Radix-3 Algorithm for the Fast Computation of Forward and Inverse MDCT

Huazhong Shu, Xudong D. Bao, Christine Toumoulin, Limin M. Luo

Radix-3 Algorithm for the Fast Computation of Forward and Inverse MDCT

Huazhong Shu, Xudong Bao, Christine Toumoulin, and Limin Luo

***Abstract*—The modified discrete cosine transform (MDCT) and the inverse MDCT (IMDCT) are two of the most computationally intensive operations in layer III of MPEG audio coding standard. In this letter, we present a radix-3 algorithm for efficiently computing the MDCT and the corresponding IMDCT of a sequence with length $N = 2 \times 3^m$. Comparison of the computational complexity with some known algorithms shows that the proposed approach reduces significantly the number of arithmetic operations.**

***Index Terms*—Modified discrete cosine transform (MDCT), MPEG audio coding, radix-3 algorithm.**

I. Introduction

THE MPEG audio coding standard uses the dynamically windowed modified discrete cosine transform (MDCT) to achieve high quality performance. However, the direct computation of the MDCT in MPEG coding and the inverse MDCT (IMDCT) in MPEG decoding is a computationally intensive task. Therefore, efficient algorithms are of great importance.

Since the introduction of MDCT by Princen *et al.* [1], many fast algorithms have been reported in the literature for computing the MDCT and IMDCT. Chiang and Liu [2] proposed a recursive algorithm, which can be implemented by parallel VLSI filters; this algorithm was further improved by Nikolajevic and Fettweis [3]. Fan *et al.* [4] developed two algorithms based, respectively, on type-II DCT and on the fast Hartley transform for performing the IMDCT quickly. Britanak and Rao [5] developed an efficient implementation of MDCT and IMDCT based on the $N/4$-point type-II DCT and corresponding $N/4$-point type-II DST. Lee [6] then suggested an improvement in the computation speed of this algorithm. Recently, a systematic approach for investigating the MDCT and IMDCT, using a matrix representation, has been presented by Cheng and Hsu [7]. Other recent works on this subject can be found in [8] and [9]. It is worth mentioning that the MDCT is equivalent to the modulated lapped transform introduced by Malvar [10].

H. Shu, X. Bao, and L. Luo are with the Laboratory of Image Science and Technology, Department of Computer Science and Engineering, Southeast University, 210096 Nanjing, China (e-mail: shu.list@seu.edu.cn; bao.list@seu.edu.cn; luo.list@seu.edu.cn).

C. Toumoulin is with the Laboratoire Traitement du Signal et de l'Image, Université de Rennes I-INSERM U642, 35042 Rennes, France (e-mail: christine.toumoulin@univ-rennes1.fr).

Some of the existing algorithms dealt with data sequences whose length is a power of 2, but in layer III of MPEG-I and MPEG-II, the length of the data blocks is $N \neq 2^m$, so that the data sequence must be zero-padded. Since the layer III specifies two different MDCT block sizes: a long block ($N = 36$) and a short block ($N = 12$), we present in this letter a radix-3 algorithm for efficiently computing the MDCT and IMDCT for data sequences with length $N = 2 \times 3^m$, where m is a positive integer. Such a strategy was recently adopted by Chan and Siu [11] in the fast computation of type-II DCT.

II. Derivation of the MDCT Algorithm

Letting $x(n)$, $n = 0, 1, 2, \ldots, N-1$, be an input data sequence, the MDCT of $x(n)$ is defined as [1]

$$X(k) = \sum_{n=0}^{N-1} x(n) \cos\left[\frac{\pi}{2N}\left(2n+1+\frac{N}{2}\right)(2k+1)\right] \qquad k = 0, 1, 2, \ldots, N/2-1. \quad (1)$$

Letting $N = 2 \times 3^m$, where m is a positive integer, we can realize the following three formulations to obtain the MDCT coefficients of $x(n)$ instead of computing (1) directly.

