Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2012, Article ID 912194, 31 pages
doi:10.1155/2012/912194

Research Article

A Unified Algorithm for Subband-Based Discrete Cosine Transform

Lu-Ting Ko,[1] Jwu-E Chen,[1] Hsi-Chin Hsin,[2] Yaw-Shih Shieh,[3] and Tze-Yun Sung[3]

[1] *Department of Electrical Engineering, National Central University, Chungli City 320-01, Taiwan*
[2] *Department of Computer Science and Information Enginering, National United University, Miaoli City 360-03, Taiwan*
[3] *Department of Electronics Engineering, Chung Hua University, Hsinchu City 300-12, Taiwan*

Correspondence should be addressed to Tze-Yun Sung, bobsung@chu.edu.tw

Received 13 April 2011; Revised 22 May 2011; Accepted 4 June 2011

Academic Editor: Shengyong Chen

Copyright © 2012 Lu-Ting Ko et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Discrete cosine transform (DCT) and inverse DCT (IDCT) have been widely used in many image processing systems and real-time computation of nonlinear time series. In this paper, the unified DCT/IDCT algorithm based on the subband decompositions of a signal is proposed. It is derived from the data flow of subband decompositions with factorized coefficient matrices in a recursive manner. The proposed algorithm only requires $(4^{(\log_2 n)-1} - 1)$ and $(4^{(\log_2 n)-1} - 1)/3$ multiplication time for n-point DCT and IDCT, with a single multiplier and a single processor, respectively. Moreover, the peak signal-to-noise ratio (PSNR) of the proposed algorithm outperforms the conventional DCT/IDCT. As a result, the subband-based approach to DCT/IDCT is preferable to the conventional approach in terms of computational complexity and system performance. The proposed reconfigurable architecture of linear array DCT/IDCT processor has been implemented by FPGA.

1. Introduction

The discrete cosine transform (DCT) first proposed by Ahemd et al. [1] is a Fourier-like transform. While the Fourier transform decomposes a signal into sine and cosine functions, DCT only makes use of cosine functions with the property of high energy compaction. As DCT is preferable for a trade-off between the optimal decorrelation known as the Karhunen-Loève transform and computational simplicity [2], it has been extensively used in many applications [3–14]. In particular, two-dimensional (2D) DCT, such as 8×8 DCT, has been adopted in some international standards such as JPEG, MPEG, and H.264 [15]. In MP3

audio codec, the subband analysis and synthesis filter banks requires the use of 32-point DCT/integer DCT to expedite computation [16]. Other audio compression standards, for example, the Dolby Digital AC-3 codec, utilize a modified DCT with 256 or 512 data points.

Many algorithms have been proposed for DCT/IDCT [17–21]. In which, the transportation matrix is factorized into products of simpler matrices. It is noted that, however, the factorized matrices are no longer as regular as the fast Fourier transform (FFT); thus, these algorithms can only achieve moderate computational speed. Specifically, the dedicated data paths deduced from the signal flow graphs (SFGs) of the above algorithms need to be optimized for performance enhancement, which is computationally intensive, and the custom-designed DCT is often complicated and cannot be easily scalable for variable data points.

In this paper, we propose a novel linear-array architecture based on the subband decomposition of a signal for scalable DCT/IDCT. The remainder of this paper proceeds as follows. First, the subband-based 8-point DCT/IDCT algorithm [22] is reviewed in Section 2. Its extension to n-point DCT/IDCT called the unified subband-based algorithm is proposed in Section 3. Section 4 presents the analysis of system complexity. The reconfigurable architecture of linear-array DCT/IDCT processor implemented by FPGA (field programmable gate array) is proposed in Section 5, and the conclusion can be found in Section 6.

2. The Subband-Based 8-Point DCT/IDCT Algorithm

The discrete cosine transform (DCT) of an 8-point signal, [8], is defined as

$$c[k] = \alpha[k] \sum_{n=0}^{7} x[n] \cos\left[\frac{(2n+1)k\pi}{16}\right], \quad k = 0, 1, \ldots, 7, \tag{2.1}$$

where $\alpha[0] = 1/(2\sqrt{2})$, and $\alpha[k] = 1/2$ for $k > 0$. It can be rewritten in the following matrix form:

$$\mathbf{c}_8 = \mathbf{D}_8 \cdot \mathbf{x}_8, \tag{2.2}$$

where $\mathbf{c}_8 = [c[0] \cdots c[7]]^T$, $\mathbf{x}_8 = [x[0] \cdots x[7]]^T$, and the transformation matrix $\mathbf{D}_8$ is as follows:

$$\mathbf{D}_8 = \frac{1}{\sqrt{8}} \cdot \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ a & c & d & f & -f & -d & -c & -a \\ b & e & -e & -b & -b & -e & e & b \\ c & -f & -a & -d & d & a & f & -c \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ d & -a & f & c & -c & -f & a & -d \\ e & -b & b & -e & -e & b & -b & e \\ f & -d & c & -a & a & -c & d & -f \end{bmatrix}, \tag{2.3}$$

where $a = \sqrt{2}\cos(\pi/16)$, $b = \sqrt{2}\cos(2\pi/16)$, $c = \sqrt{2}\cos(3\pi/16)$, $d = \sqrt{2}\cos(5\pi/16)$, $e = \sqrt{2}\cos(6\pi/16)$, and $f = \sqrt{2}\cos(7\pi/16)$.

Let $x_L[n]$ and $x_H[n]$ denote the low-frequency and high-frequency subband signals of $x[n]$, respectively [22], which can be obtained by

$$\begin{aligned} x_L[n] &= \frac{1}{2}\{x[2n] + x[2n+1]\}, \\ x_H[n] &= \frac{1}{2}\{x[2n] - x[2n+1]\}, \end{aligned} \tag{2.4}$$

where $n = 0, 1, 2, 3$. As one can see, the DCT of $x[8]$ can be rewritten as

$$\begin{aligned} c[k] &= \sum_{n=0}^{3} \alpha[k]x[2n]\cos\left(\frac{(4n+1)k\pi}{16}\right) + \sum_{n=0}^{3} \alpha[k]x[2n+1]\cos\left(\frac{(4n+3)k\pi}{16}\right) \\ &= \underbrace{2\cos\left(\frac{\pi k}{16}\right)\sum_{n=0}^{3} \alpha[k]x_L[n]\cos\left(\frac{(2n+1)k\pi}{8}\right)}_{c_L[k]} \\ &\quad + \underbrace{2\sin\left(\frac{\pi k}{16}\right)\sum_{n=0}^{3} \alpha[k]x_H[n]\sin\left(\frac{(2n+1)k\pi}{8}\right)}_{s_H[k]}, \end{aligned} \tag{2.5}$$

where $c_L[k]$ and $s_H[k]$ are the subbands DCT and DST (discrete sine transform) of $x[n]$, respectively. Its vector form is as follows:

$$\begin{aligned} \mathbf{c}_8 &= \left[\mathbf{T}_{\text{SB\_DCT},8} \;\; \mathbf{T}_{\text{SB\_DCT},8}\right]_{8\times 8} \cdot \mathbf{M}_8 \cdot \mathbf{x}_8 \\ &= \left[\mathbf{T}_{\text{SB\_DCT},8} \;\; \mathbf{T}_{\text{SB\_DCT},8}\right]_{8\times 8} \cdot \begin{bmatrix} \mathbf{x}_L \\ \mathbf{x}_H \end{bmatrix}_{8\times 1} \\ &= \underbrace{\mathbf{T}_{\text{SB\_DCT},8} \cdot \mathbf{x}_L}_{\widehat{\mathbf{c}}_{L,8}} + \underbrace{\mathbf{T}_{\text{SB\_DST},8} \cdot \mathbf{x}_H}_{\widehat{\mathbf{s}}_{H,8}}, \end{aligned} \tag{2.6}$$

where $\mathbf{T}_{\text{SB\_DCT},8}$ and $\mathbf{T}_{\text{SB\_DST},8}$ denote the 8×4 matrices of the subband DCT and DST, respectively, $\mathbf{x}_L = [x_L[0] \cdots x_L[3]]^T$, $\mathbf{x}_H = [x_H[0] \;\; \cdots x_H[3]]^T$ and $\mathbf{c}_8 = [c[0] \cdots c[7]]^T$.

According to (2.4), the 8-point matrix $\mathbf{M}_8$ can be written as

$$\mathbf{M}_8 = \begin{bmatrix} 0.5 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.5 & 0.5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.5 & 0.5 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.5 & 0.5 \\ 0.5 & -0.5 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.5 & -0.5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.5 & -0.5 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.5 & -0.5 \end{bmatrix}. \tag{2.7}$$

Due to the orthogonality between $\mathbf{T}_{\text{SB\_DCT},8}$ and $\mathbf{T}_{\text{SB\_DST},8}$, $x_L[n]$ and $x_H[n]$ can be obtained from $c[k]$ by

$$\begin{aligned} x_L[n] &= \sum_{n=0}^{7} \alpha[k] \cos\left(\frac{\pi k}{16}\right) c[k] \cos\left(\frac{(2n+1)k\pi}{8}\right), \quad n = 0, \ldots, 7, \\ x_H[n] &= \sum_{n=0}^{7} \alpha[k] \sin\left(\frac{\pi k}{16}\right) c[k] \sin\left(\frac{(2n+1)k\pi}{8}\right), \quad n = 0, \ldots, 7. \end{aligned} \tag{2.8}$$

In [22], the multistage subband decomposition is as follows. $\mathbf{c}_{L,4}$ and $\mathbf{c}_{H,4}$ denote the DCTs of $\mathbf{x}_L$ and $\mathbf{x}_H$, respectively, which can be obtained by $\mathbf{c}_{L,4} = \mathbf{D}_4 \cdot \mathbf{x}_L$ and $\mathbf{c}_{H,4} = \mathbf{D}_4 \cdot \mathbf{x}_H$, where $\mathbf{D}_4$ is the transformation matrix

$$\mathbf{D}_4 = \frac{1}{\sqrt{4}} \cdot \begin{bmatrix} 1 & 1 & 1 & 1 \\ \sqrt{2}\cos\frac{\pi}{8} & \sqrt{2}\cos\frac{3\pi}{8} & -\sqrt{2}\cos\frac{3\pi}{8} & -\sqrt{2}\cos\frac{\pi}{8} \\ 1 & -1 & -1 & 1 \\ \sqrt{2}\cos\frac{3\pi}{8} & -\sqrt{2}\cos\frac{\pi}{8} & \sqrt{2}\cos\frac{\pi}{8} & -\sqrt{2}\cos\frac{3\pi}{8} \end{bmatrix}. \tag{2.9}$$

We have

$$\begin{aligned}
\widehat{\mathbf{c}}_{L,8} &= \mathbf{T}_{\text{SB_DCT},8} \cdot \mathbf{D}_4^{-1} \cdot \mathbf{c}_{L,4} = \begin{bmatrix} 1.412 & 0 & 0 & 0 \\ 0 & 1.3870 & 0 & 0 \\ 0 & 0 & 1.3066 & 0 \\ 0 & 0 & 0 & 1.1759 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -0.7857 \\ 0 & 0 & -0.5412 & 0 \\ 0 & -0.2759 & 0 & 0 \end{bmatrix} \cdot \mathbf{c}_{L,4}, \\
\widehat{\mathbf{s}}_{H,8} &= \mathbf{T}_{\text{SB_DST},8} \cdot \mathbf{D}_4{}^{-1} \cdot \mathbf{c}_{H,4} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0.2549 & 0 & -0.1056 & 0 \\ 0 & 0.5 & 0 & -0.2071 \\ 0.3007 & 0 & 0.7259 & 0 \\ 0 & 0.5412 & 0 & 1.3066 \\ 0.4500 & 0 & 1.0864 & 0 \\ 0 & 1.2071 & 0 & -0.5 \\ 1.2815 & 0 & -0.5308 & 0 \end{bmatrix} \cdot \mathbf{c}_{H,4}.
\end{aligned} \tag{2.10}$$

Similarly, the 4-point DCT computations of $\mathbf{c}_{L,4}$ and $\mathbf{c}_{H,4}$ can be obtained by

$$\begin{aligned}
\mathbf{c}_{L,4} &= \left[\mathbf{T}_{\text{SB_DCT},4} \;\; \mathbf{T}_{\text{SB_DCT},4}\right]_{4\times 4} \cdot \mathbf{M}_4 \cdot \mathbf{x}_L \\
&= \underbrace{\mathbf{T}_{\text{SB_DCT},4} \cdot \mathbf{x}_{LL}}_{\widehat{\mathbf{c}}_{LL,4}} + \underbrace{\mathbf{T}_{\text{SB_DST},4} \cdot \mathbf{x}_{LH}}_{\widehat{\mathbf{s}}_{LH,4}},
\end{aligned} \tag{2.11}$$

$$\begin{aligned}
\mathbf{c}_{H,4} &= \left[\mathbf{T}_{\text{SB_DCT},4} \;\; \mathbf{T}_{\text{SB_DCT},4}\right]_{4\times 4} \cdot \mathbf{M}_4 \cdot \mathbf{x}_H \\
&= \underbrace{\mathbf{T}_{\text{SB_DCT},4} \cdot \mathbf{x}_{HL}}_{\widehat{\mathbf{c}}_{HL,4}} + \underbrace{\mathbf{T}_{\text{SB_DST},4} \cdot \mathbf{x}_{HH}}_{\widehat{\mathbf{s}}_{HH,4}},
\end{aligned} \tag{2.12}$$

where $\mathbf{x}_{LL} = [x_{LL}[0] \; x_{LL}[1]]^T$, $\mathbf{x}_{LH} = [x_{LH}[0] \; x_{LH}[1]]^T$, $\mathbf{x}_{HL} = [x_{HL}[0] \; x_{HL}[1]]^T$, $\mathbf{x}_{HH} = [x_{HH}[0] \;\; x_{HH}[1]]^T$, and

$$x_{LL}[n] = \frac{1}{2}\{x_L[2n] + x_L[2n+1]\}, \tag{2.13}$$

$$x_{LH}[n] = \frac{1}{2}\{x_L[2n] - x_L[2n+1]\}, \tag{2.14}$$

$$x_{HL}[n] = \frac{1}{2}\{x_H[2n] + x_H[2n+1]\}, \tag{2.15}$$

$$x_{HH}[n] = \frac{1}{2}\{x_H[2n] - x_H[2n+1]\} \tag{2.16}$$

for $n = 0, 1$.

