

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/3343693>

Convolution Using Discrete Sine and Cosine Transforms

Article in IEEE Signal Processing Letters · August 2007

DOI: 10.1109/LSP.2006.891312 · Source: IEEE Xplore

CITATIONS

23

READS

1,040

3 authors:



V.G. Reju

Nanyang Technological University

26 PUBLICATIONS 244 CITATIONS

SEE PROFILE



Soo Ngee Koh

Nanyang Technological University

123 PUBLICATIONS 1,177 CITATIONS

SEE PROFILE



I.Y. Soon

Nanyang Technological University

70 PUBLICATIONS 1,017 CITATIONS

SEE PROFILE

Convolution Using Discrete Sine and Cosine Transforms

V. G. Reju, Soo Ngee Koh, *Senior Member, IEEE*, and Ing Yann Soon

Abstract—In this paper, we derive a relation for the circular convolution operation in the discrete sine and cosine transform domains. The transform coefficients are either symmetric or asymmetric and hence we need to calculate only half of the total coefficients. Since fast algorithms are available for the computation of discrete sine and cosine transforms, the proposed method is an alternative to the discrete Fourier transform method for filtering applications.

Index Terms—Discrete transforms, filtering, Fourier transforms.

I. INTRODUCTION

THE convolution multiplication property of discrete Fourier transform (DFT) is well known. For discrete cosine transforms (DCTs) and discrete sine transforms (DSTs), called discrete trigonometric transforms (DTTs), such a nice property does not exist. Martucci [1], [2] derived the convolution multiplication properties of all the families of DCTs and DSTs, in which the convolution is a special type called a symmetric convolution. For the symmetric convolution, the sequences to be convolved must be either symmetric or asymmetric. The general form of the equation for the symmetric convolution in the DTT domain is $s(n) * h(n) = T_c^{-1} \{T_a \{s(n)\} \times T_b \{h(n)\}\}$, where $s(n)$ and $h(n)$ are the input sequences, \times represents the element-wise multiplication operation, and $*$ represents the convolution operation. The type of the transforms T_a , T_b , and T_c to be used depends on the type of the symmetry of the sequences to be convolved (see [1] and [2] for more details). In [1], [2], and [3], it is also shown that by proper zero-padding of the sequences, symmetric convolution can be used to perform linear convolution.

In this letter, we derive a relation for circular convolution in the DTT domain. The advantage of this new relation is that the input sequences to be convolved need not be symmetrical or asymmetrical, and the computational time is less than that of the symmetric convolution method.

II. CONVOLUTION IN THE DTT DOMAIN

The DSTs and DCTs used in this paper are the same as those used in [1] and [2], which are given as follows:

$$S_{C_1}(k) = 2 \sum_{n=0}^N \zeta_n s(n) \cos\left(\frac{\pi k n}{N}\right) \quad k = 0, 1, \dots, N \quad (1)$$

Manuscript received August 11, 2006; revised October 30, 2006. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Hakan Johansson.

The authors are with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore 639798. (e-mail: PE2630688@ntu.edu.sg; esnkoh@ntu.edu.sg; eiyssoon@ntu.edu.sg).

Digital Object Identifier 10.1109/LSP.2006.891312

$$S_{C_2}(k) = 2 \sum_{n=0}^{N-1} s(n) \cos\left(\frac{\pi k(2n+1)}{2N}\right) \quad k = 0, 1, \dots, N-1 \quad (2)$$

$$S_{S_1}(k) = 2 \sum_{n=1}^{N-1} s(n) \sin\left(\frac{\pi k n}{N}\right) \quad k = 1, 2, \dots, N-1 \quad (3)$$

$$S_{S_2}(k) = 2 \sum_{n=0}^{N-1} s(n) \sin\left(\frac{\pi k(2n+1)}{2N}\right) \quad k = 1, 2, \dots, N$$

$$\zeta_n = \begin{cases} \frac{1}{2} & n = 0 \text{ or } N \\ 1 & n = 1, 2, \dots, N-1 \end{cases} \quad (4)$$

where $S_{C_1}(k)$, $S_{C_2}(k)$, $S_{S_1}(k)$, $S_{S_2}(k)$ denotes the type I even DCT (DCTIe) coefficients, type II even DCT (DCTIIe) coefficients, type I even DST (DSTIe) coefficients, and type II even DST (DSTIIe) coefficients, respectively, of the sequence $s(n)$.

