FAST FACTORIZATIONS OF DISCRETE SINE TRANSFORMS OF TYPES VI AND VII

Ravi K. Chivukula

Qualcomm Inc.
5775 Morehouse Drive, San Diego, CA
rchivuku@qualcomm.com

*Yuriy A. Reznik**

InterDigital, Inc.
9710 Scranton Road, San Diego, CA
yreznik@ieee.org

ABSTRACT

Discrete Sine Transforms of types VI and VII (DST-VI/VII) have recently received considerable interest in video coding. In particular, it was shown that DST-VII offers good approximation for KLT of residual signals produced by spatial (Intra) prediction process. In this paper, we offer an additional argument for use of such transforms by showing that they allow fast computation. Specifically, we establish a mapping between N-point DST-VI/VII and an $2N+1$-point Discrete Fourier Transform (DFT), apply known factorization techniques for the DFT, and show how unused parts of the resulting flowgraph can be pruned, producing factorizations of DST-VI/VII.

Index Terms— Video coding, intra prediction, sinusoidal transforms, DCT, DST, DST-VII, DFT, FFT, factorization, multiplicative complexity.

1. INTRODUCTION

The Discrete Cosine Transforms of types II and IV (DCT-II/IV) are among fundamental, well understood, and much appreciated tools in data compression. The DCT-II is used at the core of standards for image and video compression, such as JPEG, ITU-T H.26x-series, and MPEG 1-4 standards [2]. The DCT-IV is used in audio coding algorithms, such as ITU-T Rec. G.722.1, MPEG-4 AAC, and others [3]. Such transforms are very well studied, and a number of efficient technique exists for their computation [2, 4, 5, 6, 7, 8].

Much less known is are so-called "odd" sinusoidal transforms: Discrete Cosine and Sine Transforms of types V, VI, VII, and VIII. Existence of some of such transforms was discovered by A. Jain in 1979 [11]. A complete tabulation was developed later by Wang and Hunt [12]. However, not much work has followed. Surveys of related results can be found in [13, 4].

Nevertheless, DST of types VI and VII have recently surfaced as useful tool in image and video coding. Thus, Han, Saxena, and Rose have shown that DST-VII produce good approximations of Karhunen-Loeve Transform (KLT) for model of residual signals after Intra-prediction [1]. This was subsequently validated in the course of experimental work on ISO/IEC/ITU-T High Efficiency Video Coding (HEVC) standard [14, 15, 16]. The adoption of DST-VII in HEVC has also prompted a discussion on the existence of fast algorithms for computing of such transforms [14].

The purpose of this paper is to show that fast algorithms for computing DST-VI/VII indeed exist, and offer general technique for their construction. Next section contains definitions. Section 3 establishes mapping between N-point DST-VII and $2N+1$-point DFT. Section 4 describes our proposed method for construction of fast factorizations of DST-VII. Examples of fast factorizations of DST-VI/VII of lengths $N = 4, 8$ are also provided in Section 4.

2. DEFINITIONS

Hereafter, by N we will denote the length of input data sequence, by $\Re(.)$ and $\Im(.)$ we will denote real and imaginary parts of complex numbers, and by $j = \sqrt{-1}$ we will denote imaginary unit.

Let $x = x_0, \ldots, x_{N-1}$ be a sequence of real numbers (input signal). The Discrete Fourier Transform (DFT) over sequence x will be defined as[1]:

$$X_k^F = \sum_{n=0}^{N-1} x_n e^{-j\frac{2\pi kn}{N}}; \quad k = 0, \ldots, N-1.$$

The Discrete Sine Transform of types VI and VII (DST-VI/VII) over x will be defined as follows:

$$\begin{aligned} X_k^{VI} &= \sum_{n=0}^{N-1} x_n \sin\frac{\pi(2n+1)(k+1)}{2N+1}, \ k = 0, \ldots, N-1\,, \\ X_k^{VII} &= \sum_{n=0}^{N-1} x_n \sin\frac{\pi(2k+1)(n+1)}{2N+1}, \ k = 0, \ldots, N-1\,. \end{aligned}$$

We immediately notice, that DST-VII is simply a transpose of DST-VI, and so finding factorization for either one of them will be sufficient to show how to factorize both.

*The work on this paper was done when second author was with Qualcomm Inc, San Diego, CA.

[1] For simplicity, we omit all normalization factors.

3. MAPPING BETWEEN DST-VI/VII AND DFT

In this section we will prove the following statement.

