ALGORITHMS FOR COMPUTING ODCT (ODST) INDEPENDENT OF ODST (ODCT) COEFFICIENTS FOR WINDOWED DATA

Synopsis submitted in fulfillment of the requirements for the Degree of

DOCTOR OF PHILOSOPHY

By

SHAMIM AKHTER

Department of Electronics and Communication

JAYPEE INSTITUE OF INFORMATION TECHNOLOGY
(Declared Deemed to be University U/S 3 of UGC Act)
A-10, SECTOR-62, NOIDA, INDIA

August, 2014

DECLARATION BY THE SCHOLAR

I hereby declare that the work reported in the Ph.D. thesis entitled **"Algorithms for Computing ODCT (ODST) Independent of ODST (ODCT) Coefficients for Windowed Data"** submitted at **Jaypee Institute of Information Technology, Noida, India**, is an authentic record of my work carried out under the supervision of **Prof. R. C. Jain** and **Dr. Vikram Karwal**. I have not submitted this work elsewhere for any other degree or diploma. I am fully responsible for the contents of my Ph.D. Thesis.

Shamim Akhter

Department of Electronics and Communication Engineering

Jaypee Institute of Information Technology, Noida, India

Date

SUPERVISOR’S CERTIFICATE

This is to certify that the work reported in the Ph.D. thesis entitled **“Algorithms for Computing ODCT (ODST) Independent of ODST (ODCT) Coefficients for Windowed Data”**, submitted by **Shamim Akhter** at **Jaypee Institute of Information Technology, Noida**, **India,** is a bonafide record of his original work carried out under our supervision. This work has not been submitted elsewhere for any other degree or diploma.

(Prof. R. C. Jain)

Professor,

ECE Department,

JIIT, Noida (U.P.)

India

Date…………………

(Dr. Vikram Karwal)

Assistant Professor,

ECE Department,

JIIT, Noida (U.P.)

India

Date………………….

SYNOPSIS

The synopsis being put forward is based on my research work that develops fast algorithms for computing Odd Discrete Cosine Transform (ODCT-II) and Odd Discrete Sine Transform (ODST-II) coefficient of a running data sequence.

Introduction: The DCT transform has been extensively used in various digital image coding schemes and image compression standards. These transforms provide performance comparable to that of ideal Karhunen-Loeve Transform (KLT) [1] with the advantage of lower computational complexity. This is the main reason for popularity of these transforms. Discrete Cosine Transform (DCT) and Discrete Sine Transform (DST) of Type I–IV form a group known as Even DCT/DST [EDCT/EDST] [1-3]. For store-and-forward applications, DCT coefficients are computed for a chosen input data matrix, which is normally a group of 4X4 or 8X8 pixels. In high speed real time processing, direct computation of so many pixels in a group would cause too much computational overhead and may take unacceptably high computation time. In practice, a real time data sequence is sampled using a window function and then, the desired transform coefficient of a limited number of pixels is computed [4-9]. The window function is shifted in time and the transform coefficient of updated sequence is computed as and when the new input data is available. This substantially reduces the computational complexity.

Another group of transforms, namely, DCT/DST V–VIII are called “Odd” DCT/DST [ODCT/ODST] [10]. Recently ODCT/ODST transforms have been used in applications such as ISO/IEC/ITU-T high efficiency video coding (HEVC) standards [11-14]. ODCT-II/ODST-II transform are also used for intra-coding in the HEVC standards [14]. Despite our exhaustive search, we have come across only one paper in the last decade giving ‘Simultaneous Update Algorithm’ for ODCT-II/ODST-II coefficients computation [15]. We have proposed fast ‘Independent Updating Algorithm’ for ODCT-II/ODST-II coefficients’ computation for real time data sequence sampled using rectangular and split-triangular window functions. By independent updating, we mean ODCT-II (ODST-II) coefficient computation of updated data does not require ODST-II (ODCT-II) respectively. The proposed algorithm takes lesser time to compute ODCT-

II/ODST-II coefficients of windowed data as compared to that taken by earlier proposed Simultaneous Update Algorithm [15]. Our proposed algorithm also requires lesser memory during coefficient computation since we store only one type of coefficient. These independent update algorithms have been derived analytically and their functional validity has been done by Matlab implementation. The prototype model has also been developed by VHDL implementation **and synthesized using ISE software (Integrated Software Environment) for vertex 4 FPGA**. Chapter wise details are given below.

