Low-complexity 8 × 8 transform for image compression

S. Bouguezel, M.O. Ahmad and M.N.S. Swamy

An efficient 8 × 8 sparse orthogonal transform matrix is proposed for image compression by appropriately introducing some zeros in the 8 × 8 signed discrete cosine transform (SDCT) matrix. An algorithm for its fast computation is also developed. It is shown that the proposed transform provides a 25% reduction in the number of arithmetic operations with a performance in image compression that is much superior to that of the SDCT and comparable to that of the approximated discrete cosine transform.

Introduction: Even though a number of algorithms for fast computation of the floating-point discrete cosine transform (DCT) are available, recent applications such as multimedia, mobile communications and Internet require faster transform algorithms for data compression. Since there is almost no scope for further reduction of the complexity in the computation of the floating-point DCT, there has been a lot of interest in recent years in finding approximate versions of the floating-point DCT [1–5]. Specifically, the 8 × 8 versions reported in [2] and the matrix $\hat{\boldsymbol{D}}_5$ proposed in [5] provides finer approximations of the floating point DCT than the matrix $\hat{\boldsymbol{D}}_1$ in [5] and the signed DCT (SDCT) in [3]. However, this is achieved at the expense of a substantial increase in the computational complexity. For instance, savings of 20% in the number of additions can be obtained using the matrix $\hat{\boldsymbol{D}}_1$ [5] compared to the matrix binDCT-C in [2]. It is noted that at low bit rates, low-complexity transforms are, in general, preferable.

Proposed 8 × 8 transform matrix: The approach introduced in [3] is interesting and simple. It consists of approximating the DCT by applying a signum function operator to the forward DCT matrix in order to obtain a matrix the entries of which are only 1, 0 or −1; the resulting SDCT still has good de-correlation and power compaction properties. The proposed 8 × 8 transform matrix is obtained by appropriately introducing some 0s and 1/2 s in the 8 × 8 SDCT matrix reported in [3], and is given as

$$\boldsymbol{C} = \boldsymbol{D}\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 & -1 & -1 \\ 1 & 1/2 & -1/2 & -1 & -1 & -1/2 & 1/2 & 1 \\ 0 & 0 & -1 & 0 & 0 & 1 & 0 & 0 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & -1 & 0 & 0 & 0 & 0 & 1 & -1 \\ 1/2 & -1 & 1 & -1/2 & -1/2 & 1 & -1 & 1/2 \\ 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 \end{bmatrix} = \boldsymbol{D}\boldsymbol{T} \tag{1}$$

where $\boldsymbol{D} = \mathrm{diag}\left(\frac{1}{2\sqrt{2}}, \frac{1}{2}, \frac{1}{\sqrt{5}}, \frac{1}{\sqrt{2}}, \frac{1}{2\sqrt{2}}, \frac{1}{2}, \frac{1}{\sqrt{5}}, \frac{1}{\sqrt{2}}\right)$. It can be verified that the proposed 8 × 8 transform matrix given by (1) is orthogonal, i.e. $\boldsymbol{C}^{-1} = \boldsymbol{C}^{\mathrm{t}} = \boldsymbol{T}^{\mathrm{t}}\boldsymbol{D}$, where t denotes the matrix transpose operation. It should be noted that, in contrast, the 8 × 8 SDCT matrix in [3] is not orthogonal.

We now discuss the use of the proposed transform in image compression applications. In transform-based image compression techniques including the above mentioned ones, images are usually divided into 8 × 8 blocks. Let $\boldsymbol{X}$ be an 8 × 8 block matrix of an image and $\boldsymbol{F}$ its corresponding block matrix in the transform domain. Then, the forward and inverse transformation operations can be performed using the proposed transform as $\boldsymbol{F} = \boldsymbol{C}\boldsymbol{X}\boldsymbol{C}^{\mathrm{t}} = \boldsymbol{D}(\boldsymbol{T}\boldsymbol{X}\boldsymbol{T}^{\mathrm{t}})\boldsymbol{D}$ and $\boldsymbol{X} = \boldsymbol{C}^{\mathrm{t}}\boldsymbol{F}\boldsymbol{C} = \boldsymbol{T}^{\mathrm{t}}(\boldsymbol{D}\boldsymbol{F}\boldsymbol{D})\boldsymbol{T}$, respectively. Since the quantisation/dequantisation is applied on the block matrix in the transform domain, the diagonal matrix $\boldsymbol{D}$ can be merged into the quantisation/dequantisation matrix. Such a merging is also suggested in many papers, including [2], [5] and [6]. Therefore, the matrix $\boldsymbol{T}$ is the only source of arithmetic operations in the transformation step of an image compression procedure that is based on the proposed transform.

As can be seen from (1), the matrix $\boldsymbol{T}$ is sparse and has 20 zero entries. Besides this sparse property, the proposed transform has an efficient algorithm for its fast computation. The matrix can be decomposed as $\boldsymbol{T} = \boldsymbol{T}_3 \times \boldsymbol{T}_2 \times \boldsymbol{T}_1$, where

$$\boldsymbol{T}_1 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & -1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & -1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & -1 \end{bmatrix},$$

$$\boldsymbol{T}_2 = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 \\ 1 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \end{bmatrix}, \text{ and}$$

$$\boldsymbol{T}_3 = \begin{bmatrix} 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1/2 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 1/2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Therefore, the fast computation of the proposed transform matrix requires only 18 additions and two bit-shift operations. In addition, the algorithm has an in-place computation property. Since the SDCT matrix in [3] requires 24 additions, it is clear that about 25% savings in the number of arithmetic operations can easily be obtained using the proposed transform matrix.

