

DIRECT N-POINT DCT COMPUTATION FROM THREE ADJACENT N/3-POINT DCT COEFFICIENTS

Soo-Chang Pei and Meng-Ping Kao

Department of Electrical Engineering
National Taiwan University
Taipei, Taiwan 10617, ROC
E-mail: pei@cc.ee.ntu.edu.tw , Fax: 886-2-23671909

ABSTRACT

An efficient method for computing a length- N DCT given three consecutive length- $N/3$ DCTs is proposed. This method differs from previous ones in that it reduces considerable arithmetic operations and uses only length- $N/3$ DCTs instead of length- N DCTs. We also find its great applications in fractional scaling of a DCT-based image by the factor of $N/2^{\alpha}3^{\beta}$. This would be very useful in HDTV standard, whose display size is usually 16:9. The comparison with conventional methods is provided in this paper.

1. INTRODUCTION

Since “Discrete Cosine Transform (DCT)” was first introduced in 1974, it has been widely used in various fields among digital signal processing (DSP). For example, it is the foundation stone of many prevailing image and video compression techniques, such as JPEG, MPEGs and H.26x. Moreover, most of the images and videos nowadays are stored in the DCT domain through the above techniques. The question, however, arises that how can we directly manipulate or process such a compressed media stored in DCT domain.

To avoid unnecessary computations in decompression and recompression, a number of algorithms have been proposed to complete the operations directly in the DCT domain. Among the various operations, scaling of an image, i.e. interpolation or decimation is probably the most common ones we might encounter in relevant applications. For example, Park [4] has proposed an algorithm using the symmetric convolution property of DCT. Zhao [5] utilizes the relationship between linear transforms in spatial domain and those in DCT domain to develop an alternative approach

In compressed image or video domain, it is necessary to construct a long DCT sequence from several short DCT sequences or to decompose a long DCT sequence into

several short DCT sequences. Since Skodras [1] has proposed a corresponding method for the case of two sequences, we now focus on the case of three consecutive sequences in this paper. Combining with Skodras’s method, many DCT domain operations, such as fractional scaling by the factor of $N/2^{\alpha}3^{\beta}$, are now realizable in a simple and efficient way.

2. PROPOSED APPROACH

Let us readdress the problem as follows. Given three sequences of the same length $N/3$, i.e. A_k , B_k and C_k , which are the DCT coefficients of three consecutive sequences a_n , b_n and c_n respectively. Suppose that a_n , b_n and c_n together form the sequence x_n of length N , i.e. $a_n = x_n$, $b_n = x_{n+(N/3)}$, $c_n = x_{n+(2N/3)}$ for $n = 0, 1, \dots, N/3 - 1$, how can we find an efficient way to compute X_k from A_k , B_k and C_k ?

The traditional way to solve this problem is depicted in Fig.1. According to this scheme, three $N/3$ -point DCTs are required followed by a N -point DCT. However, even if the fast DCT structure [3] is used, the computation complexity is approximately $N \log_2 N$ proportional to N , which means the N -point DCT in Fig.1 will always dominate the whole computational load.

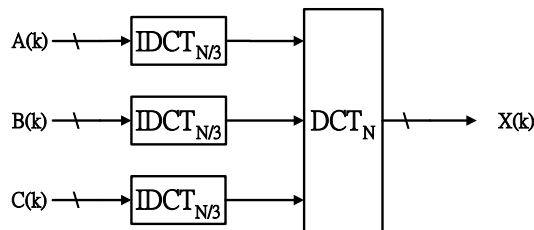


Fig.1 Schematic representation of the traditional approach

In order to modify the traditional approach, we propose a new structure based on DCT decomposition to avoid the N -point DCT computation. In other words, we

use only $N/3$ -point DCTs to find X_k , which is much more efficient than before.

