful to drive the direction search in a rate-distortion optimized way. However, in practice we found that for volumetric medical images, the bit-rate spent on directionality coding ranges from 0.001 to 0.05 bits per pixel (bpp) depending on the chosen DA-block dimensions. This is extremely low, rendering any potential gains from more advanced techniques from small to negligible.

3.2. *Volumetric coding using block-based intra-band prediction*

It is observed that resulting wavelet coefficients generated by a forward wavelet transform of medical images still contain anatomical symmetry related correlations. As shown in [42], such remaining correlations can be effectively exploited by applying a block-based intra-band prediction scheme to further reduce the energy of the sub-bands. We note that the work of [42] only exploits intra-band redundancies in 2D slices, yet showing improvements of up to 15% in bit-rate reduction compared to JPEG 2000 2D for lossless compression. In light of our search to try to improve the overall compression efficiency for medical volumetric datasets, it is relevant to test the performance of such an intra-band prediction scheme in combination with and compared to the presented volumetric and directional extensions. Thus, based on [42] a generic block-based prediction step was implemented to take place just before EBCOT encodes each sub-band. With squared prediction blocks of 16 × 16, 8 × 8 or 4 × 4 coefficients, let b_i denote the current prediction block under investigation. Then, let $C=\{b_0, b_1, \ldots, b_{i-1}\}$ denote the set of previously encoded blocks in raster-scan order, serving as candidates for the prediction of b_i. Furthermore, let $T_k(b)$ with $k \in [0;7]$ represent the spatial transform operator that takes as input a prediction block b and creates as output any of the eight geometrical possible permutations through 90°-rotation and/or mirroring operations. The encoder then searches for the best prediction candidate block $b_p \in C$ with a corresponding T_t for b_i that satisfies $L_1(T_t(b_p)-b_i)=\min_{k,b \in C}(L_1(T_k(b)-b_i))$.

Once b_p and T_t are found, both the prediction residual $T_t(b_p)-b_i$ and the original block b_i are lossless encoded with EBCOT and the comparison of the resulting rates serves as the decision criteria to enable prediction or not for b_i. Additionally, to be able to decode the predicted blocks, the prediction parameters (b,t) are encoded in the final bit-stream using Exponential Golomb ($k=0$) codes [54] and arithmetic coding [55].

4. Volumetric compression framework, JP3D+DA

In the context of this work, a complete compression system, called JP3D+DA, was designed and implemented in C++.[1] It is in fact a JPEG 2000 Part 1 and Part 10 compliant codec extended to additionally support all of the coding techniques investigated in this paper. Moreover, it can be used to generate results matching the JPEG 2000 Part 2 MCT setup as used by DICOM by disabling the use of volumetric code-blocks. Fig. 2 schematically presents the relevant parts of such a JPEG 2000 compliant codec. Fig. 3, on the other hand, shows the extended JP3D+DA codec employed in this work. Initially, the input image is sent to the transform module, where a mode switch allows to select the type of transform action to perform. This is either an Edgementation based SD-DA-DWT, a block-based DA-DWT, a non-directional DWT or no transform action. With the selection of the first three modes, the output represents the generated sub-band(s) of a single-level decomposition step, along with side-information. These sub-bands are subsequently sent back as input data to the transform module, enabling multi-level wavelet decompositions. Only in the case that no transform was selected, the input data passes through unaltered for further entropy coding in the entropy module. The entropy module contains a switch for the intra-band prediction step to allow further de-correlation of the wavelet coefficients, before encoding them with EBCOT. The module also handles coding the side-information and multiplexes all the data into the final code-stream. We would like to note that the optional interpolation for the DA-DWT occurs in the direction search and the DA-DWT blocks. The implemented wavelet filters for the experiments are the reversible Haar and 5 × 3 kernels and the irreversible 9 × 7 and (6, 6) kernels. Table 1 lists for each of these four wavelet kernels their respective parameters and kernel lifting coefficients.

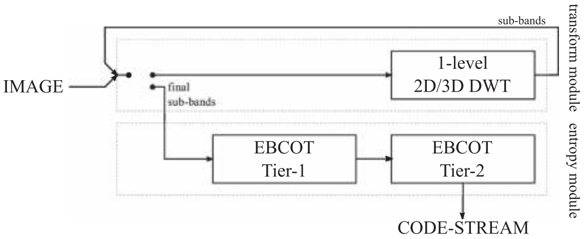

Fig. 2. Overview schematic of the standard JP3D encoder.

For the DA-DWT modes, our codec implementation works with a number of direction vectors that approximate angle steps of 22.5° in the *XY*-plane, while for directions involving the *Z*-dimension, angle steps of 45° were used, as shown in Fig. 4. For the Lanczos interpolated mode, the direction vectors used represent the same direction vectors as in the non-interpolated mode, but with the coefficients normalized such that the leading dimension coordinate becomes exactly 1 (i.e. $x=1$ for H, $y=1$ for V and $z=1$ for A).

The presented framework puts no restrictions on the search for optimal directions for the DA-DWT. Unlike in [32–34], it performs an exhaustive search, using the predefined set of vectors, to achieve transform optimality by maximizing its energy compaction into the low-pass subbands. Doing so guarantees that our DA-DWT implementation will deliver optimal results in MSE sense. The increase in time complexity for the exhaustive search is linearly dependent on the number of employed direction vectors, per applied directional 1D-DWT. Specifically, for our experiments we used a set of 9 directions along all three dimensions. Thus, for slice-based DA-DWT, this will lead to a time-complexity increase relative to the conventional

[1] The SD-DA-DWT implementation was done in MatLab as a separate module.

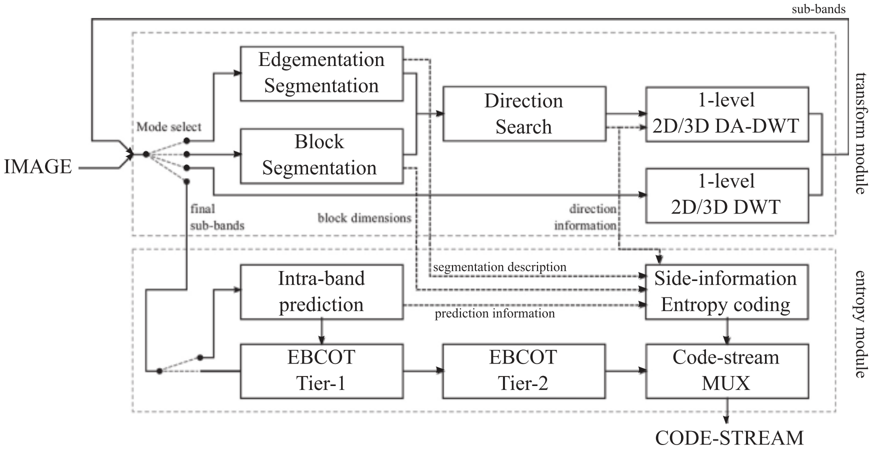

Fig. 3. Schematic representation of the JP3D+DA encoder.

Table 1
Description of the wavelet kernels used for the experiments.

Kernel	M	K_P	K_U	G_L	G_H	i	k	$c_{P,i,k}$	$c_{U,i,k}$
5 × 3	1	1	1	1	1	0	−1, 1	0.5	0.25
Haar	1	1	1	1	1	0	−1	1	0
						0	1	0	0.5
9 × 7	2	1	1	0.812893066	1.230174105	0	−1, 1	1.586134342	−0.052980119
						1	−1, 1	−0.882911076	0.443506852
(6, 6)	1	3	3	1.306139349	1.140657705	0	−1, 1	150/256	150/512
						0	−2, 2	−25/256	−25/512
						0	−3, 3	3/256	3/512

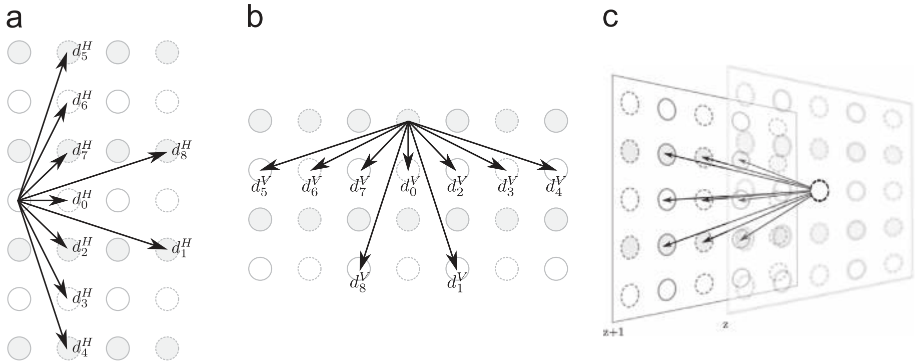

Fig. 4. Direction vectors for the non-interpolated 1D-DWT$_D$. (a) 1D-DWT$_H$, (b) 1D-DWT$_V$ and (c) 1D-DWT$_A$.

DWT by a factor of 18 per decomposed sub-band, while for the axial DA-DWT, this factor will be 9. The encoding of the direction vectors represent a negligible computational cost compared to the other codec components.

Our previous work [56] shows that it is possible to significantly reduce the computational complexity of the direction vector search by a factor of 2.75, with only a very modest penalty in the resulting rate-distortion performance. Practical implementations of the DA-DWT can and probably should make use of such complexity-reduction techniques. Nonetheless, we focus in this paper on maximizing and analyzing the rate-distortion performance, optimizing the performance-complexity trade-off being left as topic of further investigation.

5. Experimental results

5.1. Image data set

In order to provide representative results for medical volumetric image coding, the experiments in this work uses a data set consisting of various commonly used modalities, such as CT, MRI, and Ultrasound (US) scanners (see Table 2 and Fig. 6).

5.2. Bjøntegaard metric

For the evaluation of the lossy compression performance, it is custom to provide a set of rate-distortion curves where PSNR and bit-rate differences between two or more methodologies can be read. However, given the fact that we performed a huge amount of experiments for a multitude of techniques and settings, it quickly became impractical to simply provide the rate-distortion curves. Instead we opted to make use of the Bjøntegaard Delta Peak Signal Noise Ratio [57] (BD-PSNR) metric and in a lesser extent also the Bjøntegaard Delta Bit Rate (BD-BR) metric. Both metrics were originally designed within the Video Coding Experts Group (VCEG) – ITU SG 16 – for the comparison of various video coding techniques. BD-PSNR measures the difference in compression performance by calculating the average difference in quality between two rate-distortion curves, represented by the gray area as illustrated in Fig. 5. The result is a single PSNR value (in dB) that facilitates the objective comparison of the investigated compression techniques. Similarly, the average bit-rate difference can be calculated to obtain a single BD-BR value, representing the average relative difference in bit-rate between the assessed and reference compression techniques. Our particular implementation for calculating the BD-PSNR and BD-BR values was written in Excel VBA [58].

