Radix-2 Algorithms for Implementation of Type-II Discrete Cosine Transform and Discrete Sine Transform

M.N.MURTY
Department of Physics, National Institute of Science and Technology, Palur Hills, Berhampur-761008, Odisha (INDIA).

ABSTRACT

In this paper radix-2 algorithms for computation of type-II discrete cosine transform (DCT) and discrete sine transform (DST) of length $N = 2^n$ $(n \geq 2)$ are presented. The DCT/DST can be computed from two DCT/DST sequences, each of length *N*/2. The odd-indexed output components of DCT/DST can be realized using simple recursive relations. The proposed algorithms require a reduced number of arithmetic operations compared with some existing methods.

Keywords – Discrete cosine transform, discrete sine transform, radix-2 algorithm, recursive.

I.INTRODUCTION

Discrete transforms play a significant role in digital signal processing. Discrete cosine transform (DCT) and discrete sine transform (DST) are used as key functions in many signal and image processing applications. There are four types of DCT and DST. Of these, the DCT-II, DST-II, DCT-IV, and DST-IV have gained popularity.

The original definition of the DCT introduced by Ahmed *et al*. in 1974 [1] was one-dimensional (1-D) and suitable for 1-D digital signal processing. The DCT has found wide applications in speech and image processing as well as telecommunication signal processing for the purpose of data compression, feature extraction, image reconstruction, and filtering. Thus, many algorithms and VLSI architectures for the fast computation of DCT have been proposed [2]-[7]. Among those algorithms [6] and [7] are believed to be most efficient two-dimensional DCT algorithms in the sense of minimizing any measure of computational complexity.

The DST was first introduced to the signal processing by Jain [8], and several versions of this original DST were later developed by Kekre *et al.* [9], Jain [10] and Wang *et al.* [11]. Ever since the introduction of the first version of the DST, the different DST's have found wide applications in several areas in Digital signal processing (DSP), such as image processing[8,12,13], adaptive digital filtering[14] and interpolation[15]. The performance of DST can be compared to that of the DCT and it may therefore be considered as a viable alternative to the DCT. For images with high correlation, the DCT yields better results; however, for images with a low correlation of coefficients, the DST yields lower bit rates [16]. Yip and Rao [17] have proven that for large sequence length ($N \geq 32$) and low correlation coefficient ($\rho < 0.6$), the DST performs even better than the DCT.

In this paper radix-2 algorithms for computation of type-II discrete cosine transform (DCT) and discrete sine transform (DST) of length $N = 2^n$ ($n \geq 2$) are presented. The DCT/DST can be computed from two DCT/DST sequences, each of length *N*/2. The odd-indexed output components of DCT/DST are realized using recursive relations.

The rest of the paper is organized as follows. The proposed radix-2 algorithm for DCT-II is presented in Section-II. The computation complexity for DCT is given in Section-III. The proposed radix-2 algorithm for type-II DST is presented in Section-IV. The computation complexity for DST is given in Section-V. The comparison of the proposed algorithms with related works is given in Section-VI. Conclusion is given in Section-VII.

II.PROPOSED RADIX-2 ALGORITHM FOR DCT-II

The type-II DCT of input sequence $\{y(i): i = 0,1,2,....,N-1\}$ is defined as

$$X(k) = \sqrt{\frac{2}{N}}\,\varepsilon(k)\sum_{i=0}^{N-1} y(i)\cos\left[\frac{k(2i+1)\pi}{2N}\right] \quad (1)$$

for $k = 0, 1, 2, \ldots, N-1$

where, $\varepsilon(k) = \begin{cases} \frac{1}{\sqrt{2}} if\ k = 0 \\ 1 if\ k = 1, 2, ..., N-1 \end{cases}$ The X values represent the transformed data. Without loss of generality, the scale factors may be ignored in the rest of the paper.

In order to derive the radix-2 algorithm for DCT, the computation of output sequence is decomposed into even-indexed and odd-indexed output sequences.