Now we get straight to investigate the key concept of this new structure. First of all, the normalized forward DCT-II of length- N sequence x_n is given as follows,

$$X_k = \sqrt{\frac{2}{N}} \varepsilon_k \sum_{n=0}^{N-1} x_n \cos \frac{(2n+1)k\pi}{2N}, \quad k = 0, 1, \dots, N-1 \quad (1)$$

and the corresponding inverse DCT is,

$$x_n = \sqrt{\frac{2}{N}} \sum_{k=0}^{N-1} \varepsilon_k X_k \cos \frac{(2n+1)k\pi}{2N}, \quad n = 0, 1, \dots, N-1 \quad (2)$$

where

$$\varepsilon_k = \begin{cases} 1/\sqrt{2}, & \text{for } k = 0 \\ 1, & \text{for } k \neq 0 \end{cases} \quad (3)$$

Note that $\varepsilon_{3k} = \varepsilon_k$ and $\varepsilon_{3k+1} = \varepsilon_{3k+2} = 1$. We now decompose X_k to three equal-length parts, X_{3k} , $X_{3k+1} + X_{3k-1}$ and $X_{3k+2} + X_{3k-2}$ separately.

2.1. X_{3k} COEFFICIENTS CALCULATION

$$\begin{aligned} X_{3k} &= \sqrt{\frac{2}{N}} \varepsilon_{3k} \sum_{n=0}^{N-1} x_n \cos \frac{(2n+1)3k\pi}{2N} \\ &= \sqrt{\frac{2}{N}} \varepsilon_k \left\{ \sum_{n=0}^{(N/3)-1} a_n \cos \frac{(2n+1)k\pi}{2(N/3)} \right. \\ &\quad + \sum_{n=(N/3)-1}^0 b_{N/3-1-n} \cos \frac{(2(N/3-1-n+N/3)+1)k\pi}{2(N/3)} \\ &\quad \left. + \sum_{n=0}^{(N/3)-1} c_n \cos \left(\frac{(2n+1)k\pi}{2(N/3)} + 2k\pi \right) \right\} \\ &= \sqrt{\frac{1}{3}} \sqrt{\frac{2}{N/3}} \varepsilon_k \left\{ \sum_{n=0}^{(N/3)-1} a_n \cos \frac{(2n+1)k\pi}{2(N/3)} \right. \\ &\quad \left. + \sum_{n=0}^{(N/3)-1} b'_n \cos \frac{(2n+1)k\pi}{2(N/3)} + \sum_{n=0}^{(N/3)-1} c_n \cos \frac{(2n+1)k\pi}{2(N/3)} \right\} \\ &= \sqrt{\frac{1}{3}} \{A_k + B'_k + C_k\}, \quad k = 0, 1, \dots, \frac{N}{3} - 1 \end{aligned} \quad (4)$$

where $a_n = x_n$, $b_n = x_{n+(N/3)}$, $c_n = x_{n+(2N/3)}$, and

$$b'_n = b_{(N/3)-1-n} = x_{(2N/3)-1-n}, \quad (5)$$

for $n = 0, 1, \dots, (N/3)-1$

As a consequence,

$$B'_k = DCT_{N/3} \{b'_n\} = DCT_{N/3} \{b_{(N/3)-1-n}\} = (-1)^k B_k, \quad (6)$$

for $k = 0, 1, \dots, (N/3)-1$

2.2. $X_{3k+1} + X_{3k-1}$ COEFFICIENTS CALCULATION

$$\begin{aligned} X_{3k+1} + X_{3k-1} &= \sqrt{\frac{2}{N}} \sum_{n=0}^{N-1} x_n \left\{ \cos \frac{(2n+1)(3k+1)\pi}{2N} \varepsilon_{3k+1} \right. \\ &\quad \left. + \cos \frac{(2n+1)(3k-1)\pi}{2N} \varepsilon_{3k-1} \right\} \\ &= \sqrt{\frac{2}{N}} \sum_{n=0}^{N-1} x_n 2 \cos \frac{(2n+1)\pi}{2N} \cos \frac{(2n+1)3k\pi}{2N} \\ &= 2\sqrt{\frac{2}{N}} \left\{ \sum_{n=0}^{(N/3)-1} \left(a_n \cos \frac{(2n+1)\pi}{2N} \right) \cos \frac{(2n+1)k\pi}{2(N/3)} \right. \\ &\quad + \sum_{n=0}^{(N/3)-1} \left(b_n \cos \frac{(2(n+N/3)+1)\pi}{2N} \right) \cos \frac{(2(n+N/3)+1)k\pi}{2(N/3)} \\ &\quad \left. + \sum_{n=0}^{(N/3)-1} \left(c_n \cos \frac{(2(n+2N/3)+1)\pi}{2N} \right) \cos \frac{(2(n+2N/3)+1)k\pi}{2(N/3)} \right\} \end{aligned}$$