5.3. JPEG 2000 Parts 1, 2 and 10

5.3.1. Volumetric decomposition structures

There is a limit on the useful number of decomposition levels that can be performed on the data to achieve quasi-optimal energy compaction. Although performing more decompositions than what is strictly necessary will not worsen the compression efficiency, it is also preferable to limit the number of decomposition levels in order to reduce processor and memory loads of the DWT and to facilitate efficient random-access functionality. Especially the latter will be jeopardized by the fact that more decomposition levels result in larger support widths and hence poorer random access behavior. For labeling purposes, we introduce the (x, y, z)−notation, with x, y and z representing the number of 1D-DWT decompositions along H, V and A respectively, using the Mallat decomposition structure.

Fig. 7(a) shows the impact of the number of in-slice decompositions (XY), obtained with the 5×3 wavelet kernel in the lossless compression scenario. The bars in the graph each represent per image the loss in bit-rate (bpp) relative to the result obtained with the (6, 6, 0) decomposition. Moreover, the image names are annotated with the absolute bit-rate (in bpp) when using the (6, 6, 0) decomposition. These results clearly show that, generally, four XY decomposition levels for the slices suffice to efficiently compress the images.

Subsequently, Fig. 7(b) shows that along the axial (or Z-) dimension near-optimal efficiency for lossless 5×3 is already reached at two decomposition levels. For some of the volumetric images, even a single decomposition along Z suffices. Moreover, the difference in bit-rate between having two decompositions and the best test case is well below 0.01 bpp for all images. Notice that here the annotated rates refer to the (4, 4, 4) decomposition.

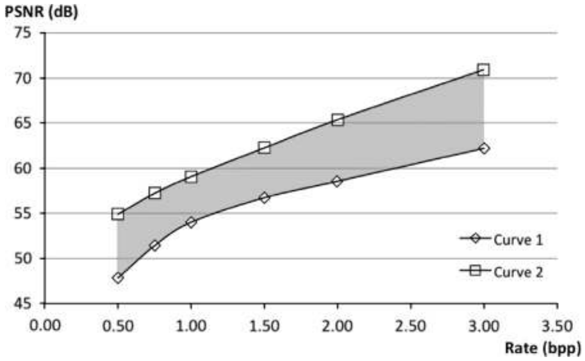

Fig. 5. Visual example of the area (or difference) between two curves, used in order to determine the BD-PSNR.

Table 2
Overview of the volumetric images used in the experiments.

Label	Resolution (w/h/d in px)	Dynamic range (bpp)	0-order entropy (bpp)	Pixel pitch (w/h/d in mm)	Content
CT1	512/512/201	12	8.35	1.000/1.000/1.000	Axial thoracic lung scan
CT2	512/512/242	12	9.32	1.000/1.000/1.000	Axial thoracic lung scan
CT3	512/512/75	12	8.98	1.000/1.000/1.000	Axial spiral arterial scan
CT4	512/512/100	12	7.37	1.000/1.000/1.000	Axial scan of female cadaver (slices 100–199)
CT5	512/512/672	12	6.62	0.977/0.977/2.500	Axial scan of human cadaver (full scan)
CT6	512/512/44	12	8.10	0.660/0.660/5.000	Helical scan of normal chest and mediastinum
MRI1	432/432/250	12	5.13	0.579/0.579/0.579	Normal brain at 3.0 T
MRI2	256/256/200	12	6.15	0.860/0.860/0.800	Normal brain at 1.0 T
MRI3	256/256/100	12	6.41	0.977/0.977/2.000	Normal brain at 1.5 T
US1	500/244/201	8	7.05	NA	Fetal spine Ultrasound
US2	352/242/136	8	7.27	NA	Fetal brain Ultrasound

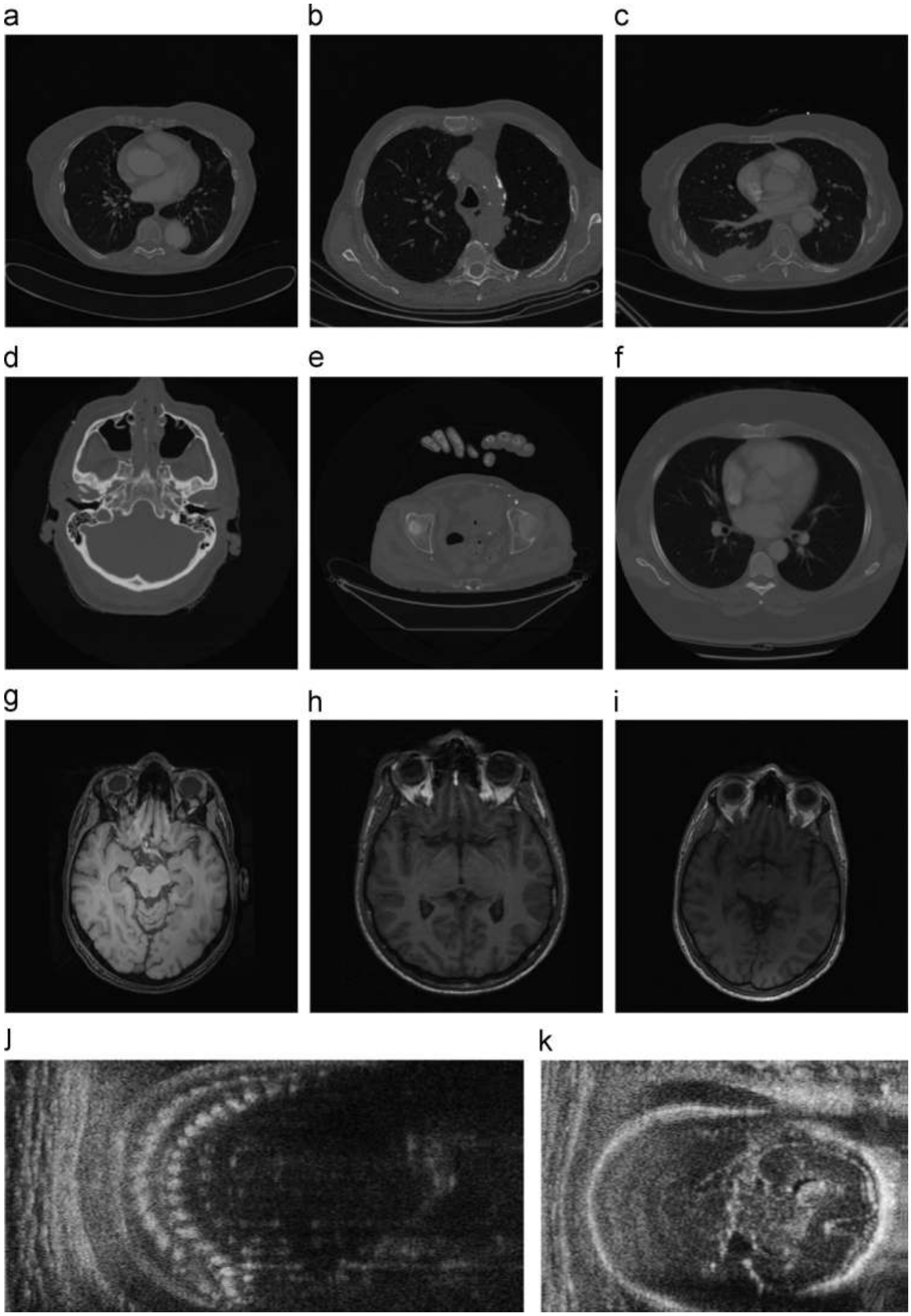

Fig. 6. Selected single slice from each of the images in the data set, resampled to 8-bit for displaying. (a) CT1, (b) CT2, (c) CT3, (d) CT4, (e) CT5, (f) CT6, (g) MRI1, (h) MRI2, (i) MRI3, (j) US1 and (k) US2.

Similar results were found for the 9 × 7 (Fig. 8) and the (6, 6) (Fig. 9) wavelet floating-point kernels. The bars in these figures represent BD-PSNR values, using bit-rates of 0.50, 0.75, 1.0, 1.5, 2.0 and 3.0 bpp, to indicate the loss in quality (dB) w.r.t. the best decomposition structure of each respective image.

5.3.2. Impact of selected wavelet filter

For the lossless compression case, it is well known that the 5 × 3 wavelet kernel performs overall better than the Haar wavelet, largely due to its longer support length. Fig. 10 shows that, even along the axial dimension where inter-pixel

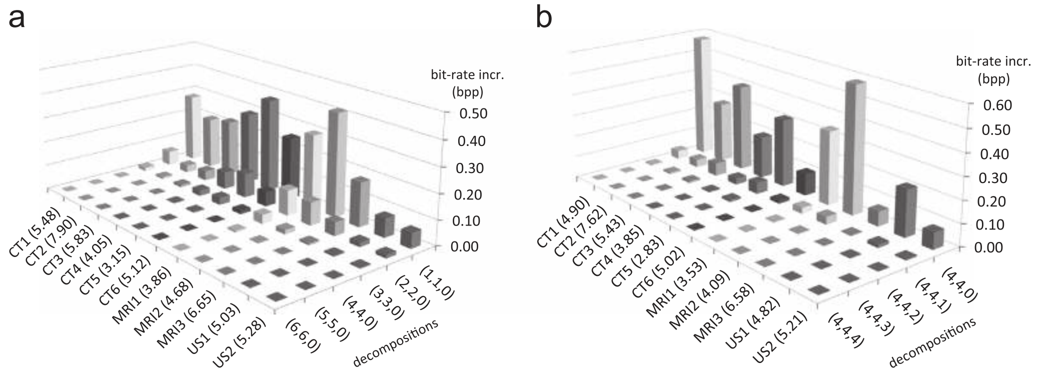

Fig. 7. Performance impact due to the applied decomposition structure with the lossless 5 × 3 wavelet kernel. (a) Varying in-slice only decompositions. The bars represent the increase in bit-rate relative to the (6, 6, 0) decomposition, with the respective reference bit-rate (in bpp) for each image given between parentheses. (b) Fixed 4 in-slice (*XY*) and varying axial (*Z*) decompositions. The bars represent the increase in bit-rate relative to the (4, 4, 4) decomposition, with the respective reference bit-rate (in bpp) for each image given between parentheses.

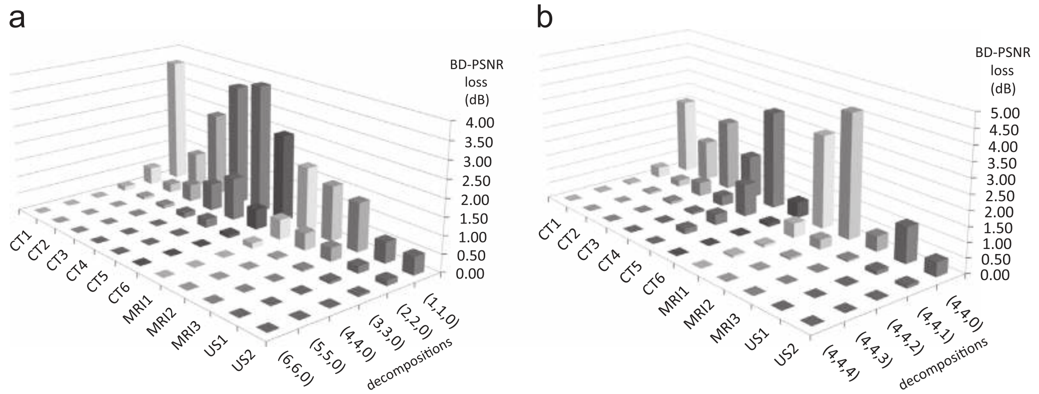

Fig. 8. Performance impact due to the applied decomposition structure in the lossy (0.50 bpp) to near-lossless (3.00 bpp) bit-rate range with the 9 × 7 wavelet kernel. (a) Varying in-slice only decompositions. The bars represent the loss in BD-PSNR (dB) relative to the (6, 6, 0) decomposition. (b) Fixed 4 in-slice (*XY*) and varying axial (*Z*) decompositions. The bars represent the loss in BD-PSNR (dB) relative to the (4, 4, 4) decomposition.