Organization of the Thesis

Chapter 1 deals with DCT and DST origin and applications. Different types of DCT and DST are discussed [1-3]. The efficient and fast coefficient computation techniques are discussed for Even DCT (EDCT) and Even DST (EDST) [4-9]. The different configuration of Update algorithm are also given by others. Out of which we have concentrated on development of 'Independent Update Algorithm' for fast and efficient computation of ODCT-II and ODST-II coefficient.

Chapter 2 deals with analytical derivation of Independent Updating Algorithm for ODCT-II/ODST-II coefficient computation. By independent updating, we mean, that the computation of ODCT-II or ODST-II coefficients of shifted data sequence does not require any ODST-II or ODCT-II coefficients respectively. The proposed algorithm is tested by Matlab implementation for different values of sample length N and r, where r is number of updated data points and ($1 \leq r \leq N-1$). The proposed algorithm has been analytically compared with existing 'Simultaneous Update Algorithm' [15] in terms of multiplication and memory requirements. A paper based on this work has been published and given in the list of publications [1]. In these set of algorithms, rectangular window is implicit.

In **Chapter 3** 'Independent Updating Algorithms' for ODCT-II/ODST-II coefficient computation of one-point updated data in presence of split-triangular windowing function are analytically derived. It means coefficients of windowed updated sequence are independently computed each time a new pixel enters into the system. A paper based on this work has been published and given in the list of publications [2].

If we need more than one-point update, we would require using one–point update algorithm r times. For such applications we propose r-point Independent Update Algorithm. In this algorithm computation is initiated after r new pixels are available. A paper based on this work is submitted to Journal of Information Processing Systems as mentioned in the list of publications [5].

Chapter 4 deals with HDL Implementation of ODCT-II coefficient computation using distributed arithmetic (DA) approach. The coefficients computation involves sum-of-product operation between input pixel data and pre-defined Cosine values for a given value of N (number of input data points). Direct ODCT-II coefficient computation involves multiply-accumulate operations. We have chosen DA approach since DA is an efficient method for computing inner products when one of the input vectors is fixed and other is variable [16-19]. VHDL implementation is done for ODCT-II coefficient computation for $N = 8$, using distributed arithmetic (DA) approach. A paper based on this work has been published and given in the list of publications [3].

Also Prototype model is designed using VHDL for Independent Update Algorithm based ODCT-II coefficient computation as discussed in Chapter 2. We have considered the input data sample size of $N = 8$. The prototype is developed for $r = 1$. It can be redesigned for other values of N and r. Update algorithm requires ODCT-II coefficient for first two sampled data set during pre-processing. Hence ODCT-II module designed above has been used to compute coefficients during pre-processing stage of update algorithm. This work has been published and given in the list of publications [4].

In **Chapter 5,** simulation results are discussed. We have used **Matlab version: R 2013.** We have given simulation results of Matlab programs for computing ODCT-II and ODST-II coefficients directly from definition as well as by Independent Update Algorithm. Generalized program for Independent Updating of coefficients is given where user provides[r: Number of new updated data points, and first input sequence of sampled data]. The program returns the ODCT-II/ODST-II coefficients of updated data sequence.

For split-triangular window data for one-point update, transform coefficients obtained using update algorithm are compared with those obtained using conventional definition for ODCT-II and ODST-II coefficient.

We have also given simulation results of the generalized program for computing ODCT-II coefficients of split-triangular windowed data where user provides [*r*: Number of new updated data points, n_0: Tail length of split-triangular window and first input data sequence].

We have used ModelSim 5.8 for simulating VHDL models for ODCT-II coefficient computation block, as proposed in Chapter 4. The simulation results are given for different data sets with brief explanation.

Chapter 6 gives conclusion and future scope of this work.

