

論文95-32B-9-8

# 최적화 정수형 여현 변환

## (Optimized Integer Cosine Transform)

李宗河\*, 金惠淑\*\*, 宋寅俊\*\*\*, 郭勳星\*\*\*

(Jong Ha Lee, Hye Sook Kim, In Jun Song, and Hoon Sung Kwak)

## 요 약

본 논문에서는 고성능 2차원 DCT 프로세서 구현을 위해 변환 계수가 정수 값으로 구성되고, 정방향과 역방향 변환 모두 정수형 연산을 수행하는 최적화 정수형 여현 변환(OICT)과 이의 고속 계산 알고리즘을 제시하였다. OICT는 OIT의 구현 방법을 이용하였으며, 정방향과 역방향 변환 후 정규화 계수를 처리할 수 있도록, 정규화 계수로 인한 오차를 무시할 수 있는 범위에서 정규화 계수를 단일 값으로 근사화 시키고 그 중에서 Hilbert-Schmidt norm 값이 작으면서 고속 알고리즘 구현이 가능한 정수 값을 선정하여 구현하였다. 또한 고속 알고리즘은 행렬 분해를 이용하여 정수형 연산을 수행하는 20개의 곱셈 수를 갖도록 하였다. 본 논문에서 제시한 N=8인 OICT의 성능 비교를 위해 변환의 효율성, 최대감소비트, 윈너 필터에 의한 평균자승오차를 구하여 DCT, OIT와 비교한 결과 OICT가 DCT나 OIT보다 우수 하였다. 뿐만 아니라 DCT 계수를 OICT와 같은 7비트로 구성하여 복원 영상의 평균자승오차를 비교한 결과 OICT는 0이었으나 DCT는 직교 변환의 성질이 상실되어 복원 영상의 화질에 치명적인 영향을 미치게 되었다.

## Abstract

We present an optimized integer cosine transform(OICT) as an alternative approach to the conventional discrete cosine transform(DCT), and its fast computational algorithm. In the actual implementation of the OICT, we have used the techniques similar to those of the orthogonal integer transform(OIT). The normalization factors are approximated to single one while keeping the reconstruction error at the best tolerable level. By obtaining a single normalization factor, both forward and inverse transform are performed using only the integers. However, there are so many sets of integers that are selected in the above manner, the best OICT matrix obtained through value minimizing the Hilbert-Schmidt norm and achieving fast computational algorithm. Using matrix decomposing, a fast algorithm for efficient computation of the order-8 OICT is developed, which is minimized to 20 integer multiplications. This enables us to implement a high performance 2-D DCT processor by replacing the floating point operations by the integer number operations. We have also run the simulation to test the performance of the order-8 OICT with the transform efficiency, maximum reducible bits, and mean square error for the Wiener filter. When the results are compared to those of the DCT and OIT, the OICT has out-performed them all. Furthermore, when the conventional DCT coefficients are reduced to 7-bit as those of the OICT, the resulting reconstructed images were critically impaired losing the orthogonal property of the original DCT. However, the 7-bit OICT maintains a zero mean square reconstruction error.

\* 正會員, 全州工業專門大學 電子科  
(Dept. of Electronics Jeonju Technical College)

\*\* 正會員, 崇實大學校 電子計算院  
(Soong Sil Computer Institute)

\*\*\* 正會員, 全北大學校 컴퓨터工學科  
(Dept. of Computer Eng. Chonbuk National University)

接受日字: 1994年9月14日, 수정완료일: 1995年9月1日

## I. 서론

영상 압축 표준에서 소스 코딩 알고리즘에 이용되는 2차원 DCT의 실시간 계산을 위해서는 전용 집적회로 소자의 사용이 필수적이다. 특히 실시간 영상 처리가 요구되는 HDTV와 같은 시스템 개발을 위해서는 고성능 2차원 DCT 프로세서가 사용되어야 하므로 VLSI 기술을 이용한 2차원 DCT 프로세서의 개발에 대한 관심이 집중되고 있다<sup>[1]</sup>.

