Multidim Syst Sign Process (2016) 27:87–104
DOI 10.1007/s11045-014-0291-6

An orthogonal 16-point approximate DCT for image and video compression

Thiago L. T. da Silveira · Fábio M. Bayer · Renato J. Cintra · Sunera Kulasekera · Arjuna Madanayake · Alice J. Kozakevicius

Received: 2 February 2014 / Revised: 22 May 2014 / Accepted: 18 July 2014 /
Published online: 20 August 2014

Abstract A low-complexity orthogonal multiplierless approximation for the 16-point discrete cosine transform (DCT) was introduced. The proposed method was designed to possess a very low computational cost. A fast algorithm based on matrix factorization was proposed requiring only 60 additions. The proposed architecture outperforms classical and state-of-the-art algorithms when assessed as a tool for image and video compression. Digital VLSI hardware implementations were also proposed being physically realized in field programmable gate array technology and implemented in 45 nm up to synthesis and place-route levels. Additionally, the proposed method was embedded into a high efficiency video coding (HEVC) reference software for actual proof-of-concept. Obtained results show negligible video degradation when compared to Chen DCT algorithm in HEVC.

T. L. T. da Silveira
Programa de Pós-Graduação em Informática, Universidade Federal de Santa Maria,
Santa Maria, RS, Brazil
e-mail: thiago@inf.ufsm.br

F. M. Bayer
Departamento de Estatística and LACESM, Universidade Federal de Santa Maria,
Santa Maria, RS, Brazil
e-mail: bayer@ufsm.br

R. J. Cintra (✉)
Signal Processing Group, Departamento de Estatística, Universidade Federal de Pernambuco,
Recife, PE, Brazil
e-mail: rjdsc@stat.ufpe.org

S. Kulasekera · A. Madanayake
Department of Electrical and Computer Engineering, University of Akron,
Akron, OH, USA
e-mail: arjuna@uakron.edu

A. J. Kozakevicius
Departamento de Matemática, Universidade Federal de Santa Maria,
Santa Maria, RS, Brazil
e-mail: alicek@ufsm.br

Keywords 16-Point DCT approximation · Low-complexity transforms · Image compression · Video coding

1 Introduction

The discrete cosine transform (DCT) (Britanak et al. 2007; Rao and Yip 1990) is a pivotal tool for digital image processing (Bouguezel et al. 2010, 2011; Lin et al. 2006). Indeed, the DCT is an important approximation for the optimal Karhunen–Loève transform (KLT), being employed in a multitude of compression standards due to its remarkable energy compaction properties (Bouguezel et al. 2008, 2011; Chang et al. 2000; Cintra and Bayer 2011; Potluri et al. 2014). Because of this, the DCT has found at applications in image and video coding standards, such as JPEG (Pennebaker and Mitchell 1992), MPEG-1 (Roma and Sousa 2007), MPEG-2 (International Organisation for Standardisation 1994), H.261 (International Telecommunication Union 1990), H.263 (International Telecommunication Union 1995), and H.264 (Joint Video Team 2003). Moreover, numerous fast algorithms were proposed for its computation (Arai et al. 1988; Feig and Winograd 1992; Hou 1987; Lee 1984; Loeffler et al. 1989; Vetterli and Nussbaumer 1984; Wang 1984).

Designing fast algorithms for the DCT is a mature area of research (Edirisuriya et al. 2012; Heideman and Burrus 1988; Liang and Tran 2001; Loeffler et al. 1989); thus it is not realistic to expect major advances by means of standards techniques. On the other hand, the development of low-complexity approximations for DCT is an open field of research. In particular, the 8-point DCT was given several approximations, such as the signed DCT (Haweel 2001), the level 1 DCT approximation (Lengwehasatit and Ortega 2004), the Bouguezel–Ahmad–Swamy (BAS) series of transforms (Bouguezel et al. 2008, 2009, 2010, 2011, 2013), the rounded DCT (RDCT) (Cintra and Bayer 2011), the modified RDCT (Bayer and Cintra 2012), the multiplier-free DCT approximation for RF imaging (Potluri et al. 2012), and the improved approximate DCT proposed in Potluri et al. (2014). Such approximations reduce the computational demands of the DCT evaluation, leading to low-power consumption and high-speed hardware realizations (Potluri et al. 2014). At the same time, approximate transforms can provide adequate numerical accuracy for image and video processing.

In response to the growing need for higher compression rates related to real time applications (International Telecommunication Union 2013), the high efficiency video coding (HEVC) video compression format (Pourazad et al. 2012) was proposed. Different from its predecessors, the HEVC employs not only 8×8 size blocks, but also 4×4, 16×16, and 32×32. Several approximations for 16-point DCT based on the integer cosine transform (Cham 1989) were proposed in Cham and Chan (1991), Dong et al. (2009) and Fong and Cham (2012). These transformations are derived from the exact DCT after scaling the elements of the DCT matrix and approximating the resulting real-numbered entries to integers (Dong et al. 2009). Therefore, real multiplications can be completely eliminated, at the expense of a noticeable increase in both the additive complexity and the number of required bit-shifting operations (Britanak et al. 2007).

A more restricted class of DCT approximations prescribe transformation matrices with entries defined on the set $\mathcal{C} = \{0, \pm 1/2, \pm 1, \pm 2\}$. Because the elements of $\mathcal{C}$ imply almost null arithmetic complexity, resulting transformations defined over $\mathcal{C}$ have *very* low-complexity, requiring no multiplications and a reduced number of bit-shifting operations. In this context, methods providing 16-point low-cost orthogonal transforms include the Walsh–Hadamard transform (WHT) (Valova and Kosugi 2000; Yarlagadda and Hershey 1997), BAS-2010 (Bouguezel et al. 2010), BAS-2013 (Bouguezel et al. 2013), and the approximate transform proposed in Bayer et al. (2012), here referred to as BCEM approximation. To

the best of our knowledge, these are the only 16-point DCT approximations defined over $\mathcal{C}$ archived in literature. Approximations over $\mathcal{C}$ are adequate the HEVC structure (Potluri et al. 2014) and are capable of minimizing the associated hardware power consumption as required by current multimedia market (International Telecommunication Union 2013).

The aim of this paper is to contribute to image and video compression methods related to JPEG-like schemes and HEVC. Thus, we propose a 16-point approximate DCT, that requires neither multiplications nor bit-shifting operations. Additionally, a fast algorithm is sought, aiming to minimize the overall computation complexity. The proposed transform is assessed and compared with competing 16-point DCT approximations. The realization of the propose DCT approximation in digital VLSI hardware as well as into a HEVC reference software is sought.

This paper unfolds as follows. In Sect. 2, we propose a new 16-point DCT approximation and detail its fast algorithm. Section 3 presents the performance analysis of the introduced transformations and compare it to competing tools in terms of the computational complexity coding measures, and similarity metrics with respect to the exact DCT. In Sect. 4, a JPEG-like image compression simulation is described and results are presented. In Sect. 5, digital hardware architectures for the proposed algorithm are supplied for both 1-D and 2-D analysis. A practical real-time video coding scenario is also considered: the proposed method is embedded into an open source HEVC standard reference software. Conclusions and final remarks are given in the last section.