The sequences to be convolved are $s(n)$ and $h(n)$ of length N , so that the convolved signal is $s(n) \star h(n)$, where \star represents the circular convolution operation. The DFT of $s(n)$ is given by [4]

$$S(k) = \sum_{n=0}^{N-1} s(n) e^{-j2\pi k n / N} \quad k = 0, 1, \dots, N-1. \quad (5)$$

Multiplying (5) by $2e^{-j\pi k / N}$, we will get

$$2e^{-j\pi k / N} S(k) = 2 \sum_{n=0}^{N-1} s(n) \left(\cos\left(\frac{\pi k(2n+1)}{N}\right) - j \sin\left(\frac{\pi k(2n+1)}{N}\right) \right). \quad (6)$$

Comparing (2) and first term of (6), it can be observed that $2 \sum_{n=0}^{N-1} s(n) \cos(\pi k(2n+1)/N)$ is the decimated and asymmetrically extended version of (2) with index $k = 0 : N-1$. Similarly comparing (4) and second term of (6), it can be observed that $2 \sum_{n=0}^{N-1} s(n) \sin(\pi k(2n+1)/N)$ is the decimated and symmetrically extended version of (4) with index $k = 1 : N$. For convenient element-wise operation in the following equations, append 0 at $k = N$ for the resulting sequence of the first term, and at $k = 0$ for the resulting sequence of the second term so as to obtain the sequences $\check{S}_{C_2}(k)$ and $\check{S}_{S_2}(k)$, respectively, of length $N+1$. Hence, (6) becomes

$$2e^{-j\pi k / N} S(k) = \check{S}_{C_2}(k) - j \check{S}_{S_2}(k). \quad (7)$$

A similar equation can be written for $h(n)$ as

$$2e^{-j\pi k / N} H(k) = \check{H}_{C_2}(k) - j \check{H}_{S_2}(k). \quad (8)$$

Element-wise multiplication of (7) and (8) gives

$$S(k)H(k) = \frac{1}{4} e^{j2\pi k/N} \left\{ \left(\check{S}_{C_2}(k) \check{H}_{C_2}(k) - \check{S}_{S_2}(k) \check{H}_{S_2}(k) \right) - j \left(\check{S}_{S_2}(k) \check{H}_{C_2}(k) + \check{S}_{C_2}(k) \check{H}_{S_2}(k) \right) \right\}. \quad (9)$$

Taking the real part of the inverse discrete Fourier transform of (9), we will get (10), as shown at the bottom of the page. Since

$$\begin{aligned} \check{S}_{C_2}(N) &= \check{H}_{C_2}(N) = \check{S}_{S_2}(N) = \check{H}_{S_2}(N) \\ &= \check{S}_{S_2}(0) = \check{H}_{S_2}(0) = 0 \end{aligned}$$

the summation range of the first term in (10) is changed from $k = 0 : N - 1$ to $k = 0 : N$ and that of the second term to $k = 1 : N - 1$.

Comparing (1), (3), and (10), it can be observed that without the scaling factor $1/4N$, the first term in (10) is the decimated and symmetrically extended version of DCTIe coefficients, $C_1\{T_1\}$, and the second term is the decimated and asymmetrically extended version of the DSTIe coefficients, $S_1\{T_2\}$, except for the shift in the resultant sequences by one sample and the absences of the constants ζ_n and 2. Considering these constants and using the fact that inverse of DCTIe is same as DCTIe and inverse of DSTIe is same as DSTIe, except for a scaling factor $2N$ [1], [2], the above equation can be rewritten as

$$\begin{aligned} s(n) \star h(n) &= \frac{1}{4} \left(\check{C}_1^{-1} \left\{ \xi_k \left(\check{C}_2\{s\} \times \check{C}_2\{h\} - \check{S}_2\{s\} \times \check{S}_2\{h\} \right) \right\} \right. \\ &\quad \left. + \check{S}_1^{-1} \left\{ \check{S}_2\{s\} \times \check{C}_2\{h\} + \check{C}_2\{s\} \times \check{S}_2\{h\} \right\} \right) \quad (11) \end{aligned}$$

where

$$\xi_k = \begin{cases} 2, & k = 0 \text{ or } N \\ 1, & k = 1, 2, \dots, N - 1. \end{cases}$$

The steps for computing (11) can be explained as follows.