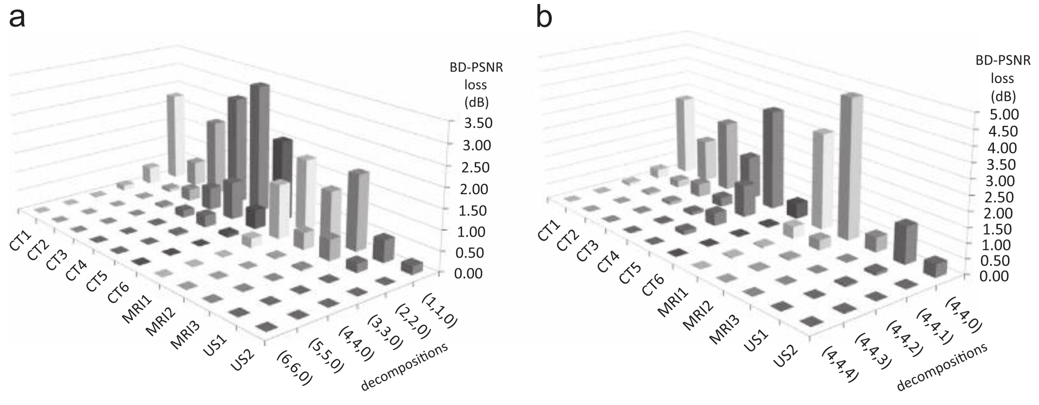

Fig. 9. Performance impact due to the applied decomposition structure in the lossy (0.50 bpp) to near-lossless (3.00 bpp) bit-rate range with the (6, 6) wavelet kernel. (a) Varying in-slice only decompositions. The bars represent the loss in BD-PSNR (dB) relative to the (6, 6, 0) decomposition. (b) Fixed 4 in-slice (*XY*) and varying axial (*Z*) decompositions. The bars represent the loss in BD-PSNR (dB) relative to the (4, 4, 4) decomposition.

correlation is typically less pronounced than for the other two dimensions, the Haar wavelet is less efficient than the 5 × 3 wavelet kernel.

We also notice that the performance gains between having and not having the axial DWT is somewhat linked with the axial resolution of the data set. Images with low axial resolutions such as CT6 or MRI3 contain less inter-slice correlations. Hence, the axial decomposition exhibits relatively small bit-rate gains when compared to those of the images with higher axial resolutions.

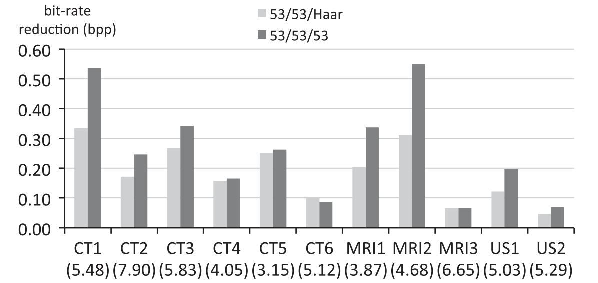

Fig. 10. Difference in lossless rates using (4, 4, 2) decomposition with 5 × 3 and Haar wavelet kernels along axial dimension and code-blocks of 32 × 32 × 32. Bars represent the decrease in bit-rate w.r.t. the (4, 4, 0) decomposition (i.e. 53/53/none, with the actual bit-rate in bpp between parentheses).

For lossy to near-lossless reconstruction, both the 9 × 7 and the (6, 6) kernels offer vastly improved compression performances over the 5 × 3 kernel. Fig. 11 clearly indicates that the two floating-point kernels easily outperform the 5 × 3 integer wavelet kernel (and as such also the Haar wavelet kernel) with overall BD-PSNR improvements of up to 2 dB for target bit-rates of 0.50, 0.75, 1.0, 1.5, 2.0 and 3.0 bits per pixel. These results also show that the (6, 6) wavelet kernel outperforms the 9 × 7 wavelet kernel for 7 out of 11 images.

5.3.3. *The impact of code-block sizes*

In this section we assess the impact of the size of the code-blocks used by EBCOT to code the resulting wavelet coefficients. It is known that large code-blocks benefit compression-wise from the relative long runs of bits sent to the arithmetic coder allowing for better adaptation of the statistical models as opposed to when using smaller code-blocks with their inherent shorter adjustment period for the arithmetic coder. Smaller code-blocks, on the other hand, improve the random access and RD-optimization granularity of the data. So, there is a trade-off to be made between this granularity and the compression efficiency. The resulting effect of the actual code-block sizes on the compression efficiency is not easily observable without experimental data due to the complex interactions between the RD-optimizer, the decoding of truncated arithmetic bit-streams and the small signaling overhead per code-block. We exhaustively encoded our volumetric dataset using a selected series of code-block dimensions, ranging from 16 × 16 × 1 to 64 × 64 × 32. To summarize the results and because all datasets showed identical behavior for the given code-block dimensions, we present the averaged results in Table 3. Please note that volumetric code-blocks are only supported by JP3D and not by JPEG 2000 Part 1 or Part 2, which only support flat code-blocks (i.e. *depth* = 1).

The numbers confirm that the compression efficiency improves by about 0.7 dB or 10% in bit-rate reduction by switching from 16 × 16 × 1 to 32 × 32 × 1, and even further by about 1 dB for 64 × 64 × 1. This is expected behavior because the arithmetic coder can fully adapt to the statistical properties of its input with larger code-blocks (256, 1024 and 4096 coefficients per code-block respectively). The same improvement can be obtained by enabling the use of volumetric code-blocks, such as 16 × 16 × 16. We observe that the optimal results are obtained using code-blocks of 32 × 32 × 32. Note that the observed drop in coding performance starting from 64 × 64 × 32 (and onwards) is caused by the inherent loss of granularity available to EBCOT for code-stream truncation.

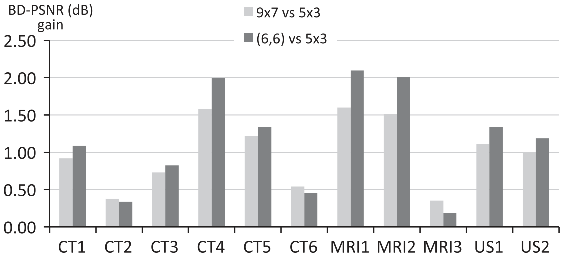

Fig. 11. BD-PSNR gain for 9 × 7 and (6, 6) kernels versus the 5 × 3 kernel while deploying a (4, 4, 2) decomposition.

Table 3
Averaged Bjøntegaard results of all volumetric medical data sets, given per kernel w.r.t. EBCOT with code-blocks of 16 × 16 × 1. Applied decomposition structure was (4, 4, 2) and target bit-rates ranged between 0.25 and 3.0 bpp. These results show the impact of varying code-block block sizes for a given kernel, when compared with a 16 × 16 × 1 setting. For each kernel, best results were observed using code-blocks of 32 × 32 × 32.

Code-block size (W × H × D)	BD-PSNR for 5 × 3 (dB)	BD-PSNR for 9 × 7 (dB)	BD-PSNR for (6, 6) (dB)
32 × 32 × 1	0.78	0.82	1.48
64 × 64 × 1	0.98	1.03	1.69
16 × 16 × 16	1.06	1.09	1.75
16 × 16 × 32	1.10	1.14	1.79
32 × 32 × 32	**1.12**	**1.16**	**1.82**
64 × 64 × 32	1.08	1.13	1.79

5.3.4. *Intermediate conclusions*

The experiments with volumetric medical images show that it is possible to achieve significant improvements in compression performance of 2 dB or more, using only JP3D and compared to JPEG 2000 Part 1. The largest compression performance gain is caused by the extra axial wavelet decomposition step. The results also indicate that for medical images the (4,4,2) decomposition structure is near-optimal and that performing more decompositions would only waste computational resources. We note that recent work [59] provides a novel analysis that models a correlation coefficient for CT images, based on the acquisition parameters of the imaging device. This provides a methodology that predicts whether an axial transform is appropriate or not, before encoding the CT image and potentially saving computational resources. Finally, the introduction of a volumetric EBCOT entropy coder improves the overall compression efficiency and it facilitates improved random access and scalability features without any performance costs.

5.4. *Block-based intra-band prediction*

Looking at the various sub-bands after the multi-level wavelet decomposition of medical volumetric images reveals that the respective wavelet coefficients still show structured symmetries. In order to further de-correlate these remaining symmetries, we tested the potential of applying the block-based intra-band prediction algorithm as described in Section 3.2.

Table 4 shows the results obtained for lossless compression with the 5 × 3 wavelet kernel using a (4, 4, 0) decomposition for J2K with code-blocks of 32 × 32 × 1 and a (4, 4, 2) decomposition with code-blocks of 32 × 32 × 32 for JP3D. The two columns show the respective relative bit-rate reductions by enabling the slice-based prediction algorithm. We point out that these results can still be improved by designing a better tuned algorithm for encoding the prediction parameters (b, t) (see Section 3.2). Even though the intra-band prediction clearly fails on the MRI and US images, the results show that it is indeed possible to achieve significant bit-rate reductions on CT images by means of an extra intra-band prediction step in the wavelet domain. Hence, it might be considered as an extra coding mode, but not as a generic coding tool.

It is worth noting that the intra-band prediction algorithm applies prediction only in the case that the resulting energy of the predicted block is lower than that of the original block. Thus the compression performance gain mainly depends on the initial energy within a sub-band before prediction. On one hand, images containing high amounts of strong edges will still exhibit relatively high energy content in the high-frequency wavelet sub-bands, and thus also have a high potential for further energy reduction by prediction. In contrast, images with primarily smooth regions will have low energy content in the high frequency bands, and hence, they will benefit less from an additional intra-band prediction step. Moreover, larger sub-bands will have more prediction blocks, given their fixed minimum block size, meaning that more prediction block candidates are available than with smaller sub-bands.

5.5. Enabling DA-DWT

5.5.1. SD-DA-DWT

The application of the SD-DA-DWT methodology [34] on volumetric medical datasets is more complicated due to the fact that the described segmentation algorithm only works on 2D images. As such, we opted to use the slice-based approach where the slices of the volumetric datasets are separately segmented and decomposed with the 2D SD-DA-DWT using the (6, 6) kernel. Subsequently, we apply EBCOT coding – instead of QT-L [60,61] – to encode the coefficients. We selected code-blocks of 64 × 64 × 1 in order to avoid inter-slice side effects caused by the 2D SD-DA-DWT. We also decided to ignore the signaling overhead of the segmentation in the final coding bit-rate because no suitable volumetric segmentation compression scheme was readily available. This means that the results presented in Table 5 are only useful to determine whether the SD-DA-DWT will be worth pursuing or not.

