A Fast Block-Pruned 4x4 DTT Algorithm for Image Compression

Hassan I. Saleh

***Abstract*— The Discrete Tchebichef transform (DTT) is a linear orthonormal version of the orthogonal Tchebichef polynomials, which is recently used in image analysis and compression. This paper presents a new fast block-pruned 4x4 DTT algorithm which is suitable for pruning the output coefficients in block fashion. The principle idea behind the proposed algorithm is the utilization of the distributed-arithmetic and the symmetry properties of 2-d DTT in order to combine similar terms of the linear combination of each computed pruned output. As well as, some trivial multiplications are represented by shifts or add-shift operations to reduce the number of required computations. The proposed algorithm requires the smallest computation complexity with respect to other recently proposed algorithms. Different block-pruning sizes are considered in the comparative analysis of the proposed algorithm vs. others. Furthermore, the experimental results show that the DTT is a good alternative for the Discrete Cosine Transform (DCT) in image compression especially for artificial diagrams images.**

***Index Terms*— DTT, Fast algorithm, Image Compression, Pruned.**

I. INTRODUCTION

Huge amounts of image data such as digital photographs, webpage pictures, and videos are created and transmitted via the internet. Due to the limited data storage and network capabilities, image compression is a great interest field of research. Therefore, there is a strong demand towards the efficient compression systems. The lossy and lossless compressions are the two classes of the image compressions. Lossless compression completely recovers the original data when decompressing the compressed data. On the contrary, lossy compression loses some information over the compression-decompression process. Human eyes can't recognize small differences in two similar pictures. Thus, Images can be compressed using lossy compression.

Lossy image compression applies the transformation techniques in order to transform the original image data from the spatial domain into a different domain such as the frequency domain. The transformation methods such as Fourier transform and Wavelet transform which have both analog and discrete transforms into another type of frequency domain based on trigonometric functions, and compactly supported functions called wavelets, respectively. The Discrete Cosine Transform (DCT) is a transform method used in JPEG image compression.

DCT algorithms can be classified into two classes; indirect and direct methods. A direct 2-D DCT method based on polynomial transform techniques was proposed in [1]. Another direct method is a matrix factorization algorithm of the 2-D DCT matrix [2]. For (NxN)-point 2-D DCTs, the conventional direct method follows the row-column method which requires 2N sets of N-point 1-D DCTs. However, true 2-D techniques are more efficient than the conventional row-column approach. In [3], Vetterli proposed an indirect method to calculate 2-D DCT by mapping it into a 2-D DFT plus a number of rotations.

In many DCT based applications, the most useful information is kept by the low-frequency DCT coefficients. Exploiting this characteristic, additional speed-up is possible by taking into account the statistics of the signal to be transformed. One method to achieve this is usually called 'pruning', where only a subset of all the DCT coefficients are computed, generally the low-frequency DCT components. There are several algorithms for pruning the 1-D DCT, [4]-[8], but in [9]-[12], the pruning of the 2-D DCT is addressed.

In [5], a pruned DCT algorithm was computed by a modified real-valued output-pruned FFT algorithm for appropriately permuted data samples. A recursive pruned DCT algorithm was presented [6], with a structure that allows the generation of the next higher order pruned DCT from two identical lower order pruned DCTs. The recursive pruned DCT (N1 pruned out of N point) was utilized to compute 2D pruned DCT (N1x N1) based on row-column decomposition for image compression applications. Different pruned block sizes were computed and applied for image compression [6]. The pruned 2D DCT based on row-column decomposition has complexity levels depends on the data pruning patterns [9]. With respect to full 8x8 2D DCT block, the complexity levels for computing the upper-left pruned 4x4, and 6x6 sub-blocks, are about 55%, and 72%, respectively [9]. The pruning algorithms in [10], and [12] compute a set of coefficients included in a top-left triangle. It corresponds to a zig-zag scanning where all coefficients in each diagonal are computed. In [9] and [5], the pruning algorithm computes N1xN1 block coefficients out of NxN. The pruning algorithms in [10], and [12] are more "useful" since it can be used in practical image and video coding algorithms where the zig-zag scanning pattern is used.