Let $N \geq 4$ be a power of 2. Using (1), the even-indexed output data $X(2m)$ can be computed as given below.

$$X(2m) = \sum_{i=0}^{\frac{N}{2}-1} [y(i) + y(N-1-i)] \cos\left[\frac{2m(2i+1)\pi}{2N}\right] \qquad (2)$$

for $m = 0, 1, 2, ..., \frac{N}{2} - 1$

Eq.(2) can be expressed as

$$X(2m) = P(m) + Q(m) \qquad (3)$$

where, $P(m) = \sum_{i=0}^{\frac{N}{2}-1} y(i) \cos\left[\frac{m(2i+1)\pi}{2\left(\frac{N}{2}\right)}\right]$ (4)

and

$$Q(m) = \sum_{i=0}^{\frac{N}{2}-1} y(N-1-i) \cos\left[\frac{m(2i+1)\pi}{2\left(\frac{N}{2}\right)}\right] \qquad (5)$$

Eq. (4) and (5) represent DCTs of size $N/2$ .From (1), the odd-indexed output data sequence $X(2m+1)$ can be realized as follows

$$X(2m+1) = \sum_{i=0}^{\frac{N}{2}-1} [y(i) - y(N-1-i)] \cos\left[\frac{(2m+1)(2i+1)\pi}{2N}\right] \qquad (6)$$

for $m = 0, 1, 2, ..., \frac{N}{2} - 1$

Eq.(6) can also be written as

$$X(2m-1) = \sum_{i=0}^{\frac{N}{2}-1} [y(i) - y(N-1-i)] \cos\left[\frac{(2m-1)(2i+1)\pi}{2N}\right] \qquad (7)$$

for $m = 1, 2, ..., \frac{N}{2}$

Adding (6) and (7), we get

$$X(2m+1) + X(2m-1) = 2\sum_{i=0}^{\frac{N}{2}-1} [y(i) - y(N-1-i)] \cos\left[\frac{m(2i+1)\pi}{2\left(\frac{N}{2}\right)}\right] \cos\left[\frac{(2i+1)\pi}{2N}\right]$$

$$= 2T(m) \quad (8)$$

where,

$$T(m) = \sum_{i=0}^{\frac{N}{2}-1} \left[y(i) - y(N-1-i) \right] \cos\left[\frac{m(2i+1)\pi}{2\left(\frac{N}{2}\right)} \right] \cos\left[\frac{(2i+1)\pi}{2N} \right] \quad (9)$$

The following recursive relation for computation of odd-indexed output components can be obtained from (8).

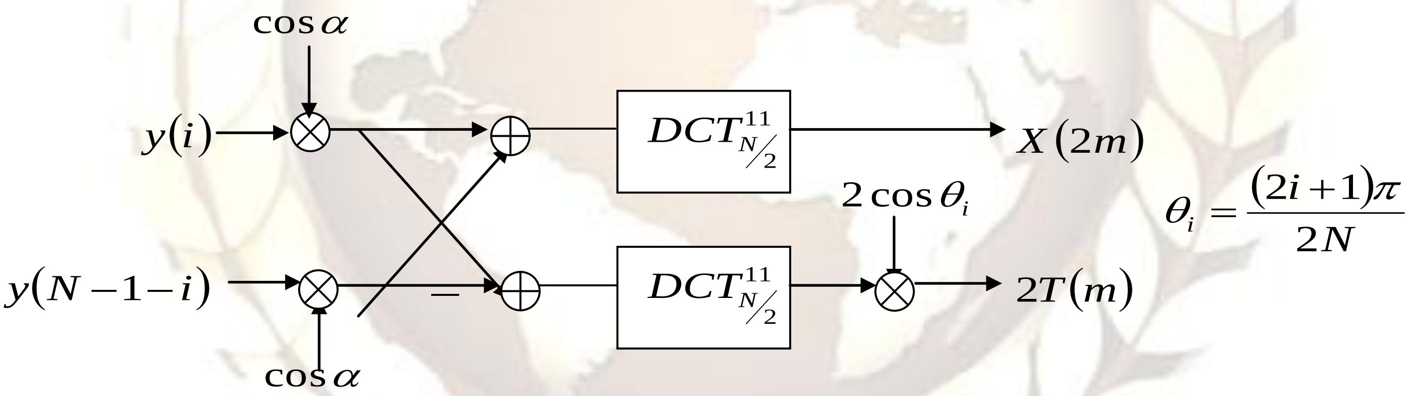
$$X(2m+1) = 2T(m) - X(2m-1) \quad (10)$$

$$\text{for} \quad m = 1, 2, ..., \frac{N}{2} - 1$$

From (10), we get the following recursive relations.