Table 4
Bit-rate reduction after applying block-based intra-band prediction.

Image	J2K bit-rate reduction (%)	JP3D bit-rate reduction (%)
CT1	1.34	2.14
CT2	1.80	2.31
CT3	3.35	4.43
CT4	0.00	0.12
CT5	0.00	0.00
CT6	0.18	0.54
MRI1	0.00	0.00
MRI2	0.00	0.00
MRI3	0.00	0.00
US1	0.00	0.00
US2	0.00	0.00
AVG	0.61	0.87

We observe that for medical volumetric images SD-DA-DWT is unable to deliver any significant improvements in overall compression efficiency. Only CT5 and CT6 show modest compression performance gains, but we recall that the signaling overhead is not included here. Moreover, for some of the images, the SD-DA-DWT algorithm as used here fails completely. This can be explained by the fact that (a) the segmentation step is not able to always match regions with similar directional features, and (b) the non-alignment of the segmentation with the code-block structure of EBCOT negatively influences the entropy coder efficiency. Hence, we conclude that no practical compression gain can be obtained from the segmentation-driven transform for the compression of volumetric medical images.

5.5.2. Block-based DA-DWT

The following set of experiments is used to assess the potential gain in compression performance when switching from a traditional DWT to a block-based DA-DWT. Results were obtained using the optimal DWT decomposition structure as determined by the previous experiments, i.e. having 4 decompositions in the slice plane (along the *X*- and *Y*-axes) and 2 decompositions along the axial dimension (i.e. the (4, 4, 2) decomposition). Additionally, as reference we use the (4,4,0) decomposition. The directionally enabled results have their first two decomposition levels of the slices generated with the DA-DWT with DA-blocks of 32 × 32 × 1. For each DA-block, the optimal direction is selected from the set of 9 available vectors by minimizing the L_1-norm of the high-pass coefficients. Recall that two vectors are selected per DA-block, i.e. one for the 1D-DWT$_H$ and one for the 1D-DWT$_V$. Entropy coding of the coefficients is done with EBCOT using

Table 5
Results using one level of SD-DA-DWT on slices without overhead and (4, 4, 0) decomposition w.r.t. classical DWT using the (6, 6) kernel.

Image	BD-PSNR (dB)	BD-BR (%)
CT1	0.172	−2.820
CT2	−0.036	0.963
CT3	0.062	−0.977
CT4	0.025	−0.338
CT5	0.451	−3.883
CT6	0.204	−3.540
MRI1	−0.039	0.332
MRI2	−0.010	0.097
MRI3	0.029	−0.344

volumetric code-blocks of 32 × 32 × 32 pixels. Furthermore, we decided to include the coding overhead of the directions in the results. As such, Fig. 12 shows the impact on the lossless 5 × 3 compression efficiency for volumetric medical images when enabling the slice-based DA-DWT. From these results it can be clearly observed that the slice-based DA-DWT has close to no effect on the lossless compression efficiency. In fact, the only real compression gain comes from the application of the volumetric decomposition structure.

Similarly, for lossy to near-lossless compression, we give results in Table 6 using the (6, 6) wavelet kernel, calculated with bit-rates ranging from 0.25 to 3.0 bpp. All decomposition structures from (1, 1, 0) up to (4, 4, 2) were tested in bulk for each image, after which we selected the best performing decomposition structure using JP3D compliant settings (i.e. without DA-DWT) and the best performing decomposition structure with slice-based DA-DWT enabled. Again, the directional coding overhead was accounted for this test. In line with the lossless coding results, these numbers again confirm that enabling block-based DA-DWT gives at best very modest compression efficiency improvements.

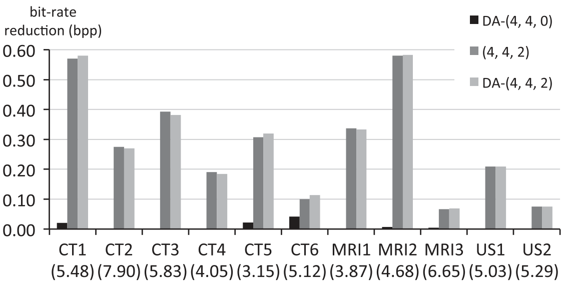

Fig. 12. Bars represent bit-rate reductions (in bpp) and relative to the (4, 4, 0) reference for lossless 5 × 3 using various decomposition structures with and without DA-DWT in slices. The respective reference rates (in bpp) are given between parentheses.

Table 6
Best-of results showing the impact of enabling the non-directional axial decomposition with and without slice-based DA-DWT on medical volumetric images. Directions selection uses L_1-mode. Entropy coding is EBCOT with 32 × 32 × 32 code-blocks. All values are in reference to (4,4,0) coding.

(6, 6) **kernel**	**Best JP3D**		**Best JP3D+DA**	
	BD-PSNR (dB)	**BD-BR (%)**	**BD-PSNR (dB)**	**BD-BR (%)**
CT1	2.68	−29.27	2.72	−30.99
CT2	1.40	−19.44	1.40	−19.44
CT3	2.51	−31.42	2.51	−31.51
CT4	1.49	−15.20	1.50	−15.44
CT5	3.33	−29.35	3.49	−31.40
CT6	0.50	−5.62	0.58	−6.74
MRI1	3.16	−20.05	3.16	−20.05
MRI2	4.52	−38.56	4.52	−38.66
MRI3	0.51	−5.38	0.52	−5.44
US1	1.26	−14.43	1.26	−14.43
US2	0.46	−5.38	0.46	−5.38

5.5.3. *Interpolated DA-DWT*

In [33] the authors have already shown that the introduction of interpolated coefficients to enable using arbitrary direction vectors in the prediction and the update steps of a DA-DWT does not help to improve the compression efficiency. In fact, our results in Table 7 reconfirm this statement. The overall compression efficiency improvement is insignificant, especially when compared to the increased algorithmic complexity. We even noted a slightly negative impact for some of the datasets on the respective image quality when using interpolation.

5.5.4. *Axial DA-DWT*

Fig. 13 shows the impact of enabling DA-DWT for the axial decomposition on volumetric medical images. The reported results were generated using the 5 × 3 wavelet kernel for horizontal and vertical decompositions (4 levels). For the axial decomposition both 5 × 3 and Haar wavelet kernels were tested, where the selection of optimal directions was based on minimizing the L_1-norm of the high-pass coefficients. Coefficients were lossless encoded with code-blocks of 32 × 32 × 32.

These results indicate that with the current DA-DWT implementation, no directional features can be effectively exploited between slices and that in fact the optimal direction vector appears to be d_0^A in most cases. The small performance penalty between 53/53/53 and 53/53/DA-53 is caused by the fact that the L_1-norm selection criterion is only an approximating metric for the final coefficient coding bit-rate and, as such, will not necessarily select the best direction vector in all situations.

5.5.5. *Intermediate conclusions*

Looking at both the block-based DA-DWT and the SD-DA-DWT results, it is clear that typical volumetric medical images, such as those in our test set, contain very few

Table 7
Average Bjøntegaard results of all volumetric images when switching to interpolated direction vectors with in-slice (6, 6) DA-DWT. Entropy coding was done using code-blocks of 32 × 32 × 32 and target bit-rates between 0.50 and 3.0 bpp.

Mode	**BD-PSNR (dB)**	**BD-BR (%)**
iDA-(4, 4, 0)	0.023	−0.215
iDA-(4, 4, 2)	0.008	−0.087

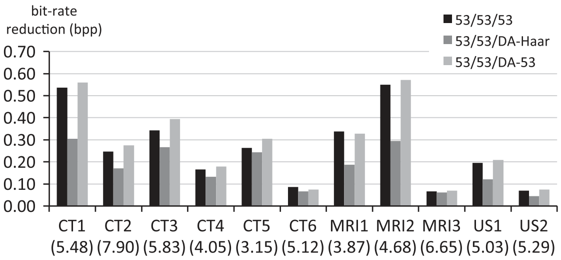

Fig. 13. Shows the effect of enabling DA-DWT for the axial decomposition. The decomposition structure is (4, 4, 2) and bars represent the decrease in bit-rate w.r.t. the (4, 4, 0) decomposition (larger is better). The respective reference rates (in bpp) are given between parentheses.

directional features that can be exploited. Moreover, SD-DA-DWT requires specifying a set of parameters, such as a smallest region size or the number of final regions that are difficult to optimize for. The presented results were obtained by varying these segmentation parameters and only keeping the best found results (to no effect) to report. As such, it can be concluded that the usefulness of the investigated directional DWTs essentially depends on the tradeoff between algorithmic complexities and potential compression efficiency gains. In cases where the data does contain high directionality features, it might make sense to apply the DA-DWT, provided that the generated directional overhead can be managed effectively [32–34]. However, for typical volumetric medical images like the ones we tested on, the potential gain in compression efficiency seems small.

5.6. Comparison with H.265/MPEG-H HEVC and JPEG-LS

To put the previously presented compression results into perspective, we also include results obtained with (1) the upcoming H.265/MPEG-H HEVC [41] video coding standard and (2) the JPEG-LS [11,12] image coding standard. H.265/MPEG-H HEVC is a promising state-of-the-art video codec that supports lossy-to-lossless compression for 2D and 2D+T sequences. Moreover, it also provides an excellent intra-frame coder that competes well with image coding standards, such as JPEG 2000 and JPEG-LS. Since volumetric medical images can be considered to be 2D+T, with T being the slice dimension, an evaluation of the rate-distortion performances of H.265/HEVC is valuable. JPEG-LS, on the other hand, is a low-complexity near-lossless to lossless image codec that offers good lossless compression performance.

For our experiments, we used the H.265/HEVC reference software (SVN revision 4089, range extensions branch, version 15.0_RExt8.0, compiled with high bit-depth support) which is freely available for download [62]. We used two of the default configurations that accompany the source code; 1) the all intra-frame (HEVC-AI) configuration and 2) the random-access (HEVC-RA) configuration. In order to maximize the compression performance, we enabled full-frame search (SearchRange=0) on both configurations and in the case of the HEVC-RA we changed the intra period to encode only a single initial key-frame (IntraPeriod=−1). We recognize that this sacrifices the original random access intention of that configuration, but for this work we prioritize on measuring the maximum compression performance potential of H.265/HEVC. Please note that enabling lossless compression in the H.265/HEVC reference software additionally requires enabling both the TransquantBypassEnableFlag and CUTransquantBypassFlagValue options [62].

The results in Table 8 show that for the given content JP3D delivers the overall best lossless compression performance. When combined with the intra-band block-prediction the compression performance improves even more. Only on CT5, MRI1 and MRI3 H.265/HEVC was able to outperform JP3D and JP3D+BP. It needs to be mentioned that JPEG 2000 with Part 2 performs very similar to JP3D.