The 4-point distributed matrix $\mathbf{M}_4$, can be defined as

$$\mathbf{M}_4 = \begin{bmatrix} 0.5 & 0.5 & 0 & 0 \\ 0 & 0 & 0.5 & 0.5 \\ 0.5 & -0.5 & 0 & 0 \\ 0 & 0 & 0.5 & -0.5 \end{bmatrix}. \tag{2.17}$$

Let $\mathbf{c}_{LL,2}$, $\mathbf{c}_{LH,2}$, $\mathbf{c}_{HL,2}$, and $\mathbf{c}_{HH,2}$ be the 2-point DCT of $\mathbf{x}_{LL}$, $\mathbf{x}_{LH}$, $\mathbf{x}_{HL}$, and $\mathbf{x}_{HH}$, respectively, which can be computed by using the following 2-point transformation matrix, $\mathbf{D}_2$:

$$\mathbf{D}_2 = \frac{1}{\sqrt{2}} \cdot \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}. \tag{2.18}$$

We have

$$\widehat{\mathbf{c}}_{LL,4} = \mathbf{T}_{\text{SB\_DCT},4} \cdot \mathbf{x}_{LL} = \mathbf{T}_{\text{SB\_DCT},4} \cdot \mathbf{D}_2^{-1} \cdot \mathbf{c}_{LL,2} = \begin{bmatrix} 1.4142 & 0 \\ 0 & 1.3066 \\ 0 & 0 \\ 0 & -0.5412 \end{bmatrix} \cdot \mathbf{c}_{LL,2}, \tag{2.19}$$

$$\widehat{\mathbf{c}}_{HL,4} = \mathbf{T}_{\text{SB\_DCT},4} \cdot \mathbf{x}_{HL} = \mathbf{T}_{\text{SB\_DCT},4} \cdot \mathbf{D}_2^{-1} \cdot \mathbf{c}_{HL,2} = \begin{bmatrix} 1.4142 & 0 \\ 0 & 1.3066 \\ 0 & 0 \\ 0 & -0.5412 \end{bmatrix} \cdot \mathbf{c}_{HL,2}, \tag{2.20}$$

$$\widehat{\mathbf{s}}_{LH,4} = \mathbf{T}_{\text{SB\_DST},4} \cdot \mathbf{x}_{LH} = \mathbf{T}_{\text{SB\_DST},4} \cdot \mathbf{D}_2^{-1} \cdot \mathbf{c}_{LH,2} = \begin{bmatrix} 0 & 0 \\ 0.5412 & 0 \\ 0 & 1.4142 \\ 1.3066 & 0 \end{bmatrix} \cdot \mathbf{c}_{LH,2}, \tag{2.21}$$

$$\widehat{\mathbf{s}}_{HH,4} = \mathbf{T}_{\text{SB\_DST},4} \cdot \mathbf{x}_{HH} = \mathbf{T}_{\text{SB\_DST},4} \cdot \mathbf{D}_2^{-1} \cdot \mathbf{c}_{HH,2} = \begin{bmatrix} 0 & 0 \\ 0.5412 & 0 \\ 0 & 1.4142 \\ 1.3066 & 0 \end{bmatrix} \cdot \mathbf{c}_{HH,2}. \tag{2.22}$$

For the 2-point DCT computations of $\mathbf{c}_{LL,2}$, $\mathbf{c}_{LH,2}$, $\mathbf{c}_{HL,2}$, and $\mathbf{c}_{HH,2}$, let

$$x_{LLL}[n] = \frac{1}{2}\{x_{LL}[2n] + x_{LL}[2n+1]\}, \tag{2.23}$$

$$x_{LLH}[n] = \frac{1}{2}\{x_{LL}[2n] - x_{LL}[2n+1]\}, \tag{2.24}$$

$$x_{LHL}[n] = \frac{1}{2}\{x_{LH}[2n] + x_{LH}[2n+1]\}, \tag{2.25}$$

$$x_{LHH}[n] = \frac{1}{2}\{x_{LH}[2n] - x_{LH}[2n+1]\}, \tag{2.26}$$

$$x_{HLL}[n] = \frac{1}{2}\{x_{HL}[2n] + x_{HL}[2n+1]\}, \tag{2.27}$$

$$x_{HLH}[n] = \frac{1}{2}\{x_{HL}[2n] - x_{HL}[2n+1]\}, \tag{2.28}$$

$$x_{HHL}[n] = \frac{1}{2}\{x_{HH}[2n] + x_{HH}[2n+1]\}, \tag{2.29}$$

$$x_{HHH}[n] = \frac{1}{2}\{x_{HH}[2n] - x_{HH}[2n+1]\}, \tag{2.30}$$

where $n = 0$, we then have

$$\mathbf{c}_{LL,2} = \left[\mathbf{T}_{\text{SB\_DCT},2} \;\; \mathbf{T}_{\text{SB\_DCT},2}\right]_{2\times 2} \cdot \mathbf{M}_2 \cdot \mathbf{x}_{LL} = \underbrace{\mathbf{T}_{\text{SB\_DCT},2} \cdot \mathbf{x}_{LLL}}_{\widehat{\mathbf{c}}_{LLL,2}} + \underbrace{\mathbf{T}_{\text{SB\_DST},2} \cdot \mathbf{x}_{LLH}}_{\widehat{\mathbf{s}}_{LLH,2}}, \tag{2.31}$$

$$\mathbf{c}_{LH,2} = \left[\mathbf{T}_{\text{SB\_DCT},2} \;\; \mathbf{T}_{\text{SB\_DCT},2}\right]_{2\times 2} \cdot \mathbf{M}_2 \cdot \mathbf{x}_{LH} = \underbrace{\mathbf{T}_{\text{SB\_DCT},2} \cdot \mathbf{x}_{LHL}}_{\widehat{\mathbf{c}}_{LHL,2}} + \underbrace{\mathbf{T}_{\text{SB\_DST},2} \cdot \mathbf{x}_{LHH}}_{\widehat{\mathbf{s}}_{LHH,2}}, \tag{2.32}$$

$$\mathbf{c}_{HL,2} = \left[\mathbf{T}_{\text{SB\_DCT},2} \;\; \mathbf{T}_{\text{SB\_DCT},2}\right]_{2\times 2} \cdot \mathbf{M}_2 \cdot \mathbf{x}_{HL} = \underbrace{\mathbf{T}_{\text{SB\_DCT},2} \cdot \mathbf{x}_{HLL}}_{\widehat{\mathbf{c}}_{HLL,2}} + \underbrace{\mathbf{T}_{\text{SB\_DST},2} \cdot \mathbf{x}_{HLH}}_{\widehat{\mathbf{s}}_{HLH,2}}, \tag{2.33}$$

$$\mathbf{c}_{HH,2} = \left[\mathbf{T}_{\text{SB\_DCT},22} \;\; \mathbf{T}_{\text{SB\_DCT},2}\right]_{2\times 2} \cdot \mathbf{M}_2 \cdot \mathbf{x}_{HH} = \underbrace{\mathbf{T}_{\text{SB\_DCT},2} \cdot \mathbf{x}_{HHL}}_{\widehat{\mathbf{c}}_{HHL,2}} + \underbrace{\mathbf{T}_{\text{SB\_DST},2} \cdot \mathbf{x}_{HHH}}_{\widehat{\mathbf{s}}_{HHH,2}}. \tag{2.34}$$

$\mathbf{M}_2$ is a 2-point matrix defined as

$$\mathbf{M}_2 = \begin{bmatrix} 0.5 & 0.5 \\ 0.5 & -0.5 \end{bmatrix}. \tag{2.35}$$

Finally, according to equations (2.6)~(2.33) together with (2.4), we have

$$\mathbf{c}_8 = \widetilde{\mathbf{K}} \cdot \widehat{\mathbf{K}} \cdot \mathbf{M}_8 \cdot \widehat{\mathbf{M}}_8 \cdot \widetilde{\mathbf{M}}_8 \cdot \mathbf{x}_8, \tag{2.36}$$

where

$$\widetilde{\mathbf{K}} = \left[\left(\mathbf{T}_{\text{SB\_DCT},8} \cdot \mathbf{D}_4^{-1}\right)_{8\times 4} \ \left(\mathbf{T}_{\text{SB\_DCT},8} \cdot \mathbf{D}_4^{-1}\right)_{8\times 4} \right]_{8\times 8},$$

$$\widehat{\mathbf{K}} = \begin{bmatrix} \left(\mathbf{T}_{\text{SB\_DCT},4} \cdot \mathbf{D}_2^{-1}\right)_{4\times 2} & \left(\mathbf{T}_{\text{SB\_DCT},4} \cdot \mathbf{D}_2^{-1}\right)_{4\times 2} & \mathbf{0}_{4\times 2} & \mathbf{0}_{4\times 2} \\ \mathbf{0}_{4\times 2} & \mathbf{0}_{4\times 2} & \left(\mathbf{T}_{\text{SB\_DCT},4} \cdot \mathbf{D}_2^{-1}\right)_{4\times 2} & \left(\mathbf{T}_{\text{SB\_DCT},4} \cdot \mathbf{D}_2^{-1}\right)_{4\times 2} \end{bmatrix}_{8\times 8}. \tag{2.37}$$

The following matrix, $\widehat{\mathbf{M}}_8$, can derived from (2.12)~(2.15).

$$\widehat{\mathbf{M}}_8 = \begin{bmatrix} 0.5 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.5 & 0.5 & 0 & 0 & 0 & 0 \\ 0.5 & -0.5 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.5 & -0.5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.5 & 0.5 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.5 & 0.5 \\ 0 & 0 & 0 & 0 & 0.5 & -0.5 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.5 & -0.5 \end{bmatrix}. \tag{2.38}$$

Similarly, the following matrix, $\widetilde{\mathbf{M}}_8$, can be derived from (2.21)~(2.28)

$$\widetilde{\mathbf{M}}_8 = \begin{bmatrix} 0.5 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.5 & -0.5 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.5 & 0.5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.5 & -0.5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.5 & 0.5 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.5 & -0.5 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.5 & 0.5 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.5 & -0.5 \end{bmatrix}. \tag{2.39}$$

The decomposition matrix $\mathbf{R}_8$ can be defined as

$$\mathbf{R}_8 = 8 \cdot \mathbf{M}_8 \cdot \widehat{\mathbf{M}}_8 \cdot \widetilde{\mathbf{M}}_8 = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 \end{bmatrix} \tag{2.40}$$

and the coefficient matrix $\mathbf{F}_8$ can be defined as

$$\mathbf{F}_8 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.9061 & 0.3753 & 0 & 0.1802 & 0 & 0 & -0.0747 \\ 0 & 0 & 0 & 0.9239 & 0 & 0.3827 & 0 & 0 \\ 0 & -0.3182 & 0.7682 & 0 & 0.2126 & 0 & 0 & 0.5133 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0.2126 & -0.5133 & 0 & 0.3182 & 0 & 0 & 0.7682 \\ 0 & 0 & 0 & -0.3827 & 0 & 0.9239 & 0 & 0 \\ 0 & -0.1802 & -0.0747 & 1 & 0.9061 & 0 & 0 & -0.3753 \end{bmatrix}. \tag{2.41}$$

According to (2.34), (2.38), and (2.39), we have

$$\mathbf{c}_8 = \frac{\sqrt{2}}{4} \cdot \mathbf{F}_8 \cdot \mathbf{R}_8 \cdot \mathbf{x}_8. \tag{2.42}$$