2 Proposed transform

Several fast algorithm for the DCT allow recursive structures, for which the computation of the N-point DCT can be split into the computation of $\frac{N}{2}$-point DCT (Blahut 2010; Britanak et al. 2007; Chen et al. 1977; Loeffler et al. 1989; Rao and Yip 1990; Yip and Rao 1988). This is usually the case for algorithms based on decimation-in-frequency methods (Blahut 2010; Heideman and Burrus 1988).

In account of the above observation and judiciously considering permutations and signal changes, we designed a 16-point transformation that splits itself into two instantiations of the low-complexity matrix associated to the 8-point RDCT (Cintra and Bayer 2011). The proposed transformation, denoted as $\mathbf{T}$, is given by:

$$\mathbf{T} = \begin{bmatrix}
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\
1 & 1 & 1 & 0 & 0 & -1 & -1 & -1 & -1 & -1 & -1 & 0 & 0 & 1 & 1 & 1 \\
1 & 1 & 0 & 0 & 0 & 0 & -1 & -1 & 1 & 1 & 0 & 0 & 0 & 0 & -1 & -1 \\
1 & 0 & 0 & -1 & -1 & 0 & 0 & 1 & 1 & 0 & 0 & -1 & -1 & 0 & 0 & 1 \\
1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 & 1 & 1 & -1 & -1 \\
1 & 0 & -1 & -1 & 1 & 1 & 0 & -1 & -1 & 0 & 1 & 1 & -1 & -1 & 0 & 1 \\
0 & 0 & -1 & 1 & 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 & -1 & 1 & 0 & 0 \\
1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\
1 & -1 & -1 & 1 & 0 & 0 & 1 & -1 & 1 & -1 & 0 & 0 & -1 & 1 & 1 & -1 \\
1 & -1 & 0 & 1 & -1 & 0 & 1 & -1 & -1 & 1 & 0 & -1 & 1 & 0 & -1 & 1 \\
0 & 0 & 1 & 1 & -1 & -1 & 0 & 0 & 0 & 0 & 1 & 1 & -1 & -1 & 0 & 0 \\
0 & -1 & 1 & 0 & 0 & 1 & -1 & 0 & 0 & -1 & 1 & 0 & 0 & 1 & -1 & 0 \\
1 & -1 & 1 & -1 & 1 & -1 & 0 & 0 & 0 & 0 & 1 & -1 & 1 & -1 & 1 & -1 \\
0 & -1 & 1 & -1 & 1 & -1 & 1 & 0 & 0 & 1 & -1 & 1 & -1 & 1 & -1 & 0 \\
1 & -1 & 0 & 0 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & 0 & 0 & 1 & -1
\end{bmatrix}.$$

Because the entries of $\mathbf{T}$ are in $\{0, \pm 1\} \subset \mathcal{C}$, the proposed matrix is a multiplierless operator. Bit-shifting operations are also unnecessary; only simple additions are required. Additionally, the above matrix obeys the condition: $\mathbf{T} \cdot \mathbf{T}^{\top} = [\text{diagonal matrix}]$, where superscript $\top$ denotes matrix transposition. Thus, the necessary conditions for orthogonalizing it according to the methods described in Cintra (2011), Cintra and Bayer (2011) and Cintra et al. (2014) are satisfied. Such procedure yields the following orthogonal 16-point DCT approximation matrix:

$$\hat{\mathbf{C}} = \mathbf{S} \cdot \mathbf{T},$$

where

$$\mathbf{S} = \text{diag}\left(\frac{1}{4}, \frac{1}{4}, \frac{1}{\sqrt{12}}, \frac{1}{\sqrt{8}}, \frac{1}{\sqrt{8}}, \frac{1}{4}, \frac{1}{\sqrt{12}}, \frac{1}{\sqrt{12}}, \frac{1}{4}, \frac{1}{\sqrt{12}}, \frac{1}{\sqrt{12}}, \frac{1}{\sqrt{8}}, \frac{1}{\sqrt{8}}, \frac{1}{\sqrt{12}}, \frac{1}{\sqrt{12}}, \frac{1}{\sqrt{12}}\right).$$

In the context of image and video compression, the diagonal matrix $\mathbf{S}$ can be absorbed into the quantization step (Bouguezel et al. 2008, 2010, 2011; Cintra and Bayer 2011; Cintra et al. 2014; Edirisuriya et al. 2012; Tran 2000). Therefore, under these conditions, the complexity of the approximation $\hat{\mathbf{C}}$ can be equated to the complexity of the low-complexity matrix $\mathbf{T}$ (Bayer et al. 2012; Tran 2000).

Matrix-based fast algorithm design techniques yield a sparse matrix factorization of $\mathbf{T}$ as given below:

$$\mathbf{T} = \mathbf{P_2} \cdot \mathbf{M_4} \cdot \mathbf{M_3} \cdot \mathbf{M_2} \cdot \mathbf{P_1} \cdot \mathbf{M_1},$$

where

$$\mathbf{M}_1 = \begin{bmatrix} \mathbf{I}_8 & \bar{\mathbf{I}}_8 \\ \bar{\mathbf{I}}_8 & -\mathbf{I}_8 \end{bmatrix}, \quad \mathbf{P}_1 = \text{diag}\left(\mathbf{I}_9, \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}\right),$$

$$\mathbf{M}_2 = \text{diag}\left(\begin{bmatrix} \mathbf{I}_4 & \bar{\mathbf{I}}_4 \\ \bar{\mathbf{I}}_4 & -\mathbf{I}_4 \end{bmatrix}, \begin{bmatrix} \mathbf{I}_4 & \bar{\mathbf{I}}_4 \\ \bar{\mathbf{I}}_4 & -\mathbf{I}_4 \end{bmatrix}\right),$$

$$\mathbf{M}_3 = \text{diag}\left(\begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & -1 & 0 \\ 0 & -1 & 1 & 0 \\ 1 & 0 & 0 & -1 \end{bmatrix}, \begin{bmatrix} 0 & 1 & 1 & -1 \\ -1 & -1 & 0 & 1 \\ -1 & 1 & -1 & 0 \\ 1 & 0 & -1 & 1 \end{bmatrix}, \begin{bmatrix} 1 & 0 & -0 & -1 \\ 0 & 1 & 1 & 0 \\ 0 & -1 & 1 & 0 \\ -1 & 0 & 0 & 1 \end{bmatrix}, \begin{bmatrix} 0 & -1 & -1 & -1 \\ 1 & 1 & 0 & -1 \\ 1 & -1 & 1 & 0 \\ 1 & 0 & -1 & 1 \end{bmatrix}\right),$$

$$\mathbf{M}_4 = \text{diag}\left(\begin{bmatrix} 1 & -1 \\ 1 & -1 \end{bmatrix}, \mathbf{I}_6, \begin{bmatrix} 1 & -1 \\ 1 & -1 \end{bmatrix}, \mathbf{I}_6\right) \tag{1}$$

Table 1 Arithmetic complexity assessment

Transform	Operation count			
	Multiplication	Addition	Bit-shifting	Total
Chen DCT (Chen et al. 1977)	44	74	0	118
WHT	0	64	0	64
BAS-2010	0	64	8	72
BAS-2013	0	64	0	64
BCEM	0	72	0	72
Proposed	0	**60**	0	**60**

and matrix $\mathbf{P_2}$ performs the simple permutation (0)(1 8)(2 4 3 11 10 7 12 2)(5 9 13 14 6 5)(15) in cyclic notation (Herstein 1975, p. 77). Matrices $\mathbf{I}_n$ and $\bar{\mathbf{I}}_n$ denote the identity and counter-identity matrices of order n, respectively.