Lossy results are given in Fig. 14 where we compared HEVC-AI, HEVC-RA, JPEG 2000 Part 1 and JP3D. The JPEG 2000 results were obtained using the 9 × 7 wavelet kernel with code-blocks of 64 × 64 × 1 for JPEG 2000 Part 1 and 32 × 32 × 32 for JP3D. We also point out that the curves are drawn as-is, based on the numeric results that were measured. From these results JP3D is the best performing codec for all high bit-depth images – i.e. all but the UltraSound images. Only with CT3 it seems that H.265/HEVC surpasses JP3D around the 1.5 bpp bit-rate. As shown, many of the HEVC curves stop early at relatively low bit-rates, without reaching the requested high quality point, even though the chosen QP value in these cases was equal to 1. For example, the H.265/MPEG-H HEVC codec was unable to compress the CT5 image at any bit-rates above 1 bpp. This suggests that the current H.265/HEVC standard still needs tuning with respect to the quantization parameter values for high bit-depth image content. We expect that this issue can and will be resolved. However, this signifies that using H.265/HEVC for compression of volumetric medical images is not yet straightforward, while the decade old JPEG 2000 standard just works as designed, regardless of the sample bit-depth.

Table 8
Comparison of lossless rates (in bpp) obtained with respectively JPEG-LS, H.265/HEVC All-Intra, H.265/HEVC Random Access, JPEG 2000 Part 1 with (4, 4, 0) decomposition, JPEG 2000 Part 2 using (4, 4, 2), JP3D using (4, 4, 2) and JP3D with intra-band block prediction enabled. For the JPEG 2000-based compression, the results use the 5 × 3 wavelet kernel and code-blocks sizes that performed best (being 64 × 64 × 1 and 32 × 32 × 32). The best result for each image is presented in bold.

Image	JPEG-LS (bpp)	HEVC-AI (bpp)	HEVC-RA (bpp)	J2K-P1 (bpp)	J2K-P2 (bpp)	JP3D (bpp)	JP3D+BP (bpp)
CT1	5.42	5.55	5.01	5.52	4.93	4.91	**4.80**
CT2	7.83	7.93	7.73	7.93	7.65	7.63	**7.45**
CT3	5.89	6.14	5.69	5.87	5.46	5.44	**5.20**
CT4	4.01	4.28	4.12	4.09	3.88	3.86	**3.85**
CT5	2.99	3.07	**2.75**	3.20	2.87	2.84	2.84
CT6	**4.91**	5.12	5.08	5.17	5.05	5.02	5.00
MRI1	3.73	3.71	**3.37**	3.89	3.55	3.52	3.56
MRI2	4.58	4.68	4.11	4.72	4.14	**4.09**	4.12
MRI3	6.53	6.50	**6.46**	6.69	6.62	6.58	6.63
US1	5.20	5.48	5.23	5.07	4.85	**4.82**	**4.82**
US2	5.47	5.77	5.68	5.33	5.25	**5.21**	**5.21**

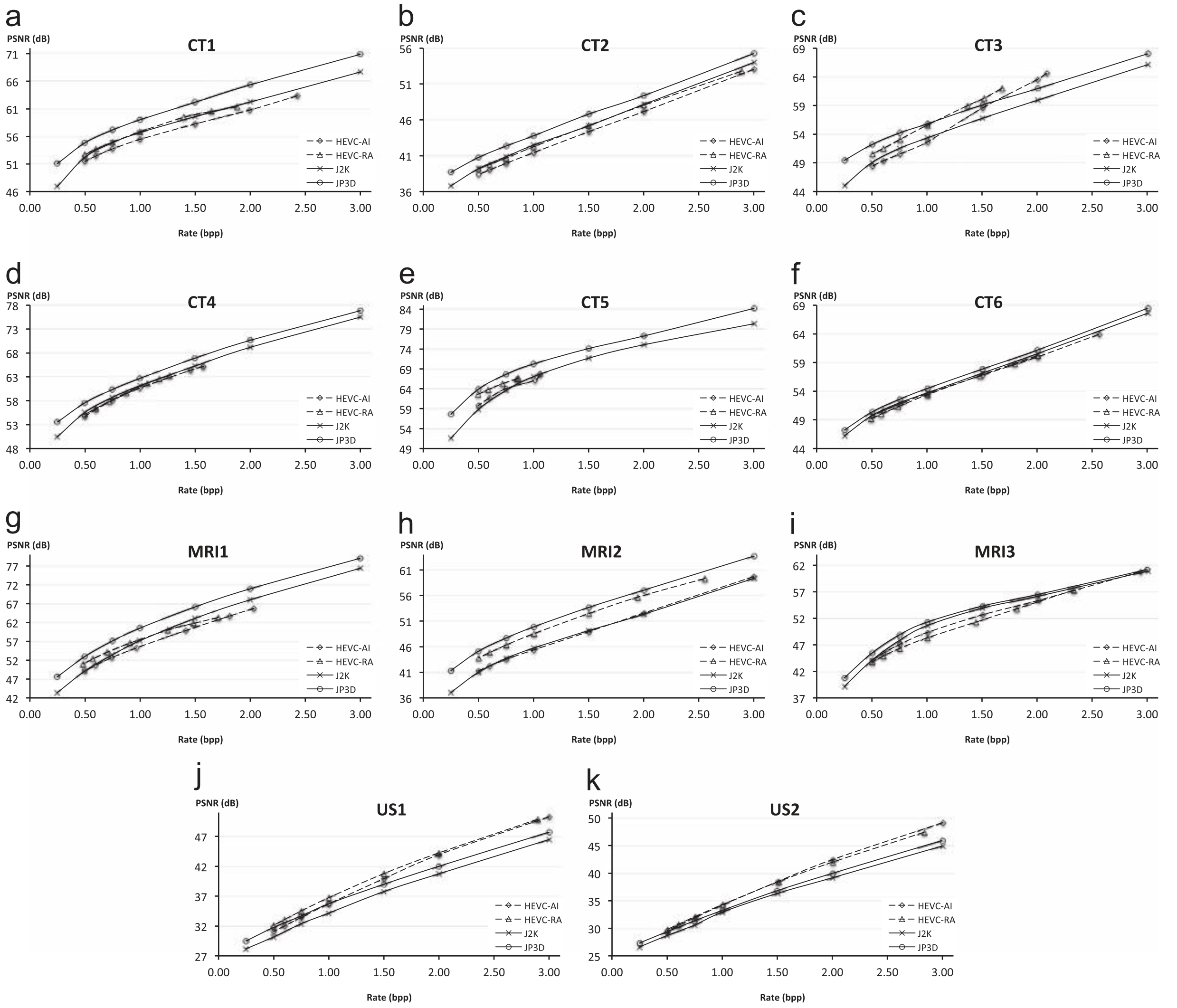

Fig. 14. Comparison of lossy (between 0.50 and 3.00 bpp) compression with H.265/HEVC All-Intra, H.265/HEVC Random Access, JPEG 2000 Part 1 with (4, 4, 0) decomposition, JPEG 2000 Part 2 using (4, 4, 2), JP3D using (4, 4, 2) and JP3D with intra-band block prediction enabled. For the JPEG 2000-based compression, the results use the 9 × 7 wavelet kernel and code-blocks sizes that performed best (being 64 × 64 × 1 and 32 × 32 × 32). (a) CT1, (b) CT2, (c) CT3, (d) CT4, (e) CT5, (f) CT6, (g) MRI1, (h) MRI2, (i) MRI3, (j) US1 and (k) US2.

Table 9
Execution times in million pixels per second (MP/s) for the encoding of volumetric medical images using respectively JPEG-LS, H.265/HEVC All-Intra and H.265/HEVC Random Access, followed by respectively JPEG 2000 Part 1, JP3D and JP3D+DA with 5 × 3 and 9 × 7 wavelet kernels. Bottom row reports the averages over the complete data set.

Image	JPEG-LS (MP/s)	HEVC-AI (MP/s)	HEVC-RA (MP/s)	5 × 3 kernel			9 × 7 kernel		
				J2K-P1 (MP/s)	JP3D (MP/s)	JP3D+DA (MP/s)	J2K-P1 (MP/s)	JP3D (MP/s)	JP3D+DA (MP/s)
CT1	12.43	0.11	0.02	1.87	1.76	0.11	1.41	1.30	0.07
CT2	10.92	0.10	0.02	1.67	1.52	0.11	1.27	1.17	0.07
CT3	11.81	0.11	0.02	2.01	1.83	0.11	1.41	1.35	0.07
CT4	14.68	0.12	0.02	2.30	2.07	0.11	1.60	1.47	0.07
CT5	15.95	0.12	0.02	2.16	2.04	0.11	1.55	1.40	0.07
CT6	14.20	0.12	0.02	2.20	1.82	0.11	1.45	1.42	0.07
MRI1	17.95	0.12	0.03	1.91	1.79	0.12	1.41	1.34	0.06
MRI2	13.70	0.12	0.02	2.22	2.00	0.11	1.50	1.46	0.07
MRI3	10.76	0.12	0.02	1.57	1.40	0.11	1.17	1.10	0.07
US1	13.80	0.10	0.02	1.86	1.75	0.11	1.41	1.27	0.07
US2	13.33	0.10	0.02	1.96	1.79	0.11	1.45	1.30	0.07
Average	13.59	0.11	0.02	1.98	1.80	0.11	1.42	1.33	0.07

