

Fast Algorithm for Modulated Complex Lapped Transform

Xingdong Dai and Meghanad D. Wagh

Abstract—A new algorithm for the modulated complex lapped transform (MCLT) with a sine windowing function is presented. It is shown that by merging the windowing operation with the main computation, both the real and the imaginary parts of the MCLT with $2N$ inputs can be obtained from two N -point discrete cosine transforms of type II (DCTs-II) of appropriate inputs. The resulting algorithm is computationally very efficient. In general, the value of N is an even number. When N is a power of 2, the proposed algorithm uses only $N \log N + 2$ real multiplications (including the scaling factors in the DCT computation), with none of those being outside the DCT blocks.

Index Terms—Audio coding, fast algorithm, modified discrete cosine transform, modified discrete sine transform, modulated complex lapped transform.

I. INTRODUCTION

THE modulated complex lapped transform (MCLT) is structured as a cosine-/sine-modulated filter bank that maps overlapping blocks of a real-valued signal into blocks of complex-valued transform coefficients [1]. It is a special form of a two times oversampled discrete Fourier transform (DFT) filter bank to perform frequency decomposition. Since the reconstruction formula of the MCLT is not unique, the MCLT allows more flexible implementations of audio enhancement and encoding systems than the DFT. Recent MCLT applications include acoustic echo cancellation [1] and audio watermarking [2] by using the phase information from the imaginary coefficients.

The N -point MCLT of a $2N$ -point input sequence $\{x(n)\}$ is defined as [1]

$$y(k) = \sum_{n=0}^{2N-1} [p_c(n, k) - jp_s(n, k)]x(n), \quad 0 \leq k < N$$

where the real and imaginary parts of the MCLT kernel are, respectively, defined as

$$\begin{aligned} p_c(n, k) &= \sqrt{\frac{2}{N}} h(n) \cos\left(\frac{(2n+1+N)(2k+1)\pi}{4N}\right) \\ p_s(n, k) &= \sqrt{\frac{2}{N}} h(n) \sin\left(\frac{(2n+1+N)(2k+1)\pi}{4N}\right). \end{aligned} \quad (1)$$

Manuscript received July 18, 2007; revised August 02, 2008. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Yuan-Pei Lin.

X. Dai is with LSI Corporation, 1110 American Parkway NE, Allentown, PA 18109 USA (e-mail: xingdong.dai@lsi.com).

M. D. Wagh is with the Department of Electrical and Computer Engineering, Lehigh University, Bethlehem, PA 18015 USA (e-mail: mdw0@lehigh.edu).

Digital Object Identifier 10.1109/LSP.2008.2008434

Function $h(n)$ in (1) is the windowing function. The most common choice for $h(n)$ is the sine windowing function

$$h(n) = -\sin\left(\frac{(2n+1)\pi}{4N}\right) \quad (2)$$

which is used in many audio coding applications including MPEG-1/2 layer III (MP3) since it permits a perfect reconstruction [3], [4], [6].

The real part of MCLT is the forward/direct modulated lapped transform (MLT) [3], which is used to implement transform coding in video and audio compression applications [5]. The MLT represents the perfect reconstruction cosine-modulated filter bank based on the concept of time domain aliasing cancellation (TDAC) [6]. The MLT has also been referred to as the oddly stacked modified discrete cosine transform (MDCT) [6], [7]. Similarly, the imaginary part of MCLT is the oddly stacked modified discrete sine transform (MDST). However, following [8], we use in this letter rather the MDCT to refer to the MLT without a windowing function. The calculation of MLT often involves scaling the input by applying the windowing function and then using a fast MDCT algorithm based on either FFT or the DCT with appropriate pre-and post-permutations [8]–[10]. Computational efficiency of the MLT can be improved by combining the windowing function with the MDCT [3], [9], [11], [12].

The original sequence $\{x(n)\}$ can be recovered from MCLT by using either its real or the imaginary part, or both [13]. If only the real part is used, the inverse transform is the same as the inverse MLT (IMLT) which has been studied in detail [3]. Here we focus on fast algorithm for the forward/direct MCLT where both real and imaginary parts are required.