The coefficient matrix $\mathbf{F}_8$ can be represented by the reordered coefficient matrix $\widehat{\mathbf{F}}_8$, prepermutation matrix $\widehat{\mathbf{T}}_8$ and post-permutation matrix $\widetilde{\mathbf{T}}_8$, and can be written as

$$\mathbf{F}_8 = \widehat{\mathbf{T}}_8 \cdot \widehat{\mathbf{F}}_8 \cdot \widetilde{\mathbf{T}}_8, \tag{2.43}$$

where the matrices $\widehat{\mathbf{F}}_8$, $\widehat{\mathbf{T}}_8$, and $\widetilde{\mathbf{T}}_8$ can be defined as

$$\widehat{\mathbf{F}}_8 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.9238 & 0.3826 & 0 & 0 & 0 & 0 \\ 0 & 0 & -0.3826 & 0.9238 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.9061 & 0.1802 & 0.3753 & -0.0746 \\ 0 & 0 & 0 & 0 & -0.1802 & 0.9061 & -0.0746 & -0.3753 \\ 0 & 0 & 0 & 0 & 0.2126 & 0.3181 & -0.5132 & 0.7682 \\ 0 & 0 & 0 & 0 & -0.3181 & 0.2126 & 0.7682 & 0.5132 \end{bmatrix},$$

$$\widehat{\mathbf{T}}_8 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}, \tag{2.44}$$

$$\widetilde{\mathbf{T}}_8 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

The reordered coefficient matrix $\widehat{\mathbf{F}}_8$ can be represented as

$$\widehat{\mathbf{F}}_8 = \begin{bmatrix} \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{A} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{B} & \mathbf{C} \\ \mathbf{0} & \mathbf{0} & \mathbf{D} & \mathbf{E} \end{bmatrix}. \tag{2.45}$$

The computation of sub-coefficient matrix **A** can be written as

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \mathbf{A} \cdot \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} a & b \\ -b & a \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}, \tag{2.46}$$

where $a = 0.9239$ and $b = 0.3827$. The above can be rewritten as [20]

$$y_1 = (b - a) \cdot x_2 + a \cdot (x_1 + x_2), \tag{2.47}$$

$$y_2 = -(a + b) \cdot x_1 + a \cdot (x_1 + x_2). \tag{2.48}$$

Thus, the number of multiplications can be reduced to 3 for matrix **A**; this technique can also be applied to matrices **B**, **C**, **D**, and **E**. As a result, the total number of multiplications of the subband-based 8-point DCT is only 15.

Based on (2.40) and (2.41), the corresponding subband-based IDCT can be obtained by

$$\mathbf{x}_8 = 2\sqrt{2} \cdot \mathbf{R}_8^{-1} \cdot \tilde{\mathbf{T}}_8^{-1} \cdot \hat{\mathbf{F}}_8^{-1} \cdot \hat{\mathbf{T}}_8^{-1} \cdot \mathbf{c}_8, \tag{2.49}$$

where $\mathbf{R}_8^{-1}$ is the inverse decomposition matrix. As the decomposition matrix $\mathbf{R}_8$ is orthonormal, $\mathbf{R}_8^{-1}$ can be derived from the transportation of $\mathbf{R}_8$

$$\mathbf{R}_8^{-1} = \mathbf{R}_8^T = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 \end{bmatrix}. \tag{2.50}$$

$\hat{\mathbf{F}}_8^{-1}$ is the inverse reordered coefficient matrix, $\hat{\mathbf{T}}_8^{-1}$ is the inverse prepermutation matrix and $\tilde{\mathbf{T}}_8^{-1}$ is the inverse post-permutation matrix. Since the reordered coefficient matrix $\hat{\mathbf{F}}_8$, prepermutation matrix $\hat{\mathbf{T}}_8$ and post-permutation matrix $\tilde{\mathbf{T}}_8$ are all orthonormal, three matrices $\hat{\mathbf{F}}_8^{-1}$, $\hat{\mathbf{T}}_8^{-1}$, and $\tilde{\mathbf{T}}_8^{-1}$ can be written as

$$\widehat{\mathbf{F}}_8^{-1} = \widehat{\mathbf{F}}_8^T = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.9238 & -0.3826 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.3826 & 0.9238 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.9061 & -0.1802 & 0.2126 & -0.3181 \\ 0 & 0 & 0 & 0 & 0.1802 & 0.9061 & 0.3181 & 0.2126 \\ 0 & 0 & 0 & 0 & 0.3753 & -0.0746 & -0.5132 & 0.7682 \\ 0 & 0 & 0 & 0 & -0.0746 & -0.3753 & 0.7682 & 0.5132 \end{bmatrix}, \tag{2.51}$$

$$\widehat{\mathbf{T}}_8^{-1} = \widehat{\mathbf{T}}_8^T = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}, \tag{2.52}$$

$$\widetilde{\mathbf{T}}_8^{-1} = \widetilde{\mathbf{T}}_8^T = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}. \tag{2.53}$$

The inverse reordered coefficient matrix $\widehat{\mathbf{F}}_8^{-1}$ can be represented as

$$\widehat{\mathbf{F}}_8^{-1} = \begin{bmatrix} \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{A}^T & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{B}^T & \mathbf{D}^T \\ \mathbf{0} & \mathbf{0} & \mathbf{C}^T & \mathbf{E}^T \end{bmatrix}. \tag{2.54}$$

As a result, the total number of multiplications of the subband based 8-point IDCT is only 15.

3. The Unified Subband-Based n-Point DCT/IDCT Algorithm

The subband-based DCT algorithm [22] can be unified for n-point DCT/IDCT due to the inherent regular pattern. For an n-point signal, $\mathbf{x}_n$, the unified subband-based discrete cosine transform can be defined as

$$\mathbf{c}_n = \frac{\sqrt{n}}{n} \cdot \widehat{\mathbf{T}}_n \cdot \widehat{\mathbf{F}}_n \cdot \widetilde{\mathbf{T}}_n \cdot \mathbf{R}_n \cdot \mathbf{x}_n, \tag{3.1}$$

where $n = \{2^m | m = 3, 4, 5, \ldots\}$, $\mathbf{R}_n$ is the decomposition matrix, $\widehat{\mathbf{F}}_n$ is the reordered coefficient matrix, $\widehat{\mathbf{T}}_n$ is the pre-permutation matrix, and $\widetilde{\mathbf{T}}_n$ is the post-permutation matrix. The unified decomposition matrix $\mathbf{R}_n$ can be written as

$$\begin{aligned}\mathbf{R}_n = n \cdot \mathbf{M}_{n,n\times n} \cdot \begin{bmatrix} \mathbf{M}_{n/2,(n/2)\times(n/2)} & \mathbf{0}_{(n/2)\times(n/2)} \\ \mathbf{0}_{(n/2)\times(n/2)} & \mathbf{M}_{n/2,(n/2)\times(n/2)} \end{bmatrix}_{n\times n} \\ \cdot \begin{bmatrix} \mathbf{M}_{n/4,(n/4)\times(n/4)} & \mathbf{0}_{(n/4)\times(n/4)} & \mathbf{0}_{(n/4)\times(n/4)} & \mathbf{0}_{(n/4)\times(n/4)} \\ \mathbf{0}_{(n/4)\times(n/4)} & \mathbf{M}_{n/4,(n/4)\times(n/4)} & \mathbf{0}_{(n/4)\times(n/4)} & \mathbf{0}_{(n/4)\times(n/4)} \\ \mathbf{0}_{(n/4)\times(n/4)} & \mathbf{0}_{(n/4)\times(n/4)} & \mathbf{M}_{n/4,[(n/4)\times(n/4)]} & \mathbf{0}_{(n/4)\times(n/4)} \\ \mathbf{0}_{(n/4)\times(n/4)} & \mathbf{0}_{(n/4)\times(n/4)} & \mathbf{0}_{(n/4)\times(n/4)} & \mathbf{M}_{n/4,(n/4)\times(n/4)} \end{bmatrix}_{n\times n} \cdots \\ \cdot \begin{bmatrix} \mathbf{M}_{2,2\times2} & \mathbf{0}_{2\times2} & \cdots & \mathbf{0}_{2\times2} \\ \mathbf{0}_{2\times2} & \mathbf{M}_{2,2\times2} & & \vdots \\ \vdots & & \ddots & \mathbf{0}_{2\times2} \\ \mathbf{0}_{2\times2} & \cdots & \mathbf{0}_{2\times2} & \mathbf{M}_{2,2\times2} \end{bmatrix}_{n\times n},\end{aligned} \tag{3.2}$$

where the basic 2×2 matrix, $\mathbf{M}_2$, consists of two submatrices, $\mathbf{M}'_{2,1\times2}$ and $\mathbf{M}''_{2,1\times2}$

$$\mathbf{M}_2 = \begin{bmatrix} 0.5 & 0.5 \\ 0.5 & -0.5 \end{bmatrix} = \begin{bmatrix} \mathbf{M}'_{2,1\times2} \\ \mathbf{M}''_{2,1\times2} \end{bmatrix}. \tag{3.3}$$

As noted, $\mathbf{M}_4$ can be represented by the sub-matrices of $\mathbf{M}_2$, or sub-matrices, $\mathbf{M}'_{4,2\times4}$ and $\mathbf{M}''_{4,2\times4}$ as follows:

$$\mathbf{M}_4 = \begin{bmatrix} 0.5 & 0.5 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & 0.5 & 0.5 \\ 0.5 & -0.5 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & 0.5 & -0.5 \end{bmatrix} = \begin{bmatrix} \mathbf{M}'_{2,1\times2} & \mathbf{0}_{1\times2} \\ \mathbf{0}_{1\times2} & \mathbf{M}'_{2,1\times2} \\ \mathbf{M}''_{2,1\times2} & \mathbf{0}_{1\times2} \\ \mathbf{0}_{1\times2} & \mathbf{M}''_{2,1\times2} \end{bmatrix} = \begin{bmatrix} \mathbf{M}'_{4,2\times4} \\ \mathbf{M}''_{4,2\times4} \end{bmatrix}. \tag{3.4}$$

According to (3.1) and (3.2), the unified distributed matrix $\mathbf{M}_n$ can be derived as

$$\mathbf{M}_n = \begin{bmatrix} \mathbf{M}'_{n/2,(n/4)\times(n/2)} & \mathbf{0}_{(n/4)\times(n/2)} \\ \mathbf{0}_{(n/4)\times(n/2)} & \mathbf{M}'_{n/2,(n/4)\times(n/2)} \\ \mathbf{M}''_{n/2,(n/4)\times(n/2)} & \mathbf{0}_{(n/4)\times(n/2)} \\ \mathbf{0}_{(n/4)\times(n/2)} & \mathbf{M}''_{n/2,(n/4)\times(n/2)} \end{bmatrix} = \begin{bmatrix} \mathbf{M}'_{n,(n/2)\times n} \\ \mathbf{M}''_{n,(n/2)\times n} \end{bmatrix}. \tag{3.5}$$

According to (2.2) and (2.40), we have

$$\mathbf{F}_n = \mathbf{D}_n \cdot \mathbf{R}_n^{-1}. \tag{3.6}$$

The 4-point reordered coefficient matrix $\widehat{\mathbf{F}}_4$ can be derived as

$$\widehat{\mathbf{F}}_{4,4\times4} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0.9239 & 0.3827 \\ 0 & 0 & -0.3827 & 0.9239 \end{bmatrix} = \begin{bmatrix} \widehat{\mathbf{F}}_{2,2\times2} & \mathbf{0}_{2\times2} \\ \mathbf{0}_{2\times2} & \widehat{\mathbf{F}}'_{4,2\times2} \end{bmatrix}. \tag{3.7}$$

According to (3.5), the 8-point reordered coefficient matrix $\widehat{\mathbf{F}}_8$ can be derived as

$$\widehat{\mathbf{F}}_{8,8\times8} = \begin{bmatrix} \widehat{\mathbf{F}}_{2,2\times2} & \mathbf{0}_{2\times2} & \mathbf{0}_{2\times4} \\ \mathbf{0}_{2\times2} & \widehat{\mathbf{F}}'_{4,2\times2} & \mathbf{0}_{2\times4} \\ \mathbf{0}_{4\times2} & \mathbf{0}_{4\times2} & \widehat{\mathbf{F}}'_{8,4\times4} \end{bmatrix} = \begin{bmatrix} \widehat{\mathbf{F}}_{4,4\times4} & \mathbf{0}_{4\times4} \\ \mathbf{0}_{4\times4} & \widehat{\mathbf{F}}'_{8,4\times4} \end{bmatrix}. \tag{3.8}$$

Hence, the unified reordered coefficient matrix can be written as

$$\widehat{\mathbf{F}}_{n,n\times n} = \begin{bmatrix} \widehat{\mathbf{F}}_{2,2\times2} & \mathbf{0}_{2\times2} & \cdots & \mathbf{0}_{2\times n} \\ \mathbf{0}_{2\times2} & \widehat{\mathbf{F}}'_{4,2\times2} & & \vdots \\ \vdots & & \ddots & \mathbf{0}_{(n/2)\times n} \\ \mathbf{0}_{n\times2} & \cdots & \mathbf{0}_{n\times(n/2)} & \widehat{\mathbf{F}}'_{n,(n/2)\times(n/2)} \end{bmatrix} = \begin{bmatrix} \widehat{\mathbf{F}}_{n/2,(n/2)\times(n/2)} & \mathbf{0}_{n/2,(n/2)\times(n/2)} \\ \mathbf{0}_{n/2,(n/2)\times(n/2)} & \widehat{\mathbf{F}}'_{n,(n/2)\times(n/2)} \end{bmatrix}. \tag{3.9}$$