3 Computational complexity and evaluation

In this section, we aim at (1) assessing the computational complexity of the proposed approximation, (2) evaluating it in terms of approximation error, and (3) measuring its coding performance (Britanak et al. 2007). For comparison purposes, we selected the following state-of-the-art 16-point DCT approximations: BAS-2010 (Bouguezel et al. 2010), BAS-2013 (Bouguezel et al. 2013) and the BCEM method (Bayer et al. 2012). We also considered the classical WHT (Yarlagadda and Hershey 1997) and the exact DCT as computed according to the Chen DCT algorithm (Chen et al. 1977). This latter method is the algorithm employed in the HEVC codec by Capelo (2011).

3.1 Arithmetic complexity

The computational cost of a given transformation is traditionally measured by its arithmetic complexity, i.e, the number of required arithmetic operations for its computation (Blahut 2010; Britanak et al. 2007; Proakis and Manolakis 2006). Considered operations are multiplications, additions, and bit-shifting operations (Blahut 2010). Table 1 lists the operation count for each arithmetical operation for all considered methods. Total operation count is also provided. Values in bold are the correspond to the best cases.

The proposed transform showed 6.25 % less total operation count when compared with the WHT or BAS-2013 approximation. Considering BAS-2010 or the BCEM approximation, the introduced approximation required 16.67 % less operation overall. As a more strict complexity assessment, even if we take only the additive complexity into account, the proposed transformation can still outperform all considered methods. It is also noteworthy that the proposed method has the lowest multiplicative complexity among all considered methods. Moreover, to the best of our knowledge, the proposed transformation outperforms any meaningful 16-point DCT approximation archived in literature.

3.2 Similarity measures

For the approximation error analysis, we considered three tools: the DCT distortion (Wien and Sun 2001), the total error energy (Cintra and Bayer 2011), and the mean square error (MSE)

(Britanak et al. 2007; Rao and Yip 1990). This set of measures determines the similarity between the exact DCT matrix and a given approximation. These quality metrics are briefly described as follows.

Let $\mathbf{C}$ be the exact N-point DCT matrix and $\tilde{\mathbf{C}}$ be a given N-point DCT approximation. Adopting the notation employed in Fong and Cham (2012), the DCT distortion of $\tilde{\mathbf{C}}$ is given by:

$$d_2(\tilde{\mathbf{C}}) = 1 - \frac{1}{N} \cdot \left\| \operatorname{diag}\left(\mathbf{C} \cdot \tilde{\mathbf{C}}^\top\right) \right\|^2,$$

where $\|\cdot\|$ is the Euclidean norm (Watkins 2004). The DCT distortion captures the difference between the exact DCT matrix and a candidate approximation by quantifying the orthogonality among the basis vectors of both transforms (Fong and Cham 2012).

Taking the basis vectors of the exact DCT and a given approximation as filter coefficients, the total error energy (Cintra and Bayer 2011), measures the spectral proximity between the corresponding transfer functions (Haweel 2001). Invoking Parseval theorem (Oppenheim 2006), the total error energy can be evaluated according to:

$$\epsilon(\tilde{\mathbf{C}}) = \pi \cdot \left\| \mathbf{C} - \tilde{\mathbf{C}} \right\|_{\mathrm{F}}^2,$$

where $\|\cdot\|_{\mathrm{F}}$ is the Frobenius norm (Watkins 2004).

The MSE is a well-established proximity measure (Britanak et al. 2007). The MSE between $\mathbf{C}$ and $\tilde{\mathbf{C}}$ is given by (Britanak et al. 2007; Liang and Tran 2001):

$$\mathrm{MSE}(\tilde{\mathbf{C}}) = \frac{1}{N} \cdot \mathrm{tr}\left((\mathbf{C} - \tilde{\mathbf{C}}) \cdot \mathbf{R} \cdot (\mathbf{C} - \tilde{\mathbf{C}})^\top\right),$$

where $\mathrm{tr}(\cdot)$ is the trace function (Seber 2007) and $\mathbf{R}$ is the covariance matrix of the input signal. Assuming the first-order stationary Markov process model for the input data, we have that $\mathbf{R}_{[i,j]} = \rho^{|i-j|}$, for $i, j = 1, 2, \ldots, N$, and the correlation coefficient ρ is set to 0.95 (Britanak et al. 2007; Liang and Tran 2001). This particular model is suitable for real signals and natural images (Britanak et al. 2007; Cintra et al. 2014; Liang and Tran 2001). The minimization of MSE values indicates proximity to the exact DCT (Britanak et al. 2007).

3.3 Coding measures

We adopted two coding measures: the transform coding gain (Fong and Cham 2012) and the transform efficiency (Britanak et al. 2007). The transform coding gain quantifies the coding or data compression performance of an orthogonal transform (Britanak et al. 2007; Fong and Cham 2012). This measure is given by (Britanak et al. 2007; Malvar 1992):

$$C_g(\tilde{\mathbf{C}}) = 10 \cdot \log_{10}\left\{ \frac{\frac{1}{N}\sum_{i=0}^{N-1} s_{ii}}{\left[\prod_{i=0}^{N-1}\left(s_{ii} \cdot \sqrt{\sum_{j=0}^{N-1} \tilde{\mathbf{c}}_{ij}^2}\right)\right]^{\frac{1}{N}}} \right\},$$

where s_{ij} and $\tilde{\mathbf{c}}_{ij}$ are the (i, j)-th entry of $\tilde{\mathbf{C}} \cdot \mathbf{R} \cdot \tilde{\mathbf{C}}^\top$ and $\tilde{\mathbf{C}}$, respectively.

On the other hand, the transform efficiency (Britanak et al. 2007) is an alternative method to compute the compression performance. Denoted by η, the transform efficiency is furnished by:

$$\eta(\tilde{\mathbf{C}}) = \frac{\sum_{i=0}^{N-1} |s_{ii}|}{\sum_{i=0}^{N-1}\sum_{j=0}^{N-1} |s_{ij}|} \times 100.$$

Table 2 Performance analysis

Transform	Measures				
	Approximation			Coding	
	d_2	ϵ	MSE	C_g	η
DCT	0	0	0	9.4555	88.4518
WHT	0.8783	92.5631	0.4284	8.1941	70.6465
BAS-2010	0.6666	64.749	0.1866	**8.5208**	**73.6345**
BAS-2013	0.5108	54.6207	0.132	8.1941	70.6465
BCEM	**0.1519**	**8.0806**	**0.0465**	7.8401	65.2789
Proposed	**0.3405**	**30.323**	**0.0639**	**8.295**	**70.8315**

Quantity $\eta(\tilde{\mathbf{C}})$ indicates the data decorrelation capability of the transformation. The KLT achieves optimality with respect to this measure, presenting a transform efficiency of 100 (Britanak et al. 2007).