As a complex extension of the MLT, the MCLT shares many fast algorithms with MLT. Malvar has shown that the real part of the MCLT with arbitrary windowing function can be obtained from a discrete cosine transform of type IV (DCT-IV) and the imaginary, from a discrete sine transform of type IV (DST-IV) [1]. However, the computational complexity is affected by applying the windowing function to the input before using the DCT-IV or DST-IV. Later, FFT-based MCLT algorithms have been developed that merge a sine windowing function [13], [14] with the main computation to improve the computational efficiency. Another recently proposed MCLT algorithm applicable to arbitrary but symmetric windowing function uses two DCTs of type II (DCTs-II) to compute the real and imaginary parts of the MCLT separately [15]. However, this algorithm employs permutation stages with nonconstant multiplications outside the main computation module limiting its efficiency. An improved version of the MCLT algorithm [15] can be found in [20].

This letter focuses on MCLT computation with a sine windowing function merged with the main computational modules.

Our algorithm for the $2N$ -point MCLT is based on the evenly stacked MDCT and the corresponding MDST, each with $2N$ inputs and $N+1$ outputs. The real and complex parts of the MCLT can be obtained by adding the outputs of these evenly stacked MDCT and MDST. This allows one to use any algorithm for the evenly stacked MDCT and MDST in MCLT computation with a sine window function. We further show that the evenly stacked MDCT and MDST may be computed with two N -point DCTs-II of appropriately folded input sequences. This results in a simple derivation of a new MCLT algorithm with lower computational complexity than almost all the algorithms available today.

II. PROPOSED ALGORITHM

Recall that the real and imaginary parts of the MCLT are the oddly stacked MDCT and MDST, respectively. When one applies the windowing function separately, the oddly stacked MDCT and MDST can be computed through evenly stacked MDCT and MDST [16]. We apply a similar strategy directly to the MCLT with a sine windowing function. For this purpose, we use the intermediate transforms with $2N$ input and $N+1$ output points defined as

$$c^E(k) = \frac{1}{2\sqrt{N}} \sum_{n=0}^{2N-1} \cos\left(\frac{\pi k(2n+1+N)}{2N}\right) x(n) \quad (3)$$

$$s^E(k) = \frac{1}{2\sqrt{N}} \sum_{n=0}^{2N-1} \sin\left(\frac{\pi k(2n+1+N)}{2N}\right) x(n) \quad (4)$$

$0 \leq k \leq N$. Note that (3) and (4) describe the evenly stacked MDCT and corresponding MDST [4], [16], respectively, with a constant scaling factor.

One can show that the transforms given by (3) and (4) are related to the DCT-II when N is an even integer. To convert evenly stacked MDCT into an N -point DCT-II, note that in the computation of (3), samples $x(n+3N/2)$ and $x(3N/2-n-1)$ for $0 \leq n < N/2-1$ are multiplied with the same cosine term $\cos(\pi k(2n+1)/(2N))$. Similarly, the samples $x(n-N/2)$ and $x(3N/2-n-1)$ for $N/2 \leq n < N$ are also multiplied with the same cosine term, $\cos(\pi k(2n+1)/(2N))$. Further, as n goes over the stated ranges, the sample indices of $\{x(n)\}$ span the entire range from 0 to $2N-1$. To take advantage of this, define a sequence $\{x_c(n)\}$ as

$$x_c(n) = \begin{cases} x\left(n + \frac{3N}{2}\right) + x\left(\frac{3N}{2} - n - 1\right), & \text{if } 0 \leq n < \frac{N}{2} \\ x\left(n - \frac{N}{2}\right) + x\left(\frac{3N}{2} - n - 1\right), & \text{if } \frac{N}{2} \leq n < N. \end{cases} \quad (5)$$

It is then obvious that

$$c^E(k) = \frac{1}{2\sqrt{N}} \sum_{n=0}^{N-1} \cos\left(\frac{\pi k(2n+1)}{2N}\right) x_c(n) \quad 0 \leq k \leq N. \quad (6)$$