A. Computation of $A(k) = X(3k+1)$

$$\begin{aligned} A(k) &= \sum_{n=0}^{N-1} x(n) \cos\left[\frac{\pi}{2N}\left(2n+1+\frac{N}{2}\right)(6k+3)\right] \\ &= A_1(k) + A_2(k) + A_3(k) \end{aligned} \quad (2)$$

where

$$A_1(k) = \sum_{n=0}^{N/3-1} x(n) \cos\left[\frac{3\pi}{2N}\left(2n+1+\frac{N}{2}\right)(2k+1)\right] \quad (3)$$

$$A_2(k) = \sum_{n=N/3}^{2N/3-1} x(n) \cos\left[\frac{3\pi}{2N}\left(2n+1+\frac{N}{2}\right)(2k+1)\right] \quad (4)$$

$$A_3(k) = \sum_{n=2N/3}^{N-1} x(n) \cos\left[\frac{3\pi}{2N}\left(2n+1+\frac{N}{2}\right)(2k+1)\right]. \quad (5)$$

For computation of $A_1(k)$, by making a change of variable $n' = N/3 - 1 - n$, we obtain

$$A_1(k) = \sum_{n=0}^{N/3-1} x(N/3-1-n) \cos\phi_{n,k} \quad (6)$$

with $\phi_{n,k} = \frac{3\pi}{2N}\left(2n+1+\frac{N}{6}\right)(2k+1)$. Equation (6) shows that $A_1(k)$ is MDCT of a sequence with length $N/3$. Similarly, we have

$$A_2(k) = -\sum_{n=0}^{N/3-1} x(2N/3-1-n)\cos\phi_{n,k} \tag{7}$$

$$A_3(k) = \sum_{n=0}^{N/3-1} x(N-1-n)\cos\phi_{n,k}. \tag{8}$$

Letting $a_n = x(n)$, $b_n = x(N/3+n)$, and $c_n = x(2N/3+n)$, for $n = 0, 1, \ldots, N/3-1$, (2) becomes

$$A(k) = \sum_{n=0}^{N/3-1} (a'_n - b'_n + c'_n)\cos\phi_{n,k} \tag{9}$$

with $a'_n = a_{N/3-1-n}$, $b'_n = b_{N/3-1-n}$, and $c'_n = c_{N/3-1-n}$.

B. Computation of $B(k) = X(3k) + X(3k+2)$

$$B(k) = \sum_{n=0}^{N-1} 2x(n)\sin\left[\frac{\pi}{N}(2n+1)\right] \times\cos\left[\frac{3\pi}{2N}\left(2n+1+\frac{N}{2}\right)(2k+1)\right]. \tag{10}$$

Proceeding in a similar way as for $A(k)$, we obtain

$$B(k) = \sum_{n=0}^{N/3-1} \Big[(2c'_n - a'_n + b'_n)\sin\theta_n - \sqrt{3}(a'_n + b'_n)\cos\theta_n\Big]\cos\phi_{n,k} \tag{11}$$

with $\theta_n = (2n+1)\pi/N$.

C. Computation of $C(k) = X(3k) - X(3k+2)$

$$\begin{aligned} C(k) &= \sum_{n=0}^{N-1} 2x(n)\cos\Big[\frac{\pi}{N}(2n+1)\Big] \times\sin\left[\frac{3\pi}{2N}\left(2n+1+\frac{N}{2}\right)(2k+1)\right] \\ &= \sum_{n=0}^{N/3-1} 2x(n)\cos\theta_n \times\sin\Big[\frac{\pi}{2}(2k+1)+\phi_{n,k}\Big] \\ &\quad + \sum_{n=0}^{N/3-1} 2x\left(\frac{N}{3}+n\right)\cos\left(\frac{2\pi}{3}+\theta_n\right) \times\sin\left[\frac{3\pi}{2}(2k+1)+\phi_{n,k}\right] \\ &\quad + \sum_{n=0}^{N/3-1} 2x\left(\frac{2N}{3}+n\right)\cos\left(\frac{4\pi}{3}+\theta_n\right) \times\sin\left[\frac{5\pi}{2}(2k+1)+\phi_{n,k}\right] \end{aligned} \tag{12}$$