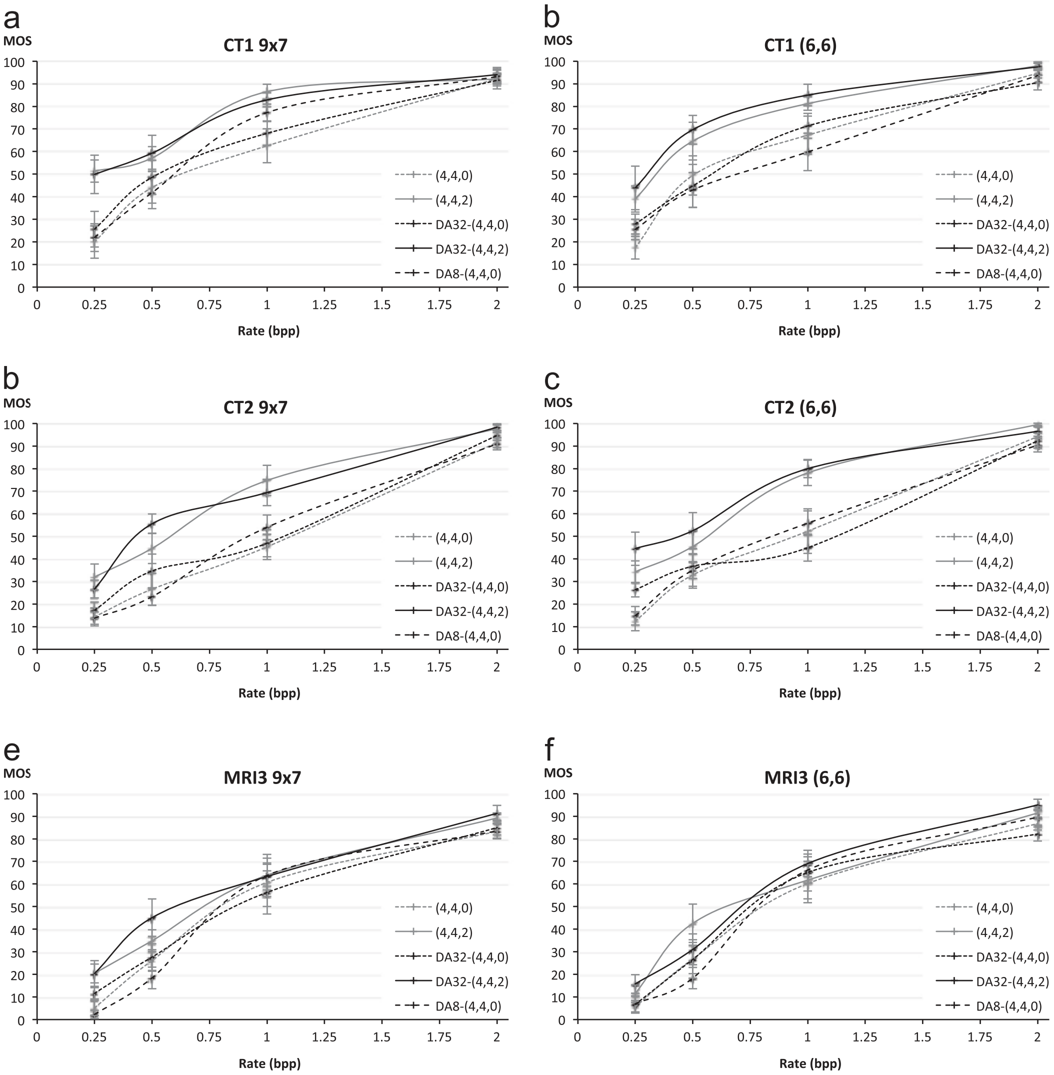

Fig. 15. MOS results using 15 observers. (a) CT1 (slice 20) with 9 × 7, (b) CT1 (slice 20) with (6, 6), (c) CT2 (slice 20) with 9 × 7, (d) CT2 (slice 20) with (6, 6), (e) MRI3 (slice 20) with 9 × 7 and (f) MRI3 (slice 20) with (6 ,6).

Finally, it has to be noted that JP3D is a scalable codec depicting an extremely fine granularity while HEVC is not. Scalable extensions of H.265/HEVC are proposed though delivering a coarse scalability at the expense of some loss in terms of rate-distortion performance. Hence, with this respect the reported experimental results were obtained under very favorable test conditions for H.265/HEVC.

5.7. Computational complexity

In this section we present a selection of binary execution times on a common hardware platform (Dual Intel Xeon E5620 with 144 GB RAM) in order to provide some indication of the algorithmic complexity of the aforementioned techniques. We opted to also include H.265/HEVC results to provide an indication about the complexities of the tested algorithms. However, it is of utmost importance to understand that a direct comparison between two very different implementations, in this case being our JP3D + DA codec and the H.265/HEVC codec, is not straightforward to do, especially with these codecs representing two fundamentally different techniques. Moreover, the implementations used in this paper were not optimized for computational performance. Hence, the reader should interpret the reported computational complexity results with caution and rather consider them as indicative measurements. Comparisons between the different JPEG 2000 implementations are of course much more informative, since they all build upon the same code base.

The results, presented in Table 9, show that the H.265/HEVC codec perform at consistent speeds, regardless of the input data. As expected, the All-Intra mode is a magnitude faster than the Random-Access mode. The pixel processing speed of our codec, on the other hand, is dependent on the image content. This is caused by the variation in the number of bit-planes that EBCOT needs to process while encoding the wavelet coefficients, which heavily depend on the input data, the transform and the decomposition

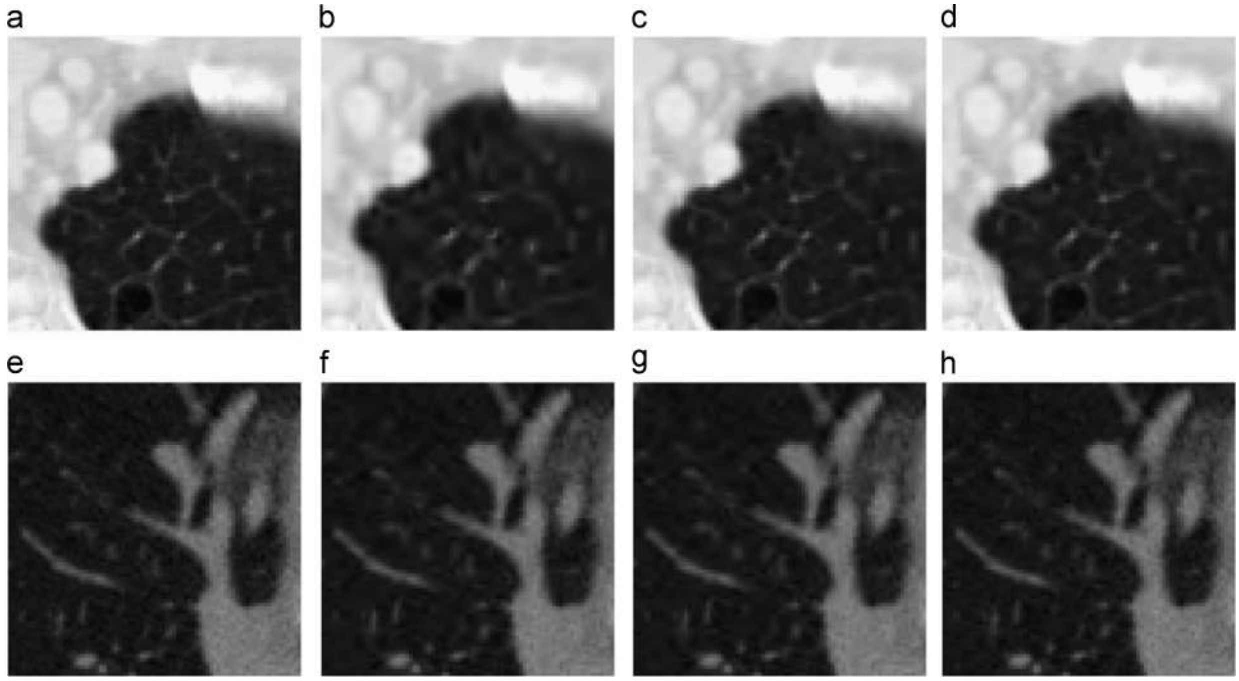

Fig. 16. Visual crops of CT1 (a–d) and CT2 (e–h), using the (6, 6) kernel and resampled to 8 bit. (a) Crop of original CT1, (b) (4, 4, 0) at 0.25 bpp, (c) (4, 4, 2) at 0.25 bpp, (d) DA-(4, 4, 0) at 0.25 bpp, (e) Crop of original CT2, (f) (4, 4, 0) at 1 bpp, (g) DA-(4, 4, 0) at 1 bpp and (h) (4, 4, 2) at 1 bpp.

structure. Furthermore, the difference between the 5 × 3 and the 9 × 7 wavelet kernels is mainly due to the difference in processing speed between integer and floating-point calculations on Intel CPU architectures. The integer 5 × 3 kernel is clearly faster than the 9 × 7 kernel. JPEG-LS is extremely fast and easily outperforms the other codecs in computational load.

5.8. *Visual comparisons*

To complete this work we performed subjective experiments in order to assess the perceptual quality of the discussed methodologies. In particular we show the impact on the perceptual quality of medical images when using directional wavelet transforms, specifically at bit-rates below 2.0 bpp. The observers were all experts in the field of image and video coding and are able to easily recognize compression artifacts that typically occur at low bit-rates due to the applied compression techniques. We advise that further studies be performed with professional radiologists as observers if clinical validation is required. Still, our perceptual study is a critical first step to deliver unbiased verification of the usefulness of the described coding methodologies in the context of medical volumetric image coding.

The experiments were conducted in a well-controlled test environment that complies with the specifications of ITU-R BT.500-11 [63]. The test room is illuminated to 64 lux with indirect fluorescent tube lights having a color temperature of 6500 K and is completely sealed to block external light sources. The walls and ceiling are painted in mid-gray (RAL 7047). As such, the room conforms to the CIE Standard Illuminant D65 specification. The actual display device used during the experiments is a medical grade Barco Coronis Fusion 6MP with a DICOM calibrated color profile.

The experiments use a single slice (slice 20) from a selection of 3 medical images (CT1, CT2 and MRI3) encoded at bit-rates of 0.25, 0.5, 1 and 2 bpp. For the wavelet transform we applied (6, 6) and 9 × 7 kernels using (4, 4, 0) and (4, 4, 2) decomposition modes, both with and without DA-DWT. The DA-DWT enabled compression uses DA-blocks of 8 × 8 × 1 and 32 × 32 × 1 as is indicated in the results. The presented images were manually converted to 8 bit pixel depth enabling them to be properly shown on the monitor. This conversion was done in a comparable way as used in medical image viewers by manually selecting the lower and upper luminance values that map to 0 and 255 respectively, taking care that the resulting image clearly shows useful content.

For the experiments, we employed the simultaneous double-stimulus test [63] in which 15 observers were asked to score the quality of the impaired image with a discrete 0–100 scale score. A score of 0 represents visible and extremely annoying compression artifacts while a score of 100 means that visually no difference with the original image is noticeable. The presented results are the Mean Observer Scores (MOS), representing the average score per test-point over all observers. The raw scores were first normalized and outliers were removed [64]. All presented results are accompanied with the 95% confidence interval (CI95), indicated by error bars.

Fig. 15 shows the MOS of the observed images. These subjective results confirm what was observed from the previously described objective BD-PSNR-based results.

Regardless of the content, no significant visual differences exists between the classic DWT and the directional adaptive DWT. E.g. in the case of the (4, 4, 2) decomposition with and without DA-DWT the curves are within the CI95 boundaries of each other. This shows that the visual differences are extremely moderate to non-existent. On the other hand, the subjective results also show a noticeable difference between the (4, 4, 0) and the (4, 4, 2) decompositions at bit-rates below 2 bpp, suggesting that the final image quality indeed benefits from the extra axial wavelet decompositions. At very low bit-rates, compression artifacts such as wavelet ringing and blurring effects manifested more clearly. Fig. 16 provides some crops to visualize the observed quality for two of the tested images.

5.9. Summarizing rate-distortion performance tables

Finally, we provide plain rate-distortion figures in Tables 10 and 11 to supplement the previously reported BD-PSNR values.

Table 10
Summary of the best of lossy compression performance results for 9 × 7 and (6,6) kernels at target bit-rates 0.5, 1.00, 1.50 and 3.00 bpp. J2K-P1 = JPEG 2000 Part 1, J2K-P2 = JPEG 2000 Part 2, JP3D = JPEG 2000 Part 10, JP3D + DA = JP3D with DA-DWT enabled. Code-block sizes for J2K-P1 and J2K-P2 were 64 × 64 × 1, while JP3D and JP3D + DA used 32 × 32 × 32.