Thus, for $0 \leq k \leq N-1$, evenly stacked MDCT of the sequence $\{x(n)\}$ is the same as the N -point DCT-II of the sequence $\{x_c(n)\}$ except for a constant multiplication factor $1/(2\sqrt{N})$. Therefore, transform coefficients $\{c^E(k)\}$ can be

computed using any available DCT-II algorithms [17], [18] without incurring additional computational penalty except for one multiplication. Further, (6) also shows that

$$c^E(N) = 0. \quad (7)$$

Evenly stacked MDST can also be related to the DCT-II in a similar way. In the definition (4), samples $x(n+3N/2)$ and $-x(3N/2-n-1)$ when $0 \leq n < N/2$ as well as $x(n-N/2)$ and $-x(3N/2-n-1)$ when $N/2 \leq n < N$ are all multiplied with the same sine term, $\sin(\pi k(2n+1)/(2N))$. Again, the indices of these four $x(n)$ terms span the entire input index range, 0 to $2N-1$, when n goes over the specified ranges. To take advantage of this, define the sequence $\{x_s(n)\}$ as

$$x_s(n) = \begin{cases} x\left(n + \frac{3N}{2}\right) - x\left(\frac{3N}{2} - n - 1\right), & \text{if } 0 \leq n < \frac{N}{2} \\ x\left(n - \frac{N}{2}\right) - x\left(\frac{3N}{2} - n - 1\right), & \text{if } \frac{N}{2} \leq n < N. \end{cases} \quad (8)$$

One can then see that

$$s^E(k) = \frac{1}{2\sqrt{N}} \sum_{n=0}^{N-1} \sin\left(\frac{\pi k(2n+1)}{2N}\right) x_s(n) \quad 0 \leq k \leq N. \quad (9)$$

Equation (9) shows that except for the scaling factor $1/(2\sqrt{N})$, evenly stacked MDST is converted to the N -point discrete sine transform of type II (DST-II) and can be obtained using any available algorithm for the DST-II [18]. However, since we are already using the DCT-II for the computation of $\{c^E(k)\}$, we prefer to use the same for computation of $\{s^E(k)\}$ as well. For this, we employ an approach similar to that of [19]. Define a new sequence \bar{x}_s as

$$\bar{x}_s(n) = (-1)^n x_s(n). \quad (10)$$

Then (9) can be then rewritten as

$$\begin{aligned} s^E(k) &= \frac{1}{2\sqrt{N}} \sum_{n=0}^{N-1} \sin\left((2n+1)\frac{\pi}{2} - \frac{\pi(N-k)(2n+1)}{2N}\right) (-1)^n \bar{x}_s(n) \\ &= \frac{1}{2\sqrt{N}} \sum_{n=0}^{N-1} \cos\left(\frac{\pi(N-k)(2n+1)}{2N}\right) \bar{x}_s(n) \quad 0 \leq k \leq N. \end{aligned} \quad (11)$$

Equation (11) shows that when $1 \leq k \leq N$, except for the constant multiplication factor $1/(2\sqrt{N})$, the k th coefficient $s^E(k)$ is the same as the $(N-k)$ th coefficient of the N -point DCT-II of $\{\bar{x}_s(n)\}$. As before, coefficients $\{s^E(k)\}$ can be computed using any available DCT-II algorithm with at most one additional multiplication. Equation (11) also shows that

$$s^E(0) = 0. \quad (12)$$

In the following two subsections, we show that the real and imaginary parts of the MCLT can be obtained directly from $c^E(k)$ and $s^E(k)$. This then allows us to compute MCLT from the DCTs-II of $\{x_c(n)\}$ and $\{\bar{x}_s(n)\}$.

A. Real Part of the MCLT

The real part of the MCLT kernel $p_c(n, k)$ given by (1) with the sine windowing function given by (2) can be simplified to

$$p_c(n, k) = p'_c(n, k) + p''_c(n, k)$$

where

$$\begin{aligned} p'_c(n, k) &= \frac{1}{\sqrt{2N}} \\ &\quad \times \sin\left(\frac{k\pi(2n+1+N)}{2N} + \frac{\pi}{4}\right) \\ p''_c(n, k) &= -\frac{1}{\sqrt{2N}} \\ &\quad \times \sin\left(\frac{(k+1)\pi(2n+1+N)}{2N} - \frac{\pi}{4}\right). \end{aligned}$$

Using trigonometric identities, p'_c and p''_c can be simplified to

$$\begin{aligned} p'_c(n, k) &= \frac{1}{2\sqrt{N}} \left[\cos\left(\frac{k\pi(2n+1+N)}{2N}\right) \right. \\ &\quad \left. + \sin\left(\frac{k\pi(2n+1+N)}{2N}\right) \right] \\ p''_c(n, k) &= \frac{1}{2\sqrt{N}} \left[\cos\left(\frac{(k+1)\pi(2n+1+N)}{2N}\right) \right. \\ &\quad \left. - \sin\left(\frac{(k+1)\pi(2n+1+N)}{2N}\right) \right]. \end{aligned}$$