3.4 Results

Table 2 summarizes the results for the above detailed similarity and coding measures. For each figure of merit, we emphasize in bold the two best measurements. The proposed transform displays consistently good performance according to all considered criteria. This fact contrasts with existing transformations, which tend to excel in terms of similarity measures, but perform limitedly in terms of coding performance; and vice-versa. Therefore, the proposed transform offers a compromise, while still achieving state-of-the-art performance.

4 Application to image compression

The proposed approximation was submitted to the image compression simulation methodology originally introduced in Haweel (2001) and employed in Bayer et al. (2012); Bouguezel et al. (2008, 2010, 2011, 2013) and Cintra and Bayer (2011). In our experiments a set of 45 512 × 512 8-bit grayscale images obtained from a standard public image bank (USC-SIPI 2011) was considered to validate the proposed algorithm. We adapted the JPEG-like compression scheme for the 16 × 16 matrix case, as suggested in Bouguezel et al. (2010).

The adopted image compression method is detailed as follows. An input 512 × 512 image was divided into 16 × 16 disjoint blocks $\mathbf{A}_k$, $k = 0, 1, \ldots, 31$. Each block $\mathbf{A}_k$ was 2-D transformed according to $\mathbf{B}_k = \tilde{\mathbf{C}} \cdot \mathbf{A}_k \cdot \tilde{\mathbf{C}}^\top$, where $\mathbf{B}_k$ is a frequency domain image block and $\tilde{\mathbf{C}}$ is a given approximation matrix. Matrix $\mathbf{B}_k$ contains the 256 transform domain coefficients for each block. Adapting the zigzag sequence (Pao and Sun 1998) for the 16 × 16 case, we retained only the r initial coefficients and set the remaining coefficients to zero (Bayer et al. 2012; Bouguezel et al. 2008, 2010, 2011, 2013; Cintra and Bayer 2011; Haweel 2001), generating $\mathbf{B}'_k$. Subsequently, the inverse transformation was applied to $\mathbf{B}'_k$ according to: $\mathbf{A}'_k = \tilde{\mathbf{C}}^\top \cdot \mathbf{B}'_k \cdot \tilde{\mathbf{C}}$. The above procedure was repeated for each block. The rearrangement of all blocks $\mathbf{A}'_k$ reconstructs the image, which can be assessed for quality.

Image degradation was evaluated using two different quality measures: (1) the peak signal-to-noise ratio (PSNR) and (2) the structural similarity index (SSIM) (Wang et al. 2004)—a generalization of the universal image quality index (Wang and Bovik 2002). In

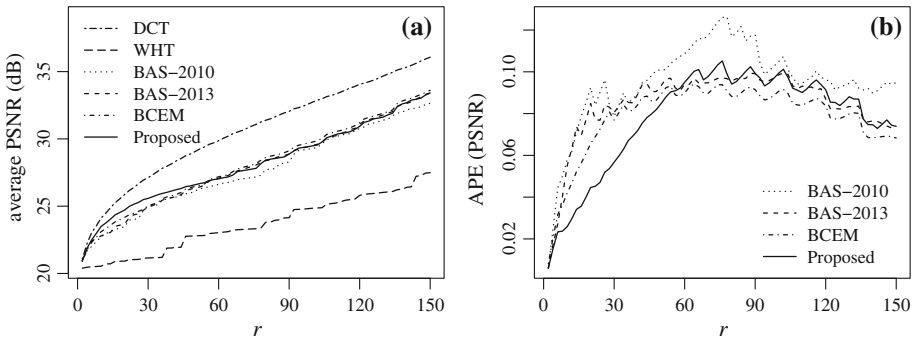

Fig. 1 PSNR results for all considered transforms under several compression rates. **a** Average PSNR, **b** absolute percentage error of PSNR

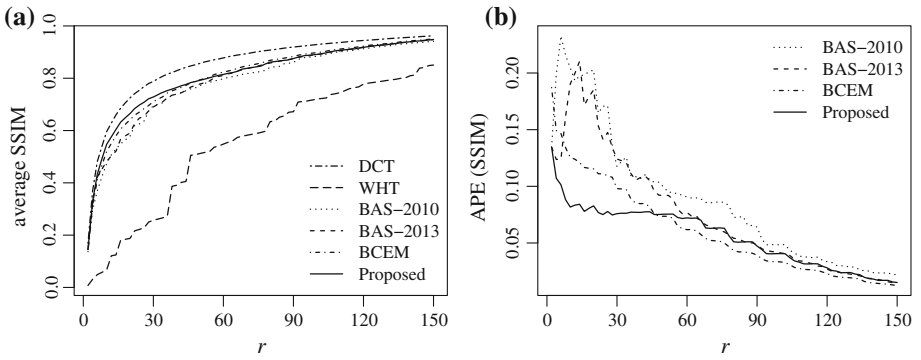

Fig. 2 SSIM results for all considered transforms under several compression rates. **a** Average SSIM, **b** absolute percentage error of SSIM

contrast to the PSNR, SSIM definition takes advantage of known characteristics of the human visual system (Wang et al. 2004). Following the methodology adopted in Cintra and Bayer (2011) and Bayer et al. (2012), we calculated average PSNR and SSIM values for all 45 images.

Figures 1a and 2a show average PSNR and SSIM measurements, respectively. Additionally, we considered absolute percentage error (APE) measurements of PSNR and SSIM with respect to the exact DCT. Results are displayed in Figs. 1b and 2b, for PNSR and SSIM, respectively. APE figures for the WHT are absent because their values were exceedingly high, being located outside of the plot range.

According to Figs. 1 and 2, the proposed transform outperforms other methods for $r \leq 50$, which correspond to high-compression rates. Therefore, the proposed transform is in consonance with ITU recommendation for high-compression coding in real time applications (International Telecommunication Union 2013). For $r > 50$, discussed methods are essentially comparable in terms of image degradation.

As a qualitative comparison, Fig. 3 shows the compressed *Lena* image at $r = 16$ (93.75 % compression) obtained from each considered method. The proposed transform offered less pixelation and block artifacts; demonstrating its adequacy for high-compression rate scenarios.