Theorem 1. *Let $x = x_0, \dots, x_{N-1}$ be a real-valued input sequence. Define an intermediate $2N+1$-point sequence $y = y_0, \dots, y_{2N}$ as follows:*

$$\begin{aligned} y_n &= 0, & n &= 0, \dots, N, \\ y_{N+1+n} &= x_{2n}, & n &= 0, \dots, \left\lceil \tfrac{N}{2} \right\rceil - 1, \\ y_{N+1+\left\lceil \frac{N}{2} \right\rceil + n} &= x_{2\left\lfloor \frac{N}{2} \right\rfloor - 1 - 2n}, & n &= 0, \dots, \left\lfloor \tfrac{N}{2} \right\rfloor - 1, \end{aligned}$$

and compute DFT over it

$$Y_k = \sum_{n=0}^{2N+1} y_n e^{-j\frac{2\pi k n}{2N+1}}; \quad k = 0, \dots, 2N.$$

Then:

$$X_k^{VII} = \Im\left(Y_{2k+1}\right), \quad k = 0, \dots, N-1. \tag{1}$$

is the DST-VII over x.

Proof. Let's take a look at DFT outputs ($k = 0, \dots, N-1$):

$$\begin{aligned} \Im\left[Y_{2k+1}\right] &= -\sum_{n=1}^{2N} y_n \sin\frac{2\pi(2k+1)n}{2N+1} \\ &= -\sum_{n=1}^{N}\left[y_n \sin\frac{2\pi(2k+1)n}{2N+1}\right. \\ &\quad \left. + y_{2N+1-n}\sin\frac{2\pi(2k+1)(2N+1-n)}{2N+1}\right]. \end{aligned}$$

Since further $y_n = 0,\ n = 1, \dots, N$, we have

$$\begin{aligned} \Im\left[Y_{2k+1}\right] &= \sum_{n=1}^{N} y_{2N+1-n} \sin\frac{\pi(2k+1)(2N+1-2n)}{2N+1} \\ &= \sum_{n=1}^{N} y_{2N+1-n} \sin\frac{2\pi(2k+1)n}{2N+1}, \end{aligned}$$

or by using substitution $n' = N - n$:

$$\Im\left[Y_{2k+1}\right] = \sum_{n=N-1}^{0} y_{N+1+n} \sin\frac{\pi(2k+1)(2n+1)}{2N+1}.$$

Let's now assume that N is even. Similar argument holds for odd N. We write

$$\begin{aligned} \Im\left[Y_{2k+1}\right] &= \sum_{n=0}^{\frac{N}{2}-1} y_{N+1+n} \sin\frac{\pi(2k+1)(2n+1)}{2N+1} \\ &+ \sum_{n=0}^{\frac{N}{2}-1} y_{\frac{3N}{2}+1+n} \sin\frac{\pi(2k+1)(2n+N+1)}{2N+1} \end{aligned}$$

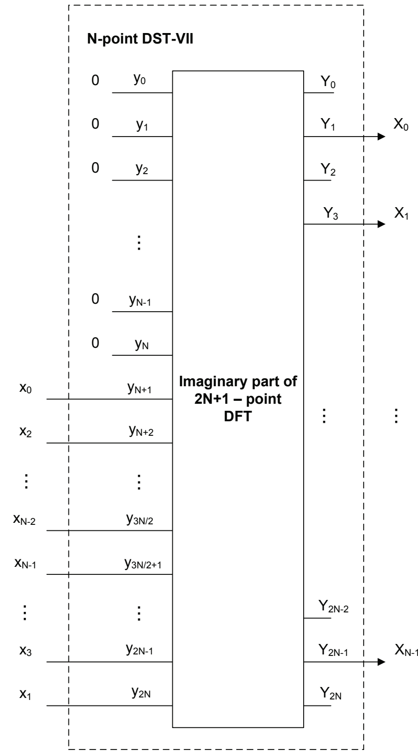

Fig. 1. Flow-graph of proposed mapping between N-point DST-VII and $2N+1$-point DFT, drawn when N is even.

where, based on our input mapping, the first sum receives quantities $y_{N+1+n} = x_{2n}$, while the second sum receives $y_{\frac{3N}{2}+1+n} = x_{N-1-2n}$.

By putting these facts together, we obtain

$$\begin{aligned} \Im\left[Y_{2k+1}\right] &= \sum_{n=0}^{\frac{N}{2}-1} x_{2n} \sin\frac{\pi(2k+1)(2n+1)}{2N+1} \\ &+ \sum_{n=0}^{\frac{N}{2}-1} x_{N-1-2n} \sin\frac{\pi(2k+1)(2n+N+1)}{2N+1} \\ &= \sum_{n=0}^{\frac{N}{2}-1} x_{2n} \sin\frac{\pi(2k+1)(2n+1)}{2N+1} \\ &+ \sum_{n=0}^{\frac{N}{2}-1} x_{2n+1} \sin\frac{\pi(2k+1)((2n+1)+1)}{2N+1} \\ &= X_k^{VII}. \end{aligned}$$