The 4-point pre-permutation matrix $\widehat{\mathbf{T}}_4$ and post-permutation matrix $\widetilde{\mathbf{T}}_4$ can be written as

$$\widehat{\mathbf{T}}_4 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \left[\widehat{\mathbf{T}}_{41,4\times1} \;\; \widehat{\mathbf{T}}_{42,4\times1} \;\; \widehat{\mathbf{T}}_{43,4\times1} \;\; \widehat{\mathbf{T}}_{44,4\times1}\right],$$
$$\widetilde{\mathbf{T}}_4 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \end{bmatrix} = \left[\widetilde{\mathbf{T}}_{41,4\times1} \;\; \widetilde{\mathbf{T}}_{42,4\times1} \;\; \widetilde{\mathbf{T}}_{43,4\times1} \;\; \widetilde{\mathbf{T}}_{44,4\times1}\right]. \tag{3.10}$$

According to (2.50), (2.51), (3.8), and (3.9), the 8-point pre- and post-permutation matrices can be written as

$$\widehat{\mathbf{T}}_8 = \begin{bmatrix} \widehat{\mathbf{T}}_{41,4\times1} & \mathbf{0}_{4\times1} & \widehat{\mathbf{T}}_{42,4\times1} & \mathbf{0}_{4\times1} & \widehat{\mathbf{T}}_{43,4\times1} & \mathbf{0}_{4\times1} & \widehat{\mathbf{T}}_{44,4\times1} & \mathbf{0}_{4\times1} \\ \mathbf{0}_{4\times1} & \widehat{\mathbf{T}}'_{81,4\times1} & \mathbf{0}_{4\times1} & \widehat{\mathbf{T}}'_{82,4\times1} & \mathbf{0}_{4\times1} & \widehat{\mathbf{T}}'_{83,4\times1} & \mathbf{0}_{4\times1} & \widehat{\mathbf{T}}'_{84,4\times1} \end{bmatrix},$$
$$\widetilde{\mathbf{T}}_8 = \begin{bmatrix} \widetilde{\mathbf{T}}_{41,4\times1} & \mathbf{0}_{4\times1} & \widetilde{\mathbf{T}}_{42,4\times1} & \mathbf{0}_{4\times1} & \widetilde{\mathbf{T}}_{43,4\times1} & \mathbf{0}_{4\times1} & \widetilde{\mathbf{T}}_{44,4\times1} & \mathbf{0}_{4\times1} \\ \mathbf{0}_{4\times1} & \widetilde{\mathbf{T}}'_{81,4\times1} & \mathbf{0}_{4\times1} & \widetilde{\mathbf{T}}'_{82,4\times1} & \mathbf{0}_{4\times1} & \widetilde{\mathbf{T}}'_{83,4\times1} & \mathbf{0}_{4\times1} & \widetilde{\mathbf{T}}'_{84,4\times1} \end{bmatrix}. \tag{3.11}$$

Hence, the unified pre- and post-permutation matrices can be represented as

$$\widehat{\mathbf{T}}_n = \begin{bmatrix} \widehat{\mathbf{T}}_{(n/2)1,\mathcal{A}} & \mathbf{0}_{\mathcal{A}} & \widehat{\mathbf{T}}_{(n/2)2,\mathcal{A}} & \mathbf{0}_{\mathcal{A}} & \widehat{\mathbf{T}}_{(n/2)3,\mathcal{A}} & \mathbf{0}_{\mathcal{A}} & \widehat{\mathbf{T}}_{(n/2)4,\mathcal{A}} & \mathbf{0}_{\mathcal{A}} \\ \mathbf{0}_{\mathcal{A}} & \widehat{\mathbf{T}}'_{n1,\mathcal{A}} & \mathbf{0}_{\mathcal{A}} & \widehat{\mathbf{T}}'_{n2,\mathcal{A}} & \mathbf{0}_{\mathcal{A}} & \widehat{\mathbf{T}}'_{n3,\mathcal{A}} & \mathbf{0}_{\mathcal{A}} & \widehat{\mathbf{T}}'_{n4,\mathcal{A}} \end{bmatrix},$$
$$\widetilde{\mathbf{T}}_n = \begin{bmatrix} \widetilde{\mathbf{T}}_{(n/2)1,\mathcal{A}} & \mathbf{0}_{\mathcal{A}} & \widetilde{\mathbf{T}}_{(n/2)2,\mathcal{A}} & \mathbf{0}_{\mathcal{A}} & \widetilde{\mathbf{T}}_{(n/2)3,\mathcal{A}} & \mathbf{0}_{\mathcal{A}} & \widetilde{\mathbf{T}}_{(n/2)4,\mathcal{A}} & \mathbf{0}_{\mathcal{A}} \\ \mathbf{0}_{(n/2)\times(n/8)} & \widetilde{\mathbf{T}}'_{(n/2)1,\mathcal{A}} & \mathbf{0}_{\mathcal{A}} & \widetilde{\mathbf{T}}'_{(n/2)2,\mathcal{A}} & \mathbf{0}_{\mathcal{A}} & \widetilde{\mathbf{T}}'_{(n/2)3,\mathcal{A}} & \mathbf{0}_{\mathcal{A}} & \widetilde{\mathbf{T}}'_{(n/2)4,\mathcal{A}} \end{bmatrix}. \tag{3.12}$$

where $\mathcal{A}$ denotes $(n/2) \times (n/8)$.
According to (2.47) and (2.53), the unified subband-based IDCT can be obtained.

$$\mathbf{x}_n = \frac{n}{\sqrt{n}} \cdot \mathbf{R}_n^{-1} \cdot \widetilde{\mathbf{T}}_n^{-1} \cdot \widehat{\mathbf{F}}_n^{-1} \cdot \widehat{\mathbf{T}}_n^{-1} \cdot \mathbf{c}_n, \tag{3.13}$$

where $\mathbf{R}_n^{-1}$ is the inverse decomposition matrix, $\widehat{\mathbf{F}}_n^{-1}$ is the inverse coefficient matrix, $\widehat{\mathbf{T}}_n^{-1}$ is the inverse pre-transportation matrix, and $\widetilde{\mathbf{T}}_n^{-1}$ is the inverse post-transportation matrix. Note that

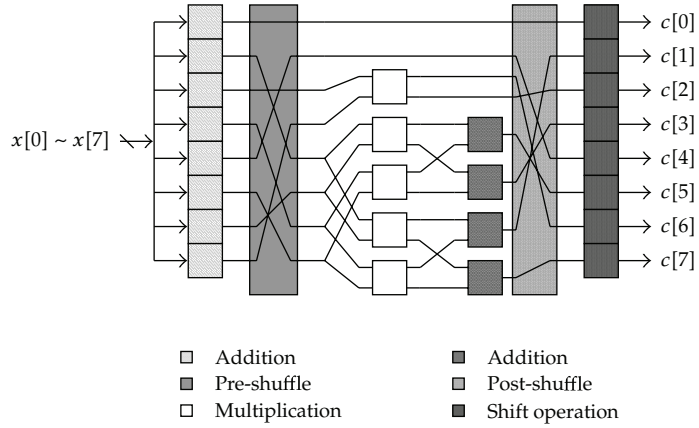


Figure 1: Data flow of the subband-based 8-point DCT with six pipelined stages.

the decomposition matrix, reordered coefficient matrix, pre- and post-permutation matrices are all orthonormal. Hence, we have

$$\begin{aligned}
\mathbf{R}_n^{-1} &= \mathbf{R}_n^T, \\
\widehat{\mathbf{F}}_n^{-1} &= \widehat{\mathbf{F}}_n^T, \\
\widehat{\mathbf{T}}_n^{-1} &= \widehat{\mathbf{T}}_n^T, \\
\widetilde{\mathbf{T}}_n^{-1} &= \widetilde{\mathbf{T}}_n^T.
\end{aligned} \tag{3.14}$$

4. Analysis of Computation Complexity and System Performance

Based on the 8-point subband-based DCT and IDCT algorithm, the data flow of parallel-pipelined processing for 8-point DCT and IDCT are described as follows. The data flow of the subband-based 8-point DCT with six pipelined stages is shown in Figure 1. In which, $\mathbf{y}_8 = \mathbf{R}_8 \cdot \mathbf{x}_8$, $\mathbf{z}_8 = \widehat{\mathbf{T}}_8 \cdot \widehat{\mathbf{F}}_8 \cdot \widetilde{\mathbf{T}}_8 \cdot \mathbf{y}_8$ and $\mathbf{c}_8 = (\sqrt{2}/4) \cdot \mathbf{z}_8$, the matrix-vector multiplication of $\mathbf{R}_8 \cdot \mathbf{x}_8$ in the first stage, takes one simple-addition time for each element of $\mathbf{y}_8$. The preshuffle performs the prepermutation matrix $\widetilde{\mathbf{T}}_8$ operation in the second stage. The matrix-vector multiplication is used to compute $\widehat{\mathbf{F}}_8 \cdot \mathbf{y}_8$ in the third and fourth stages. In the fifth stage, the postshuffle is used for the post-permutation matrix $\widehat{\mathbf{T}}_8$. The final stage is to compute $(\sqrt{2}/4) \cdot \mathbf{z}_8$ by using simple shift operation with the Booth recoded algorithm.

Similarly, Figure 2 shows the data flow of the subband-based 8-point IDCT with seven pipelined stages.

In which, $\mathbf{u}_8 = (\sqrt{2}/4) \cdot \mathbf{c}_8$, $\mathbf{w}_8 = \widehat{\mathbf{F}}_8^{-1} \cdot \widehat{\mathbf{T}}_8^{-1} \cdot \mathbf{u}_8$, $\mathbf{v}_8 = \widetilde{\mathbf{T}}_8^{-1} \cdot \mathbf{w}_8$, and $\mathbf{x}_8 = \mathbf{R}_8{}^{-1} \cdot \mathbf{v}_8$, it performs $\mathbf{u}_8 = (\sqrt{2}/4) \cdot \mathbf{c}_8$ by using shift operation with the Booth recoded algorithm in the first stage. The preshuffle performs the pre-permutation matrix $\widehat{\mathbf{T}}_8^{-1}$ operation in the second stage. The matrix-vector multiplication is used to compute $\mathbf{w}_8$ in the third and fourth stages.

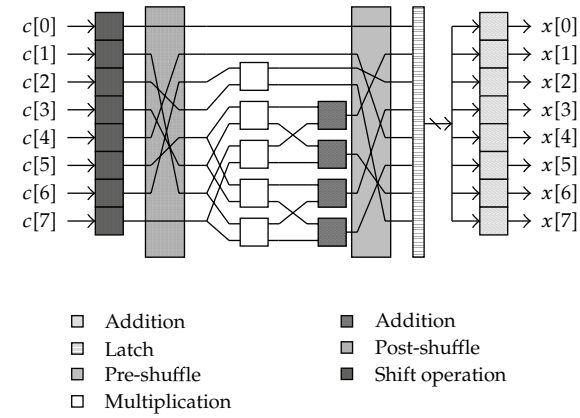


Figure 2: Data flow of the subband-based 8-point IDCT with seven pipelined stages.

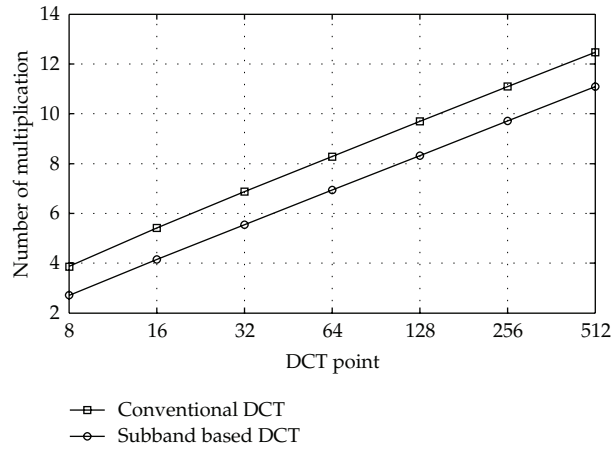


Figure 3: Log plot of the number of multiplications versus the number of DCT points.