Theoretical Background

The mathematical expressions for computing ODCT-II, C(*k*), and ODST-II coefficient, *S*(*k*) , for input data sequence *f*(*x*) of length *N* , are given in Equation (1).

$$C(k) = \frac{2}{\sqrt{2N-1}} P_k \sum_{x=0}^{N-1} P_{x+1} f(x) \cos \frac{2x+1}{2N-1} \pi k \qquad \text{for } k = 0,1,\ldots, N-1 .$$

$$S(k) = \frac{2}{\sqrt{2N-1}} \sum_{x=0}^{N-2} f(x) \sin \frac{(2x+1)k\pi}{2N-1} \qquad \text{for } k = 1,\ldots, N-1,$$

where,

$$P_k = \begin{cases} \frac{1}{\sqrt{2}} & for \quad k=0 \quad or \quad N \\ 1, & otherwise \end{cases} \qquad P_{x+1} = \begin{cases} \frac{1}{\sqrt{2}} & for \quad x = N-1 \\ 1, & otherwise \end{cases} \qquad (1)$$

In order to achieve fast and efficient computation of ODCT-II and ODST-II coefficients, a class of Simultaneous Update Algorithm is proposed in [15]. The expression for the same is give below:

$$C_+(k) = C(k)\cos\frac{2kr\pi}{M} + P_K S(k)\sin\frac{2kr\pi}{M} + aP_K\left[(1-\sqrt{\frac{1}{2}})f(N-1)(-1)^k \cos\frac{2kr\pi}{M} - f(r-1)\cos\frac{2kr\pi}{M} + \sqrt{\frac{1}{2}}(-1)^k f(N-1+r) + A\right]$$

$$S_+(k) = S(k)\cos\frac{2kr\pi}{M} - C(k)\sin\frac{2kr\pi}{M} + a\left[\sqrt{\frac{1}{2}} f(N-1)(-1)^k \sin\frac{2kr\pi}{M} + f(r-1)\sin\frac{k\pi}{M} - (-1)^k f(N-1)\sin\frac{2kr\pi}{M} + A\right] \quad (2)$$

where, $$A = \sum_{y=0}^{r-2}\left[f(y+N)\cos\left(\frac{2(y-r+N)+1}{M}k\pi\right) - f(y)\cos\left(\frac{2(y-r)+1}{M}k\pi\right)\right],$$

$$a = \frac{2}{\sqrt{2N-1}},$$

$$P_K = \begin{cases}\frac{1}{\sqrt{2}} & for \quad k = 0 \quad or \quad N, \\ 1, & otherwise.\end{cases}$$

- $C_+(k)$: ODCT-II coefficients of *r*-point updated input data sequence,
- $S_+(k)$: ODST-II coefficients of *r*-point updated input data sequence,
- $C(k)$: ODCT-II coefficient of current data sequence,
- S(k) : ODST-II coefficient of current data sequence.

As shown in Equation (2), the computation of ODCT-II coefficients of the updated data sequence requires both ODCT-II and ODST-II coefficients of current data sequence. Equation (2) can be used to compute ODCT-II and ODST-II coefficients for *r*-points updated data at a time, where $1 \leq r \leq N-1$. It can be seen that the 'Simultaneous Update Algorithm' based computation requires regular updating of ODCT-II as well as ODST-II coefficients. The above coefficient computation technique has following two main drawbacks:

(i) Requires storing both ODCT-II & ODST-II coefficient,

(ii) More computation to obtain the desired coefficients.

We have also proposed 'Simultaneous Update Algorithm' for computing ODCT-II/ODST-II coefficient of updated input data sequence and the expression for the same is given below in Equation (3).

$$
\begin{aligned}
C_{+}(k) = {} & A_r C(k) + B_r P_k S(k) \\
& + (-1)^k a P_k A_r \left(1 - \frac{1}{\sqrt{2}}\right) f(n-1) \\
& - a P_k \sum_{x=0}^{r-1} f(n-N-1+r-x) \cos\frac{(2x+1)k\pi}{2N-1} \\
& + a P_k (-1)^k \sum_{x=0}^{r-1} f(n-1+r-x) \cos\frac{2xk\pi}{2N-1} \\
& + a P_k (-1)^k \left(\frac{1}{\sqrt{2}} - 1\right) f(n-1+r) \qquad \text{for } k = 0,1,\ldots,N-1,
\end{aligned}
$$

$$
\begin{aligned}
S_{+}(k) = {} & A_r S(k) - B_r C(k) \\
& + a \sum_{x=0}^{r-1} f(n-N-1+r-x) \sin\frac{(2x+1)k\pi}{2N-1} \\
& - a(-1)^k \sum_{x=0}^{r-1} f(n-1+r-x) \sin\frac{2xk\pi}{2N-1} \\
& + a(-1)^k \left(\frac{1}{\sqrt{2}} - 1\right) B_r f(n-1) \qquad \text{for } k = 1,\ldots,N-1,
\end{aligned}
$$