일반적으로, 알고리즘의 VLSI 구현은 현재의 NMOS/CMOS 기술에서는 칩 면적과 수행 시간이 가장 중요한 요소가 된다<sup>[2]</sup>. 따라서 효율적인 구현을 위해서는 칩 면적과 수행 시간에 대한 면밀한 분석이 필요하게 되며, DCT 프로세서의 경우도 예외는 아니다. 뿐만 아니라 DCT 프로세서 구현시 고려하여야 할 중요한 요소 중의 하나는 연산의 정밀도(computational accuracy)이다. DCT는 연산시에 발생되는 오차가 그 평균값이 0이 되지 않으면 DCT의 직교 변환 성질이 상실되고, 이는 역방향 DCT 과정에 누적되어 복원 영상의 질에 치명적인 영향을 미치게 된다. 특히 하드웨어의 비용을 절감하기 위하여 DCT 계수를 작은 비트로 표현할 경우 이와 같은 성질은 더욱 심화되어 프로세서 구현이 어렵게 된다. 따라서, 유한 레지스터 길이(finite register length)를 갖는 DCT 프로세서의 경우 연산의 정밀도를 높이기 위해서는 오차의 평균값을 최소화 하는 것이 중요하게 된다<sup>[3][4]</sup>.

문헌에 발표된 대부분의 DCT 프로세서<sup>[5][6]</sup>들은 20~30MHz 범위의 pixel rate를 가져 50MHz 이상의 pixel rate를 갖는 HDTV 신호처리 등의 분야에서 요구되는 성능에는 미치지 못하는 것으로 평가되고 있다. 따라서, 고성능 2차원 DCT 프로세서의 구현을 위해서는 DCT 계수가 작은 비트 수로 구성되고, 이로 인한 연산의 오차가 복원 영상의 질에 영향을 미치지 않아야 하며, 이의 고속 알고리즘 구현이 가능해야 한다. 또한 이를 실리콘에 효율적으로 매핑하기 위한 algorithm-specific 아키텍처의 고안 그리고 아키텍처 레벨의 병렬성을 극대화하기 위한 최적화 방안 등이 종합적으로 연구되어야 한다<sup>[1]</sup>.

이러한 연구의 일환으로 Cham 등은 dyadic symmetry 원리를 이용하여 변환 계수가 작은 비트의 정수 값으로 구성되는 ICT(Integer Cosine Transform)<sup>[7][8]</sup>를 제시하였으나 변환의 성능이 DCT 보

다 낮으며 고속 알고리즘 구현시 부분적인 실수형 연산을 수행하는 단점이 있다. 뿐만 아니라 ICT는 정방향 변환 수행시 정수형 연산을 수행하나 단일하지 않은 정규화 계수로 인하여 역방향 변환에서는 실수형 연산을 수행하게 된다.

또한 Lee와 Kwak은 DCT의 구조 특성인 직교성과 크기 순서를 이용하여 변환 계수가 정수 값을 갖는 OIT(Orthogonal Integer Transform)<sup>[9]</sup>를 제시하였다. 이 OIT는 N=8일 때 변환의 성능이 DCT 보다 우수하며 4비트로 구성되는 장점을 가지고 있으나 ICT와 마찬가지로 고속 알고리즘 구현시 부분적인 실수형 연산과 역방향 변환시 실수형 연산을 수행하는 단점이 있다.

본 논문에서는 고성능 2차원 DCT 프로세서 구현을 위해 변환 계수가 작은 비트의 정수 값으로 구성되고, 정방향과 역방향 변환 모두 정수형 연산을 수행하는 최적화 정수형 여현 변환(Optimized Integer Cosine transform : OICT)과 이의 고속 계산 알고리즘을 제시한다.

OICT는 OIT 구현 방법을 이용하여 정방향과 역방향 변환 후 정규화 계수를 처리할 수 있도록 정규화 계수로 인한 오차를 무시할 수 있는 범위에서 단일한 정규화 계수로 근사화 시키고, 그 중에서 Hilbert-Schmidt norm 값이 작으면서 고속 알고리즘 구현이 가능한 정수 값을 선정하여 구현하였다.

본 논문에서 제시한 OICT의 성능 비교를 위해 일반적으로 많이 사용하는 변환의 효율성, 최대감소비트, 윈너 필터에 의한 평균자승오차를 이용하여 DCT, OIT와 비교 평가 하였다. 뿐만 아니라 한 화소(picture element) 당 8 비트의 해상도를 갖고 256×256개의 화소로 구성된 실제 영상인 "GIRL", "LENA", "COUPLE", "CRONKITE" 등에 DCT 계수를 OICT와 같은 비트로 구성하여 복원 영상의 평균자승오차도 비교하였다.