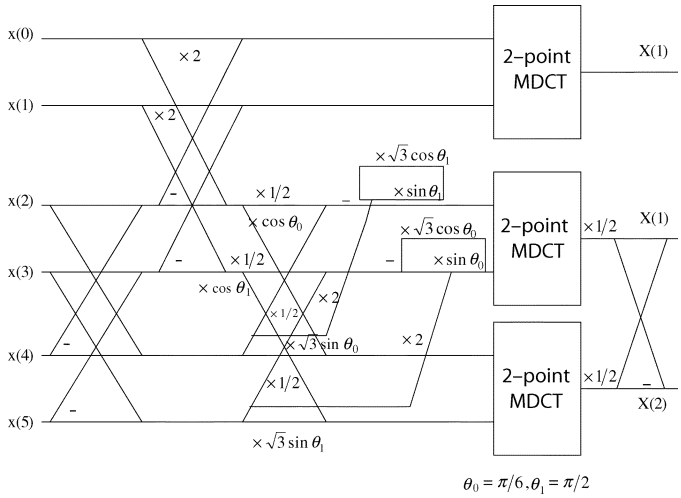

Fig. 1. Flowgraph of a 6-point MDCT.

thus

$$C(k) = (-1)^k \sum_{n=0}^{N/3-1} \Big[(2a_n + b_n - c_n)\cos\theta_n + \sqrt{3}(b_n + c_n)\sin\theta_n\Big] cos\phi_{n,k}. \tag{13}$$

When $B(k)$ and $C(k)$ are computed, the values of $X(3k)$ and $X(3k+2)$ can be obtained by

$$\begin{aligned} X(3k) &= \frac{1}{2}[B(k) + C(k)] \\ X(3k+2) &= \frac{1}{2}[B(k) - C(k)] \end{aligned} \qquad k = 0, 1, \ldots, N/6-1. \tag{14}$$

Since $A(k)$, $B(k)$, and $C(k)$ are all the $N/3$-length MDCTs, the above discussion shows that the coefficients $X(k)$ can be obtained from $A(k)$, $B(k)$, and $C(k)$, $k = 0, 1, \ldots, N/6-1$, with $N/3$ additions. Hence, we can calculate an N-length MDCT via the computation of three $N/3$-length MDCTs. Note that to obtain the sequences $(2c'_n - a'_n + b'_n)\sin\theta_n - \sqrt{3}(a'_n + b'_n)\cos\theta_n$ in (11) and $(2a_n + b_n - c_n)\cos\theta_n + \sqrt{3}(b_n + c_n)\sin\theta_n$ in (13), four multiplications are required for each n. However, when $n = (N/6-1)/2$, we have $\theta_n = \pi/6$, $\cos\theta_n = \sqrt{3}/2$, $\sin\theta_n = 1/2$. In this case, we can save three multiplications and five additions. Fig. 1 shows the flowgraph of the realization of 6-point MDCT.

The decomposition method described above is categorized as "decimation in frequency." Its computational complexity is given in both recursive and nonrecursive forms as follows:

$$\begin{aligned} M_N^{\mathrm{MDCT}} &= 3M_{N/3}^{\mathrm{MDCT}} + 4N/3 - 3 \\ &= \frac{4}{3}N\log_3 N - \frac{3}{2}(N-1) \\ &\quad \text{for } N = 2\times 3^m, m \geq 1 \end{aligned} \tag{15}$$

$$\begin{aligned} A_N^{\mathrm{MDCT}} &= 3A_{N/3}^{\mathrm{MDCT}} + 10N/3 - 5 \\ &= \frac{10}{3}N\log_3 N - \frac{5}{2}(N-1) \\ &\quad \text{for } N = 2\times 3^m, m \geq 1. \end{aligned} \tag{16}$$