(3)

where, $A_r = \cos\frac{2\pi rk}{2N-1}$, $B_r = \sin\frac{2\pi rk}{2N-1}$, $a = \frac{2}{\sqrt{2N-1}}$,

$$
P_K = \begin{cases} \frac{1}{\sqrt{2}} & \textit{for} \quad k = 0 \quad \textit{or} \quad N, \\ 1, & \textit{otherwise}. \end{cases}
$$

The Interdependency in the coefficients' computation during 'Simultaneous Update Algorithm' is eliminated by Independent Updating Algorithm for ODCT-II/ODST-II proposed by us. By Independent updating, we mean ODCT-II coefficient of updated data sequence is computed without using ODST-II coefficient of previous sampled data. Due to this reason, our proposed algorithm is termed as 'Independent Update Algorithm'. This algorithm is useful in applications where we need only one of the coefficients i.e. either ODCT-II or ODST-II. This will significantly reduce memory requirements as we do not need to store ODST-II coefficients while computing updated ODCT-II coefficient and vice-versa. The expression derived for ODCT-II coefficient computation of *r*-point updated input data sequence using Independent Update Algorithm in the case of rectangular windowed data is given in Equation (4) below:

$$
\begin{aligned}
C(n+r,k) &= 2A_r C(n,k) - C(n-r,k) \\
&+ aP_k(-1)^k\left(\frac{1}{\sqrt{2}}-1\right)\left[f(n-1-r) - f(n-1)A_r + f(n-1+r)\right] \\
&+ aP_k\sum_{x=0}^{r-1}\left[f(n-N-1-x)A_r - f(n-N-1+r-x)\right]\cos\frac{(2x+1)k\pi}{2N-1} \\
&+ aP_k(-1)^k\sum_{x=0}^{r-1}\left[f(n-1+r-x) - P_{N-x}A_r f(n-1-x)\right]\cos\frac{2xk\pi}{2N-1} \\
&- aP_k B_r(-1)^k\sum_{x=0}^{r-1}P_{N-x} f(n-1-x)\sin\left(\frac{2xk\pi}{2N-1}\right) \\
&+ aP_k B_r\sum_{x=0}^{r-1} f(n-N-1-x)\sin\left(\frac{2x+1}{2N-1}k\pi\right)
\end{aligned}
$$

$$\text{for } k = 0,1,\ldots,N-1 \qquad (4)$$

where,

$$A_r = \cos\frac{2rk\pi}{2N-1} \quad B_r = \sin\frac{2rk\pi}{2N-1} \quad a = \frac{2}{\sqrt{2N-1}},$$

$$P_K = \begin{cases} \frac{1}{\sqrt{2}} & \textit{for} \quad k=0 \quad \textit{or} \quad N, \\ 1, & \textit{otherwise} \end{cases}$$

- $C(n-r,k)$ are ODCT-II coefficient of the previous time step input sequence *f*(*x*-*r*),
- $C(n,k)$ are ODCT-II coefficient in the current time step input sequence *f*(*x*),
- $C(n+r,k)$ are ODCT-II coefficients of *r*-point updated input sequence *f*(*x*+*r*),

- Summation term $f(n-N-1-x)$ represents previous data points shifted out,
- Summation term $f(n-N-1+r-x)$ represents recent data shifted out,
- Summation term $f(n-1+r-x)$ indicates recent data points shifted in,
- Summation term $f(n-1-x)$ shows previous data shifted in,
- $f(n-1-r)$ represents last data value of the input data sequence *f*(*x*-*r*).