## II. 최적화 정수형 여현 변환(OICT)의 구현

OIT는 N=8일 때 변환의 성능이 DCT 보다 우수하며 4비트로 구성되는 장점을 가지고 있으나 ICT와 마찬가지로 역방향 변환시 실수형 연산을 수행하며, 고속 알고리즘<sup>[10]</sup> 구현시에도 부분적인 실수형 연산을 수행하는 문제점이 있다.

본 논문에서 제시한 OICT는 OIT의 구현 방법<sup>[9]</sup>을 이용하고, OIT의 문제점을 보완한 정수형 변환이다. 따라서 먼저 OIT에 대한 문제점을 기술하고 OICT의 구현 방법을 기술한다.

1. Orthogonal Integer Transform(OIT)의 문제점

$N \times 1$  데이터 벡터  $[f]$ 의 1차원 이산 변환에 대한 정방향 변환과 이의 역방향 변환은 각각 식 (1), (2)와 같다.

$$[F] = [K_N] [A_N] [f] \tag{1}$$

$$[f] = [A_N]^T [K_N] [F] \tag{2}$$

여기서  $[K_N]$ 는  $[K_N] = \text{diag} \{k_{11}, k_{22}, \dots, k_{NN}\}$ 인 대각 행렬로서 직교 변환 행렬  $[A_N]$ 을 정규화 하는 크기 인자(정규화 계수)이며  $[A_N]^T$ 는 전치 행렬이다.

일반적으로 직교 변환 행렬은 식(3)과 같이 정의된다.

$$[A_N] [A_N]^T = \frac{1}{[K_N^2]} \tag{3}$$

변환 행렬은 대각 행렬( $[K_N]$ ) 요소가 같은 값( $k_{11} = k_{22} = \dots = k_{NN} = k$ )을 갖는 경우와 같지 않은 값( $k_{11} \neq k_{22} \neq \dots \neq k_{NN}$ )을 갖는 경우로 나눌 수 있다.

대각 행렬 요소 값이 같은 경우 식(3)의 변환 행렬은 식(4)와 같이 나타낼 수 있다.

$$[A_N] [A_N]^T = \frac{1}{k^2} [I_N] \tag{4}$$

여기서  $[I_N]$ 은 단일 행렬이다.

따라서 식(4)를 이용하여 식(1)의 정방향 변환과 식(2)의 역방향 변환은 각각 식(5), (6)으로 나타낼 수 있다.

$$[F] = k [I_N] [A_N] [f] \tag{5}$$

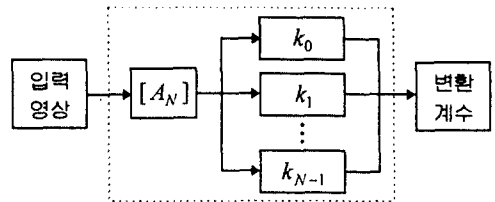
$$[f] = k [I_N] [A_N]^T [F] \tag{6}$$

대부분의 이산 변환은 식(5), (6)의 형태를 갖으므로 변환 과정에서의 정규화 계수는 변환 후 처리할 수 있게 된다.

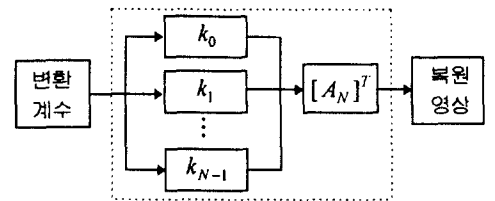
대각 행렬 요소 값이 같지 않은 이산 변환의 경우는

먼저 식(1)에 나타난 바와 같이 정방향 변환의 경우에는 정규화 계수를 변환 후 처리할 수 있으나 식(2)의 역방향 변환에서는 정규화 계수를 먼저 처리하고 변환을 수행해야 한다.

변환 행렬  $[A_N]$ 이 정수 값들로 구성되는 OIT는 정규화 계수  $[K_N]$ 이  $\text{diag} \{1/\sqrt{8}, 1/\sqrt{442}, 1/\sqrt{232}, 1/\sqrt{442}, 1/\sqrt{8}, 1/\sqrt{442}, 1/\sqrt{232}, 1/\sqrt{442}\}$  이어서 역방향 변환의 경우 정규화 계수  $[K_N]$ 의 값에 따라서 열별로 정규화 해야 하므로 실수 값을 갖는 정규화 계수로 인하여 변환의 계수가 정수일 지라도 실수 연산이 수행된다. 1차원 OIT의 정방향 변환과 역방향 변환에 대한 이 과정을 각각 그림 1(a), 그림 1(b)에 나타내었다.