- Compute $\check{C}_2\{s\}$ and $\check{S}_2\{s\}$ as

$$\begin{aligned} [\check{C}_2]_{k,n} &= 2 \cos \left(\frac{\pi k(2n+1)}{N} \right) \quad k, n = 0, 1, \dots, N-1 \\ [\check{S}_2]_{k,n} &= 2 \sin \left(\frac{\pi k(2n+1)}{N} \right) \quad \begin{matrix} k = 1, 2, \dots, N \\ n = 0, 1, \dots, N-1 \end{matrix} \\ \check{C}_2\{s\} &= \begin{bmatrix} \check{S}_{C_2} \end{bmatrix}_{(N+1) \times 1} \\ &= \begin{bmatrix} 0 & 0 & \dots & 0 & 0 \end{bmatrix}_{(N+1) \times N} \begin{bmatrix} s \end{bmatrix}_{N \times 1} \end{aligned}$$

$$\begin{aligned} \check{S}_2\{s\} &= \begin{bmatrix} \check{S}_{S_2} \end{bmatrix}_{(N+1) \times 1} \\ &= \begin{bmatrix} 0 & 0 & \dots & 0 & 0 \\ & \check{S}_2 & & & \end{bmatrix}_{(N+1) \times N} \begin{bmatrix} s \end{bmatrix}_{N \times 1}. \end{aligned}$$

Alternatively, $\check{C}_2\{s\}$ and $\check{S}_2\{s\}$ can be found from the sequences $C_2\{s\}$ and $S_2\{s\}$ respectively after decimation and extending them asymmetrically and symmetrically as shown in Fig. 1. The square markings in the figure show the appended zeros.

- Similarly compute $\check{C}_2\{h\}$ and $\check{S}_2\{h\}$.
- Compute $T_1(k)$ and $T_2(k)$ as

$$\begin{aligned} [T_1]_{(N+1) \times 1} &= [\check{S}_{C_2}] \times [\check{H}_{C_2}] - [\check{S}_{S_2}] \times [\check{H}_{S_2}] \\ [T_2]_{(N+1) \times 1} &= [\check{S}_{S_2}] \times [\check{H}_{C_2}] + [\check{S}_{C_2}] \times [\check{H}_{S_2}] \end{aligned}$$

- Multiply $T_1(0)$ and $T_1(N)$ by $\xi_k = 2$ and keep all other elements the same to obtain the new sequence $T'_1(k)$ of length $N + 1$.
- Discard $T_2(0)$ and $T_2(N)$ to obtain the new sequence $T'_2(k)$ of length $N - 1$.
- Compute $\check{C}_1^{-1}\{T'_1\}$ and $\check{S}_1^{-1}\{T'_2\}$ as

$$\begin{aligned} [\check{C}_1]_{k,n} &= 2\zeta_n \cos \left(\frac{2\pi kn}{N} \right) \quad k, n = 0, 1, \dots, N \\ [\check{S}_1]_{k,n} &= 2 \sin \left(\frac{2\pi kn}{N} \right) \quad k, n = 1, 2, \dots, N-1 \\ \check{C}_1^{-1}\{T'_1\} &= [\check{T}'_{1C_1^{-1}}]_{(N+1) \times 1} \\ &= \frac{1}{2N} [\check{C}_1]_{(N+1) \times (N+1)} [T'_1]_{(N+1) \times 1} \\ \check{S}_1^{-1}\{T'_2\} &= [\check{T}'_{2S_1^{-1}}]_{(N-1) \times 1} \\ &= \frac{1}{2N} [\check{S}_1]_{(N-1) \times (N-1)} [T'_2]_{(N-1) \times 1}. \end{aligned}$$

Alternatively, $\check{C}_1^{-1}\{T'_1\}$ and $\check{S}_1^{-1}\{T'_2\}$ can be found from the sequences $C_1\{T'_1\}$ and $S_1\{T'_2\}$ after scaling, decimation, and extending them symmetrically and asymmetrically, as shown in Fig. 1.

- Discard the first element of $\check{T}'_{1C_1^{-1}}$, append one zero at the end of $\check{T}'_{2S_1^{-1}}$, add the resultant sequences together and scale them with the scaling factor $1/4$ to obtain the convolved signal as

$$s(n) \star h(n) = \frac{1}{4} \left(\check{T}'_{1C_1^{-1}}(1:N) + [\check{T}'_{2S_1^{-1}}; 0] \right).$$

$$\begin{aligned} \text{real} \left(\frac{1}{N} \sum_{k=0}^{N-1} S(k)H(k) e^{j2\pi kn/N} \right) &= \frac{1}{4N} \sum_{k=0}^N \underbrace{\left(\check{S}_{C_2}(k) \check{H}_{C_2}(k) - \check{S}_{S_2}(k) \check{H}_{S_2}(k) \right)}_{T_1(k)} \cos \left(\frac{2\pi k(n+1)}{N} \right) \\ &\quad + \frac{1}{4N} \sum_{k=1}^{N-1} \underbrace{\left(\check{S}_{S_2}(k) \check{H}_{C_2}(k) + \check{S}_{C_2}(k) \check{H}_{S_2}(k) \right)}_{T_2(k)} \sin \left(\frac{2\pi k(n+1)}{N} \right). \quad (10) \end{aligned}$$