□

We show the flow-graph of mapping (1) in Figure 1. It can be observed that DST-VII can be computed by simply producing particularly re-ordered and zero-padded sequence as input

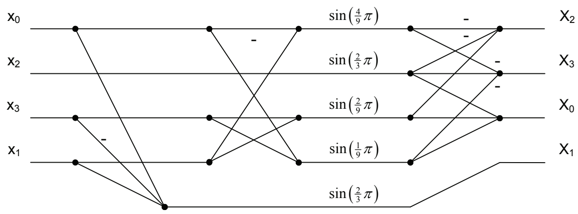

Fig. 2. Fast factorization of DST-VII of length 4.

to DFT, and collecting imaginary parts of odd-indexed DFT output values. In next section, we show that this process can be further simplified by ”pruning” DFT flow-graph such that only paths necessary for computing of DST-VII output values are remaining.

4. FAST ALGORITHMS FOR COMPUTING DST-VI/VII OF LENGTHS N=4,8

Based on previous discussion, it follows that DST-VII transform can be constructed by following these steps:

- Use mapping between DST-VII and DFT;
- Select fast factorization of DFT of length $2N + 1$;
- Prune flow-graph of the DFT, leaving only paths leading to odd-indexed imaginary values, corresponding to outputs of DST-VII.

This produces the flow-graph for DST-VII. By reversing the direction we obtain flow-graph for DST-VI.

We now show how these steps can be carried out for construction of fast transforms of length N=4. We start with mapping (1). Then, we pick fast factorization of DFT of length 9. In this case, we use Winograd’s DFT module of length 9 described in [17, 18]. We show flow-graph of this algorithm in Figure 3. We use red color to show paths that are needed for computation of DST-VII. It can be easily observed that the remaining paths are irrelevant because they either receive zero input, or lead to real portion of DFT’s output. Final flow-graph for computing DST-VII is show in Figure 2. Based on Figure 2 we can see that DST-VII of length 4 can be computed by using only 5 multiplications and 11 additions. Same complexity is required for computing DST-VI of length 4.

Same steps can also be repeated for construction of fast transforms of length N=8. In this case, we can use 17-point Winograd DFT module described in [18, 19]. We show the final flow-graph of length-8 DST-VII in Figure 4. This transform requires 21 multiplications and 77 additions.

5. REFERENCES

[1] J. Han, A. Saxena, and K. Rose, “Towards jointly optimal spatial prediction and adaptive transform in video/image coding,” in *IEEE Int. Conf. Acoust., Speech, Signal Processing (ICASSP)*, March 2010, pp. 726–729.

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

[3] R. K. Chivukula and Y. A. Reznik, “Efficient implementation of a class of MDCT/IMDCT filterbanks for speech and audio coding applications,” in *IEEE Int. Conf. Acoust., Speech, Signal Processing (ICASSP)*, July 2008, pp. pp. 213–216.

[4] V. Britanak, K. R. Rao, and P. Yip, *Discrete Cosine and Sine Transforms: General Properties, Fast Algorithms and Integer Approximations*, Academic Press - Elsevier, Oxford, UK, 2007.

[5] M. T. Heideman, “Computation of an odd-length DCT from a real-valued DFT of the same length,” *IEEE Trans. Signal Processing*, vol. 40, no. 1, pp. 54–61, 1992.

[6] C. W. Kok, “Fast algorithm for computing Discrete Cosine Transform,” *IEEE Trans. Signal Processing*, vol. 45, no. 3, pp. 757–760, 1997.

[7] E. Feig and S. Winograd, “On the multiplicative complexity of Discrete Cosine Transforms (Corresp.),” *IEEE Trans. Information Theory*, vol. IT-38, pp. 1387–1391, 1992.

[8] C. Loeffler, A. Ligtenberg, and G.S. Moschytz, “Practical fast 1-D DCT algorithms with 11 multiplications,” in *IEEE Int. Conf. Acoust., Speech, Signal Processing (ICASSP)*, May 1989, vol. 2, pp. 988–991.

[9] Y. A. Reznik and R. K. Chivukula, “Design of fast transforms for high-resolution image and video coding,” in *Applications of Digital Image Processing XXXII*, A. G. Tescher, Ed., 2009, vol. 7443 of *Proc. SPIE*, pp. 744312–1–18.

[10] Y. Reznik R. Joshi and M. Karczewicz, “Efficient large size transforms for high-performance video coding,” in *Applications of Digital Image Processing XXXIII*, A. G. Tescher, Ed., August 2010, vol. 7798 of *Proc. SPIE*, pp. 779831–1–7.