TABLE I
COMPUTATIONAL COMPLEXITY OF THE PROPOSED RADIX-3 ALGORITHM AND THE FAST RADIX-2 ALGORITHM WITH ZERO-PADDING FOR THE COMPUTATION OF MDCT OF LENGTH $N = 2 \times 3^m$

	Britanak's algorithm [5]			Proposed method		
N	Mul	Add	Mul+Add	Mul	Add	Mul+Add
2×3 = 6	8	24	32	5	15	20
2×9 = 18	48	144	192	36	100	136
2×27 = 54	112	336	448	177	475	652
2×81=162	576	1728	2304	744	1960	2704

In the recursive method proposed by Nikolajevic and Fettweis [3], the computational complexity for computing an N-point MDCT is

$$M_N^{\mathrm{MDCT}} = \frac{N}{2}(N+2), \quad A_N^{\mathrm{MDCT}} = \frac{N}{2}(2N+1). \quad (17)$$

Thus, our method appears more efficient than the recursive algorithm. We also compare our algorithm with the fast radix-2 algorithm presented in [5] for which the zero-padding is included. Table I lists the arithmetic operations needed by the two approaches for $N = 2\times 3^m$. It shows that these algorithms require about the same computational complexity in terms of the total number of arithmetic operations. However, for the radix-2 algorithm based on DCT/DST, the input sequence needs to be rearranged. Note that the proposed approach requires for $N = 12$, 28 multiplications and 52 additions, and for $N = 36$, 132 multiplications and 276 additions.

III. DERIVATION OF THE IMDCT ALGORITHM

The IMDCT is given by [1]

$$\widetilde{x}(n) = \sum_{k=0}^{N/2-1} X(k)\cos\left[\frac{\pi}{2N}\left(2n+1+\frac{N}{2}\right)(2k+1)\right] \quad n = 0,1,2,\ldots,N-1. \quad (18)$$

To derive an efficient algorithm for fast computation of IMDCT, we further suppose N is a multiple of 12, i.e., $N = 4\times 3^m$.

A. Computation of $A'(n) = \widetilde{x}(3n+1)$

$$A'(n) = \sum_{k=0}^{N/2-1} X(k)\cos\left[\frac{\pi}{2N}\left(6n+3+\frac{N}{2}\right)(2k+1)\right] = A'_1(n) + A'_2(n) + A'_3(n) \quad (19)$$

where

$$A'_1(n) = \sum_{k=0}^{N/6-1} X(k)\cos\phi_{n,k} \quad (20)$$

$$A'_2(n) = \sum_{k=N/6}^{N/3-1} X(k)\cos\phi_{n,k} \quad (21)$$

$$A'_3(n) = \sum_{k=N/3}^{N/2-1} X(k)\cos\phi_{n,k}. \quad (22)$$

Equation (20) shows that $A'_1(n)$ is the IMDCT of $X(k)$ with length $N/6$. For $A'_2(n)$, we have

$$A'_2(n) = \sum_{k=0}^{N/6-1} X(N/3-1-k) \times \cos\left[\pi\left(2n+1+\frac{N}{6}\right) - \phi_{n,k}\right]. \quad (23)$$

Since N is a multiple of 12, we obtain

$$A'_2(n) = -\sum_{k=0}^{N/6-1} X(N/3-1-k)\cos\phi_{n,k}. \quad (24)$$

For $A'_3(n)$, we have

$$A'_3(n) = -\sum_{k=0}^{N/6-1} X(N/3+k)\cos\phi_{n,k}. \quad (25)$$

Letting $f_k = X(k)$, $g_k = X(N/6+k)$, and $h_k = X(N/3+k)$, for $k = 0,1,\ldots,N/6-1$, then we have

$$A'(n) = \sum_{k=0}^{N/6-1} (f_k - g'_k - h_k)\cos\phi_{n,k} \quad (26)$$

with $g'_k = g_{N/6-1-k}$.