(a) 1차원 정방향 OIT



(b) 1차원 역방향 OIT

그림 1. 1차원 OIT의 블럭 선도

Fig. 1. Block diagram of 1-D OIT.

그러나 식(5), (6)과 같이 정규화 계수가  $k [I_N]$ 인 직교 정수형 변환은 정방향과 역방향 모두 변환 후 정규화 계수를 처리할 수 있으므로 변환 과정에서 정수 연산을 수행하게 된다.

그리고 OIT는 고속 알고리즘<sup>[10]</sup> 구현시에 단계 3의 변환 계수 값이 정수가 아닌 실수 값을 포함하게 되어 변환 수행시 정수 연산이 아닌 실수 연산을 수행하게 된다.

2. OICT의 구현 방법

OICT는 OIT의 구현 방법<sup>[9]</sup>에서 정규화 계수가 단

일 값을 갖도록 식(7)을 추가하여 구현한다. 이는 정방향 변환은 물론 역방향 변환에서도 변환 후에 정규화 계수를 처리하게 됨으로써 정수형 연산만을 수행할 수 있도록 하기 위한 것이다.

$$2^{n-1} [t^2(0,0)]_{A_N} = 2^{n-i-1} \sum_{j=0}^{2^i-1} [d^2(0,j)]_{D_{N/2^i}} \quad (7)$$

$$= \sum_{j=0}^{(N/2)^i-1} [q^2(0,j)]_{Q_{N/2}}$$

$$i = 1, 2, \dots, n-2; n = \log_2 N$$

여기서  $d(k,j), q(k,j)$ 는 각 분해 행렬의 열과 행이  $(k,j)$ 인 행렬 요소 값이며,  $t(0,0)$ 는 행렬  $[A_N]$ 의 열과 행이  $(0,0)$ 인 요소 값이다. 그리고 첨자  $A_N$ 은 OIT 행렬을 의미하며,  $D_{N/2^i}, Q_{N/2}$ 는 각 분해 행렬을 의미한다.

OIT의 구성 식과 식(7)을 만족하는 OICT 행렬은 여러 개 존재하나 좋은 성능을 가지는 OICT가 구성되지 않는다. 그러므로 정규화 계수로 인한 정규화 오차를 무시할 수 있는 범위에서 정규화 계수의 관계식(7)을 식(8)로 근사화 한다.

$$2^{n-1} [t^2(0,0)]_{A_N} \approx 2^{n-i-1} \sum_{j=0}^{2^i-1} [d^2(0,j)]_{D_{N/2^i}} \quad (8)$$

$$\approx \sum_{j=0}^{(N/2)^i-1} [q^2(0,j)]_{Q_{N/2}}$$

$$i = 1, 1, \dots, n-2; n = \log_2 N$$

$N=8$ 인 경우를 예로 OICT를 구하는 과정을 설명하면, 먼저 식(8)은 식(9)와 같이 된다.

$$4t^2 \approx 2(d_1^2 + d_2^2) \approx q_1^2 + q_2^2 + q_3^2 + q_4^2 \quad (9)$$

식(9)에서 각 변의 값이 같지 않기 때문에 정규화 과정에서 오차가 발생하며, 값의 차이가 클수록 오차는 커지게 된다. 따라서 정수 값을 갖는 OICT 행렬  $[A_8]$ 을 구하는데 있어서, 먼저 OIT의 직교 조건과 크기 순서를 만족하는 정수 값  $q_1, q_2, q_3, q_4$ 를 구한 다음,  $\sqrt{q_1^2 + q_2^2 + q_3^2 + q_4^2} / 2$ 를 구하여 반올림(round off)했을 때의 정수 값을  $t$ 로 한다. 그리고 OIT의 크기 순서를 만족하면서 식(9)의 조건에 가장 근사한 정수 값  $d_1, d_2$ 를 취한다.