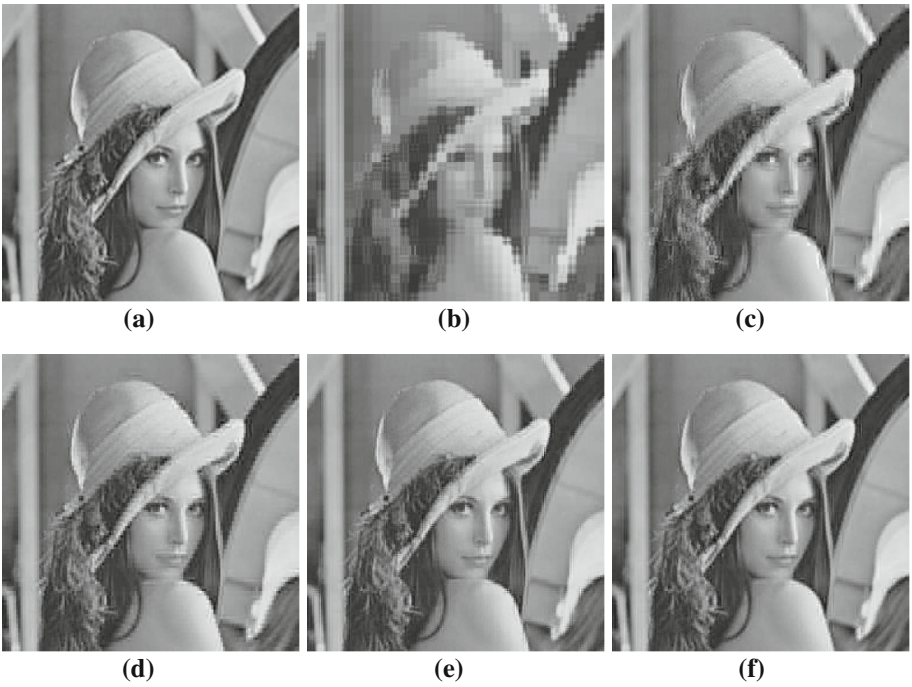

Fig. 3 Compressed *Lena* image using all considered transforms, for $r = 16$. **a** DCT (PSNR = 28.55), **b** WHT (PSNR = 21.20), **c** BAS-2010 (PSNR = 25.27), **d** BAS-2013 (PSNR = 25.79), **e** BCEM (PSNR = 25.75), **f** Proposed (PSNR = 27.13)

5 Digital architecture and realization

In this section, hardware architectures for the proposed 16-point approximate DCT are detailed. Both 1-D and 2-D transformations are addressed. Introduced architectures were submitted to (1) Xilinx field programmable gate array (FPGA) implementations and (2) CMOS 45 nm application specific integrated circuit (ASIC) implementation up to the synthesis level. Additionally, in order to assess the performance of the proposed algorithm in real time video coding, the introduced approximation was also embedded into an HEVC reference software (Joint Collaborative Team on Video Coding (JCT-VC) 2013).

5.1 Architecture for the 16-point DCT approximation

The 2-D version of the 16-point DCT approximation architecture was realized using two 1-D transforms and a transpose buffer. This is possible because the proposed approximation inherits the separable kernel property of the exact DCT (Smith 1997). The first instantiation of the approximate DCT block furnishes a row-wise transform computation of the input image, while the second implementation furnishes a column-wise transformation of the previous intermediate result. A real time row-parallel transposition buffer circuit is required in between the 1-D transformation blocks. Such block ensures data ordering for converting the row-transformed data from the first DCT approximation circuit to a transposed format as required by the second DCT approximation circuit. Both 1-D transformation blocks and the transposition buffer were initially modeled and tested in Matlab Simulink; then they were

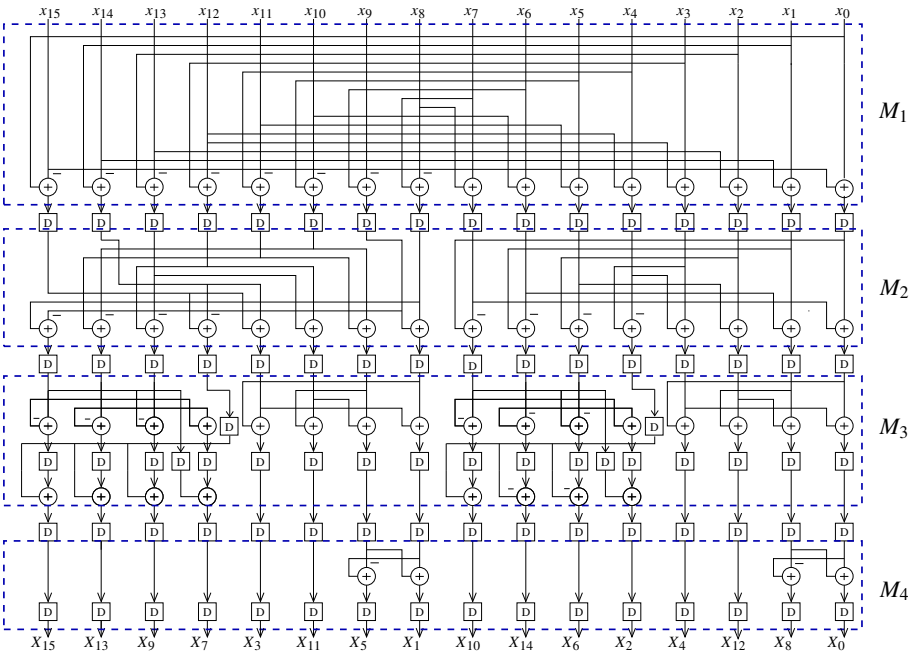

Fig. 4 Architecture of the proposed 16-point DCT approximation

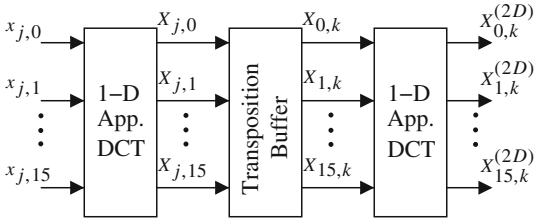

Fig. 5 Two-dimensional approximate transform by means of 1-D approximate transform. Signal $x_{k,0}, x_{k,1}, \ldots$, corresponds to the *rows* of the input image; $X_{k,0}, X_{k,1}, \ldots$ indicates the transformed rows; $X_{0,j}, X_{1,j}, \ldots$ indicates the *columns* of the transposed *row-wise* transformed image; and $X^{(2-D)}_{0,j}, X^{(2-D)}_{1,j}, \ldots$ indicates the *columns* of the final 2-D transformed image

combined to furnish the complete 2-D approximate transform. Figure 4 depicts the architecture for the proposed 1-D approximate DCT. We emphasize in dashed boxes the blocks M_1, M_2, M_4, and M_4, which correspond to the realization of sparse matrices $\mathbf{M_1}$, $\mathbf{M_2}$, $\mathbf{M_3}$, and $\mathbf{M_4}$, respectively, as shown in the equation set (1). Figure 5 shows the implementation of the 2-D transform by means of the 1-D transforms.

5.2 FPGA and ASIC realizations and results

The above discussed architecture was physically realized on a FPGA based rapid prototyping system for various register sizes and tested using on-chip hardware-in-the-loop co-simulation. The architecture was designed for digital realization within the MATLAB environment using the Xilinx System Generator. Xilinx Virtex-6 XC6VLX240T-1FFG1156

Table 3 Hardware resource consumption and power consumption for the proposed 2-D 16-point DCT approximation

CLB	FF	T_{cpd} (ns)	$F_{\max}$ (MHz)	D_p (mW/MHz)	Q_p (W)
1,408	4,600	3.7	270.27	6.91	3.481

Table 4 Hardware resource consumption for CMOS 45 nm ASIC place-route implementation of the proposed 2-D 16-point DCT approximation

Area (mm^2)	AT	AT^2	T_{cpd} (ns)	$F_{\max}$ (MHz)	D_p (mW/MHz)	Q_p (mW)
0.585	4.896	40.98	8.37	119.47	0.311	216.2

device was employed to physically realize the architecture on FPGA with fine-grain pipelining for increased throughput. The realization was verified on FPGA chip using a Xilinx ML605 board at a clock frequency of 50 MHz. The FPGA realization was tested with 10,000 random 16-point input test vectors using hardware co-simulation. Test vectors were generated from within the MATLAB environment and routed to the physical FPGA device using JTAG based hardware co-simulation. Then measured data from the FPGA was routed back to MATLAB memory space.