B. Computation of $B'(n) = \widetilde{x}(3n) + \widetilde{x}(3n+2)$

We have

$$B'(n) = \sum_{k=0}^{N/2-1} 2X(k)\cos\left[\frac{\pi}{N}(2k+1)\right]\cos\phi_{n,k}. \quad (27)$$

Proceeding with the computation of $B'(n)$ in a similar way as for $A'(n)$, we obtain

$$B'(n) = \sum_{k=0}^{N/6-1} \Big[(2f_k + g'_k + h_k)\times\cos\theta_k + \sqrt{3}(h_k - g'_k)\sin\theta_k\Big]\cos\phi_{n,k} \quad (28)$$

with $\theta_k = (2k+1)\pi/N$.

C. Computation of $C'(n) = \widetilde{x}(3n) - \widetilde{x}(3n+2)$

We have

$$C'(n) = \sum_{k=0}^{N/2-1} 2X(k)\sin\left[\frac{\pi}{N}(2k+1)\right]\sin\phi_{n,k} = \sum_{k=0}^{N/6-1} 2X\left(\frac{N}{6}-1-k\right)\sin\left[\frac{\pi}{N}\left(\frac{N}{3}-2k-1\right)\right] \times \sin\left[\frac{\pi}{2}\left(2n+1+\frac{N}{6}\right) - \phi_{n,k}\right]$$

$$+\sum_{k=0}^{N/6-1} 2X\left(\frac{N}{6}+k\right)\sin\left[\frac{\pi}{N}\left(\frac{N}{3}+2k+1\right)\right]$$
$$\times\sin\left[\frac{\pi}{2}\left(2n+1+\frac{N}{6}\right)+\phi_{n,k}\right]$$
$$+\sum_{k=0}^{N/6-1} 2X\left(\frac{N}{2}-1-k\right)\sin\left[\frac{\pi}{N}(N-2k-1)\right]$$
$$\times\sin\left[\frac{3\pi}{2N}\left(2n+1+\frac{N}{6}\right)-\phi_{n,k}\right]. \quad (29)$$

Since N is a multiple of 12, i.e., $N=12L$, (29) becomes

$$C'(n)=(-1)^{n+L}\sum_{k=0}^{N/6-1}\Big[\sqrt{3}\,(f'_k+g_k)\cos\theta_k$$
$$-(f'_k-g_k+2h'_k)\sin\theta_k\Big]\cos\phi-n,k. \quad (30)$$

Note that the assumption of N being a multiple of 12 instead of a multiple of 6 is only required in the computation of $C'(n)$. Note also that $A'(n)$, $B'(n)$, and $C'(n)$ are all $N/6$-point IMDCTs. Similarly to the previous section, we can obtain the sequence $\widetilde{x}(n)$ from $A'(n)$, $B'(n)$, and $C'(n)$ with $2N/3$ additions. The computational complexity of the above method is given by

$$M_N^{\mathrm{IMDCT}}=3M_{N/3}^{\mathrm{IMDCT}}+2N/3$$
$$=\frac{2}{3}N\log_3 N,\quad \text{for } N=4\times3^m,\ m\geq1 \quad (31)$$
$$A_N^{\mathrm{IMDCT}}=3A_{N/3}^{\mathrm{IMDCT}}+13N/6$$
$$=\frac{13}{6}N\log_3 N,\quad \text{for } N=4\times3^m,\ m\geq1. \quad (32)$$

In the recursive method proposed by Nikolajevic and Fettweis [3], the computational complexity for computing an $N/2$-point IMDCT is

$$M_N^{\mathrm{IMDCT}}=N(N/2+1),\quad A_N^{\mathrm{IMDCT}}=N(N+1). \quad (33)$$

The above discussion shows that the proposed method is more efficient than the recursive algorithm. Table II lists the computational complexity of the proposed algorithm and the fast radix-2 algorithm with zero-padding for computing the IMDCT. It shows that in most cases, the proposed method needs fewer arithmetic operations than the one required in the radix-2 algorithm.