It can be seen that ODCT-II coefficient of updated data sequence is computed by using pre-computed ODCT-II coefficient value without any need of ODST-II Coefficients of previous data sequence. Similarly, the derivation of Independent Update Algorithm for ODST-II coefficient computation of *r*-point updated input data sequence is given in [20] and final result is given in Equation (5) below:

$$
\begin{aligned}
S(n+r,k) = {} & 2A_r S(n,k) - S(n-r,k) - a(-1)^k B_r f(n-1) \\
& + a\sum_{x=0}^{r-1}\left[f(n-N-1+r-x) + f(n-N+x-r)\right]\sin\frac{2x+1}{2N-1}k\pi \\
& - a(-1)^k \sum_{x=1}^{r-1}\left[f(n-1+r-x) + f(n-1-r+x)\right]\sin\frac{2xk\pi}{2N-1} \qquad (5) \\
& \text{for } k = 1,\dots,N-1
\end{aligned}
$$

Proposed Independent Update Algorithm based ODCT-II/ODST-II coefficient computation has two advantages as compared to that of Simultaneous Update Algorithm:

(i) Reduced computation time, and
(ii) Reduced memory space to compute coefficient.

Independent Update Algorithm requires lesser hardware for multiplication operation as compared to that required during Simultaneous Update Algorithm. Also, the quantum of advantage increases as the value of *N* and *r* increases. Most real world problems these days require computation of coefficients for large values of *N*. The advantage increases with the increases in the value of *r*. For different values of *r*, the comparison of multiplication operations required for Simultaneous and Independent Update based ODCT-II computation is given in Table.1.

Table 1: Comparison of required multiplication operations

r	Simultaneous update algorithm	Independent update algorithm
1	9N	5*N*
2	13N	7*N*
3	17N	9*N*

The computational complexity is shown for different values of N in Table 2.

Table 2: Comparison of multiplication operations for different value of N

r		N (Number of Data points) 8	16	32
1	Independent update algorithm [20]	40	80	160
2		56	112	224
3		72	144	288
r		N (Number of Data points) 8	16	32
1	Simultaneous update algorithm [20]	72	144	288
2		104	208	416
3		136	272	544

Memory is required in both the algorithms to store the following data during coefficient computation:

(a) The coefficients of the recursive equations,

(b) Transform result,

(c) Input signal samples.

Table 3 below shows the memory requirement for both the update algorithms.

Table 3: Comparison of memory requirement

	Coefficient		Input Sequence		Transform domain Data		Total memory requirement	
r	Simultaneous update	Independent update	Simultaneous update	Independent update	Simultaneous update	Independent update	Simultaneous update	Independent update
1	$6N$	$5N$	$N+1$	$N+2$	$2N$	$2N$	$9N+1$	$8N+2$
2	$10N$	$7N$	$N+2$	$N+4$	$2N$	$2N$	$13N+2$	$10N+4$
3	$14N$	$9N$	$N+3$	$N+6$	$2N$	$2N$	$17N+3$	$12N+6$

Comparison of memory requirement for different values of N and r is given in Table 4.

Table 4: Comparison of memory requirement for different values of N

r		N (Number of Data points) 8	16	32
1	Independent update algorithm [20]	66	130	258
2		84	164	324
3		102	198	390
r		N (Number of Data points) 8	16	32
1	Simultaneous update algorithm [20]	73	145	289
2		106	210	418
3		139	275	547

It is seen that ‘Simultaneous Update Algorithm’ based computation requires more storage as compared to that required in ‘Independent Update Algorithm’, since in ‘Simultaneous Update Algorithm’ both ODCT-II as well as ODST-II coefficients are to be stored. Thus it can be concluded that ‘Independent Update Algorithm’ is better in terms of storage requirement as compared to ‘Simultaneous Update Algorithm’.

In the proposed Independent Update Algorithms, rectangular window is used for sampling the input data sequence. The complexity of developed algorithms is $O(N)$ which is the lowest level of complexity. The validation of both of the above proposed algorithms is done by MATLAB implementation.

Discontinuities at the edges of a rectangular windowed data result in ringing effects [7, 8]. The ringing effect is reduced if windowed signal is made smoother at the edges. Other commonly used windowing functions are: Split-Triangular, Hanning, Hamming, and Blackman. We have proposed an algorithm to compute ODCT-II/ODST-II coefficients of one-point updated input sequence sampled using Split-Triangular window function. We have developed Independent Update Algorithm for computing ODCT-II/ ODST-II coefficients of windowed data. The proposed algorithm is capable of computing ODCT-II /ODST-II coefficient of one-point update data sequence at a time. In the developed algorithms, ODCT-II coefficients of updated windowed data are computed using ODCT-II of two previously sampled windowed data. Analogous Independent Update Algorithm based ODST-II coefficient for Split-Triangular windowed data is also proposed and tested by Matlab implementation.