Image	Bit-rate	PSNR for 9 × 7 (dB)				PSNR for (6,6) (dB)			
	(bpp)	J2K-P1	J2K-P2	JP3D	JP3D + DA	J2K-P1	J2K-P2	JP3D	JP3D + DA
CT1	3.00	67.68	70.78	70.89	70.85	67.59	70.91	71.01	70.94
CT1	1.50	59.58	62.07	62.18	62.15	59.58	62.27	62.37	62.31
CT1	1.00	56.68	58.92	59.01	59.00	56.76	59.09	59.19	59.15
CT1	0.50	52.22	54.74	54.83	54.98	52.42	54.95	55.05	55.13
CT2	3.00	54.01	55.08	55.22	55.12	53.86	55.20	55.33	55.20
CT2	1.50	45.14	46.69	46.80	46.75	45.17	46.49	46.60	46.53
CT2	1.00	42.47	43.67	43.77	43.74	42.35	43.64	43.74	43.69
CT2	0.50	39.17	40.64	40.75	40.76	39.07	40.55	40.64	40.63
CT3	3.00	66.10	67.86	68.01	67.87	66.27	68.00	68.13	68.04
CT3	1.50	56.74	58.98	59.09	59.01	56.86	59.04	59.12	59.07
CT3	1.00	53.33	55.69	55.78	55.72	53.37	55.84	55.91	55.88
CT3	0.50	49.03	52.10	52.19	52.18	49.15	52.16	52.23	52.23
CT4	3.00	75.49	76.73	76.84	76.76	75.81	77.00	77.09	77.06
CT4	1.50	65.26	66.78	66.91	66.79	65.82	67.33	67.48	67.38
CT4	1.00	61.16	62.54	62.69	62.60	61.60	63.02	63.15	63.09
CT4	0.50	55.57	57.42	57.58	57.60	55.88	57.76	57.96	57.98
CT5	3.00	80.31	83.87	84.17	84.11	80.20	83.01	83.19	83.13
CT5	1.50	71.67	73.97	74.09	74.04	71.68	74.12	74.25	74.19
CT5	1.00	67.05	70.08	70.24	70.27	67.08	70.15	70.29	70.29
CT5	0.50	58.81	63.65	63.90	64.31	58.94	63.86	64.08	64.39
CT6	3.00	67.59	68.22	68.41	68.47	67.53	68.18	68.35	68.44
CT6	1.50	57.12	57.74	57.83	57.86	57.14	57.61	57.72	57.73
CT6	1.00	53.69	54.37	54.46	54.48	53.58	54.26	54.36	54.38
CT6	0.50	49.73	50.27	50.36	50.47	49.60	50.16	50.27	50.40
MRI1	3.00	76.35	78.89	79.03	78.98	76.58	79.09	79.16	79.12
MRI1	1.50	63.09	66.02	66.13	66.05	63.52	66.59	66.69	66.65
MRI1	1.00	57.17	60.37	60.51	60.44	57.62	61.02	61.16	61.11
MRI1	0.50	49.12	52.79	52.99	52.95	49.45	53.37	53.55	53.53
MRI2	3.00	59.36	63.48	63.69	63.60	59.49	64.10	64.31	64.24
MRI2	1.50	49.15	53.51	53.67	53.62	49.23	54.18	54.33	54.27
MRI2	1.00	45.73	49.71	49.86	49.82	45.72	50.29	50.42	50.37
MRI2	0.50	41.02	45.00	45.12	45.10	41.03	45.36	45.49	45.46
MRI3	3.00	60.88	61.02	61.16	61.11	60.66	60.95	61.11	61.00
MRI3	1.50	53.89	54.15	54.29	54.24	53.69	53.96	54.09	54.03
MRI3	1.00	50.61	51.16	51.26	51.26	50.42	51.08	51.18	51.16
MRI3	0.50	44.05	45.26	45.45	45.45	44.01	45.17	45.29	45.29
US1	3.00	46.46	47.60	47.74	47.72	47.46	48.38	48.53	48.51
US1	1.50	37.76	38.84	38.99	38.98	38.16	39.26	39.40	39.37
US1	1.00	34.16	35.57	35.73	35.73	34.46	35.72	35.86	35.85
US1	0.50	30.15	31.67	31.78	31.76	30.11	31.75	31.89	31.88
US2	3.00	44.94	45.80	45.93	45.92	45.90	46.42	46.59	46.57
US2	1.50	36.33	36.67	36.77	36.75	36.77	37.19	37.36	37.35
US2	1.00	32.92	33.07	33.24	33.24	32.85	33.22	33.36	33.35
US2	0.50	28.67	29.24	29.39	29.37	28.68	29.06	29.19	29.19

Table 11
Summary of the H.265/HEVC-based lossy compression performance results. HEVC-AI=All-Intra mode and HEVC-RA=Random Access mode. All results were obtained using the high bit-depth enabled compilation of HM 15.0 Rext 8.0.

Image	HEVC-AI		HEVC-RA	
	Bit-rate (bpp)	**PSNR (db)**	**Bit-rate (bpp)**	**PSNR (db)**
CT1	2.43	63.35	1.88	61.35
CT1	1.99	60.78	1.65	60.53
CT1	1.50	58.25	1.40	59.48
CT1	1.00	55.47	1.00	56.85
CT2	3.00	53.07	2.89	52.79
CT2	2.00	47.15	2.00	48.02
CT2	1.50	44.34	1.50	45.20
CT2	1.00	41.42	1.00	42.23
CT3	2.08	64.63	1.68	61.94
CT3	2.00	63.52	1.51	60.21
CT3	1.50	58.54	1.36	58.94
CT3	1.00	52.61	1.00	55.55
CT4	1.58	65.16	1.27	63.28
CT4	1.58	65.16	1.18	62.59
CT4	1.46	64.41	1.07	61.67
CT4	1.00	60.67	0.88	59.68
CT5	1.06	67.75	0.86	66.60
CT5	1.06	67.75	0.86	66.62
CT5	1.06	67.75	0.86	66.63
CT5	1.01	66.16	0.85	66.49
CT6	2.56	63.88	2.00	60.17
CT6	2.00	59.98	1.79	58.81
CT6	1.50	56.78	1.47	56.67
CT6	1.00	53.57	1.00	53.35
MRI1	2.03	65.76	1.71	63.31
MRI1	1.81	63.72	1.49	61.77
MRI1	1.42	59.89	1.25	60.00
MRI1	0.97	55.29	0.91	56.57
MRI2	3.00	59.69	2.55	59.30
MRI2	2.00	52.59	1.94	55.67
MRI2	1.50	48.94	1.50	52.47
MRI2	1.00	45.37	1.00	48.54
MRI3	2.93	60.73	2.33	57.26
MRI3	2.01	55.34	1.81	53.74
MRI3	1.50	52.63	1.44	51.26
MRI3	1.00	49.31	1.00	48.29
US1	3.00	50.30	2.90	49.86
US1	2.00	43.94	2.00	44.24
US1	1.50	39.93	1.50	40.78
US1	1.00	35.71	1.00	36.75
US2	3.00	49.14	2.83	47.38
US2	2.00	42.46	2.00	41.99
US2	1.50	38.45	1.51	38.44
US2	1.00	34.19	1.00	34.31

6. Conclusions

First of all, in medical imaging, it is not well defined what the optimal and most efficient settings for data compression are. The actual context that determines when to use lossless versus lossy compression and the actual parameters for tweaking the compression system are left to the judgment of the implementer and the use case. Compression standards, such as JPEG or JPEG-LS, have a very limited set of options regarding their setup. However, in the case of JPEG 2000 a lot more options are available. In this work, we have spent a huge effort into testing a wide range of the possible settings for the compression of volumetric medical images. Our results show that for volumetric medical images the addition of two wavelet decompositions along the axial dimension suffices to optimize the compression efficiency in all cases for all of the tested images. This means that the (4, 4, 2) wavelet decomposition structure is in practice the advisable choice for the compression of volumetric medical images.

Secondly, we also show that the addition of directional adaptive wavelet filters can improve the compression efficiency in some specific cases, but at the cost of increased time and memory complexities. However, given that the additional complexity cost is huge, and that the achieved gains are often minimal, we conclude that the use of directional wavelet transforms for the compression of volumetric CT, MRI or US image data is not worth the effort.

Third, it seems that the addition of an intra-band prediction step can slightly improve the compression performance for a wavelet based coding system for some specific images, but again not without increasing the codec's complexity. Moreover, efficiently managing the extra overhead is not without difficulties and requires further study to avoid negating the achievable transform gains.

Fourth, when comparing JPEG 2000 with H.265/HEVC we see that on most images JP3D, optionally with intra-band block prediction, outperforms H.265/HEVC for almost all images. Though, for those images where H.265/HEVC is better, the bit-rate improvement is also significant. This suggests that H.265/HEVC might not yet be well-tuned for the compression of higher bit-depth image data (12 and 15 bits per sample). We also point out that, unless the All Intra configuration is used, H.265/HEVC cannot not offer random access functionality for partial and progressive decoding of images, like JPEG 2000 inherently does. Moreover, the time-complexity experiments indicate that H.265/HEVC demands a significantly higher computational cost compared to the JPEG-2000 based coding techniques.

And lastly, given the fact that JP3D uses only core technology of JPEG 2000 Part 1, we strongly recommend it as the compression standard to use on volumetric medical images. The JP3D standard does not suffer from ambiguity problems concerning the signaling of slice versus component data, it is not more complex than JPEG 2000 Part 1 and it offers significant improved compression performances through use of both volumetric wavelets and code-blocks. Compared to JPEG 2000 Part 2 MCT, JP3D offers very similar compression performance, but it also offers volumetric support for both code-blocks and region-of-interest functionality. Even at very low bit-rates where JPEG 2000 2D fails entirely regarding visual quality, the volumetric counterpart is able to deliver images at a visual quality that can still be considered useful.

Acknowledgments

This research was supported by the Research Foundation Flanders (FWO) with projects No. G014610N and

G017712N, and the iMinds projects ICA4DT, eHip, DMO-BISA and MMIQQA.

References

[1] H.K. Huang, PACS and Imaging Informatics: Basic Principles and Applications, Wiley, USA, 2010 (ISBN: 978-0-470-37372-9).

[2] L.W. Goldman, Principles of CT: multislice CT, J. Nucl. Med. Technol. 36 (2008).

[3] C.H. Riedel, J. Zoubie, S. Ulmer, J. Gierthmuehlen, O. Jansen, Thin-Slice reconstructions of nonenhanced CT images allow for detection of thrombus in acute stroke, Stroke: J. cereb. circul. 43 (2012) 2319–2323.

[4] W.B. Pennebaker, J.L. Mitchell, JPEG Still Image Data Compression Standard, 1st edition, Kluwer Academic Publishers, 1992.

[5] ISO/IEC 10918-1 | ITU-T Rec. T.81, Information Technology – Digital Compression and Coding of Continuous-tone Still Images, 1992.