$$X(3) = 2T(1) - X(1)$$
$$X(5) = 2T(2) - X(3) \quad (11)$$
$$X(7) = 2T(3) - X(5) \text{ ,etc.}$$

The even-indexed DCT output data $\{X(2m): m = 0,1,2,...,(N/2)-1\}$ can be computed from (3) using (4) and (5). After finding *X*(1) from (6) for *m* = 0, the other odd-indexed DCT output data $\{X(2m+1): m = 1,2,......,(N/2)-1\}$ can be recursively computed using the recursive relations (11) along with (9) for $\{T(m): m = 1,2,....,(N/2)-1\}$. Fig. 1 shows the flow graph for realization of the even-indexed DCT output components *X*(2*m*) given by (2) and 2*T*(*m*) from (9).



$$0 \le i \le \frac{N}{2} - 1; \ \alpha = \frac{m(2i+1)\pi}{2\left(\frac{N}{2}\right)}$$

Fig. 1. Flow graph of the proposed algorithm for DCT.

III.COMPUTATION COMPLEXITY FOR DCT

The computation of *X*(0) requires (*N* - 1) additions only. But other even-indexed DCT output components $\{X(2m): m = 1,2,....,(N/2)-1\}$ from (2) require (*N*-1) additions and *N*/2 multiplications. For computation of *X* (2*m* – 1) from (7), we need (*N* - 1) additions and *N*/2 multiplications. The computation of 2*T*(*m*) from (9) needs an additional (*N*-1) additions and *N* multiplications. Therefore, the recursive relations (11) require (2*N* - 1) additions and 3*N*/2 multiplications for computation of odd-indexed DCT output components $\{X(2m+1): m = 0,1,2,.....,(N/2)-1\}$.

IV. PROPOSED RADIX-2 ALGORITHM FOR TYPE-II DST

Let $x(j), 1 \le j \le N$, be the input data array. The type-II DST is defined as

$$Y(k) = \sqrt{\frac{2}{N}} C_k \sum_{j=1}^{N} x(j) \sin\left[\frac{k(2j-1)\pi}{2N}\right] \tag{12}$$

for $k = 1, 2, ..., N$

where ,

$$C_k = \begin{cases} \frac{1}{\sqrt{2}} \; if \; k = N \\ 1 \; if \; k = 1, 2, ..., N-1 \end{cases}$$

The Y values represent the transformed data. The scale factors in (12) are ignored in the rest of the paper. Let $N \ge 4$ be a power of 2. The output sequence is now divided into even-indexed and odd-indexed output sequences.

Using (12), the even-indexed output sequence $Y(2n)$ can be realized as given below

$$Y(2n) = \sum_{j=1}^{\frac{N}{2}} \left[x(j) - x(N+1-j)\right] \sin\left[\frac{2n(2j-1)\pi}{2N}\right] \tag{13}$$

for $n = 1, 2, ..., N/2$

Eq.(13) can be written as

$$Y(2n) = U(n) - V(n) \tag{14}$$

Where,

$$U(n) = \sum_{j=1}^{\frac{N}{2}} x(j) \sin\left[\frac{n(2j-1)\pi}{2\left(N/2\right)}\right] \tag{15}$$

and

$$V(n) = \sum_{j=1}^{\frac{N}{2}} x(N+1-j) \sin\left[\frac{n(2j-1)\pi}{2\left(N/2\right)}\right] \tag{16}$$

Eq.(15) and (16) are DSTs of length $N/2$.

The odd-indexed output sequence $Y(2n+1)$ can be computed from (12) as given below.

$$Y(2n+1) = \sum_{j=1}^{\frac{N}{2}} \left[x(j) + x(N+1-j)\right] \sin\left[\frac{(2n+1)(2j-1)\pi}{2N}\right] \tag{17}$$

for $n = 0, 1, 2, ..., \frac{N}{2} - 1$

Eq.(17) can also be expressed as

$$Y(2n-1) = \sum_{j=1}^{\frac{N}{2}} \left[x(j) + x(N+1-j)\right] \sin\left[\frac{(2n-1)(2j-1)\pi}{2N}\right] \tag{18}$$