The algorithm proposed above is for one-point Independent updating algorithm of split-triangular windowed data. We have also proposed split-triangular windowed update algorithm for r-point data update. The proposed algorithm has been tested by Matlab implementation for most commonly used values of N, r and window parameter (n_0).

Prototype models are developed in VHDL for ODCT-II coefficient computation using direct approach as well as Independent Update Algorithm based computation for N=8 and r=1 using DA approach and successfully simulated in ModelSim 5.8.

Conclusion

The use of a recursive method is very appropriate for machine computation. The shift property of DCT and DST transforms gives a set of recursive equations that can be used for updating DCT and DST coefficients of a running data sequence as required in real time data processing. We have analytically derived a novel and efficient recursive algorithm for computation of ODST-II and ODCT-II coefficients for rectangular window. Since the use of rectangular window results in ringing effect, we have also proposed r-point update algorithm for split triangular windowed data. In the proposed algorithms, the computation of ODCT-II coefficients does not require simultaneous computation of ODST-II coefficients. The algorithms proposed by us are an improvement over existing update algorithms as these provide independence between ODCT-II and ODST-II coefficient computation, thereby reducing computation and memory requirement. Calculating the transform of the real time input data sequence using update algorithms is more efficient than using conventional definition. The complexity of the analytically derived independent update algorithms are of the order $O(N)$.

Distributed Arithmetic (DA) based implementation is a technique to implement sum of product without actually using multiplier. ODCT-II computation involved vector matrix multiplication of inputs data with the fixed cosine values. The multiplier which multiplies input with the fixed contents significantly occupies more space to store the temporary values and also increases power consumption. As the scaling of silicon devices has progressed over the past decades, semiconductor memory has become cheaper, faster and more efficient. Thus the multipliers are replaced with memory based structures to reduce area. One of the memory based technique is Distributed Arithmetic (DA). DA is cost-effective and also provides area-time efficient computing structure. ODCT-II coefficient computation using DA approach is more suitable for FPGA based implementation because it efficiently utilizes ROM available in FPGAs.

Future Scope

These update algorithms can be analytically derived for different windows like Hanning, Hamming, etc.

Also DA based technique can be extended for LUT –less implementation which will result in lesser hardware utilization.

List of Publications

[1] S. Akhter, V. Karwal, and R. C. Jain, “Improved Algorithm for ODCT Computation of a Running Data Sequence,” Journal of Electrical and Computer Engineering, vol. 2012, Article ID 879626, 10 pages, 2012. doi:10.1155/2012/879626

[2] S.Akhter, V.Karwal, R.C.Jain, “Fast Windowed Update Algorithm for ODCT Computation”, International Journal of Electronics Letters (2014), Taylor & Francis, doi:10.1080/21681724.2014.880989.

[3] S.Akhter, V. Karwal, R.C.Jain. "Implementation of Odd Discrete Cosine Transform (ODCT-II) using Distributed Arithmetic approach," Engineering (NUiCONE), 2012 Nirma University International Conference on Engineering, 6-8 Dec. 2012, pp.1-6
doi: 10.1109/NUICONE.2012.6493252.

[4] S. Akhter, V. Karwal, and R. C. Jain,” Implementation of Rectangular Windowed Odd Discrete Cosine Transform Update Algorithm Using Distributed Arithmetic Approach”, 2013 International Conference on Signal Processing and Communication (ICSC-2013) at JIIT Noida, India, 12-14 Dec-2013, pp.381-386.

[5] S.Akhter, V.Karwal, and R. C. Jain, “Real-Time *r*-point ODCT Computation Algorithm for Split-Triangular Windowed Data”, (Submitted).

REFERENCES

[1] K. R. Rao and P. Yip, Discrete Cosine Transform: Algorithms, Advantages, Applications, *Academic Press*, 1990.

[2] G. Strang, “The discrete cosine transform: SIAM review, Vol.41, No.1, pp-135-147, 1999.