The Discrete Tchebichef Transform (DTT) is another transform based on a linear orthonormal version of

Manuscript received April 29, 2009. H. I. Saleh is permanently with the National Center for Radiation Research and Technology (NCRRT), Atomic Energy Authority, Egypt (phone: 202-22738665; fax: 202-22749298). H. I. Saleh is currently with Computer Science Dept., Gurayat Community College, Jouf University, KSA (Phone:+966551553404, Fax:+96646432445).

Tchebichef polynomials and recently used in image compression [13]. The DTT and the DCT have very similar energy compactness for natural images such as photographs [14]. On the other hand, for images with high illumination variations such as artificial diagrams, the DTT has higher energy compactness than the DCT [15]. The DTT is also utilized in image feature extraction and pattern recognition [16]. Recent publications proposed fast 4x4 DTT algorithms for image compression in [17], and [18]. As well as, a 2x2 block-pruned out of 4x4 DTT algorithm which computes only the upper-left quarter of the outputs, was proposed in [19].

In this paper, a fast 4x4 algorithm which is suitable for different block-pruning sizes is proposed. The rest of the paper is organized as follows. Section II introduces the definition and the properties of the DTT algorithm and its 4x4 version. The proposed fast block-pruning 4x4 DTT algorithm is presented in Section III. In Section IV, the computation complexity of the proposed algorithm is illustrated with respect to different block-pruning sizes. As well as, the computation complexity of the proposed algorithm is compared with those of others. Section V demonstrates the reconstruction of different standard images using the pruned 2-D DTT vs. the pruned 2-D DCT. Finally, the conclusion is given in Section VI.

II. The DTT definition and properties

A. *Definition*

The DTT is defined as follows:

$$X(m) = \sum_{n=0}^{N-1} x(n) t_m(n) , \qquad (1)$$

where x is an input vector of N values, and X is the vector of the transformed coefficients.

The kernel, $t_m(n)$, of the DTT is given by:

$$t_m(n) = a_1 (n + (1-N)/2) t_{m-1}(n) + a_2 t_{m-2}(n), \quad \text{for } m = 2,.., N\text{-}1, \text{ and } n = 0, \dots, N\text{-}1 \qquad (2)$$

with

$$t_0(n) = 1/\sqrt{N} , \qquad (3)$$

and

$$t_1(n) = (2n + 1 - N)\sqrt{\frac{3}{N(N^2-1)}} \qquad (4)$$

where

$$a_1 = \frac{2}{m}\sqrt{\frac{4m^2-1}{N^2-m^2}} \qquad (5)$$

and

$$a_2 = \frac{1-m}{m}\sqrt{\frac{2m+1}{2m-3}}\sqrt{\frac{N^2-(m-1)^2}{N^2-m^2}} \qquad (6)$$

B. *The 1-D DTT*

The N-point 1-D DTT $X(m)$ of an input sequence $x(n)$, for $n, m = 0, \dots, N\text{-}1$ is defined as

$$X(m) = \sum_{n=0}^{N-1} t(m,n) x(n) , \qquad (7)$$

and the inverse 1-D DTT is given by

$$x(n) = \sum_{m=0}^{N-1} t(m,n) X(m) , \qquad (8)$$

where the kernel $t(m, n)$ represents $t_m(n)$ the Tchebichef orthogonal basis given by (2).

C. *The 2-D DTT*

The 2-D DTT can be defined in the matrix form [18] as

$$X = t\, x\, t' , \qquad (9)$$

where x is the 2-D input matrix, t is the Tchebichef transform kernel and X is the 2-D transform coefficients matrix.

C) Symmetry and separability properties

The DTT satisfies the even symmetry property [18] as from (2)

$$t_m(N-1-n) = (-1)^m t_m(n). \qquad (10)$$

The 2-D DTT satisfies the separability property as the DTT is linear transform [17]. Hence, the 2-D DTT can be evaluated using the 1-D DTT in a row-column decomposition fashion.