[6] ISO/IEC 15444-1 | ITU-T Rec. T.800, Information Technology - JPEG 2000 Image Coding System: Core Coding System, 2002.

[7] R. Leung, D. Taubman, Transform and embedded coding techniques for maximum efficiency and random accessibility in 3-d scalable compression, IEEE Trans. Image Process. 14 (2005) 1632–1646.

[8] B. Huang, Satellite Data Compression, Springer, SpringerLink, Bücher, 2011.

[9] I. Blanes, J. Serra-Sagrista, M. Marcellin, J. Bartrina-Rapesta, Divide-and-conquer strategies for hyperspectral image processing: a review of their benefits and advantages, IEEE Signal Process. Mag. 29 (2012) 71–81.

[10] X. Wu, N. Memon, Context-based, adaptive, lossless image coding, IEEE Trans. Commun. 45 (1997) 437–444.

[11] M.J. Weinberger, G. Seroussi, G. Sapiro, The LOCO-I lossless image compression algorithm: principles and standardization into JPEG-LS, IEEE Trans. Image Process. 9 (2000) 1309–1324.

[12] ISO/IEC 14495-1 | ITU-T Rec. T.87, Information Technology – Lossless and Near-lossless Compression of Continuous-tone Still Images: Baseline, 1998.

[13] T. Wiegand, G.J. Sullivan, G. Bjntegaard, A. Luthra, Overview of the h.264/avc video coding standard, IEEE Trans. Circuits Syst. Video Techn. 13 (2003) 560–576.

[14] G.J. Sullivan, J.-R. Ohm, W. Han, T. Wiegand, Overview of the high efficiency video coding (HEVC) standard, IEEE Trans. Circuits Syst. Video Technol. 22 (2012) 1649–1668.

[15] J.-R. Ohm, G.J. Sullivan, H. Schwarz, T.K. Tan, T. Wiegand, Comparison of the coding efficiency of video coding standards - including high efficiency video coding (HEVC), IEEE Trans. Circuits Syst. Video Technol. 22 (2012) 1669–1684.

[16] H. Yu, Z. Lin, F. Pan, Applications and improvement of H.264 in medical video compression, IEEE Trans. Circuits Syst. I: Regul. Pap. 52 (2005) 2707–2716.

[17] V. Sanchez, P. Nasiopoulos, R. Abugharbieh, Lossless compression of 4D medical images using H.264/AVC, in: IEEE International Conference on Acoustics, Speech and Signal Processing, vol. 2, 2006, p. 2.

[18] V. Sanchez, P. Nasiopoulos, R. Abugharbieh, Efficient lossless compression of 4-d medical images based on the advanced video coding scheme, IEEE Trans. Inf. Technol. Biomed. 12 (2008) 442–446.

[19] V. Sanchez, J. Bartrina-Rapesta, Lossless compression of medical images based on HEVC intra coding., in: IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2014, Florence, Italy, May 4–9, 2014, 2014, pp. 6622–6626.

[20] V. Sanchez, F.A. Llinàs, J. Bartrina-Rapesta, J. Serra-Sagristà, Improvements to HEVC intra coding for lossless medical image compression, in: Data Compression Conference, DCC 2014, Snowbird, UT, USA, 26–28 March, 2014, p. 423.

[21] P. Schelkens, A. Munteanu, J. Barbarien, M. Galca, X.G. i Nieto, J. Cornelis, Wavelet coding of volumetric medical datasets, IEEE Trans. Med. Imaging 22 (2003) 441–458.

[22] DICOM, Digital Imaging and Communications in Medicine, URL: ⟨http://medical.nema.org/⟩, 2013.

[23] Z. Xiong, X. Wu, S. Cheng, J. Hua, Lossy-to-Lossless compression of medical volumetric data using three-dimensional integer wavelet transforms, IEEE Trans. Med. Imaging 22 (2003) 459–470.

[24] K. Krishnan, M.W. Marcellin, A. Bilgin, M.S. Nadar, Efficient transmission of compressed data for remote volume visualization, IEEE Trans. Med. Imaging 25 (2006) 1189–1199.

[25] T. Bruylants, A. Munteanu, A. Alecu, R. Deklerck, P. Schelkens, Volumetric Image Compression with JPEG 2000, SPIE Newsroom Biomedical Optics and Medical Imaging, URL: ⟨https://spie.org/x14507.xml⟩, 2007.

[26] D.S. Taubman, M.W. Marcellin, JPEG2000: Image Compression Fundamentals, Standards, and Practice, Kluwer Academic Publishers, Boston, 2002.

[27] P. Schelkens, A. Skodras, T. Ebrahimi, The JPEG 2000 Suite, Wiley Publishing, 2009.

[28] ISO/IEC 15444-2 | ITU-T Rec. T.801, Information Technology – JPEG 2000 Image Coding System: Extensions, 2002.

[29] ISO/IEC 15444-10 | ITU-T Rec. T.809, Information Technology – JPEG 2000 Image Coding System: Extensions for Three-dimensional Data, 2011.

[30] D. Taubman, EBCOT: embedded block coding with optimized truncation, IEEE Trans. Image Process. 9 (1998) 1158–1170.

[31] P. Schelkens, C. M. Brislawn, J. Barbarien, A. Munteanu, J.P. Cornelis, Jpeg2000 Part 10: Volumetric Imaging, vol. 5203, 2003, pp. 296–305, URL: ⟨http://dx.doi.org/10.1117/12.512540⟩.

[32] W. Ding, F. Wu, X. Wu, S. Li, H. Li, Adaptive directional lifting-based wavelet transform for image coding, IEEE Trans. Image Process. 16 (2007) 416–427.

[33] C.-L. Chang, B. Girod, Direction-Adaptive discrete wavelet transform for image compression, IEEE Trans. Image Process. 16 (2007) 1289–1302.

[34] A. Munteanu, O.M. Surdu, J. Cornelis, P. Schelkens, Segmentation-driven direction-adaptive discrete wavelet transform, in: IEEE International Conference in Image Processing, vol. 1, 2007, pp. 437–440.

[35] W. Dong, G. Shi, J. Xu, Adaptive nonseparable interpolation for image compression with directional wavelet transform, IEEE Signal Process. Lett. 15 (2008) 233–236.

[36] J.-R. Ohm, Three-dimensional subband coding with motion compensation, IEEE Trans. Image Process. 3 (1994) 559–571.

[37] S.-J. Choi, J.W. Woods, Motion-compensated 3-d subband coding of video, IEEE Trans. Image Process. 8 (1999) 155–167.

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

[39] H.S. Kim, H.W. Park, Wavelet-based moving-picture coding using shift-invariant motion estimation in wavelet domain, Signal Process.: Image Commun. 16 (2001) 669–679.

[40] Y. Andreopoulos, A. Munteanu, J. Barbarien, M.V. der Schaar, J. Cornelis, P. Schelkens, In-band motion compensated temporal filtering, Signal Process.: Image Commun. 19 (2004) 653–673. Special Issue on Subband/Wavelet Interframe Video Coding.

[41] ITU-T Rec. H.265, High Efficiency Video Coding, 2013.

[42] V. Sanchez, R. Abugharbieh, P. Nasiopoulos, Symmetry-Based scalable lossless compression of 3D medical image data, IEEE Trans. Med. Imaging 28 (2009) 1062–1072.

[43] P. Hao, Q. Shi, Reversible integer klt for progressive-to-lossless compression of multiple component images, in: Proceedings of the IEEE International Conference in Image Processing, 2003, pp. 633–636.

[44] A. Petrosian, F. Meyer, Wavelets in Signal and Image Analysis. From Theory to Practice, Kluwer Academic Publishers, 2001.

[45] R. Calderbank, I. Daubechies, W. Sweldens, B.-L. Yeo, Wavelet transforms that map integers to integers, Appl. Comput. Harmon. Anal. 5 (1998) 332–369.

[46] S.G. Mallat, A theory for multiresolution signal decomposition: the wavelet representation, IEEE Trans. Pattern Anal. Mach. Intell. 11 (1989) 674–693.

[47] A. Cohen, I. Daubechies, J.C. Feauveau, Biorthogonal bases of compactly supported wavelets, Commun. Pure Appl. Math. 45 (1992) 485–560.

[48] M. Unser, T. Blu, Mathematical properties of the JPEG2000 wavelet filters, IEEE Trans. Image Process. 12 (2003) 1080–1090.

[49] M.D. Adams, F. Kossentini, Reversible integer-to-integer wavelet transforms for image compression: performance evaluation and analysis, IEEE Trans. Image Process. 9 (2000) 1010–1024.

[50] C.E. Duchon, Lanczos filtering in one and two dimensions, J. Appl. Meteor. 18 (1979) 1016–1022.

[51] R. Deklerck, J. Cornelis, M. Bister, Segmentation of medical images, Image Vis. Comput. 11 (1993) 486–502.

[52] C. Christopoulos, W. Philips, A. Skodras, J. Cornelis, Segmented image coding: techniques and experimental results, Signal Process.: Image Commun. 11 (1997) 63–80.

[53] V. Christopoulos, Segmented Still Image and Video Coding (Ph.D. thesis), Vrije Universiteit Brussel, 1998.

[54] S. Golomb, Run-length encodings (corresp.), IEEE Trans. Inf. Theory 12 (1966) 399–401.

[55] I.H. Witten, R.M. Neal, J.G. Cleary, Arithmetic coding for data compression, Commun. ACM 30 (1987) 520–540.

[56] R. Stevens, A. Munteanu, J. Cornelis, P. Schelkens, Optimized directional lifting with reduced complexity, in: EURASIP European Signal Processing Conference, EUSIPCO, 2008.

[57] G. Bjontegaard, Calculation of average PSNR differences between RD-curves, ITU-VCEG, VCEG-M33, 2001.

[58] S. Pateux, J. Jung, An Excel Add-in for Computing Bjontegaardmetric and its Evolution, ITU-VCEG, VCEG-AE07, 2007.

[59] J. Munoz-Gomez, J. Bartrina-Rapesta, M. Marcellin, J. Serra-Sagrista, Correlation modeling for compression of computed tomography images, IEEE J. Biomed. Health Inf. 17 (2013) 928–935.

[60] A. Munteanu, J. Cornelis, G. Van Der Auwera, P. Cristea, Wavelet image compression – the quadtree coding approach, Trans. Infoem. Technol. Biomed. 3 (1999) 176–185.

[61] A. Munteanu, J. Cornelis, G.V.D. Auwera, P. Cristea, Wavelet-based lossless compression scheme with progressive transmission capability, Int. J. Imaging Syst. Technol. 10 (1999) 76–85.

[62] High Efficiency Video Coding (HEVC), URI: ⟨http://hevc.hhi.fraunhofer.de⟩, 2014.

[63] ITU-R Rec. BT.500-11, Methodology for the Subjective Assessment of the Quality of Television Pictures, 2002.

[64] T. Bruylants, J. Barbarien, A. Munteanu, P. Schelkens, Perceptual quality assessment of JPEG, JPEG 2000, and JPEG XR, in: SPIE Optics, Photonics, and Digital Technologies for Multimedia Applications, vol. 7723, 2010, pp. 77230E-12.