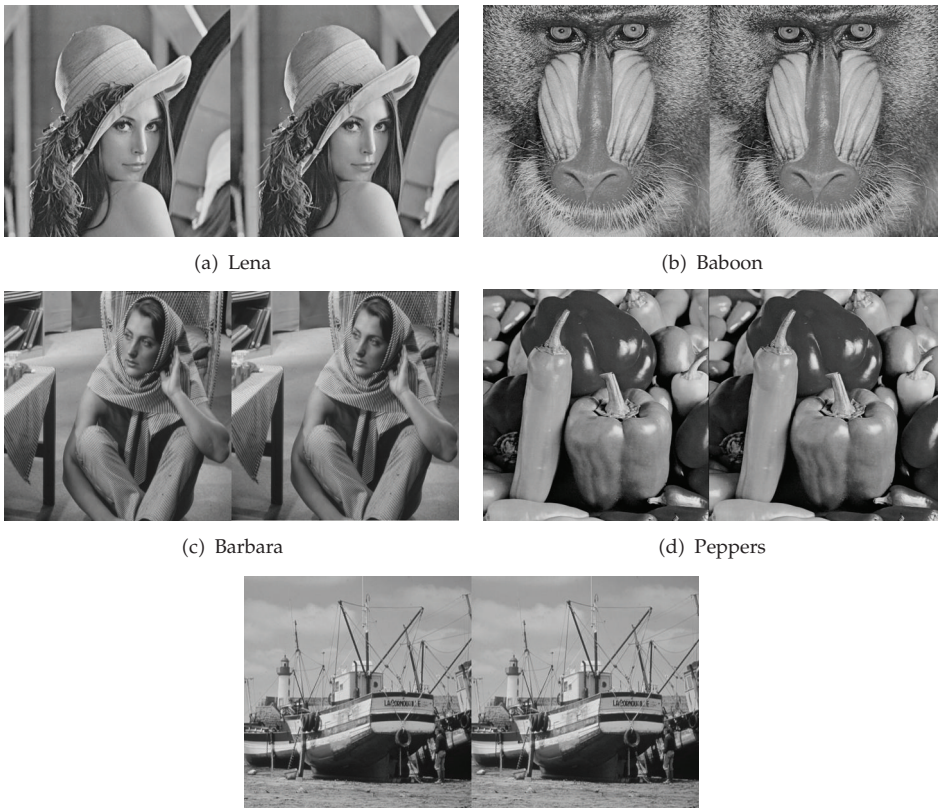
In the fifth stage, the post-shuffle performs the post-permutation matrix $\widetilde{\mathbf{T}}_8^{-1}$. The sixth and seventh stages are to perform $\mathbf{R}_8^{-1} \cdot \mathbf{v}_8$ with simple addition. Both of the subband-based DCT and IDCT algorithms need one multiplication operation with parallel-pipelined processing, in comparison to [23] using linear array, which needs five multiplication operations.

Recall that the DCT of a signal, $\mathbf{x}_n$, can be represented as $\mathbf{c}_n = (\sqrt{n}/n) \cdot \widehat{\mathbf{T}}_n \cdot \widehat{\mathbf{F}}_n \cdot \widetilde{\mathbf{T}}_n \cdot \mathbf{R}_n \cdot \mathbf{x}_n$. The multiplication time of the unified subband-based algorithm can be derived as

$$T_M = 3 \cdot \left(4^0 + 4^1 + 4^2 + \cdots + 4^{(\log_2 n)-2}\right) = 4^{(\log_2 n)-1} - 1, \tag{4.1}$$

where $n = \{2^m \mid m = 3, 4, 5, \ldots\}$.

The log plot of the subband-based DCT computations versus the number of DCT points is shown in Figure 3.



(a) Lena (b) Baboon

(c) Barbara (d) Peppers

(e) Boat

Figure 4: The original image (left side) and the reconstructed image (right side).

The multiplication time of the unified subband-based DCT with single processor can be derived as follows [23]:

$$T_M = 4^0 + 4^1 + 4^2 + \cdots + 4^{(\log_2 n)-2} = \frac{\left(4^{(\log_2 n)-1} - 1\right)}{3}. \tag{4.2}$$

The left side of Figures 4(a), 4(b), 4(c), 4(d), and 4(e) show the original 512×512 Lena, Baboon, Barbara, Peppers, and Boat images, respectively. The reconstructed Lena, Baboon, Barbara, Peppers and boat image shown in the right side of Figures 4(a), 4(b), 4(c), 4(d), and 4(e), respectively, are obtained by using the proposed subband based 8-point DCT/IDCT algorithm with 32-bit fixed-point operands; the peak signal-to-noise ratios (PSNRs) of Lena, Baboon, Barbara, Peppers, and Boat images are 149.67 dB, 142.12 dB, 143.08 dB, 143.36 dB, and 140.79 dB, respectively.

The PSNR curves of Lena, Baboon, Barbara, Peppers, and Boat images obtained by using the conventional 8-point DCT and the proposed subband-based 8-point DCT at various word lengths are shown in Figure 5. Figures 6(a), 6(b), 6(c), 6(d), and 6(e) show the PSNR curves of Lena, Baboon, Barbara, Peppers, and Boat images obtained by using

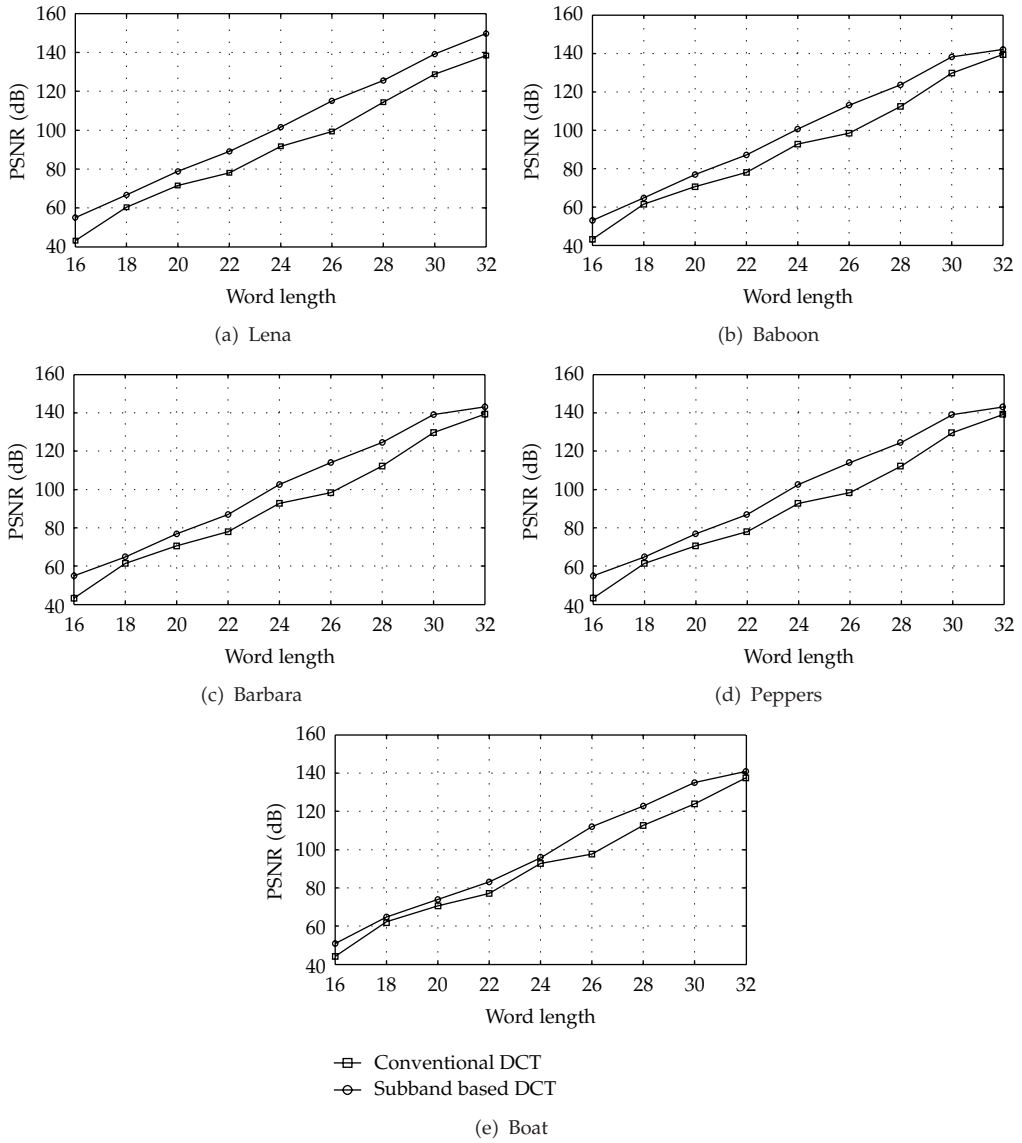


Figure 5: The PSNR curves of (a) Lena, (b) Baboon, (c) Barbara, (d) Peppers, and (e) Boat images obtained by using the conventional 8-point DCT and the proposed subband-based 8-point DCT at various word lengths.

the conventional DCT and the proposed subband-based DCT with 32-bit operand at various DCT points. As one can see, the subband-based DCT is preferable.

5. FPGA Implementation of the Reconfigurable Linear-Array DCT/IDCT Processor

The reconfigurable architecture of the fast 8-, 16-, 32- and 64-point DCT and IDCT processors based on the subband-based 8-point DCT is proposed in this section.

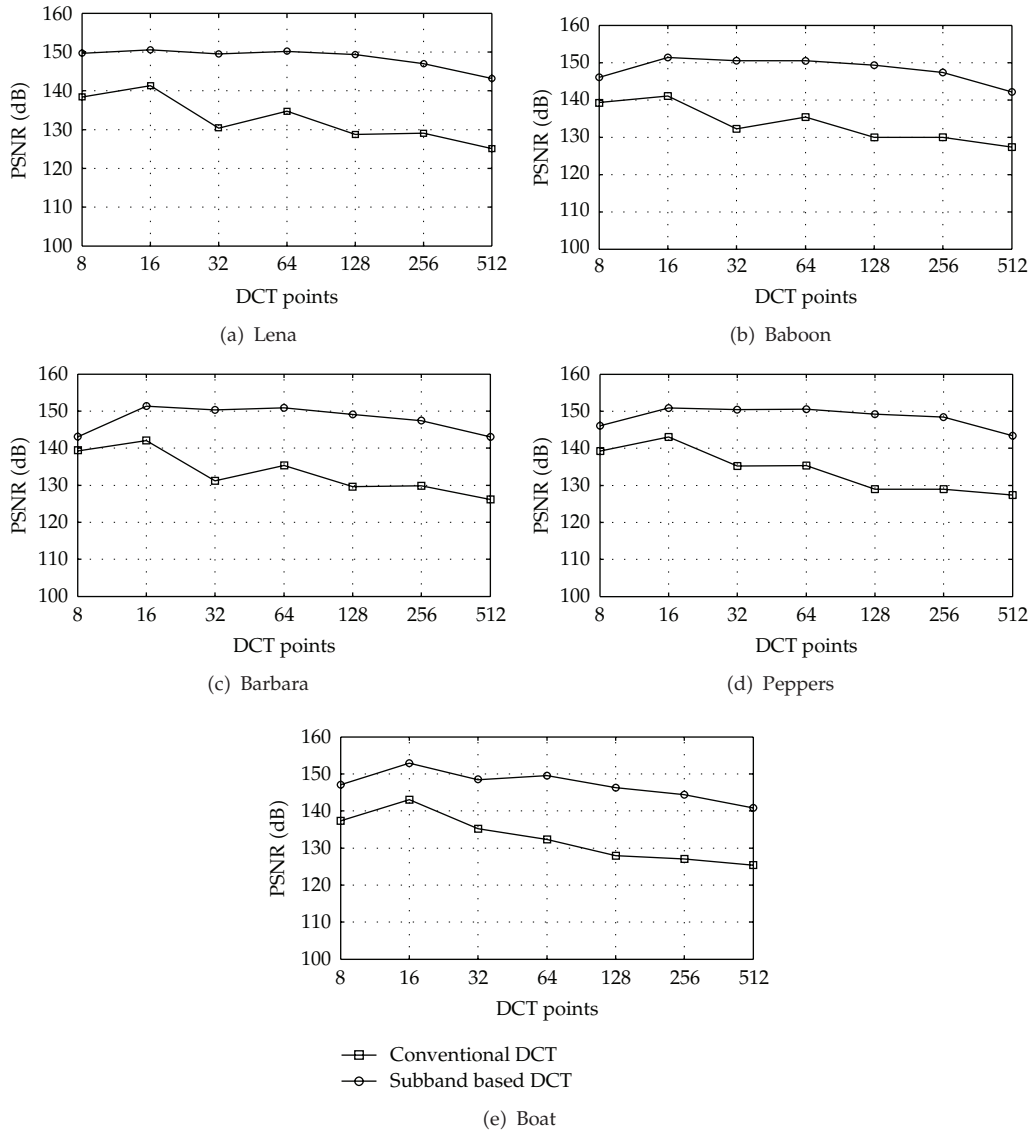


Figure 6: PSNR comparisons of (a) Lena, (b) Baboon, (c) Barbara, (d) Peppers, and (e) Boat images using the conventional DCT and the subband-based DCT with 32-bit operand at various DCT points.