III. Proposed Block-pruned 4x4 DTT Algorithm

The transform kernel for 4-point DTT can be defined from (2) as

$$t = \begin{pmatrix} 1/2 & 1/2 & 1/2 & 1/2 \\ -3/2\sqrt{5} & -1/2\sqrt{5} & 1/2\sqrt{5} & 3/2\sqrt{5} \\ 1/2 & -1/2 & -1/2 & 1/2 \\ -1/2\sqrt{5} & 3/2\sqrt{5} & -3/2\sqrt{5} & 1/2\sqrt{5} \end{pmatrix}. \qquad (11)$$

Let us define $a = 1/2$, and $b = 1/\sqrt{5}$, hence (11) can be rewritten as

$$t = \begin{pmatrix} a & a & a & a \\ -3ab & -ab & ab & 3ab \\ a & -a & -a & a \\ -ab & 3ab & -3ab & ab \end{pmatrix} \qquad (12)$$

Substituting from (12) in (9), each 2-D transform coefficient $X_{i,j}$ can be computed as linear combination of the elements of the input matrix x. Furthermore, the even symmetry property in (10) is used to reduce the number of arithmetic operations in a distributed arithmetic style. For block-pruning, only the specific transform coefficients can be computed in upper-left sub-blocks. The 3x3 block-pruned coefficients are given as:

$$X_{0,0} = a^2((x_{0,3}+x_{0,0}) + (x_{0,2}+x_{0,1}) + (x_{3,3}+x_{3,0}) + (x_{3,2}+x_{3,1}) + (x_{1,3}+x_{1,0}) + (x_{1,2}+x_{1,1}) + (x_{2,3}+x_{2,0}) + (x_{2,2}+x_{2,1})) \qquad (13.a)$$

$$X_{0,1} = a^2 b(3(x_{0,3}-x_{0,0}) + (x_{0,2}-x_{0,1}) + 3(x_{3,3}-x_{3,0}) + (x_{3,2}-x_{3,1}) + 3(x_{1,3}-x_{1,0}) + (x_{1,2}-x_{1,1}) + 3(x_{2,3}-x_{2,0}) + (x_{2,2}-x_{2,1})) \qquad (13.b)$$

$$X_{1,0} = a^2 b(3(-(x_{0,3}+x_{0,0}) - (x_{0,2}+x_{0,1}) + (x_{3,3}+x_{3,0}) + (x_{3,2}+x_{3,1})) - (x_{1,3}+x_{1,0}) - (x_{1,2}+x_{1,1}) + (x_{2,3}+x_{2,0}) + (x_{2,2}+x_{2,1})) \qquad (13.c)$$

$$X_{1,1} = a^2 b^2(-9(x_{0,3}-x_{0,0}) - 3(x_{0,2}-x_{0,1}) + 9(x_{3,3}-x_{3,0}) + 3(x_{3,2}-x_{3,1}) - 3(x_{1,3}-x_{1,0}) - (x_{1,2}-x_{1,1}) + 3(x_{2,3}-x_{2,0}) + (x_{2,2}-x_{2,1})) \qquad (13.d)$$

$$X_{0,2} = a^2((x_{0,3}+x_{0,0}) - (x_{0,2}+x_{0,1}) + (x_{3,3}+x_{3,0}) - (x_{3,2}+x_{3,1}) + (x_{1,3}+x_{1,0}) - (x_{1,2}+x_{1,1}) + (x_{2,3}+x_{2,0}) - (x_{2,2}+x_{2,1})) \qquad (13.e)$$

$$X_{1,2} = a^2b(3(-(x_{0,3}+x_{0,0})+(x_{0,2}+x_{0,1})+(x_{3,3}+x_{3,0})-(x_{3,2}+x_{3,1})) - (x_{1,3}+x_{1,0})+(x_{1,2}+x_{1,1})+(x_{2,3}+x_{2,0})-(x_{2,2}+x_{2,1})) \quad (13.f)$$

$$X_{2,0} = a^2((x_{0,3}+x_{0,0})+(x_{0,2}+x_{0,1})+(x_{3,3}+x_{3,0})+(x_{3,2}+x_{3,1}) - ((x_{1,3}+x_{1,0})+(x_{1,2}+x_{1,1})+(x_{2,3}+x_{2,0})+(x_{2,2}+x_{2,1}))) \quad (13.g)$$