From these expressions, one can obtain the real part of the MCLT of the sequence $\{x(n)\}$ as

$$\begin{aligned} \mathcal{R}e[y(k)] &= \sum_{n=0}^{2N-1} p'_c(n, k)x(n) \\ &\quad + \sum_{n=0}^{2N-1} p''_c(n, k)x(n) \\ &= c^E(k) + s^E(k) + c^E(k+1) \\ &\quad - s^E(k+1), \quad 0 \leq k < N. \end{aligned} \quad (13)$$

B. Imaginary Part of the MCLT

The imaginary part of the MCLT can also be computed through $c^E(k)$ and $s^E(k)$. The imaginary part of the MCLT kernel $p_s(n, k)$ given by (1) with the sine windowing function (2) can be expressed as

$$p_s(n, k) = p'_s(n, k) + p''_s(n, k)$$

where

$$\begin{aligned} p'_s(n, k) &= -\frac{1}{\sqrt{2N}} \cos\left(\frac{k\pi(2n+1+N)}{2N} + \frac{\pi}{4}\right) \\ p''_s(n, k) &= \frac{1}{\sqrt{2N}} \cos\left(\frac{(k+1)\pi(2n+1+N)}{2N} - \frac{\pi}{4}\right). \end{aligned}$$

Using trigonometric identities, p'_s and p''_s can be simplified to

$$\begin{aligned} p'_s(n, k) &= \frac{1}{2\sqrt{N}} \left[-\cos\left(\frac{k\pi(2n+1+N)}{2N}\right) \right. \\ &\quad \left. + \sin\left(\frac{k\pi(2n+1+N)}{2N}\right) \right] \end{aligned}$$

$$\begin{aligned} p''_s(n, k) &= \frac{1}{2\sqrt{N}} \left[\cos\left(\frac{(k+1)\pi(2n+1+N)}{2N}\right) \right. \\ &\quad \left. + \sin\left(\frac{(k+1)\pi(2n+1+N)}{2N}\right) \right]. \end{aligned}$$

The imaginary part of the MCLT of the sequence $\{x(n)\}$ is then obtained as

$$\begin{aligned} \mathcal{I}m[y(k)] &= \sum_{n=0}^{2N-1} p'_s(n, k)x(n) \\ &\quad + \sum_{n=0}^{2N-1} p''_s(n, k)x(n) \\ &= -c^E(k) + s^E(k) + c^E(k+1) \\ &\quad + s^E(k+1), \quad 0 \leq k < N. \end{aligned} \quad (14)$$

C. New MCLT Algorithm

The above discussion leads to the following MCLT algorithm.

- Create sequences $\{x_c(n)\}$ and $\{\bar{x}_s(n)\}$ using (5), (8), and (10). This step requires $2N$ additions.
- Compute the DCT-II of $\{x_c(n)\}$ and $\{\bar{x}_s(n)\}$ using any of the fast DCT-II algorithms. For example, when N is a power of 2, one may employ the procedure of [17] to compute each DCT-II in $(N/2) \log N$ multiplications and $(3N/2) \log N - N + 1$ additions. The algorithm in [17] recursively partitions the DCT-II into two DCTs-II of half size. Denote them by DCT_1 and DCT_2 . The input for DCT_1 is obtained by folding the input sequence, and its output gives the even-indexed DCT-II components. The input for DCT_2 is obtained by folding the input sequence and then multiplying each component by a proper cosine term. A linear combination of DCT_2 output gives the odd-indexed DCT-II components. The DCT-II output can be scaled by a constant factor $1/(2\sqrt{N})$ [see (6) and (11)] by combining this factor with the multipliers applied to create the input of DCT_2 . This gives *scaled* odd-indexed DCT components without increasing the multiplication count. Since the algorithm of [17] is recursive, the same process can be used repeatedly every time DCT_1 is partitioned to scale half of its outputs. Thus, the only extra multiplication needed to scale all the components of the DCT-II is the multiplication used to scale the 0th DCT-II component. Two DCTs-II with the scaling factor as required in (6) and (11) can be computed in $N \log N + 2$ multiplications and $3N \log N - 2N + 2$ additions. If the scaling factor in DCT-II calculation is ignored as in [20], then the scaled DCTs-II require only $N \log N$ multiplications. The scaled DCTs-II of $\{x_c(n)\}$ and $\{\bar{x}_s(n)\}$ provide values of $\{c^E(k)\}$ and $\{s^E(k)\}$ for $0 \leq k \leq N$ from relations (6), (7), (11), and (12).
- Finally, to obtain the MCLT coefficients, first compute