for $n = 1, 2, ..., \frac{N}{2}$

Adding (17) and (18), we obtain

$$Y(2n+1)+Y(2n-1)=2\sum_{j=1}^{\frac{N}{2}}\left[x(j)+x(N+1-j)\right]\sin\left[\frac{n(2j-1)\pi}{2\left(N/2\right)}\right]\cos\left[\frac{(2j-1)\pi}{2N}\right]$$

$$=2R(n) \qquad (19)$$

where,

$$R(n)=\sum_{j=1}^{\frac{N}{2}}\left[x(j)+x(N+1-j)\right]\sin\left[\frac{n(2j-1)\pi}{2\left(N/2\right)}\right]\cos\left[\frac{(2j-1)\pi}{2N}\right] \qquad (20)$$

From (19), we get the following recurrence relation for realization of odd-indexed output sequence.

$$Y(2n+1)=2R(n)-Y(2n-1) \qquad (21)$$

$$\text{for} \quad n=1,2,...,\frac{N}{2}-1$$

From (21), we obtain the following recursive relations.

$$\begin{aligned} Y(3)&=2R(1)-Y(1)\\ Y(5)&=2R(2)-Y(3)\\ Y(7)&=2R(3)-Y(5), etc. \end{aligned} \qquad (22)$$

The even-indexed DST output components $\{Y(2n): n=1,2,.....,N/2\}$ can be realized from (14) using (15) and (16). After finding $Y(1)$ from (17) for n = 0, other odd-indexed output components $\{Y(2n+1): n=1,2,...,(N/2)-1\}$ can be recursively computed using the recursive relations (22) along with (20) for $\{R(n): n=1,2,...,(N/2)-1\}$.Fig.2 shows the flow graph for realization of even-indexed DST components $Y(2n)$ from (13) and $2R(n)$ from (20).

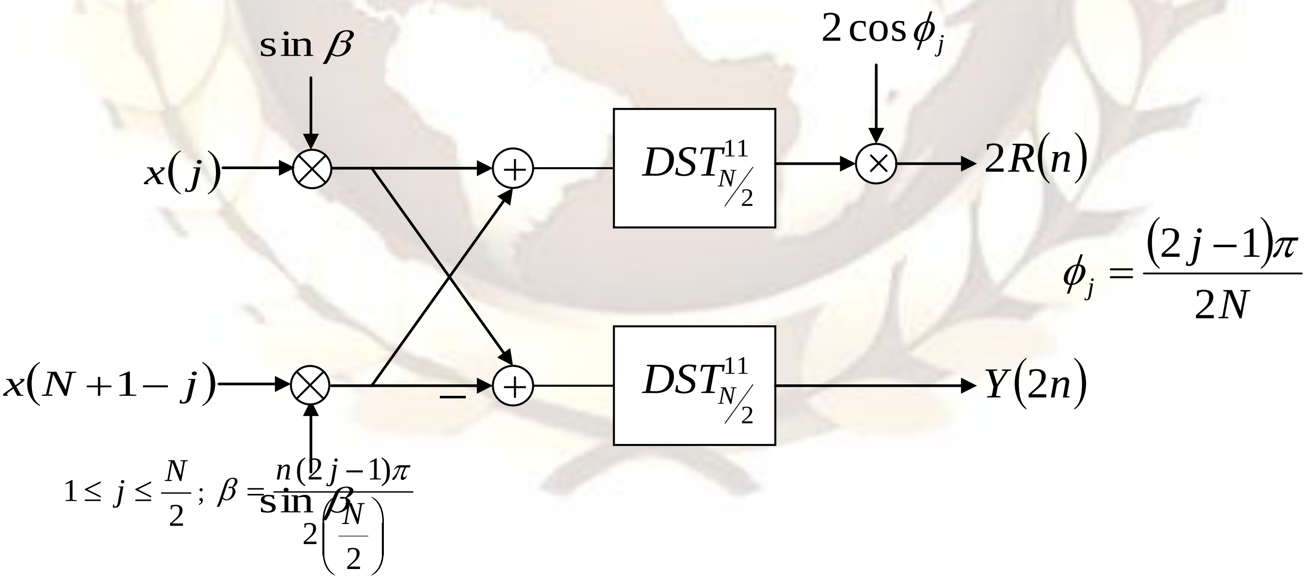


Fig. 2. Flow graph of the proposed algorithm for DST.