$$X_{2,2} = a^2((x_{0,3}+x_{0,0})-(x_{0,2}+x_{0,1})+(x_{3,3}+x_{3,0})-(x_{3,2}+x_{3,1}) - ((x_{1,3}+x_{1,0})-(x_{1,2}+x_{1,1})+(x_{2,3}+x_{2,0})-(x_{2,2}+x_{2,1}))) \quad (13.h)$$

$$X_{2,1} = a^2b(3(x_{0,3}-x_{0,0})+(x_{0,2}-x_{0,1})+3(x_{3,3}-x_{3,0})+(x_{3,2}-x_{3,1}) - (3(x_{1,3}-x_{1,0})+(x_{1,2}-x_{1,1})+3(x_{2,3}-x_{2,0})+(x_{2,2}-x_{2,1}))) \quad (13.i)$$

It is obvious that, the multiplications by the factors a^2=0.25, 3, and 9, can be computed by a shift-2bits, a shif-1bit and an addition, and a shift-3bits and an addition, operations, respectively. The full 4x4 DTT coefficients can be computed as well for the remaining points and the full 4x4 DTT computation complexity is given in the next section.

IV. The Computation Complexity

From the computation complexity point of view, the proposed block-pruned 4x4 DTT algorithm is compared with recent algorithms [17], [18], [19], and the traditional separability-symmetry algorithm. We have to mention that the scaled algorithm in [18] is considered here as a normalized full version that require 16 multiplications as a last stage to get the final output. Table I gives comparative results of the compared algorithms with the proposed one in terms of the number of multiplications, additions, and shifts operations. Different block-pruned sizes out of 4x4 are considered for our proposed algorithm complexity.

Table I. Comparative Results between Different 2-D DTT Algorithms and the proposed one.

Block-pruned DTT	Computation complexity Mults / adds / Shifts				
	Separability & Symmetry	[17]	[18]	[19]	Proposed
1x1	-	-	-	-	0/15/1
2x2	-	-	-	24/48/00	2/39/7
3x3	-	-	-	-	6/66/14
4x4 (Full)	64/96/0	32/66/00	16/80/16	-	12/80/20

It is clear that our proposed algorithm complexity is much better than that of the algorithm in [19] for pruned 2x2-block. For the full 4x4 DTT outputs, our proposed algorithm is slight better than algorithm in [18]. However, our proposed algorithm is much better than that in [17] as saving 20 multiplications by turning it into shift-add operations which are much simpler than multiplications.

V. The Expiremenatl Results of the Block-Pruned DTT vs. DCT

The proposed DTT algorithm is tested in compression of set of standard images, shown in Fig.1, which are reconstructed from different upper-left block-pruned sizes of 1-point (dc component), 2x2, 3x3, and 4x4 coefficients. The reconstructed images are compared with those reconstructed by the pruned DCT. Fig. 2 compares the PSNR of the set of images reconstructed by different block-pruning sizes of the DTT and the DCT. The performance of the DTT is very similar to that of the DCT for the natural images such as Lenna, and Boat. For artificial images such as Glass, and Ruler, the PSNR of the DTT is slightly better than that of the DCT. The mean square error (MSE) of the reconstructed Lenna, and Ruler are shown in Table II. Fig. 3 demonstrates the Lenna image reconstructed from different block-pruned out of 4x4 DTT.

Table II. MSE of the reconstructed Lenna, and Ruler images with respect to different block-pruned of the DTT and the DCT.

Block-pruned DTT	MSE of reconstructed image			
	Lenna		Ruler	
	DTT	DCT	DTT	DCT
1x1	24.29	24.51	74.63	74.63
2x2	11.35	11.18	49.40	49.28
3x3	3.65	3.47	40.15	47.62
4x4 (Full)	0	0	0	0

VI. Conclusion

In this paper, a new fast algorithm of 2-D 4x4 DTT has been proposed which is suitable for pruning in a block fashion. Our proposed algorithm requires less computation complexity in comparison with the recent published algorithms. The basic idea behind the algorithm is to utilize the symmetry property beside the distributed-arithmetic to combine similar terms in linear combinations for each pruned output. The block-pruned 4x4 DTT is used to reconstruct a set of standard images showing that the DTT compression is very similar to the DCT compression for the natural photograph images. For the artificial images which have high illumination variations, the DTT has higher energy compactness than the DCT. For Future work, the zigzag-order pruning of the 2-D DTT algorithm is an interested research area as the zigzag scanning is more suitable for image compression standard. As well as, the regularity of the DTT algorithm is highly required.