Assume that

$$\theta_n = \frac{(2n+1)\pi}{2N}, \quad (7)$$

then

$$\begin{aligned} &2\sqrt{\frac{2}{N}} \frac{1}{\varepsilon_k} \sqrt{\frac{N/3}{2}} \left\{ \sqrt{\frac{2}{N/3}} \varepsilon_k \sum_{n=0}^{(N/3)-1} (a_n \cos \theta_n) \cos \frac{(2n+1)k\pi}{2(N/3)} \right. \\ &\quad + \sqrt{\frac{2}{N/3}} \varepsilon_k \sum_{n=0}^{(N/3)-1} \left(b'_n \cos \left(\frac{(2n+1)\pi}{2N} - \frac{2\pi}{3} \right) \right) \cos \frac{(2n+1)k\pi}{2(N/3)} \\ &\quad \left. + \sqrt{\frac{2}{N/3}} \varepsilon_k \sum_{n=0}^{(N/3)-1} \left(c_n \cos \left(\frac{(2n+1)\pi}{2N} + \frac{2\pi}{3} \right) \right) \cos \frac{(2n+1)k\pi}{2(N/3)} \right\} \\ &= \sqrt{\frac{1}{3}} \frac{1}{\varepsilon_k} \left\{ DCT_{N/3} \{2a_n \cos \theta_n\} + DCT_{N/3} \{ \sqrt{3}b'_n \sin \theta_n - b'_n \cos \theta_n \} \right. \\ &\quad \left. + DCT_{N/3} \{ -\sqrt{3}c_n \sin \theta_n - c_n \cos \theta_n \} \right\} \\ &= \sqrt{\frac{1}{3}} \frac{1}{\varepsilon_k} DCT_{N/3} \{ (2a_n - d_n) \cos \theta_n + \sqrt{3}e_n \sin \theta_n \} \\ &= \sqrt{\frac{1}{3}} \frac{1}{\varepsilon_k} DCT_{N/3} \{ IDCT_{N/3} \{ 2A_k - D_k \} \cos \theta_n + IDCT_{N/3} \{ E_k \} \sqrt{3} \sin \theta_n \} \end{aligned} \quad (8)$$

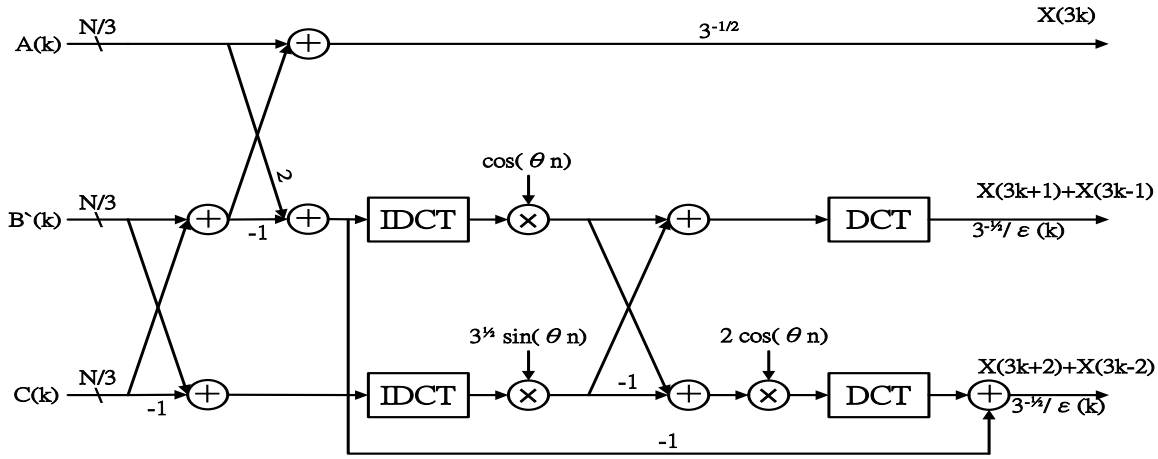


Fig.2 Flow graph of the proposed algorithm

where

$$d_n = b'_n + c_n \quad (9)$$

$$e_n = b'_n - c_n \quad (10)$$

As a consequence,

$$D_k = (-1)^k B_k + C_k \quad (11)$$

$$E_k = (-1)^k B_k - C_k \quad (12)$$

2.3. $X_{3k+2}+X_{3k-2}$ COEFFICIENTS CALCULATION

The derivation of $X_{3k+2} + X_{3k-2}$ is similar to that of $X_{3k+1} + X_{3k-1}$. Therefore, we summarize the results as follows,

$$X_{3k+2} + X_{3k-2} = \sqrt{\frac{1}{3}} \frac{1}{\epsilon_k} DCT_{N/3} \left\{ IDCT_{N/3} \left\{ 2A_k - D_k \right\} \cos 2\theta_n - IDCT_{N/3} \left\{ E_k \right\} \sqrt{3} \sin 2\theta_n \right\} \quad (13)$$