Evaluation of hardware complexity and real time performance considered the following metrics: the number of used configurable logic blocks (CLB), flip-flop (FF) count, critical path delay (T_{cpd}), and the maximum operating frequency ($F_{\max}$) in MHz. The `xflow.results` report file was accessed to obtain the above results. Dynamic (D_p) and static power (Q_p) consumptions were estimated using the Xilinx XPower Analyzer. Results are shown in Table 3.

For the ASIC implementation, the hardware description language code was ported to 45 nm CMOS technology and subject to synthesis and place-and-route steps using the Cadence Encounter. The FreePDK, a free open-source ASIC standard cell library at the 45 nm node, was used for this purpose. The supply voltage of the CMOS realization was fixed at $V_{\text{DD}} = 1.1\text{V}$ during estimation of power consumption and logic delay. The adopted figures of merit for the ASIC synthesis were: area (A) in mm^2, area-time complexity (AT) in $\text{mm}^2 \cdot \text{ns}$, area-time-squared complexity (AT^2) in $\text{mm}^2 \cdot \text{ns}^2$, dynamic ($D_p$) power in (mW/MHz) and static (Q_p) power consumption in watts, critical path delay (T_{cpd}) in ns, and maximum operating frequency ($F_{\max}$) in MHz. Results are displayed in Table 4.

Among the considered competitors, the BAS-2010 (Bouguezel et al. 2010) showed arithmetic complexity and coding performance similar to the proposed transform. For comparison purposes the 1-D versions of the BAS-2010 approximation and the proposed 16-point approximation were realized on a Xilinx Virtex-6 XC6VLX240T-1FFG1156 device as well as were ported to 45 nm CMOS technology and subject to synthesis and place-and-route steps using the Cadence Encounter. The results are shown in Tables 5 and 6. Compared to the BAS-2010, the proposed transform is faster when both the FPGA implementation and CMOS synthesis is considered while having similar performance in hardware usage and dynamic power consumption. Importantly, the proposed is better in image quality as evidenced by Fig. 3.

5.3 Real time video compression software implementation

In order to assess real-time video coding performance, the proposed approximation was embedded into the open source HEVC standard reference software by the Fraunhofer Heinrich Hertz Institute (Joint Collaborative Team on Video Coding (JCT-VC) 2013). The original transform prescribed in the selected HEVC reference software is the scaled approximation

Table 5 Hardware resource consumption of the 1-D approximations using Xilinx Virtex-6 XC6VLX240T-1FFG1156 device

Transform	CLB	FF	T_{cpd} (ns)	F_{max} (MHz)	D_p (mW/MHz)	Q_p (W)
BAS-2010	430	1,440	1.950	512.82	4.54	3.49
Proposed	421	1,372	1.900	526.31	4.22	3.49

Table 6 Hardware resource consumption for CMOS 45nm ASIC place-route implementation of the 1-D approximations

Transform	Area (mm^2)	AT	AT^2	T_{cpd} (ns)	F_{max} (MHz)	D_p (mW/MHz)	Q_p (mW)
BAS-2010	0.169	0.843	4.21	4.994	200.24	0.093	70.47
Proposed	0.183	0.895	4.38	4.895	204.29	0.095	78.73

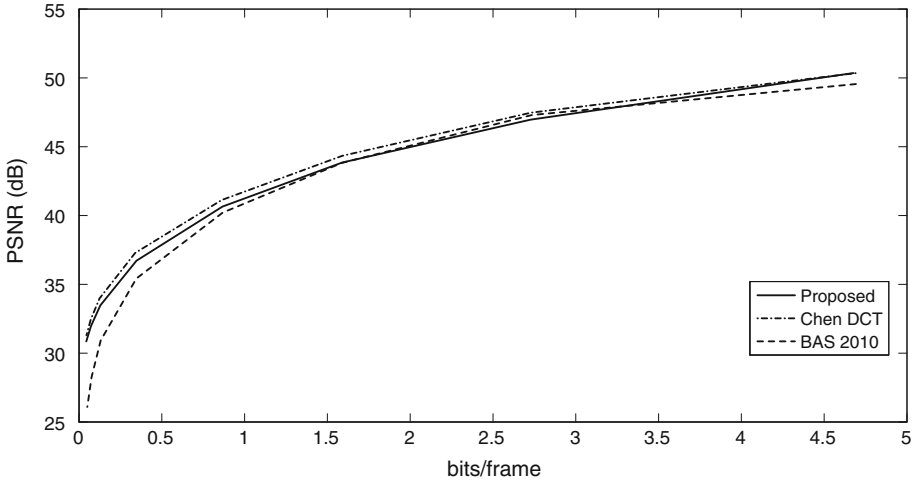

Fig. 6 RD *curves* for 'BasketballPass' test sequence

of Chen DCT algorithm (Capelo 2011; Chen et al. 1977) and the software can process image block sizes of 4×4, 8×8, 16×16, and 32×32.

Our methodology consists of replacing the 16×16 DCT algorithm of the reference software by the proposed 16-point approximate algorithm. Algorithms were evaluated for their effect on the overall performance of the encoding process. For such, we obtained rate-distortion (RD) curves for standard video sequences (Ortega and Ramchandran 1998). The quantization point (QP) varied from 0 to 50 to obtain the curves and the resulting PSNR values with the bit rate values measured in bits per frame were recorded for the proposed algorithm, the Chen DCT algorithm, and the BAS-2010 (Bouguezel et al. 2010) algorithm. Figure 6 depicts the obtained RD curves for the 'BasketballPass' test sequence. Figure 7 shows particular 416×240 frames for the test video sequence 'BasketballPass' with QP $\in \{0, 32, 50\}$.

The RD curves and selected frames reveal that the difference between the original HEVC and the implementation with the proposed approximation is negligible. In fact, in Fig. 6 the maximum PSNR difference is 0.56 dB, which is very low. Figure 7 shows that both encoded

Fig. 7 Selected frames from 'BasketballPass' test video coded by means of the Chen DCT and the proposed 16-point DCT approximation for QP = 0 (**a**–**b**), QP = 32 (**c**–**d**), and QP = 50 (**e**–**f**)

video streams are almost identical. These results confirm the adequacy of the proposed scheme.

6 Conclusion

This work proposed a new orthogonal 16-point DCT approximation. The introduced transform offers a very low computational cost, outperforming—to the best of our knowledge—all competing methods. Moreover, the proposed transform performed well as an image compression tool, specially at high compression rate scenarios. By means of (1) comprehensive computational simulations, (2) hardware implementation (both in FPGA and ASIC), and (3) software embedding, we demonstrated the adequacy and efficiency of the proposed method, which is suitable for codec schemes, like the HEVC. Additionally, the introduced transformation offers an unusual good performance balance among several metrics, as shown in

Table 2. This suggests that the applicability of proposed transform is not limited in scope to the image and video compression context.

Acknowledgments This work was partially supported by CPNq, FACEPE, FAPERGS and FIT/UFSM (Brazil), and by the College of Engineering at the University of Akron, Akron, OH, USA.

References

Arai, Y., Agui, T., & Nakajima, M. (1988). A fast DCT-SQ scheme for images. *Transactions of the IEICE*, *E–71*(11), 1095–1097.