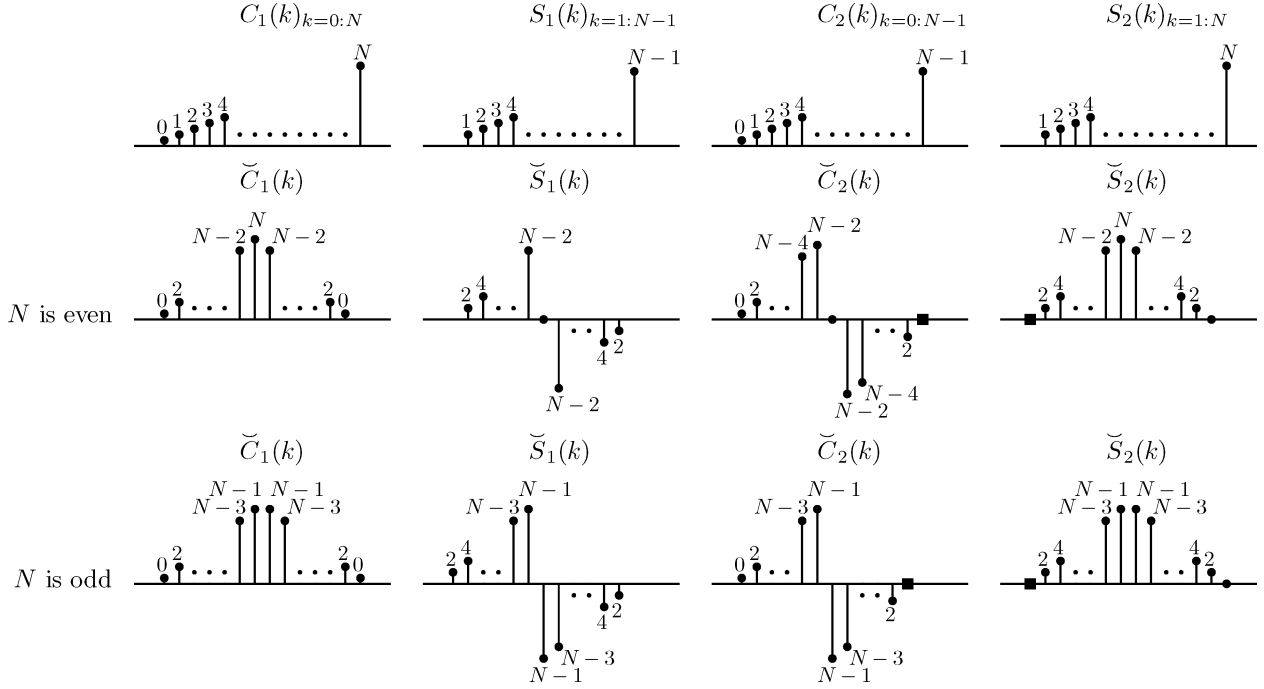


Fig. 1. Generation of $\check{C}_1(k)$, $\check{S}_1(k)$, $\check{C}_2(k)$, and $\check{S}_2(k)$ from $C_1(k)$, $S_1(k)$, $C_2(k)$, and $S_2(k)$, respectively, after decimation and symmetric or asymmetric extension. Black squares represent the appended zeros to make the length of the sequences to $N + 1$ for element-wise operation, to be discussed later.

TABLE I
COMPUTATIONAL COST COMPARISON

Method used	Number of DDT coefficients to be calculated				\times	+/-
	DCTIe	DSTIe	DCTIle	DSTIle		
Symmetric convolution	$M + 1$	0	$2M$	$2M$	$2M$	$M - 1$
Proposed	$\lfloor \frac{R}{2} \rfloor + 1$	$\lfloor \frac{R-1}{2} \rfloor$	$2\lceil \frac{R}{2} \rceil$	$2\lfloor \frac{R}{2} \rfloor$	$3\lceil \frac{R}{2} \rceil + \lfloor \frac{R}{2} \rfloor - 2$	$\lfloor \frac{R-1}{2} \rfloor + 2\lceil \frac{R}{2} \rceil - 2$

$\lfloor y \rfloor$ and $\lceil y \rceil$ round y to the nearest integer towards minus infinity and plus infinity respectively.

Filter length = Q , valid output samples per block = P , $M = P + 2Q - 1$ and $R = P + Q - 1$.