$$z(k) = c^E(k+1) + s^E(k)$$

$$z'(k) = s^E(k+1) - c^E(k), \quad 0 \leq k < N$$

and the real and imaginary parts of MCLT are obtained from (13) and (14) as

$$\mathcal{R}e[y(k)] = z(k) - z'(k)$$

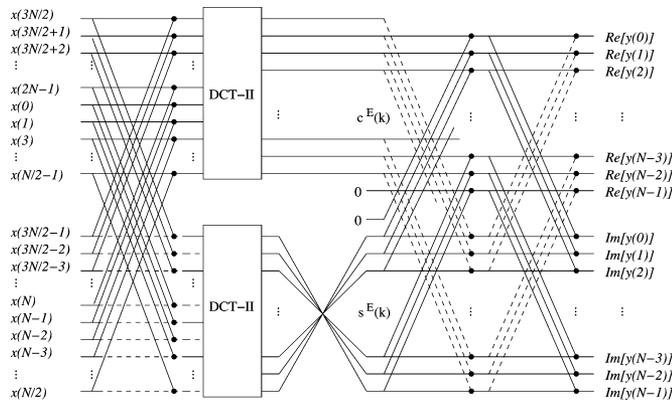


Fig. 1. Proposed algorithm for the 2N-point MCLT. The filled circles indicate additions and the dotted lines show negations of operands. The top DCT produces $c^E(k)$, $0 \leq k < N$ and the bottom DCT produces $s^E(k)$, $0 \leq k \leq N$ in reverse order.

TABLE I
COMPLEXITIES OF VARIOUS FAST MCLT ALGORITHMS FOR LENGTH $N = 2^n$

Algorithm	Window choice	Real multiplications	Real additions
Ref. [1]	any	$N \log N + 4N$	$3N \log N + 2N$
Ref. [20]	any	$N \log N + 2N$	$3N \log N + 4N$
Ref. [13]	sine	$N \log N + N$	$3N \log N + 3N - 2$
Ref. [15]	sine	$N \log N + N$	$3N \log N + 4N$
Ref. [20]	sine	$N \log N$	$3N \log N + 4N$
This paper	sine	$N \log N + 2^\dagger$	$3N \log N + 4N$

[†] If the scaling factor in DCT calculation is ignored as in [20], then the new algorithm requires $N \log N$ multiplications.

$$\text{Im}[y(k)] = z(k) + z'(k), \quad 0 \leq k < N.$$

This step requires $4N - 2$ additions, ignoring the trivial additions by $c^E(N)$ and $s^E(0)$, each of which is 0.

Thus, we can obtain the MCLT in $N \log N + 2(N \log N$ if the scaling factor in DCT-II calculation is ignored) real multiplications and $3N \log N + 4N$ real additions.

The flow chart of the algorithm is shown in Fig. 1.

III. DISCUSSION AND CONCLUSION

This letter shows that by merging the windowing operation with the MCLT computation, both the real and imaginary parts of the MCLT can be computed from the same two N -point DCTs-II. Table II compares the computational complexity of our algorithm to those of [1], [3], [15], and [20].

One can see from Table I that when N is a power of 2, our algorithm and that of [20] have the smallest number of multiplications amongst all algorithm available in literature. However, unlike [20], our algorithm exploits the relationship between the MCLT with a sine windowing function and the evenly stacked MDCT/MDST. This places our MCLT algorithm in a unique position to take advantage of any new developments in the evenly stacked MDCT and MDST computation.

When the evenly stacked MDCT and MDST are computed using the DCT-II as shown here, one may employ any efficient DCT-II software or hardware modules. For example, by using a bilinear algorithm for DCT-II, one can get a bilinear algorithm for the MCLT which can result in a very fast implementation

in VLSI. Most other algorithms use multiplications outside the main computational blocks of DCT and therefore cannot have a bilinear structure.

Finally, it should be mentioned that this letter extends the results of [16] to show that the oddly stacked MDCT/MDST can be computed through the evenly stacked MDCT/MDST even in the presence of a merged sine windowing function.