Bayer, F. M., & Cintra, R. J. (2012). DCT-like transform for image compression requires 14 additions only. *Electronics Letters*, *48*(15), 919–921. doi:10.1049/el.2012.1148.

Bayer, F. M., Cintra, R. J., Edirisuriya, A., & Madanayake, A. (2012). A digital hardware fast algorithm and FPGA-based prototype for a novel 16-point approximate DCT for image compression applications. *Measurement Science and Technology*, *23*(8), 114010–114019.

Blahut, R. E. (2010). *Fast algorithms for signal processing*. Cambridge: Cambridge University Press.

Bouguezel, S., Ahmad, M. O., & Swamy, M. N. S. (2008). Low-complexity 8×8 transform for image compression. *Electronics Letters*, *44*(21), 1249–1250. doi:10.1049/el:20082239.

Bouguezel, S., Ahmad, M. O., Swamy, M. (2009). A fast 8×8 transform for image compression. In: *2009 International conference on microelectronics (ICM)*, pp. 74–77. doi:10.1109/ICM.2009.5418584.

Bouguezel, S., Ahmad, M. O., Swamy, M. N. S. (2010). A novel transform for image compression. In: *53rd IEEE international midwest symposium on circuits and systems (MWSCAS)*, pp. 509–512. doi:10.1109/MWSCAS.2010.5548745.

Bouguezel, S., Ahmad, M. O., Swamy, M. N. S. (2011). A low-complexity parametric transform for image compression. In: *Proceedings of the 2011 IEEE international symposium on circuits and systems*.

Bouguezel, S., Ahmad, M. O., & Swamy, M. N. S. (2013). Binary discrete cosine and Hartley transforms. *IEEE Transactions on Circuits and Systems I: Regular Papers*, *60*(4), 989–1002. doi:10.1109/TCSI.2012.2224751.

Britanak, V., Yip, P., & Rao, K. R. (2007). *Discrete cosine and sine transforms*. New York: Academic Press.

Capelo, M. (2011). *Advances on transforms for high efficiency video coding*. Master's thesis, Instituto Superior Técnico, Lisboa.

Cham, W. K. (1989). Development of integer cosine transforms by the principle of dyadic symmetry. *IEE Proceedings I Communications, Speech and Vision*, *136*, 276–282.

Cham, W. K., & Chan, Y. T. (1991). An order-16 integer cosine transform. *IEEE Transactions on Signal Processing*, *39*(5), 1205–1208. doi:10.1109/78.80974.

Chang, T. S., Kung, C. S., & Jen, C. W. (2000). A simple processor core design for DCT/IDCT. *IEEE Transactions on Circuits and Systems for Video Technology*, *10*(3), 439–447. doi:10.1109/76.836290.

Chen, W. H., Smith, C., & Fralick, S. (1977). A fast computational algorithm for the discrete cosine transform. *IEEE Transactions on Communications*, *25*(9), 1004–1009. doi:10.1109/TCOM.1977.1093941.

Cintra, R. J. (2011). An integer approximation method for discrete sinusoidal transforms. *Circuits, Systems, and Signal Processing*, *30*(6), 1481–1501. doi:10.1007/s00034-011-9318-5.

Cintra, R. J., & Bayer, F. M. (2011). A DCT approximation for image compression. *IEEE Signal Processing Letters*, *18*(10), 579–582. doi:10.1109/LSP.2011.2163394.

Cintra, R. J., Bayer, F. M., & Tablada, C. J. (2014). Low-complexity 8-point DCT approximations based on integer functions. *Signal Processing*, *99*, 201–214. doi:10.1016/j.sigpro.2013.12.027.

Dong, J., Ngan, K. N., Fong, C. K., & Cham, W. K. (2009). 2-D order-16 integer transforms for HD video coding. *IEEE Transactions on Circuits and Systems for Video Technology*, *19*(10), 1462–1474. doi:10.1109/TCSVT.2009.2026792.

Edirisuriya, A., Madanayake, A., Dimitrov, V., Cintra, R. J., & Adikari, J. (2012). VLSI architecture for 8-point AI-based Arai DCT having low area-time complexity and power at improved accuracy. *Journal of Low Power Electronics and Applications*, *2*(2), 127–142. doi:10.3390/jlpea2020127.

Feig, E., & Winograd, S. (1992). Fast algorithms for the discrete cosine transform. *IEEE Transactions on Signal Processing*, *40*(9), 2174–2193.

Fong, C. K., & Cham, W. K. (2012). LLM integer cosine transform and its fast algorithm. *IEEE Transactions on Circuits and Systems for Video Technology*, *22*(6), 844–854.

Haweel, T. I. (2001). A new square wave transform based on the DCT. *Signal Processing*, *82*, 2309–2319.

Heideman, M. T., & Burrus, C. S. (1988). *Multiplicative complexity, convolution, and the DFT. Signal processing and digital filtering*. Berlin: Springer.