Further more,

$$\cos 2\theta_n = \cos(\theta_n + \theta_n) = 2 \cos^2 \theta_n - 1 \quad (14)$$

$$\sin 2\theta_n = \sin(\theta_n + \theta_n) = 2 \cos \theta_n \sin \theta_n \quad (15)$$

$$\begin{aligned} & X_{3k+2} + X_{3k-2} \\ &= \sqrt{\frac{1}{3}} \frac{1}{\epsilon_k} DCT_{N/3} \left\{ IDCT_{N/3} \left\{ 2A_k - D_k \right\} (2 \cos^2 \theta_n - 1) - IDCT_{N/3} \left\{ E_k \right\} \sqrt{3} (2 \cos \theta_n \sin \theta_n) \right\} \\ &= \sqrt{\frac{1}{3}} \frac{1}{\epsilon_k} \left\{ DCT_{N/3} \left\{ 2 \cos \theta_n \left\{ IDCT_{N/3} \left\{ 2A_k - D_k \right\} \cos \theta_n - IDCT_{N/3} \left\{ E_k \right\} \sqrt{3} \sin \theta_n \right\} \right\} - (2A_k - D_k) \right\} \end{aligned} \quad (16)$$

According to the derivation above, we can construct the entire structure in Fig.2.

2.4. X_{3k+1}, X_{3k+2} COEFFICIENTS CALCULATION

Since we have obtained $X_{3k+1} + X_{3k-1}$ and $X_{3k+2} + X_{3k-2}$, the next question is how to calculate X_{3k+1} and X_{3k+2} from them. According to (1), X_k is an even function of k because both ϵ_k and $\cos\{(2n+1)k\pi/2N\}$ are even functions of k . Therefore, for $k=0$,

$$X_{3k+1} + X_{3k-1} \Big|_{k=0} = X_1 + X_{-1} = 2X_1 \quad (17)$$

$$X_{3k+2} + X_{3k-2} \Big|_{k=0} = X_2 + X_{-2} = 2X_2 \quad (18)$$

Using sequences $X_{3k+1} + X_{3k-1}$ and $X_{3k+2} + X_{3k-2}$ together with the initial conditions X_1 and X_2 , we can derive X_{3k+1} and X_{3k+2} in an iterative approach. The flow graph is illustrated in Fig.3 where $N/3$ is chosen to be 8 for example.

3. COMPUTATIONAL COMPLEXITY

From the proposed structures in Fig.2 and Fig.3, the computation of X_{3k} , $X_{3k+1} + X_{3k-1}$ and $X_{3k+2} + X_{3k-2}$ requires 7 length- $N/3$ vector additions, 3 length- $N/3$ vector multiplications and 4 length- $N/3$ DCTs, assuming that the DCT and the IDCT are of the same complexity. Furthermore, we need 2 more scalar multiplications to accomplish the scaling of $1/\epsilon_k$ in the last stage of Fig.2.

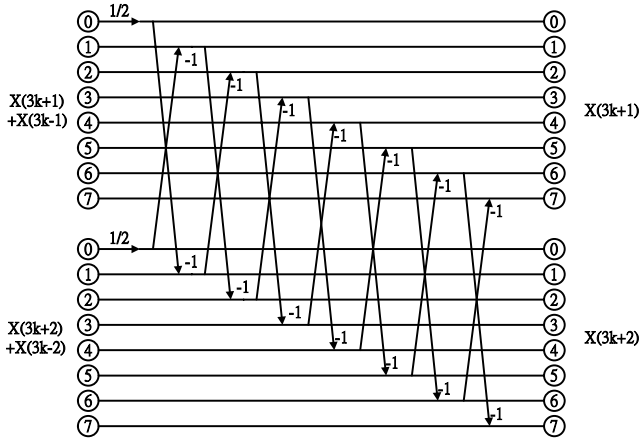


Fig.3 Flow graph of X_{3k+1} and X_{3k+2}

On the other hand, the computation of X_{3k+1} and X_{3k+2} in Fig.3 requires 2 length- $(N/3 - 1)$ vector additions. Since the scaling of $1/2$ in the first stage of Fig.3 could be absorbed into the last stage of Fig.2, there would be no more multiplications here.