ACKNOWLEDGMENT

The authors would like to thank the reviewers for suggestions which greatly improved the quality of the final manuscript.

REFERENCES

- [1] H. Malvar, "A modulated complex lapped transform and its applications to audio processing," in *Proc. IEEE ICASSP'99*, Phoenix, AZ, May 1999, pp. 1421–1424.
- [2] D. Kirovski and H. Malvar, "Spread-spectrum watermarking of audio signals," *IEEE Trans. Signal Process.*, vol. 51, no. 4, pp. 1020–1033, Apr. 2003.
- [3] H. Malvar, *Signal Processing with Lapped Transforms*. Norwood, MA: Artech House, 1992.
- [4] J. Princen and A. Bradley, "Analysis/synthesis filter bank design based on time domain aliasing cancellation," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-34, pp. 1153–1161, Oct. 1986.
- [5] S. Shlien, "The modulated lapped transform, its time-varying forms, and its applications to audio coding standards," *IEEE Trans. Speech Audio Process.*, vol. 5, pp. 359–366, Jul. 1997.
- [6] J. P. Princen, A. W. Johnson, and A. B. Bradley, "Subband/transform coding using filter bank designs based on time domain aliasing cancellation," in *Proc. IEEE ICASSP'87*, Dallas, TX, Apr. 1987, pp. 2161–2164.
- [7] T. Painter and A. Spanias, "Perceptual coding of digital audio," *Proc. IEEE*, vol. 88, pt. 4, pp. 451–515, Apr. 2000.
- [8] V. Britanak and K. Rao, "A new fast algorithm for the unified forward and inverse MDCT/MDST computation," *Signal Process.*, vol. 82, no. 3, pp. 433–459, 2002.
- [9] P. Duhamel, Y. Mahieux, and J. Petit, "A fast algorithm for the implementation of filter banks based on time domain aliasing cancellation," in *Proc. IEEE ICASSP'91*, Toronto, ON, Canada, Apr. 1991, vol. 3, pp. 2209–2212.
- [10] M. Cheng and Y. Hsu, "Fast IMDCT and MDCT algorithms—A matrix approach," *IEEE Trans. Signal Process.*, vol. 51, no. 1, pp. 221–229, Jan. 2003.
- [11] D. Sevic and M. Popovic, "A new efficient implementation of the oddly stacked Princen-Bradley filter bank," *IEEE Signal Process. Lett.*, vol. 1, no. 11, pp. 166–168, Nov. 1994.
- [12] C. Jing and H. Tai, "Fast algorithm for computing modulated lapped transform," *Electron. Lett.*, vol. 37, pp. 796–797, Jun. 2001.
- [13] H. Malvar, "Fast algorithm for the modulated complex lapped transform," *IEEE Signal Process. Lett.*, vol. 10, no. 1, pp. 8–10, Jan. 2003.
- [14] H. Tai and C. Jing, "Design and efficient implementation of a modulated complex lapped transform processor using pipelining technique," *IEICE Trans. Fundam.*, vol. E84-A, pp. 1280–1286, May 2001.
- [15] Q. Dai and X. Chen, "New algorithm for modulated complex lapped transform with symmetrical window function," *IEEE Signal Process. Lett.*, vol. 11, no. 12, pp. 925–928, Dec. 2004.
- [16] V. Britanak, "An efficient computing of oddly stacked MDCT/MDST via evenly stacked MDCT/MDST and vice versa," *Signal Process.*, vol. 85, no. 7, pp. 1353–1374, 2005.
- [17] C. Kok, "Fast algorithm for computing discrete cosine transform," *IEEE Trans. Signal Process.*, vol. 45, no. 3, pp. 757–760, Mar. 1997.
- [18] V. Britanak, Y. Yip, and K. R. Rao, *Discrete Cosine and Sine Transforms: General Properties, Fast Algorithms and Integer Approximations*. New York: Academic, 2007.
- [19] Z. Wang, "A fast algorithm for the discrete sine transform implemented by the fast cosine transform," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-30, pp. 814–815, Oct. 1982.
- [20] X. Chen and Q. Dai, "A novel DCT-based algorithm for computing the modulated lapped transform," *IEEE Trans. Signal Process.*, vol. 54, no. 11, pp. 4480–4484, Nov. 2006.