5.1. The Proposed 8-Point DCT/IDCT Processor

According to the data flow of the subband-based 8-point DCT with six pipelined stages (Figure 1), the architecture of the proposed 8-point DCT processor is shown in Figure 7. In which, the adder array (AA) with three CSA(4,2)s performing the matrix-vector multiplication of $\widehat{\mathbf{R}}_8 \cdot \mathbf{x}_8$ is shown in Figure 8. Figure 9 shows the multiplier array (MA) performing three types of operation, which are needed to compute the subcoefficient matrix computation of $\widehat{\mathbf{F}}_8$. The control signals of *swap* and *inv* determine the types of operation. The functions determined by *swap* and *inv* are shown in Table 1. Figure 10 shows the hardwired

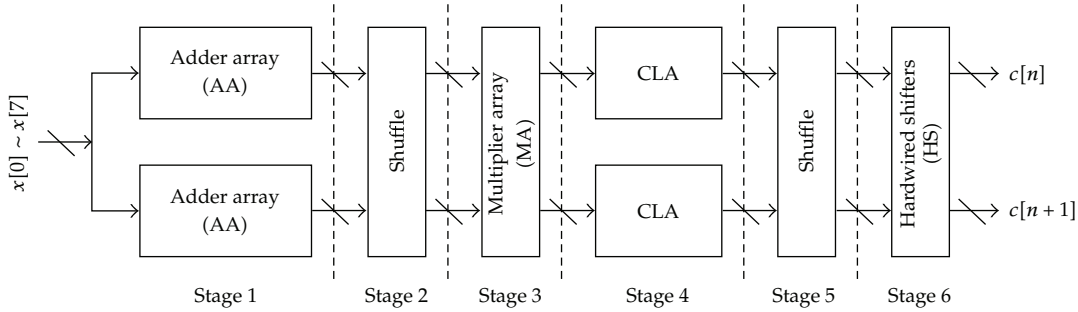


Figure 7: The proposed 8-point DCT Processor with six pipelined stages.

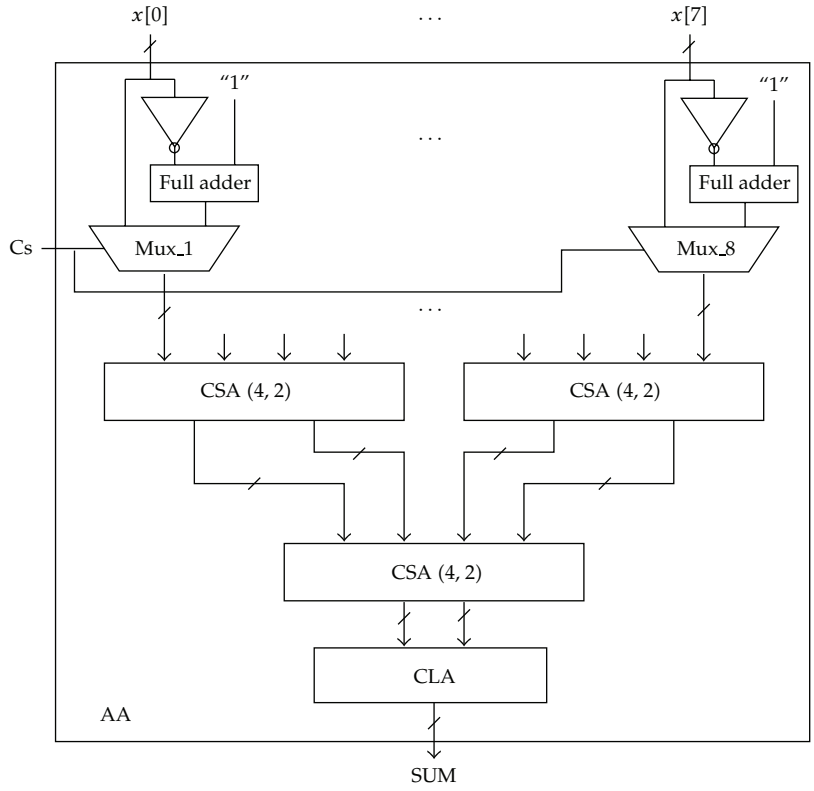


Figure 8: Adder array (AA) for the matrix-vector multiplication of $\mathbf{R}_8 \cdot \mathbf{x}_8$.

shifters used for performing $(\sqrt{2}/4)\cdot\mathbf{z}_8$ by the Booth recoded algorithm [23]. Figure 11 shows the proposed 8-point IDCT processor with seven pipelined stages. In which, the fast adder arrays, shuffle, multiplier array, CLA, and hardwired shifters for DCT architecture can also be used for performing IDCT. The latch array for retiming the input data is shown in Figure 12.

The hardware complexity of the proposed subband-based IDCT architecture is the same as that of the proposed subband-based DCT architecture. Figure 13 shows the proposed integrated 8-point DCT/IDCT processor.

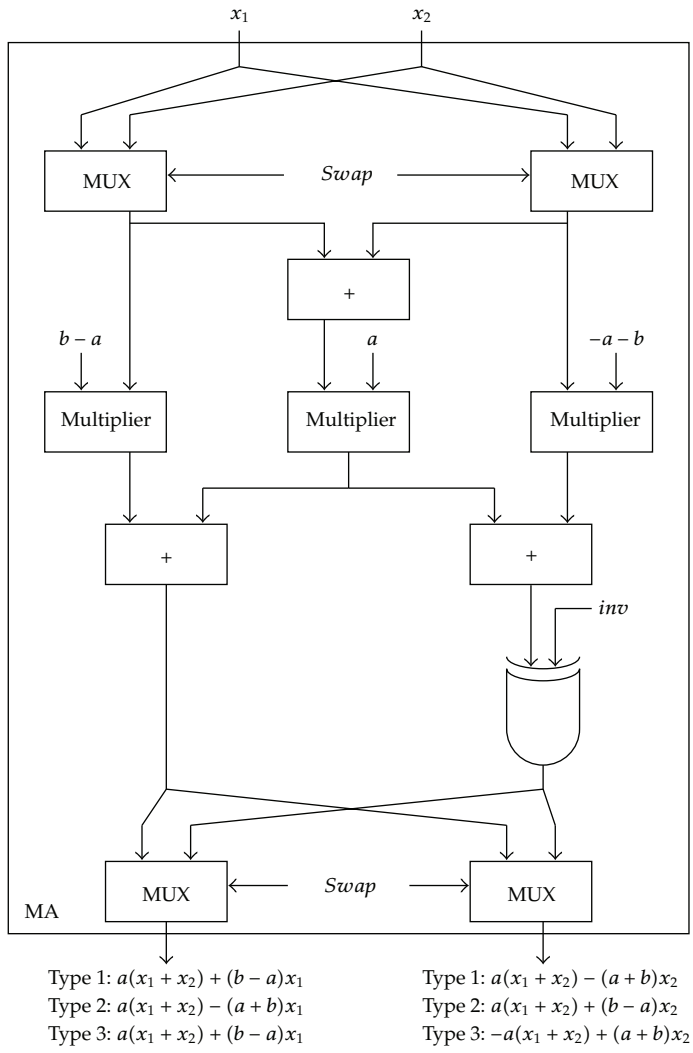


Figure 9: Multiplier array (MA) for computing the subcoefficient matrix of $\widehat{\mathbf{F}}_8$.

Table 1: The functions determined by *swap* and *inv*.

	Type 1	Type 2	Type 3
swap	0	1	0
inv	0	0	1

5.2. The Proposed Reconfigurable DCT/IDCT Processor

According to the integrated 8-point DCT/IDCT processor (Figure 13), the proposed reconfigurable 8-, 16-, 32-, and 64-point DCT/IDCT processor is shown in Figure 14. In which, the integrated adder array (IAA) for the fast computation of 8-, 16-, 32-, and 64-point DCT/IDCT is shown in Figure 15. The modified hardwired shifter (MHS) for multiplication by $\sqrt{n}/n$ (where $n = 8, 16, 32, 64$) using the Booth recoded algorithm is shown in Figure 16.

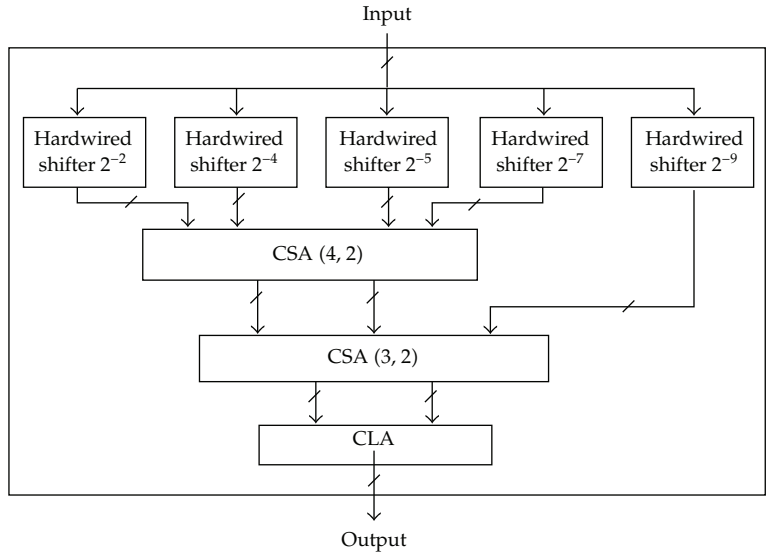


Figure 10: The proposed hardwired shifters used for performing $(\sqrt{2}/4) \cdot \mathbf{z}_8$.

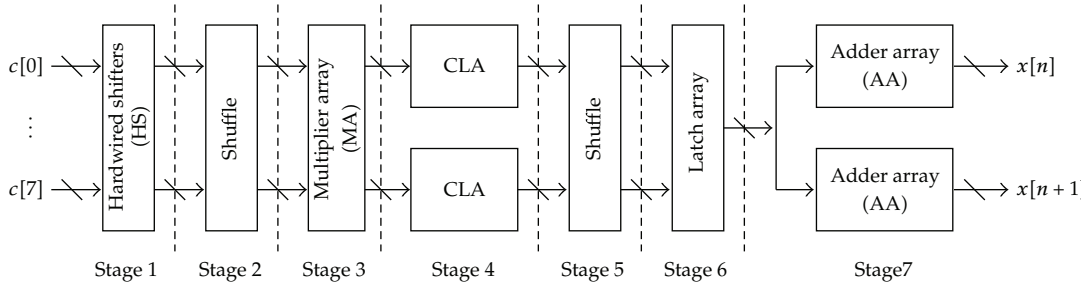


Figure 11: The proposed 8-point IDCT processor with seven pipelined stages.

In order to improve the computation efficiency, the number of multiplier arrays should be increased. The log plot of computation cycles versus number of multiplier arrays is shown in Figure 17.

5.3. FPGA Implementation of the Reconfigurable 2D DCT/IDCT Processor

The $N \times N$ DCT is defined as [29]

$$Z[u,v] = \frac{2 \cdot \alpha[u] \cdot \alpha[v]}{N} \cdot \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} x[m,n] \cdot \cos\left(\frac{2m+1}{2N} u\pi\right) \cdot \cos\left(\frac{2n+1}{2N} v\pi\right), \quad 0 \le u,\ v \le N-1, \tag{5.1}$$

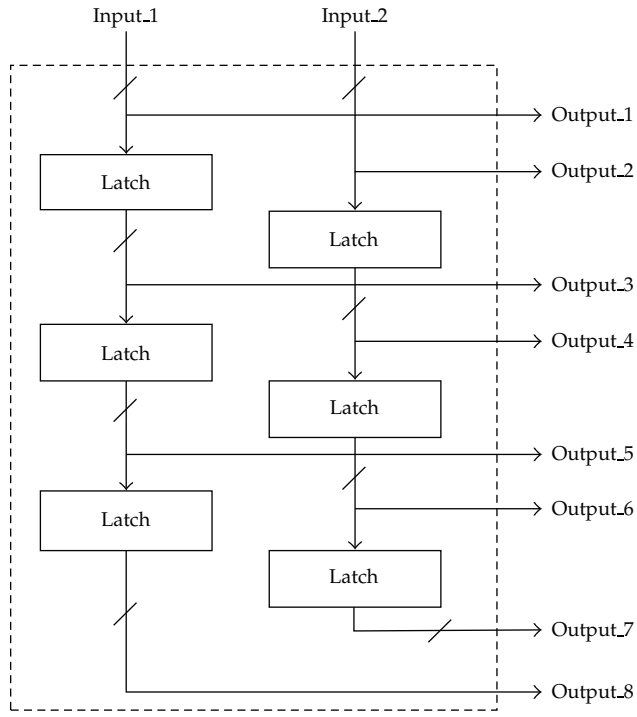


Figure 12: The latch array for retiming the input data.

where $\alpha[k] = 1/\sqrt{2}$ for $k = 0$, and $\alpha[k] = 1$ for $k > 0$. It can be rewritten as

$$Z[u,v] = \sqrt{\frac{2}{N}}\alpha[u] \sum_{m=0}^{N-1} \left[\sqrt{\frac{2}{N}}\alpha[v] \sum_{n=0}^{N-1} x(m,n) \cdot \cos\left(\frac{2n+1}{2N} v\pi\right) \right] \cdot \cos\left(\frac{2m+1}{2N} u\pi\right), \quad 0 \le u,\ v \le N-1. \tag{5.2}$$

Thus, the separable 2-D DCT can be obtained by using 1-D DCT as follows:

$$\text{2-D DCT}(\mathbf{X}) \ = \ \text{1-D DCT}\Big((\text{1-D DCT}(\mathbf{X}))^{\text{T}}\Big). \tag{5.3}$$

Similarly, the separable 2-D IDCT can be obtained by using 1-D IDCT as follows:

$$\text{2-D IDCT}(\mathbf{Z}) = \text{1-D IDCT}\Big((\text{1-D IDCT}(\mathbf{Z}))^{\text{T}}\Big). \tag{5.4}$$