Table 1: PSNR obtained by different 8 × 8 transform matrices

Test image	DCT [6]	$\hat{\boldsymbol{D}}_1$ [5]	Proposed transform	SDCT [3]
Cameraman	43.4457	42.9709	42.5775	40.9289
Lena	44.0992	42.9595	42.3735	39.9144
Bridge	37.6407	37.2364	36.8643	35.2887

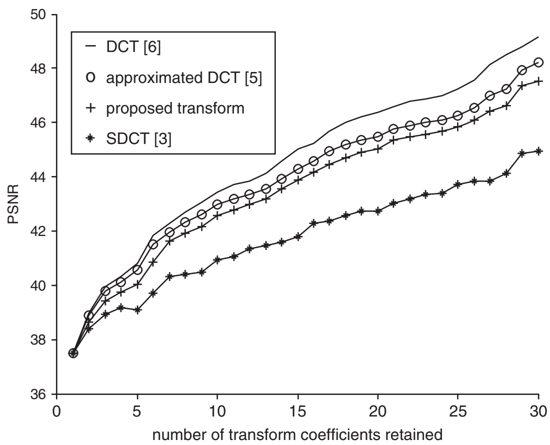

Fig. 1 *PSNR of different transforms for 256 × 256 'Cameraman' image*

To show the efficiency of the proposed 8 × 8 transform matrix in image compression, we consider the experiment used in [3]. Only ten out of the 64-transform coefficients in each block have been employed to reconstruct the image. All the other coefficients have been set to zero. The resulting peak signal-to-noise ratio (PSNR) for the transform in [3] as well as that of the proposed one is given in Table 1 for 256 × 256 'Cameraman', 512 × 512 'Lena' and 512 × 512 'Bridge' images.

Figs 1 and 2 show the PSNR obtained using the two transforms in terms of the number of transform coefficients employed to reconstruct the image. It is clear from these Figures that the PSNR obtained by the proposed transform is significantly higher than that obtained by the SDCT of [3]. Thus, in comparison to the SDCT [3], the proposed transform matrix yields a much superior performance in terms of PSNR, while at the same time it provides a substantial reduction in the number of arithmetic operations. Since the approximated DCT matrix $\hat{\boldsymbol{D}}_1$ in [5] provides a much superior performance than the SDCT of [3] with 24 additions and two bit-shift operations (about the same complexity as that of the SDCT), we compare the performance of the proposed transform with that of the former. For this purpose, we have included the corresponding results for the matrix $\hat{\boldsymbol{D}}_1$ of [5] in Table 1 as well as in Figs. 1 and 2. For the sake of reference, the corresponding results are also included for the scaled DCT [6]. It is seen from the Table and the Figures that the proposed transform provides a performance that is almost the same as that of the matrix $\hat{\boldsymbol{D}}_1$ in [5], but with a 25% reduction in the number of additions.

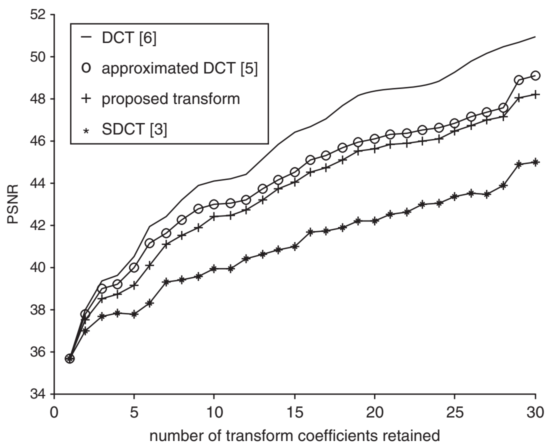

Fig. 2 *PSNR of different transforms for 512 × 512 'Lena' image*

Conclusions: An 8 × 8 sparse orthogonal transform matrix and an algorithm for its fast computation are proposed for image compression. It has been shown that the proposed transform, which is obtained by appropriately modifying the signed DCT, yields a performance much superior to that of the signed DCT and comparable to that of the approximated DCT, while at the same time it provides a 25% reduction in the number of arithmetic operations compared to that of the two transforms.

2 August 2008
Electronics Letters online no: 20082239
doi: 10.1049/el:20082239

S. Bouguezel (*Department of Electronics, Faculty of Engineering, University of Setif, Setif 19000, Algeria*)

M.O. Ahmad and M.N.S. Swamy (*Department of Electrical and Computer Engineering, Concordia University, Montreal, Canada*)

E-mail: swamy@ece.concordia.ca

References

1 Cham, W.-K.: 'Development of integer cosine transforms by the principle of dyadic symmetry', *IEE Proc. I, Commun. Speech Vis.*, 1989, **136**, (8), pp. 276–282
2 Tran, T.D.: 'The BinDCT: Fast multiplierless approximation of the DCT', *IEEE Signal Process. Lett.*, 2000, **7**, (6), pp. 141–144
3 Haweel, T.I.: 'A new square wave transform based on the DCT', *Signal Process.*, 2001, **81**, pp. 2309–2319
4 Dimitrov, V., Wahid, K., and Jullien, G.: 'Multiplication-free 8 × 8 2D DCT architecture using algebraic integer encoding', *Electron. Lett.*, 2004, **40**, (9), pp. 1310–1311
5 Lengwehasatit, K., and Ortega, A.: 'Scalable variable complexity approximate forward DCT', *IEEE Trans. Circuits Syst. Video Technol.*, 2004, **14**, (11), pp. 1236–1248
6 Feig, E., and Linzer, E.: 'Scaled DCT's on input sizes that are composite', *IEEE Trans. Signal Process.*, 1995, **43**, (1), pp. 43–50