[3] P.Yip & K.R.Rao, “On the shift property of DCT's and DST's*”* *IEEE Trans. on Acoustic Speech Signal Process*ing Vol.35, No.3, pp. 404-406,1987.

[4] N.R.Murthy & M.N.S. Swamy, “On the computation of running discrete cosine and sine transform” *IEEE Transactions on Signal Processing*, Vol.40, No. 6, pp. 1430-1437, 1992.

[5] B.G.Sherlock & Y.P.Kakad, “Windowed discrete cosine and sine transforms for shifting data” *Journal of Signal Processing, Elsevier*, Vol. 81, No. 7, pp.1465-1478, 2001.

[6] J.Xi & J.K.Chicharo, “Computing running DCTs and DSTs based on their second-order shift properties” *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, Vol.47, No.5, pp. 779-783, 2000.

[7] V. Karwal, B.G.Sherlock & Y.P.Kakad, “*Windowed DST-independent discrete cosine transforms for shifting data”, Proceedings of 20th International Conference on Systems Engineering*, Coventry, U.K, pp.252-257, 2009.

[8] V. Karwal, B.G.Sherlock & Y.P.Kakad, “Independently updating the DCT and DST for shifting windowed data”, *Journal of the Franklin Institute*, Elsevier Vol. 350, No.6, pp. 1345-1357, 2013.

[9] V. Karwal, "Discrete cosine transform-only and discrete sine transform-only windowed update algorithms for shifting data with hardware implementation" Ph.D. Dissertation, The University of North Carolina at Charlotte, ISBN: 978-1-109-34326-7, 2009.

[10] L.Wu, "Comment on the shift property of DCT's and DST's," *IEEE Trans. on Acoustic Speech Signal Processing,* Vol 38, No.1, pp. 186-188, 1990.

[11] R. K. Chivukula and Y. A. Reznik, "Fast computing of discrete cosine and sine transforms of type VI and VII," in Proceedings of SPIE, 34th Applications of Digital Image Processing, Vol. 8135, August 2011.

[12] J.Han, A.Saxena, & K.Rose, "Towards jointly optimal spatial prediction and adaptive transform in video/image coding" 2010 IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP), pp. 726-729, 2010.

[13] B. Bross, W-J. Han, G. Sullivan, J. Ohm, & T. Wiegland, "High Efficiency Video Coding (HEVC) text specification draft 6", ITU-T and ISO/IEC JCTVC-H1003, 2012.

[14] A.Saxena, F.C.Fernandes & Y.A.Reznik, "Fast Transforms for Intra-predication-based Image and Video Coding", 2013 Data Compression Conference, pp 13-22, 2013.

[15] B.G.Sherlock, Y.P.Kakad & A.Shukla, "Rapid update of odd DCT and DST for real-time signal processing., Proc. of SPIE Vol.5809, pp-464-471, 2005.

[16] Stanley A. White, "Applications of Distributed Arithmetic to Digital Signal Processing: A Tutorial Review," IEEE ASSP Magazine, pp. 4-19. 1989.

[17] S. Yu & E. E. Swartzlander, "DCT implementation with distributed arithmetic," *IEEE Transactions on Computers*, Vol. 50, No. 9, pp. 985–991, 2001.

[18] R. Ramesh & R.Nathiy, " Realization of FIR Filter using Modified Distributed Arithmetic Architecture", *Signal & Image Processing : An International Journal (SIPIJ)* Vol.3, No.1, pp. 83-94, 2012.

[19] C.G Wei & Wang Feng-ying, " The Implementation of FIR Low-pass Filter based on FGPA and DA", 2013 Fourth International Conference on Intelligent Control and Information Processing (ICICIP), Beijing,China, pp 604-608, 2012.

[20] S. Akhter, V. Karwal, & R. C. Jain, "Improved Algorithm for ODCT Computation of a Running Data Sequence," *Journal of Electrical and Computer Engineering,* vol. 2012, Article ID 879626, 10 pages, 2012. doi:10.1155/2012/879626.

[21] S.Akhter, V.Karwal, & R.C.Jain, "Fast Windowed Update Algorithm for ODCT Computation", *International Journal of Electronics Letters, Taylor & Francis*, doi:10.1080/21681724.2014.880989, 2014.