V. COMPUTATION COMPLEXITY FOR DST.

The computation of even-indexed DST output components $Y(2n)$ from (13) require (N-1) additions and N/2 multiplications. For computation of $Y(2n$-1) from (18), we require (N-1) additions and N/2 multiplications. The computation of $R(n)$ from (20) needs an additional (N-1) additions and N multiplications. Therefore, the recursive relations (22) require (2*N*-1) additions and 3*N*/2 multiplications for computation of odd-indexed DST output components *Y*(2*n*+1).

VI. COMPARISON WITH RELATED WORKS.

The computation complexities of the proposed radix-2 algorithms for DCT and DST are same. In Tables I and II, the number of multipliers and the number of adders in the proposed algorithms for DCT/DST are compared with the corresponding parameters in other methods. Table III gives the comparison of the computation complexities of the proposed algorithms for DCT/DST with other algorithms found in the related research works.

TABLE I
COMPARISON OF THE NUMBER OF MULTIPLIERS REQUIRED BY DIFFERENT ALGORITHMS OF DCT/DST

N	[18]	[22]	[5,24,25]	[26]	[19]	[30]	[27]	Proposed (even output)	Proposed (odd output)
4	6	5	4	11	2	5	4	2	6
8	16	13	12	19	8	13	8	4	12
16	44	33	32	36	30	29	16	8	24
32	116	81	80	68	54	61	32	16	48
64	292	193	192	132	130	125	64	32	96

TABLE II
COMPARISON OF THE NUMBER OF ADDERS REQUIRED BY DIFFERENT ALGORITHMS OF DCT/DST

N	[22]	[5,24,25]	[18]	[19]	[26]	[30]	[27]	Proposed (even output)	Proposed (odd output)
4	9	9	8	4	11	14	7	3	7
8	35	29	26	22	26	26	15	7	15
16	95	81	74	62	58	50	31	15	31
32	251	209	194	166	122	98	63	31	63
64	615	513	482	422	250	194	127	63	127

TABLE III
COMPUTATION COMPLEXITIES

	of multiplications	of additions
Proposed algorithm(even output)	$N/2$	N-1
Proposed algorithm(odd output)	$3N/2$	$2N$ -1
[5,20,21,25]	$(1/2)\, N \log_2 N$	$(3/2)\, N \log_2 N - N + 1$
[4,28,29]	$N \log_2 N /2 + 1$	$3\, N \log_2 N / 2 - N + 1$
[23]	$(1/2)\, N \log_2 N + (1/4)\, N - 1$	$(3/2)\, N \log_2 N + (1/2)\, N - 2$
[26]	$2(N+3)(N-1) / N$	$2(2N-1)(N-1) / N$
[27]	$(N+1)(N-1) / N$	$(2N+1)(N-1) / N$
[30]	$2N-3$	$3N+2$

VII. CONCLUSION

Radix-2 algorithms for computing type-II DCT and DST of length $N = 2^n (n \geq 2)$ are presented in this paper. Using these algorithms, the DCT/DST can be computed from two DCT/DST sequences, each of length *N*/2. The number of multiplications and additions in these algorithms are less in comparison with some existing algorithms. Therefore, saving in time can be achieved by the proposed algorithms in their realization. In the proposed methods, the odd-indexed output components of DCT/DST are realized using simple recursive relations. The recursive algorithms are suitable for parallel VLSI implementation.

REFERENCES

[1] N.Ahmed, T.Natarajan, and K.R.Rao, Discrete cosine transform, *IEEE Trans. Comput.*, vol. C-23, 1974, 90-93.

[2] R.K.Rao and P.Yip, *Discrete cosine transform: algorithm, advantages, and applications* (New York: Academic, 1990).

[3] M.J.Narasimha and A.M.Peterson, On the computation of the discrete cosine transform, *IEEE Trans. Communications*, vol.COM-26,no.6, June 1978, 934-936.

[4] H.S.Hou, A fast recursive algorithms for computing the discrete cosisine transform, *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-35(10), Oct.1987, 1455-1461.

[5] PeiZong Lee and Fang-Yu Huang, Reconstructed recursive DCT and DST algorithms, *IEEE Trans. Signal Processing*, vol.42, no.7, Jul.1994, 1600-1609.

[6] P.Duhamel and C.Guillemot, Polynomial transform computation of 2-D DCT ,*in Proc. ICASSP'90*, Apr.1990, 1515-1518.