Herstein, I. N. (1975). *Topics in algebra* (2nd ed.). London: Wiley.
Hou, H. S. (1987). A fast recursive algorithm for computing the discrete cosine transform. *IEEE Transactions on Acoustic, Signal, and Speech Processing*, *6*(10), 1455–1461.
International Organisation for Standardisation. (1994). *Generic coding of moving pictures and associated audio information—Part 2: Video*. ISO/IEC JTC1/SC29/WG11—Coding of moving pictures and audio, ISO.
International Telecommunication Union. (1990). *ITU-T recommendation H.261 version 1: Video codec for audiovisual services at* $p \times 64$ *kbits*. Tech. rep., ITU-T.
International Telecommunication Union. (1995). *ITU-T recommendation H.263 version 1: Video coding for low bit rate communication*. Tech. rep., ITU-T.
International Telecommunication Union. (2013). *Infrastructure of audiovisual services: Coding of moving video—High efficiency video coding*. Telecommunication Standarization Sector of ITU.
Joint Collaborative Team on Video Coding (JCT-VC). (2013). *HEVC references software documentation*. Fraunhofer Heinrich Hertz Institute. https://hevc.hhi.fraunhofer.de/.
Joint Video Team. (2003). *Recommendation H.264 and ISO/IEC 14 496-10 AVC: Draft ITU-T recommendation and final draft international standard of joint video specification*. Tech. rep., ITU-T.
Lee, B. G. (1984). A new algorithm for computing the discrete cosine transform. *IEEE Transactions on Acoustics, Speech and Signal Processing*, *ASSP–32*, 1243–1245.
Lengwehasatit, K., & Ortega, A. (2004). Scalable variable complexity approximate forward DCT. *IEEE Transactions on Circuits and Systems for Video Technology*, *14*(11), 1236–1248. doi:10.1109/TCSVT.2004.835151.
Liang, J., & Tran, T. D. (2001). Fast multiplierless approximation of the DCT with the lifting scheme. *IEEE Transactions on Signal Processing*, *49*, 3032–3044.
Lin, M. C., Dung, L. R., & Weng, P. K. (2006). An ultra-low-power image compressor for capsule endoscope. *BioMedical Engineering Online*, *5*(1), 1–8. doi:10.1186/1475-925X-5-14.
Loeffler, C., Ligtenberg, A., Moschytz, G. (1989), Practical fast 1D DCT algorithms with 11 multiplications. In: *Proceedings of the international conference on acoustics, speech, and signal processing*, pp. 988–991.
Malvar, H. S. (1992). *Signal processing with lapped transforms*. London: Artech House.
Oppenheim, A. V. (2006). *Discrete-time signal processing*. Upper Saddle River: Pearson Education.
Ortega, A., & Ramchandran, K. (1998). Rate-distortion methods for image and video compression. *IEEE Signal Processing Magazine*, *15*(6), 23–50. doi:10.1109/79.733495.
Pao, I. M., & Sun, M. T. (1998). Approximation of calculations for forward discrete cosine transform. *IEEE Transactions on Circuits and Systems for Video Technology*, *8*(3), 264–268. doi:10.1109/76.678620.
Pennebaker, W. B., & Mitchell, J. L. (1992). *JPEG still image data compression standard*. New York, NY: Van Nostrand Reinhold.
Potluri, U. S., Madanayake, A., Cintra, R. J., Bayer, F. M., & Rajapaksha, N. (2012). Multiplier-free DCT approximations for RF multi-beam digital aperture-array space imaging and directional sensing. *Measurement Science and Technology*, *23*(11), 114003. doi:10.1088/0957-0233/23/11/114003.
Potluri, U. S., Madanayake, A., Cintra, R. J., Bayer, F. M., Kulasekera, S., & Edirisuriya, A. (2014). Improved 8-point approximate DCT for image and video compression requiring only 14 additions. *IEEE Transactions on Circuits and Systems I*, *61*(6), 1727–1740. doi:10.1109/TCSI.2013.2295022.
Pourazad, M. T., Doutre, C., Azimi, M., & Nasiopoulos, P. (2012). HEVC: The new gold standard for video compression: How does HEVC compare with H.264/AVC? *IEEE Consumer Electronics Magazine*, *1*(3), 36–46. doi:10.1109/MCE.2012.2192754.
Proakis, J. G., & Manolakis, D. K. (2006). *Digital signal processing* (4th ed.). Englewood Cliffs: Prentice Hall.
Rao, K. R., & Yip, P. (1990). *Discrete cosine transform: Algorithms, advantages, applications*. San Diego, CA: Academic Press.
Roma, N., & Sousa, L. (2007). Efficient hybrid DCT-domain algorithm for video spatial downscaling. *EURASIP Journal on Advances in Signal Processing*, *2007*(2), 30. doi:10.1155/2007/57291.
Seber, G. A. F. (2007). *A matrix handbook for statisticians*. New York: Wiley-Interscience.
Smith, S. W. (1997). *The scientist and engineer's guide to digital signal processing*. San Diego, CA: California Technical Publishing.
Tran, T. D. (2000). The binDCT: Fast multiplierless approximation of the DCT. *IEEE Signal Processing Letters*, *6*(7), 141–144.
USC-SIPI. (2011). *The USC-SIPI image database*. University of Southern California, Signal and Image Processing Institute. http://sipi.usc.edu/database/.
Valova, I., & Kosugi, Y. (2000). Hadamard-based image decomposition and compression. *IEEE Transactions on Information Technology in Biomedicine*, *4*(4), 306–319.

Vetterli, M., & Nussbaumer, H. (1984). Simple FFT and DCT algorithms with reduced number of operations. *Signal Processing*, *6*, 267–278.
Wang, Z. (1984). Fast algorithms for the discrete W transform and for the discrete Fourier transform. *IEEE Transactions on Acoustics, Speech and Signal Processing*, *ASSP–32*, 803–816.
Wang, Z., & Bovik, A. (2002). A universal image quality index. *IEEE Signal Processing Letters*, *9*(3), 81–84.
Wang, Z., Bovik, A. C., Sheikh, H. R., & Simoncelli, E. P. (2004). Image quality assessment: From error visibility to structural similarity. *IEEE Transactions Image Processing*, *13*(4), 600–612. doi:10.1109/TIP.2003.819861.
Watkins, D. S. (2004). *Fundamentals of matrix computations. Pure and applied mathematics: A Wiley series of texts, monographs and tracts*. London: Wiley.
Wien, M., Sun, S. (2001). *ICT comparison for adaptive block transforms*. Tech. rep.
Yarlagadda, R. K., Hershey, J. E. (1997). *Hadamard matrix analysis and synthesis with applications to communications and signal/image processing*. Kluwer Academic Publishers.
Yip, P., Rao, K. R. (1988). The decimation-in-frequency algorithms for a family of discrete sine and cosine transforms. *Circuits, Systems, and Signal Processing*, 7(1), 3–19. doi:10.1007/BF01600005.

Thiago L. T. da Silveira is an M.Sc. student at the Graduate Program in Informatics at Federal University of Santa Maria (UFSM), Santa Maria, Brazil. He earned his degree in Computer Science from the same Instituition in 2013. His research interests are in the field of digital signal processing.

Fábio M. Bayer earned his B.Sc. in Mathematics, M.Sc. in Industrial Engineering from the Federal University of Santa Maria (UFSM), Santa Maria, Brazil, and D.Sc. in Statistics from the Federal University of Pernambuco (UFPE), Recife, Brazil, in 2006, 2008, and 2011, respectively. He is currently with the Department of Statistics and the Laboratory of Space Sciences of Santa Maria (LACESM), UFSM. His research interests include statistical computing, parametric inference, and digital signal processing.

Renato J. Cintra earned his B.Sc., M.Sc., and D.Sc. degrees in Electrical Engineering from Universidade Federal de Pernambuco (UFPE), Brazil, in 1999, 2001, and 2005, respectively. In 2005, he joined the Department of Statistics at UFPE. His long term topics of research include theory and methods for digital signal processing, communications systems, and applied mathematics. He performs routinely reviewing work for major peer-review journals. During 2008–2009, he worked at the University of Calgary, Canada, as a visiting research fellow. In 2010, he was a visiting professor at the University of Akron, OH. He is also member of AMS and SIAM.

Sunera Kulasekera is a M.Sc. student at the Department of Electrical and Computer Engineering at the University of Akron. He obtained his B.Sc. degree in Computer Science and Engineering from University of Moratuwa, Sri Lanka. After the bachelor's degree he worked as a Business Analyst at Millennium Information Technologies. His research interests are in the field of digital architectures.

Arjuna Madanayake (M'03) is a tenure-track assistant professor at the Department of Electrical and Computer Engineering at the University of Akron. He completed Both M.Sc. (2004) and Ph.D. (2008) Degrees, in Electrical Engineering, from the University of Calgary, Canada. Dr Madanayake obtained a B.Sc. in Electronic and Telecommunication Engineering (with First Class Honors) from the University of Moratuwa in Sri Lanka, in 2002. His research interests include multidimensional signal processing, analog/digital and mixed-signal electronics, FPGA systems, and VLSI for fast algorithms.

Alice J. Kozakevicius is a professor of Mathematics at the Federal University of Santa Maria, UFSM, Brazil. She earned a M.Sc. degree in Applied Mathematics from the Federal University of Rio Grande do Sul and a D.Sc. degree from the State University of São Paulo, both in Brazil. Since 2007, she is a member from the graduate program in Computer Science at UFSM and her research interests are wavelet methods for solving partial differential equations and signal analysis.