The Matlab code for the algorithm is available at <http://www.ntu.edu.sg/eee/home/esnkoh/>

III. DISCUSSION

It is interesting to note that in symmetric convolution [1], [2], the time sequences are symmetric or asymmetric whereas in (11), the DTT coefficients are symmetric or asymmetric except for the appended zeros in the sequences $\check{C}_2(k)$ and $\check{S}_2(k)$. Utilizing the fact that any signal can be split into symmetric and asymmetric sequences, in [1]–[3] and [5], it was shown that the symmetric convolution can be used for linear convolution. For example, if a long sequence $x(n)$ is to be convolved with filter coefficients $h(n)$ of length Q , then segment the signal $x(n)$ into blocks of length M with overlap $2Q - 1$. Let $x_b(n)$ be the b^{th} block and $h'(n)$ be the filter coefficients of length M after appending $M - Q$ zeros, then calculate $w_b(n) = C_1^{-1} \{T_c - T_s\}$; where $T_c(0 : M - 1) = C_2 \{x_b\} \times C_2 \{h'\}$, $T_c(M) = 0$, $T_s(1 : M) = S_2 \{x_b\} \times S_2 \{h'\}$, and $T_s(0) = 0$. The $P = M - 2Q + 1$ samples of $w_b(n)$ after removing Q samples from both sides of $w_b(n)$ will be the valid linear convolution coefficients. Hence, symmetric convolution can be used for linear

convolution. However, it can be seen that, since the block length of the input sequence is M , the length of the DTTs to be calculated are also of length M or $M + 1$ (M for C_2 and S_2 , $M + 1$ for C_1^{-1}) and the valid outputs will be of length $P = M - 2Q + 1$. Since (11) is for circular convolution, similar to DFT, by proper zero padding, it can be used for linear convolution also. For example, as in the previous case, to filter a long sequence $x(n)$ with filter coefficients $h(n)$ of length Q , segment the signal $x(n)$ into blocks of length P and append each block with $Q - 1$ zeros to get blocks of length $R = P + Q - 1$. Similarly, append $P - 1$ zeros to the filter coefficients to make its length equal to R . Then apply (11), and overlap and add the resultant output blocks to get the filtered signal. While computing, because of the symmetry of the DTT coefficients in (11), it is sufficient to calculate only half of the total number of coefficients. The remaining half is the symmetrically extended version of the first half. Also, for the second part of (11), the same DTT coefficients \check{S}_{C_2} , \check{H}_{C_2} , \check{S}_{S_2} , and \check{H}_{S_2} that used for the first part can be used. Similarly, for the element-wise multiplication, because of the symmetry of the DTT coefficients, only half of the coefficients need to be multiplied. The other half will be the same

as the first half with or without the sign changes. Likewise, for the addition and subtraction operations also, only half of the elements need to be added or subtracted. Moreover, unlike the symmetric convolution method, the length of the DTTs to be calculated here are $R + 1$, R , or $R - 1$ ($R + 1$ for \check{C}_1 , R for \check{C}_2 and \check{S}_2 , $R - 1$ for \check{S}_1), which is smaller than that for the symmetric convolution method. The computational cost per DTT coefficient will decrease as DTT length decreases. Hence, the computational time of (11) is less than that of the symmetric convolution method. Table I summarizes the computational cost for the two methods in filtering application, neglecting the cost involved for sign change, symmetric or asymmetric extension of the DTT coefficients and multiplication by the scaling factors.

IV. CONCLUSION

We have derived a relation for the circular convolution in the discrete trigonometric transform domain. The computational

time of the method is lower than that of the symmetric convolution method for filtering applications. Because of the availability of fast algorithms for DTTs, the new relation is an alternative to the DFT method for filtering.

REFERENCES

- [1] S. A. Martucci, "Symmetric convolution and the discrete sine and cosine transforms," *IEEE Trans. Signal Process.*, vol. 42, no. 5, pp. 1038–1051, May 1994.
- [2] —, "Symmetric Convolution and the Discrete Sine and Cosine Transforms: Principles and Applications," Ph.D. dissertation, Georgia Inst. Technol., Atlanta, 1993.
- [3] X. Zou, S. Muramatsu, and H. Kiya, "Generalized overlap-add and overlap-save methods using discrete sine and cosine transforms for FIR filtering," in *Proc. ICSP*, 1996, vol. 1, pp. 91–94.
- [4] Proakis and G. John, *Digital Signal Processing : Principles, Algorithms, and Applications*. Englewood Cliffs, NJ: Prentice-Hall, 1996.
- [5] S. A. Martucci, "Digital filtering of images using the discrete sine or cosine transform," in *Proc. SPIE*, 1994, vol. 2308, no. 2, pp. 1322–1333.