[7] E.Feig and S.Winograd, Fast algorithms for the discrete cosine transform, *IEEE Trans. Signal Processing*, vol.40, no.9, Sept.1992, 2174-2193.

[8] A.K. Jain, A fast Karhunen-Loeve transform for a class of random processes, *IEEE Trans. Commun.*, vol. COM-24, September 1976, 1023-1029.

[9] H.B. Kekre and J.K. Solanka, Comparative performance of various trigonometric unitary transforms for transform image coding, *Int. J. Electron.*, vol. 44, 1978, 305-315.

[10] A.K. Jain, A sinusoidal family of unitary transforms, *IEEE Trans. Patt. Anal. Machine Intell.*, vol. PAMI-I, September 1979,356-365.

[11] Z. Wang and B. Hunt, The discrete W transform, *Applied Math Computat.*, vol. 16, January 1985, 19- 48.

[12] S. Cheng, Application of the sine transform method in time of flight positron emission image reconstruction algorithms,*IEEE Trans. BIOMED. Eng.*, vol. BME-32, March 1985,185-192.

[13] K. Rose, A. Heiman, and I. Dinstein, DCT/DST alternate transform image coding, *Proc. GLOBE COM 87*, vol. I, November 1987, 426-430.

[14] J.L. Wang and Z.Q. Ding, Discrete sine transform domain LMS adaptive filtering, *Proc. Int. Conf. Acoust., Speech, Signal Processing*, 1985, 260-263.

[15] Z. Wang and L. Wang, Interpolation using the fast discrete sine transform, *Signal Processing*, vol. 26, 1992, 131-137.

[16] A. K. Jain, *Fundamentals of Digital Image Processing* (Englewood Cliffs, NJ: Prentice - Hall, 1989)

[17] P. Yip and K.R. Rao, On the computation and the effectiveness of discrete sine transform, *Comput. Electron.*, vol. 7, 1980, 45-55.

[18] W.H.Chen ,C.H. Smith, and,S.C. Fralick , A fast computational algorithm for the discrete cosine transform, *IEEE Trans. Communicat.*, COM-25(9), September 1977, 1004-1009.

[19] P.Yip and K.R.Rao , A fast computational algorithm for the discrete sine transform, *IEEE Trans. Commun.*, COM-28, February 1980, 304-307.

[20] P.Yip and K.R. Rao, Fast decimation-in-time algorithms for a family of discrete sine and cosine transforms, *Circuits, Syst., Signal Processing*, 3, 1984, 387-408.

[21] O.Ersoy and N.C. Hu, A unified approach to the fast computation of all discrete trigonometric transforms, *In Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, 1987, 1843-1846.

[22] H.S.Malvar, Corrections to fast computation of the discrete cosine transform and the discrete hartley transform, *IEEE Trans. Acoust., Speech, Signal Processing*, 36(4), April 1988, 610-612.

[23] P.Yip and K.R.Rao, The decimation-in-frequency algorithms for a family of discrete sine and cosine transforms. *Circuits, Syst., Signal Processing*, 7(1), 1988, 3-19.

[24] A.Gupta and K.R. Rao, A fast recursive algorithm for the discrete sine transform, *IEEE Transactions on Acoustics, Speech and Signal Processing*, 38(3), March 1990; 553-557.

[25] Z.Cvetković and M.V.Popović, New fast recursive algorithms for the computation of discrete cosine and sine transforms, *IEEE Trans. Signal Processing*, 40(8), August 1992, 2083-2086.

[26] J.Caranis, A VLSI architecture for the real time computation of discrete trigonometric transform, *J. VLSI Signal Process.*, 5, 1993, 95-104.

[27] L.P.Chau and W.C. Siu, Recursive algorithm for the discrete cosine transform with general lengths, *Electronics Letters*, 30 (3), February 1994, 197-198.

[28] V.Britanak, On the discrete cosine computation, *Signal Process.*, 40(2-3), 1994,183-194.

[29] C.W.Kok, Fast algorithm for computing discrete cosine transform, *IEEE Trans. Signal Process.*, 45, March 1997, 757-760.

[30] V.Kober, Fast recursive algorithm for sliding discrete sine transform, *Electronics Letters*, 38(25), December 2002, 1747-1748.