Acknowledgment

H. I. Saleh thanks Prof. M. Ashour for his valuable discussions and comments.

References

[1] P. Duhamel, and C. Guillemot, “Polynomial transform computation of 2-D DCT, ” Proc. ICASSP’90, pp. 1515 -1518, 1990.
[2] E. Feig, and S. Winograd, “Fast algorithms for the discrete cosine transform”, IEEE Trans. Signal Processing, vol. 40, no. 9, pp. 2174-2193, Sep. 1992.
[3] M. Vetterli, “Fast 2-D discrete cosine transform”, Proc. IEEE ICASSP’85, pp. 1538–1541, 1985.
[4] Z. Wang, “Pruning the Fast Discrete Cosine Transform”, IEEE Transactions On Communications, VOL. 39, NO. 5, pp. 640-643, May 1991.
[5] R. Stasinski, “On Pruning the Discrete Cosine and Sine Transforms”, IEEE MELECON 2004, Dubrovnik, Croatia, pp. 269-271, May 12-15, 2004.
[6] M. El-Sharkawy, and W. Eshmawy, “A fast recursive pruned DCT for image compression applications”, 37th. Midwest. Symposium. on Circuits. and Systems, pp. 887-890, 1995.
[7] A. N. Skodras, “Fast discrete cosine transform pruning”, IEEE Trans. Signal Processing, vol.42, no. 7, pp. 1833-1837, Jul. 1994.
[8] Y. Huang, J. Wu, and C. Chang, "A generalized output pruning algorithm for matrix-vector multiplication and its application to computing pruning discrete cosine transform", IEEE Trans. Signal Processing, vol. 48, no. 2, pp. 561-563, Feb. 2000.
[9] S. Peng, “Complexity scalable video decoding via IDCT data pruning”, Proc. International Conference on Consumer Electronics (ICCE'01), Los Angeles, CA, pp. 74-75, Jun. 2001.
[10] N. P. Walmsley, A. N. Skodras and K. M. Curtis, “A fast picture compression technique”, IEEE Trans. Consumer Elect., vol. 40, no. 1, pp. 11-19, Feb 1994.
[11] C. A. Christopoulos, and A. N. Skodras, “Pruning the two-dimensional fast cosine transform”, Proceedings of the VII European Signal Processing Conference (EUSIPCO), Edinburgh, Scotland, UK, pp. 596-599, September 13-16, 1994.
[12] A. Silva, and A Navarro, “Fast 8x8 DCT Pruning Algorithm”, IEEE International Conference on Image Processing (ICIP), Genova, Italy, pp. 317-320, Sept. 2005.
[13] R. Mukundan, “Improving Image Reconstruction Accuracy Using Discrete Orthonormal Moments”, In Proc. of CISST’03 International Conference on Imaging Systems, Science and Technology, Las Vegas, 2003.
[14] R. Mukundan, and O. Hunt, “A comparison of discrete orthogonal basis functions for image compression.”, In proc. of Conf. on Image and Vision Computing IVCNz’04, New Zealand, pp. 53-58, 2004.
[15] R. Mukundan, “Transform Coding Using Discrete Tchebichef Polynomials”, Proc. of IASTED International Conference on Visualization Imaging and Image Processing VIIP 2006, Palma Mallorca, Spain, 28-30 Aug. 2006, pp. 270-275.
[16] R. Mukundan, “Radial Tchebichef invariants for pattern recognition.”, In Proc. of IEEE TENCON’05 Conference, Melboroune, 21-24 Nov. 2005, pp. 2098-2103 (1 – 6).
[17] K. Nakagaki, and R. Mukundan, “A Fast 4x4 Forward Discrete Tchebichef Transform Algorithm”, IEEE Signal Processing Letters, Vol. 14, No. 10, Oct. 2007, pp. 684-687.
[18] S. Ishwar, P. K. Meher, and M. N. S. Swamy, “Discrete Tchebichef Transform – A Fast 4x4 Algorithm and its Application in Image/Video Compression”, ISCAS 2008, May 18-21, pp. 260-263, USA, 2008.
[19] S. A. Abdelwahab, “Image Compression Using a Fast and Efficient Discrete Tchebichef Transform Algorithm, ” In Proc. of INFOS2008 the 6th International Conference on Informatics and Systems, Cairo, Egypt, March 27-29, 2008, pp. 49-55.