As a result, the architecture of 2D DCT/IDCT can be implemented by using two successive 1D DCT/IDCT processors with only one transpose memory [29]. The proposed architecture of 2-D DCT and IDCT is shown in Figure 18. In which, the control signals provided by the finite state machine (FSM) controller are used to manage the data flow and the

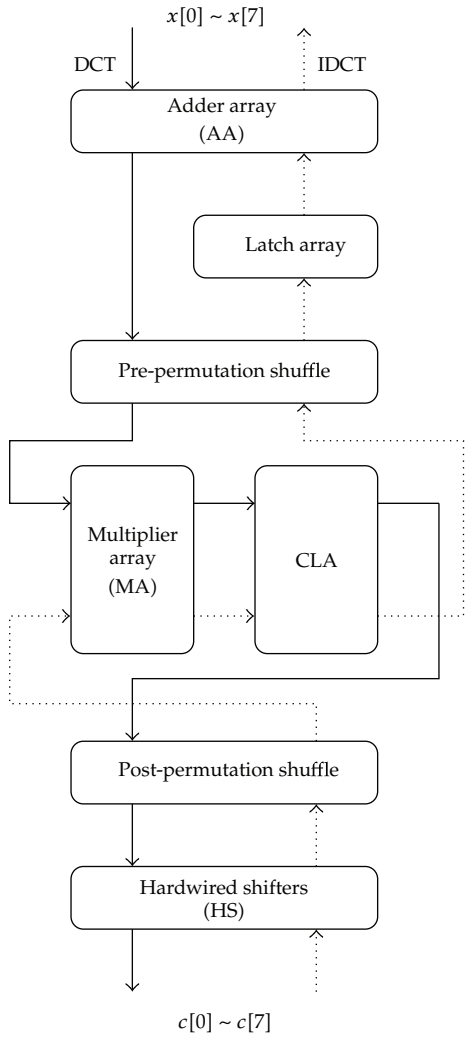


Figure 13: The integrated 8-point DCT/IDCT processor.

operation timing for the DCT/IDCT and transpose memory; the transpose memory allows simultaneous read and write operations between the two processors while performing matrix transposition. The data read and written timing diagram for 8 × 8 DCT/IDCT system is shown in Figure 19. In comparison with the conventional two transpose memories based 2-D DCT/IDCT architectures, the proposed architecture utilizes only one transpose memory.

The platform for architecture development and verification has been designed as well as implemented in order to evaluate the development cost. The architecture has been implemented on the Xilinx FPGA emulation board [30]. The Xilinx Spartan-3 FPGA has been integrated with the microcontroller (MCU) and I/O interface circuit (USB 2.0) to form the architecture development and verification platform. Figure 20 depicts block diagram and circuit board of the architecture development and evaluation platform. In which, the

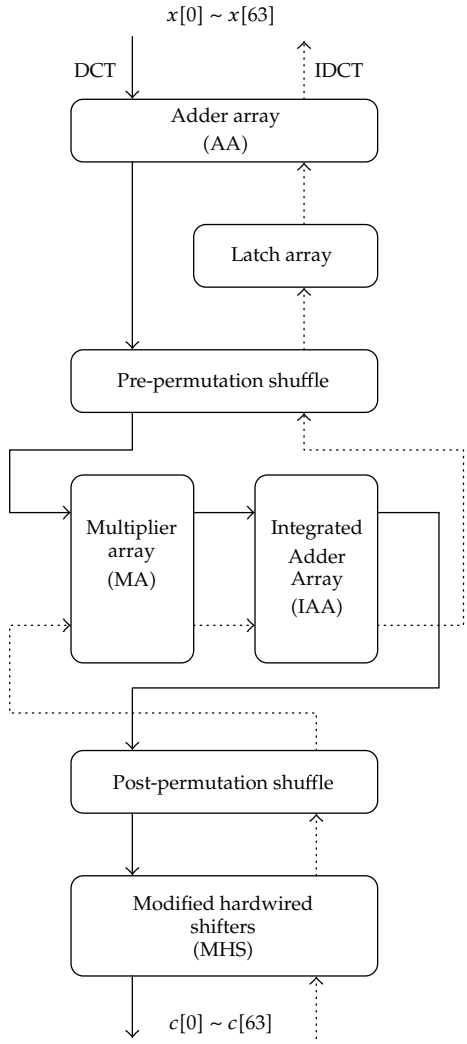


Figure 14: The reconfigurable 8-, 16-, 32-, and 64-point DCT/IDCT processor.

microcontroller reads data and commands from PC and writes the results back to PC by USB 2.0; the Xilinx Spartan-3 FPGA implements the proposed 2-D DCT/IDCT processor. The hardware code written in Verilog is for PC with the ModelSim simulation tool [31] and Xilinx ISE smart compiler [32]. It is noted that the throughput can be improved by using the proposed architecture while the computation accuracy is the same as that obtained by using the conventional one with the same word length. Thus, the proposed programmable DCT/IDCT architecture is able to improve the power consumption and computation speed significantly. The proposed processor for 8-, 16-, 32-, and 64-point DCT/IDCT is an extension of the 8-point DCT/IDCT processor. Moreover, the reusable intellectual property (IP) DCT/IDCT core has also been implemented in Verilog for the hardware realization. All the control signals are internally generated on chip. The proposed DCT/IDCT processor provides both high throughput and low gate count.

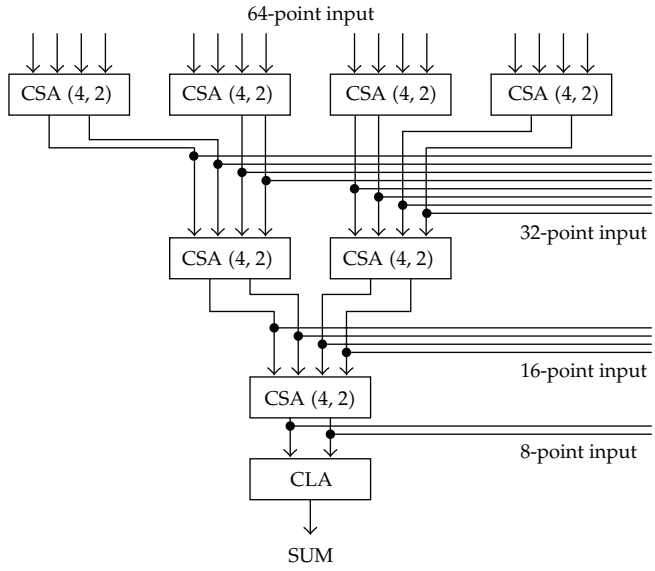


Figure 15: Integrated adder array (IAA) for 8-, 16-, 32-, and 64-point DCT/IDCT processor.

Table 2: Comparisons between the proposed algorithm and architecture and other commonly used algorithms and architectures.

DCT/IDCT	Jeong et al. [24]	Gong et al. [25]	Dimitrov et al. [26]	Alam et al. [27]	Hsiao and Tseng [28]	This Work 2011
Computation complexity	$O(N - \log_2 N + 1)$	$O(N - \log_2 N + 1)$	$O(N - \log_2 N + 1)$	$O(N - \log_2 N + 1)$	$O(\log N)$	$O(1)$
Hardware complexity	$O(2N)$	$O(2N)$	$O(2N)$	$O(2N)$	$O(N \log N)$	$O(\frac{4^{\log_2 N-1} - 1}{3})$
Pipelinability	no	no	no	no	no	good
Scalability	poor	poor	poor	poor	good	better

6. Conclusion

With the advantages of the subband decomposition of a signal, a high-efficiency algorithm with pipelined stages has been proposed for fast DCT/IDCT computations. It is noted that the proposed DCT/IDCT algorithm not only simplifies computation complexity but also improves system performance. The PSNR and system complexity of the proposed algorithm is better than those of the previous algorithms [33–36]. Table 2 shows comparisons between the proposed algorithm and architecture and other commonly used algorithms and architectures [24–28]. Thus, the proposed subband-based DCT/IDCT algorithm is suitable for the real-time signal processing applications. The proposed DCT/IDCT processor provides both high throughput and low gate count and has been applied to various images with great satisfactions.

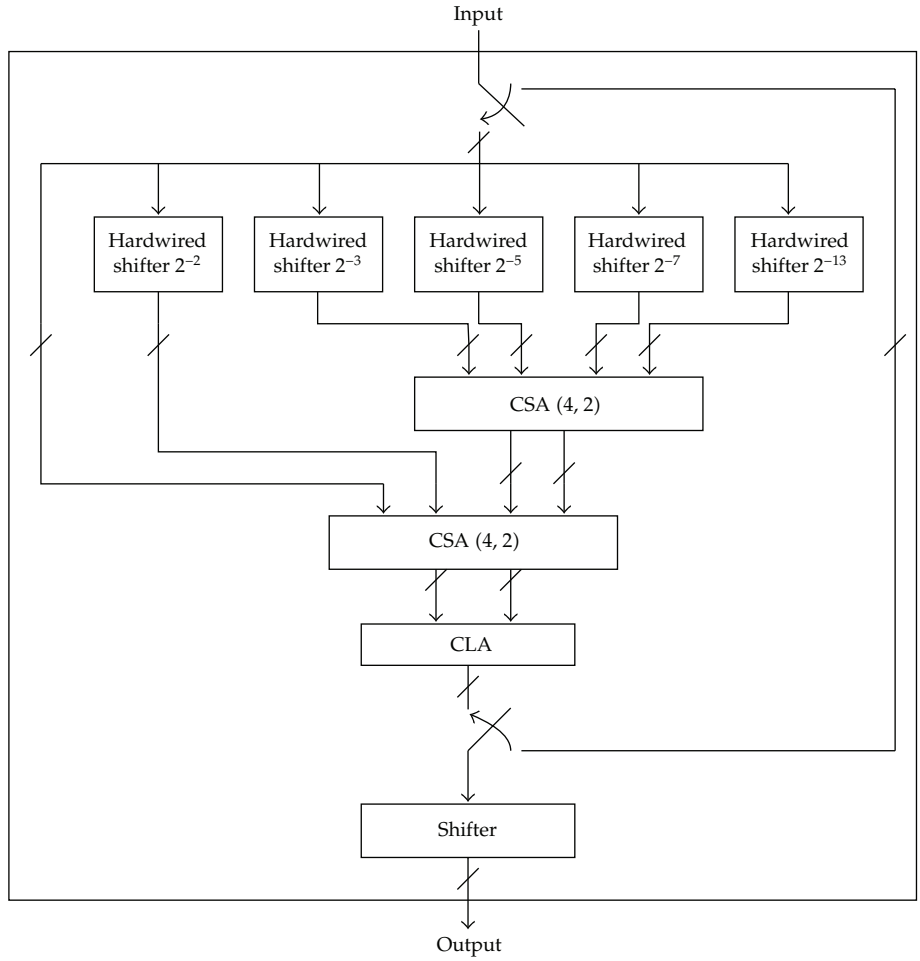


Figure 16: The modified hardwired shifter (MHS) for multiplication by $\sqrt{n}/n$ (where $n = 8, 16, 32, 64$) using the Booth recoded algorithm.

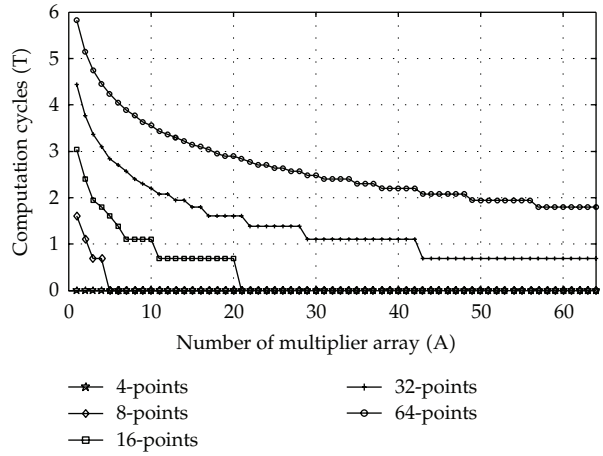


Figure 17: Log plot of number of multiplier array versus computation cycles.

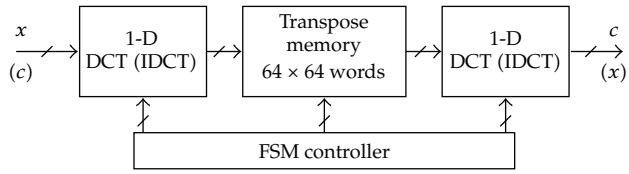


Figure 18: The proposed architecture of 2-D DCT and IDCT (FSM: finite state machine).