[11] A. K. Jain, “A sinusoidal family of unitary transforms,” *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. PAMI-1, no. 4, pp. 356 –365, Oct. 1979.

[12] Z. Wang and B. R. Hunt, “The discrete W transform,” *Applied Mathematics and Computation*, vol. 16, no. 1, pp. 19 – 48, 1985.

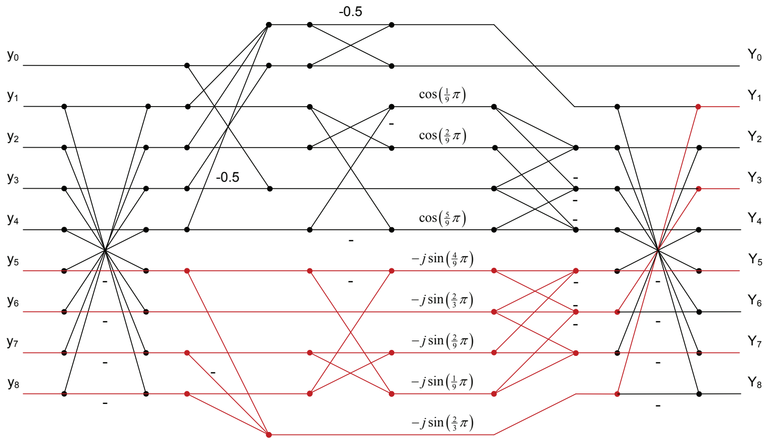

Fig. 3. Flow-graph of Winograd's factorization of DFT of length 9. Paths that are needed for computation of DST-VII are shown in red.

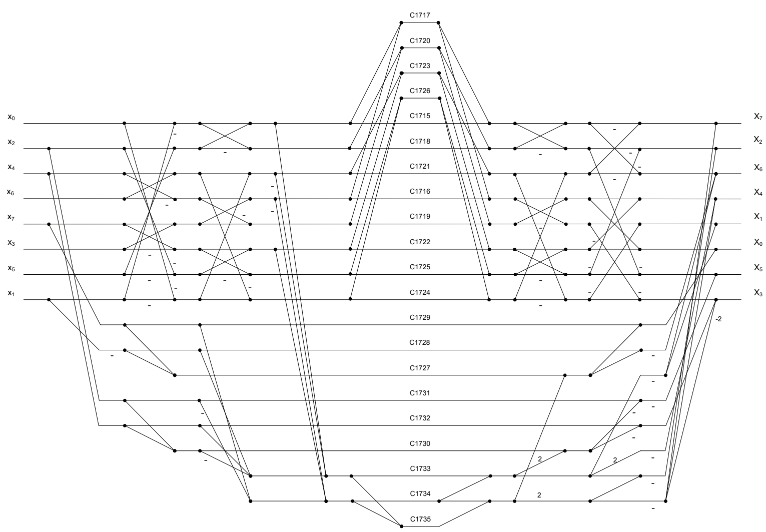

Fig. 4. Flow-graph of fast factorization of DST-VII of length 8. Factors $C1715 - C1735$ correspond to constants appearing in length $N = 17$ Winograd DFT factorization described in [19].

[13] S. A. Martucci, "Symmetric convolution and the Discrete Sine and Cosine Transforms," *IEEE Trans. Signal Processing*, vol. SP-42, pp. 1038–1051, 1994.

[14] A. Saxena and F. Fernandes, "CE7: Mode-dependent DCT/DST for intra prediction in video coding," commitee input document JCTVC-D033, ISO/IEC/ITU-T Joint Collaborative Team on Video Coding, Daegu, Korea, January 2011.

[15] B. Bross, W.-J. Han, G. S. Sullivan, J.-R. Ohm, and T. Wiegand, Eds., *WD4: Working Draft 4 of High-Efficiency Video Coding*, committee document JCTVC-F803. ISO/IEC/ITU-T Joint Collaborative Team on Video Coding, Turin, Italy, July 2011.

[16] K. McCann, B. Bross, W.-J. Han, S. Sekiguchi, and G. S. Sullivan, Eds., *HM4: HEVC Test Model 4 Encoder Description*, committee document JCTVC-F802. ISO/IEC/ITU-T Joint Collaborative Team on Video Coding, Turin, Italy, July 2011.

[17] S. Winograd, "On computing the Discrete Fourier Transform," *Mathematics of Computation*, vol. 32, no. 141, pp. 175–199, January 1978.

[18] C. Burrus, *Appendix 4, Programs for Short FFTs*, Connexions, available online at: http://cnx.org/content/m17646/1.4/, September 2009.

[19] C. Burrus and Y. A. Reznik, *N = 17 Winograd FFT module in C*, Connexions, available online at: http://cnx.org/content/m40959/1.1/, August 2011.