Hassan I. Saleh is assistant professor; He holds Ph.D. in Signal Processing, Cairo University, 2003; M.Sc. in Electronics and Electrical Engineering Cairo University, 1998; B. Sc. in Electronics and Electrical Engineering Cairo University, 1992; He is currently with the Department of Computer Science, Gyrayat Community College, Aljouf University, Saudi Arabia.
Dr. Saleh is permanently with Radiation Engineering Dept., National Center of Radiation Research and Technology (NCRRT), Egyptian Atomic Energy Authority (EAEA); Field of Interest and Research includes Digital Circuits, Signal and Image Processing, and Computer Systems.

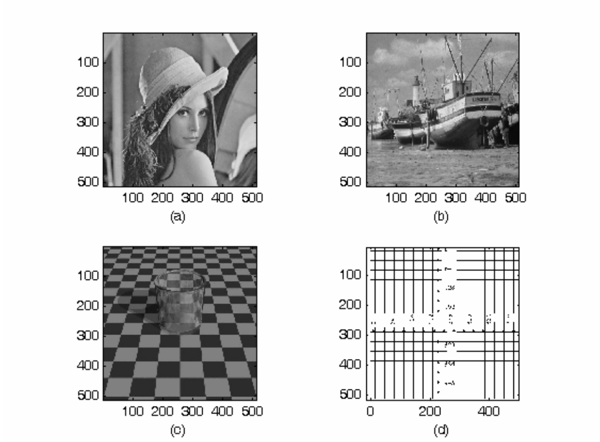

Figure 1. Set of original images, (a) Lenna, (b) Boat, (c) Glass, and (d) Ruler images, used in the experimental evaluation of the DTT and the DCT.

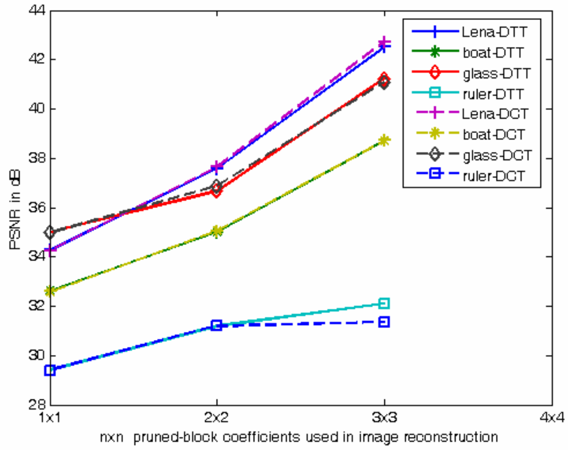

Figure 1. Comparison of reconstruction errors due to different upper-left block-pruned sizes of 4x4 DTT vs. 4x4 DCT, for Lenna, Boat, Glass, and Ruler images.

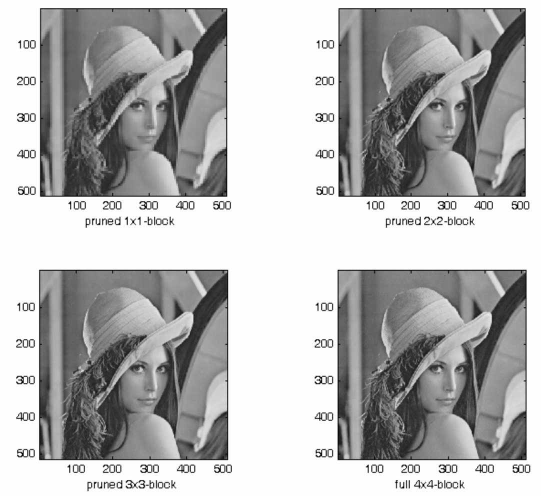

Figure 2. Lenna image reconstructed from different upper-left block-pruned DTT out of 4x4 block, (a) dc coefficient, (b) 2x2, (c) 3x3, and (d) full 4x4.