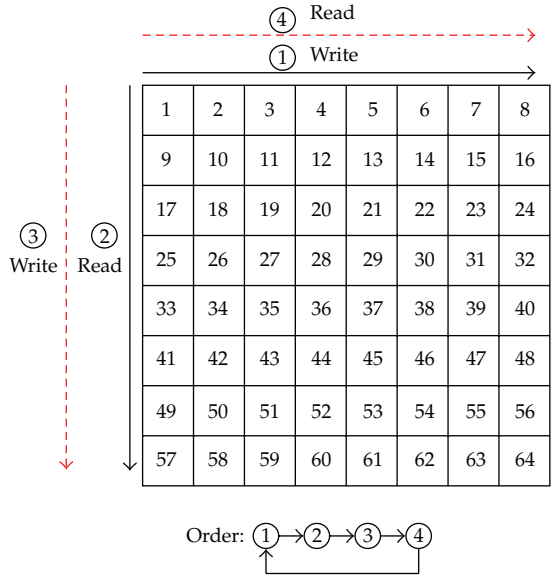


Figure 19: Data read and written timing diagram for 8 × 8 DCT/IDCT system.

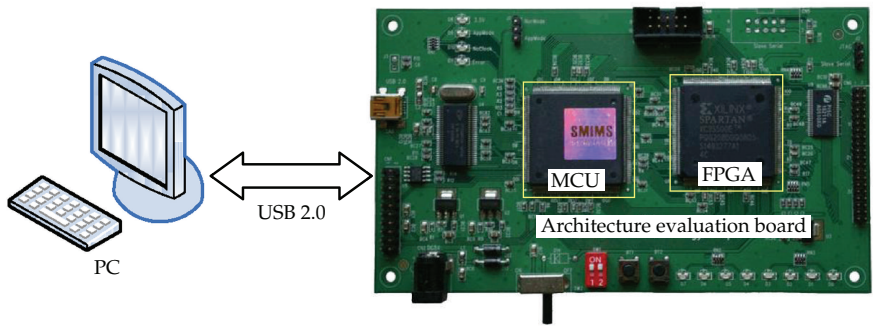


Figure 20: Block diagram and circuit board of the architecture development and verification platform.

Acknowledgment

The National Science Council of Taiwan, under Grants NSC98-2221-E-216-037 and NSC99-2221-E-239-034 supported this work.

References

[1] N. Ahmed, T. Natarajan, and K. R. Rao, "Discrete cosine transform," *IEEE Transactions on Computers*, vol. 23, pp. 90–93, 1974.

[2] A. K. Jain, "A sinusoidal family of unitary transforms," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 1, no. 4, pp. 356–365, 1979.

[3] S. Y. Chen and Y. F. Li, "Determination of stripe edge blurring for depth sensing," *IEEE Sensors Journal*, vol. 11, no. 2, Article ID 5585653, pp. 389–390, 2011.

[4] S. Y. Chen and Q. Guan, "Parametric shape representation by a deformable NURBS model for cardiac functional measurements," *IEEE Transactions on Biomedical Engineering*, vol. 58, no. 3 part 1, pp. 480–487, 2011.

[5] S. Y. Chen, Y. F. Li, and J. Zhang, "Vision processing for realtime 3-D data acquisition based on coded structured light," *IEEE Transactions on Image Processing*, vol. 17, no. 2, pp. 167–176, 2008.

[6] M. Li and W. Zhao, "Representation of a stochastic traffic bound," *IEEE Transactions on Parallel and Distributed Systems*, vol. 21, no. 9, pp. 1368–1372, 2009.

[7] M. Li, "Fractal time series—a tutorial review," *Mathematical Problems in Engineering*, vol. 2010, Article ID 157264, 26 pages, 2010.

[8] M. Li and S. C. Lim, "Modeling network traffic using generalized Cauchy process," *Physica A*, vol. 387, no. 11, pp. 2584–2594, 2008.

[9] C. Cattani, "Harmonic wavelet approximation of random, fractal and high frequency signals," *Telecommunication Systems*, vol. 43, no. 3-4, pp. 207–217, 2010.

[10] C. Cattani, "Fractals and hidden symmetries in DNA," *Mathematical Problems in Engineering*, vol. 2010, Article ID 507056, 31 pages, 2010.

[11] M. Li, "Generation of teletraffic of generalized Cauchy type," *Physica Scripta*, vol. 81, no. 2, Article ID 025007, p. 10, 2010.

[12] M. Li and J. Y. Li, "On the predictability of long-range dependent series," *Mathematical Problems in Engineering*, vol. 2010, Article ID 397454, 9 pages, 2010.

[13] E. G. Bakhoum and C. Toma, "Specific mathematical aspects of dynamics generated by coherence functions," *Mathematical Problems in Engineering*, vol. 2011, Article ID 436198, 10 pages, 2011.

[14] E. G. Bakhoum and C. Toma, "Dynamical aspects of macroscopic and quantum transitions due to coherence function and time series events," *Mathematical Problems in Engineering*, vol. 2010, Article ID 428903, 13 pages, 2010.

[15] K. R. Rao and P. Yip, *Discrete Cosine Transform: Algorithms, Advantages, Applications*, Academic Press, New York, NY, USA, 1990.

[16] V. Bhaskaran and K. Konstantinides, *Image and Video Compression Standards: Algorithms and Architectures*, kluwer academic publishers, Norwell, Mass, USA, 2nd edition, 1997.

[17] W. H. Chen, C. H. Smith, and S. C. Fralick, "A fast computational algorithm for the discrete cosine transform," *IEEE Transactions on Communications*, vol. 25, no. 9, pp. 1004–1009, 1977.

[18] B. G. Lee, "A new algorithm to compute the discrete cosine transform," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 32, no. 6, pp. 1243–1245, 1984.

[19] H. S. Hou, "A fast recursive algorithm for computing the discrete cosine transform," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 35, no. 10, pp. 1455–1461, 1987.

[20] C. Loeffer, A. Ligtenberg, and G. S. Moschytz, "Practical fast 1-D DCT algorithms with 11 multiplications," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '89)*, vol. 2, pp. 988–991, February 1989.

[21] E. Feig and S. Winograd, "Fast algorithms for the discrete cosine transform," *IEEE Transactions on Signal Processing*, vol. 40, no. 9, pp. 2174–2193, 1992.

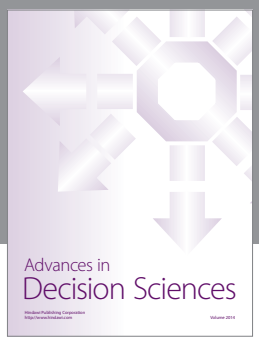
[22] T. Y. Sung, Y. S. Shieh, and H. C. Hsin, "An efficient VLSI linear array for DCT/IDCT using subband decomposition algorithm," *Mathematical Problems in Engineering*, vol. 2010, Article ID 185398, 21 pages, 2010.

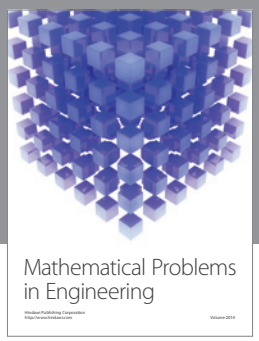
[23] H. Huang, T. Y. Sung, and Y. S. Shieh, "A novel VLSI linear array for 2-D DCT/IDCT," in *Proceedings of the 3rd International Congress on Image and Signal Processing (CISP '10)*, vol. 8, pp. 3686–3690, Yantai, China, October 2010.

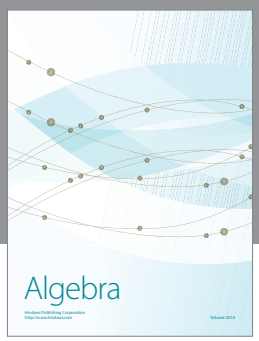
[24] H. Jeong, J. Kim, and W. K. Cho, "Low-power multiplierless DCT architecture using image data correlation," *IEEE Transactions on Consumer Electronics*, vol. 50, no. 1, pp. 262–267, 2004.

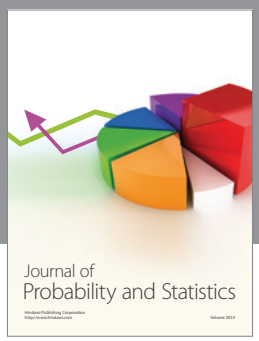
[25] D. Gong, Y. He, and Z. Cao, "New cost-effective VLSI implementation of a 2-D discrete cosine transform and its inverse," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 14, no. 4, pp. 405–415, 2004.

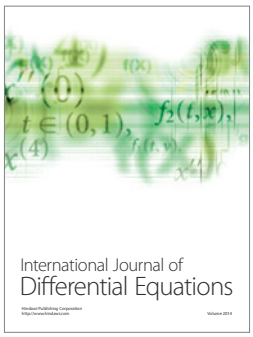
[26] V. Dimitrov, K. Wahid, and G. Jullien, "Multiplication-free 8 × 8 2D DCT architecture using algebraic integer encoding," *Electronics Letters*, vol. 40, no. 20, pp. 1310–1311, 2004.
[27] M. Alam, W. Badawy, and G. Jullien, "A new time distributed DCT architecture for MPEG-4 hardware reference model," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 15, no. 5, pp. 726–730, 2005.
[28] S. F. Hsiao and J. M. Tseng, "New matrix formulation for two-dimensional DCT/IDCT computation and its distributed-memory VLSI implementation," *IEE Proceedings: Vision, Image and Signal Processing*, vol. 149, no. 2, pp. 97–107, 2002.
[29] T. Y. Sung, "Memory-efficient and high-performance 2-D DCT and IDCT processors based on CORDIC rotation," *WSEAS Transactions on Electronics*, vol. 3, no. 12, pp. 565–574, 2006.
[30] SMIMS® Technology Corp., http://www.smims.com.
[31] ModelSim, http://model.com/content/modelsim-de-simulation-and-verification.
[32] Xilinx® FPGA products, http://www.xilinx.com/products/.
[33] A. Buemi, A. Bruna, M. Mancuso, A. Capra, and G. Spampinato, "Chroma noise reduction in DCT domain using soft-thresholding," *EURASIP Journal on Image and Video Processing*, vol. 2010, Article ID 323180, 13 pages, 2010.
[34] M. Li, C. Cattani, and S. Y. Chen, "Viewing sea level by a one-dimensional random function with long memory," *Mathematical Problems in Engineering*, vol. 2011, Article ID 654284, 13 pages, 2011.
[35] T. Bianchi, A. Piva, and M. Barni, "Encrypted domain DCT based on homomorphic cryptosystems," *EURASIP Journal on Information Security*, vol. 2009, Article ID 716357, 12 pages, 2009.
[36] F. S. Al-Kamali, M. I. Dessouky, B. M. Sallam, F. Shawki, and F. E. A. El-Samie, "Carrier frequency offsets problem in DCT-SC-FDMA system: investigation and compensation," *ISRN Communications and Networking*, vol. 2011, Article ID 842093, 7 pages, 2011.

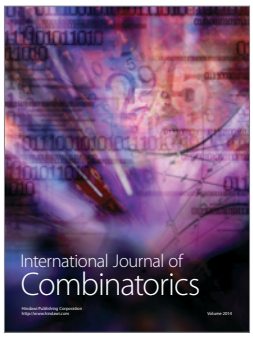


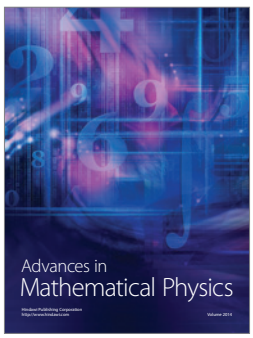


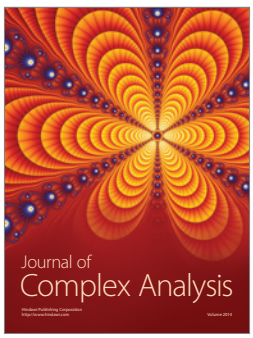


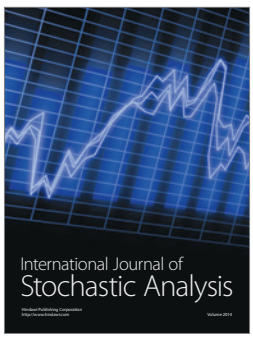


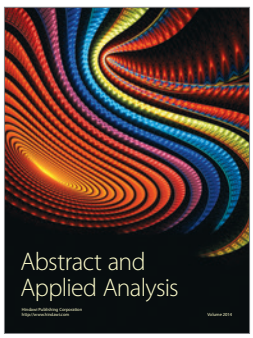


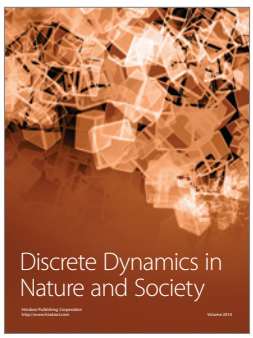


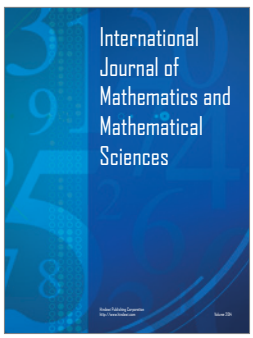


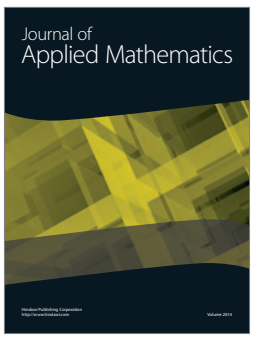












Hindawi
Submit your manuscripts at
http://www.hindawi.com