TABLE II
COMPUTATIONAL COMPLEXITY OF THE PROPOSED RADIX-3 ALGORITHM AND THE FAST RADIX-2 ALGORITHM WITH ZERO-PADDING FOR THE COMPUTATION OF IMDCT OF LENGTH $N=4\times3^m$

	Britanak's algorithm [5]			Proposed method		
N	Mul	Add	Mul+Add	Mul	Add	Mul+Add
4×3 = 12	20	42	62	20	42	62
4×9 = 36	112	304	416	84	174	258
4×27 = 108	256	704	960	324	756	1080
4×81=324	1280	3854	5134	1188	2970	4158

IV. CONCLUSIONS

In this letter, we present a new radix-3 algorithm for computing the MDCT of a sequence with length 2×3^m and the IMDCT for a sequence of length 4×3^m. The computational complexity of the new radix-3 algorithm is lower than some existing algorithms. Notice that the combination of the proposed method with the existing algorithms provides more flexibility in the choice of the sequence lengths for which the fast computation of MDCT and IMDCT can be easily realized.

REFERENCES

[1] J. P. Princen, A. W. Johnson, and A. B. Bradley, "Subband/transform coding using filter bank design based on time domain aliasing cancellation," in *Proc. IEEE ICASSP*, Dallas, TX, 1987, pp. 2161–2164.

[2] H. C. Chiang and J. C. Liu, "Regressive implementations for the forward and inverse MDCT in MPEG audio coding," *IEEE Signal Process. Lett.*, vol. 3, no. 4, pp. 116–118, Apr. 1996.

[3] V. Nikolajevic and G. Fettweis, "Computation of forward and inverse MDCT using Clenshaw's recurrence formula," *IEEE Trans. Signal Process.*, vol. 51, no. 5, pp. 1439–1444, May 2003.

[4] Y. H. Fan, V. K. Madisetti, and R. M. Mersereau, "On fast algorithms for computing the inverse modified discrete cosine transform," *IEEE Signal Process. Lett.*, vol. 6, no. 3, pp. 61–64, Mar. 1999.

[5] V. Britanak and K. R. Rao, "An efficient implementation of the forward and inverse MDCT in MPEG audio coding," *IEEE Signal Process. Lett.*, vol. 8, no. 2, pp. 48–51, Feb. 2001.

[6] S. W. Lee, "Improved algorithm for efficient computation of the forward and backward MDCT in MPEG audio coder," *IEEE Trans. Circuits Syst. II*, vol. 48, no. 10, pp. 990–994, Oct. 2001.

[7] M. H. Cheng and Y. H. Hsu, "Fast IMDCT and MDCT algorithms—a matrix approach," *IEEE Trans. Signal Process.*, vol. 51, no. 1, pp. 221–229, Jan. 2003.

[8] V. Britanak and K. R. Rao, "A new fast algorithm for the unified forward and inverse MDCT/MDST computation," *Signal Process.*, vol. 82, pp. 433–459, 2002.

[9] V. Britanak, "An efficient computing of oddly stacked MDCT/MDST via evenly stacked MDCT/MDST and vice-versa," *Signal Process.*, vol. 85, pp. 1353–1374, 2005.

[10] H. S. Malvar, *Signal Processing With Lapped Transforms*. Norwood, MA: Artech House, 1992.

[11] Y. H. Chan and W. C. Siu, "Mixed-radix discrete cosine transform," *IEEE Trans. Signal Process.*, vol. 41, no. 11, pp. 3157–3161, Nov. 1993.