For the purpose of simplicity, we choose $N = 2^m \times 3$ in the following comparisons such that the fast DCT could be used to implement length- $N/3$ DCTs, where m is a positive integer. According to [3], the arithmetic complexity of a length- $N/3$ fast DCT is

$$M_{N/3}^{DCT} = M_{2^m}^{DCT} = (m/2) \times (N/3) \quad (19)$$

$$A_{N/3}^{DCT} = A_{2^m}^{DCT} = (3m/2) \times (N/3) - (N/3) + 1 \quad (20)$$

The total numbers of multiplications M_N^P and additions A_N^P of the proposed structure then become

$$M_N^P = 3(N/3) + 2 + 4M_{N/3}^{DCT} = N(2m/3 + 1) + 2 \quad (21)$$

$$A_N^P = 7(N/3) + 2(N/3 - 1) + 4A_{N/3}^{DCT} = N(2m + 5/3) + 2 \quad (22)$$

As for the conventional approach, the complexity is equal to 3 length- $N/3$ DCTs and 1 length- N DCT. The total numbers of multiplications M_N^T and additions A_N^T of the conventional structure then become [2]

$$M_N^T = 3 \times M_{N/3}^{DCT} + M_N^{DCT} = N(m + 1) - 2 \quad (23)$$

$$A_N^T = 3 \times A_{N/3}^{DCT} + A_N^{DCT} = N(3m + 1) + 1 \quad (24)$$

We summarize the computational complexity of these two approaches in Table 1. Note that the computational saving percentage increases with N . Moreover, for $N = 8 \times 3$, which is the standard block dimension for image and video coding, about 30% operations are saved.

N	Mul Save	Add Save	Conventional		Proposed	
			Mul	Add	Mul	Add
12	13%	21%	34	85	30	70
24	27%	30%	94	241	74	186
48	34%	34%	238	625	178	466
96	38%	37%	574	1537	417	1122

Table 1 Comparison of two approaches

4. CONCLUSIONS

We propose an efficient approach to compute a length- N DCT given three consecutive length- $N/3$ DCTs. The operations could be saved up to 30% when the block size is chosen to be 8, and the larger the block size is, the more operations we save. Another advantage is that only length- $N/3$ DCTs and IDCTs are required instead of length- N DCTs. Combining with Skodras's method, scaling a DCT-based image by the factor of $N/2^\alpha 3^\beta$ is easily realizable by cascading α Skodras's structures and β our structures. This provides more flexibility and efficiency in the application of fractional scaling, especially in the rising standard of HDTV whose screen size is usually set to be 16:9.

5. REFERENCES

- [1] Athanassios N. Skodras, "Direct transform to transform computation," IEEE Signal Processing Letters, Volume: 6, Issue: 8, Aug. 1999, Pages: 202 - 204
- [2] Yuk-Hee Chan and Wan-Chi Siu, "Fast radix-3/6 algorithms for the realization of the discrete cosine transform," IEEE International Symposium on Circuits and Systems, Volume: 1, 3-6 May 1992, Pages: 153 - 156 vol.1
- [3] C. W. Kok, "Fast algorithm for computing discrete cosine transform," IEEE Transactions on Signal Processing, Volume: 45, Issue: 3, March 1997, Pages: 757 - 760
- [4] HyunWook Park, YoungSeo Park, and Seung-Kyun Oh, "L/M-fold image resizing in block-DCT domain using symmetric convolution," IEEE Transactions on Image Processing, Volume: 12, Issue: 9, Sept. 2003, Pages: 1016 - 1034
- [5] Yunlong Zhao, M.S. Kankanhall, Tat-Seng Chua, "Fractional scaling of image and video in DCT domain," Proc. of IEEE International Conference on Image Processing, Barcelona, Spain, Volume: 1, Sept. 14-17, 2003, Pages: 185 - 188