실제로 이러한 조건을 만족하는 정수 값의 집합은 여러 개가 존재하나 Hilbert-Schmidt norm 값이 작으면서 고속 알고리즘 구현이 가능한 정수 값을 선정하면  $t = 91, d_1 = 119, d_2 = 49, q_1 = 125, q_2 = 108, q_3 = 72, q_4 = 25$ 인 7 비트로 구성된다. 그리고 이 정

수 값을 이용하여 OICT 행렬을 구성하면 식(10)과 같으며, 이 행렬을 scale 하여 정규 직교 행렬로 만드는 대각 행렬은 식(11)과 같다.

$$[A_8] = \begin{bmatrix} 91 & 91 & 91 & 91 & 91 & 91 & 91 & 91 \\ 125 & 108 & 72 & 25 & -25 & -72 & -108 & -125 \\ 119 & 49 & -49 & -119 & -119 & -49 & 49 & 119 \\ 108 & -25 & -125 & -72 & 72 & 125 & 25 & -108 \\ 91 & -91 & -91 & 91 & 91 & -9 & 91 & 91 \\ 72 & -125 & 25 & 108 & -108 & -25 & 125 & -72 \\ 49 & -119 & 119 & -49 & -49 & 119 & -119 & 49 \\ 25 & -72 & 108 & -125 & 125 & -108 & 72 & -25 \end{bmatrix} \quad (10)$$

$$[K_8] = \frac{1}{k} [I_8], k = \sqrt{66222} \quad (11)$$

OICT의 정규화 오차 정도를 알아보기 위하여  $256 \times 256 \times 8$ 비트의 해상도를 갖는 임의의 실제 영상 10개를  $8 \times 8$ 의 블록으로 나눈 총 10240개의 블록에 대하여 정방향 변환을 수행한 다음, 역방향 변환을 수행하여 원 영상과 복원 영상에 대한 평균자승오차(MSE)를 측정한 결과 MSE가 "0"이었다. 따라서 식(10)의 OICT는 정규화 계수의 처리에서 발생하는 정규화 오차를 무시할 수 있다.

### 3. OICT의 고속 알고리즘

OICT의 고속 알고리즘을 구현하기 위하여 OICT 행렬  $[A_N]$ 은 행렬분해를 이용하여  $N=8$ 일 때 식(12)와 같이 나타낼 수 있다.

$$[A_8] = [P_8] \begin{bmatrix} A_4 & 0 \\ 0 & \Omega_4 \end{bmatrix} [B_8] \quad (12)$$

여기서  $[P_8]$ 은 변환 계수 벡터를 비트 역순에서 정 순서로 바꾸어 주는 행렬이고  $[B_8]$ 은 단일 행렬  $[I_4]$ 로 이루어지는 행렬이다. 그리고  $[A_4], [\Omega_4]$ 는 행렬 분해 방식<sup>[12]</sup>을 이용하여 각각 식(13),(14)와 같이 분해 행렬의 곱으로 나타낼 수 있다.

$$[A_4] = \begin{bmatrix} 91 & 91 & 91 & 91 \\ 91 & -91 & -91 & 91 \\ 119 & 49 & -49 & -119 \\ 49 & -119 & 119 & -49 \end{bmatrix} = \begin{bmatrix} 91 & 0 & 0 & 0 \\ 0 & 91 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (13)$$

$$\begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & -1 & 0 & 0 \\ 0 & 0 & 49 & 119 \\ 0 & 0 & -119 & 49 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & -1 & 0 \\ 1 & 0 & 0 & -1 \end{bmatrix}$$

$$[\Omega_4] = \begin{bmatrix} 25 & 72 & 108 & 125 \\ 103 & 25 & -125 & 72 \\ -72 & -125 & -25 & -108 \\ -125 & 108 & -72 & 25 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 5 \\ 0 & 3 & 2 & 0 \\ 0 & -2 & 3 & 0 \\ -5 & 0 & 0 & 1 \end{bmatrix} \quad (14)$$

$$\begin{bmatrix} 25 & 18 & 0 & 0 \\ 36 & -25 & 0 & 0 \\ 0 & 0 & -25 & 36 \\ 0 & 0 & 18 & 25 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

따라서 식(13),(14)의 분해 행렬의 요소 값들이 0을

많이 포함하고 있으므로 이들을 이용하여 OICT의 고속 알고리즘을 구현하면 곱셈 수를 64개에서 20개로 감소시킬 수 있다. 이 방법을 이용한 OICT의 고속 연산을 위한 신호 흐름도를 그림 2에 나타내었다. 그림 2의 신호 흐름도에서 정규화 계수  $[K_8]$ 은 포함되지 않았다. 이는 정규화 계수가 단일 값을 가지므로 정방향과 역방향 변환 후 정규화 계수를 처리할 수 있기 때문이다.

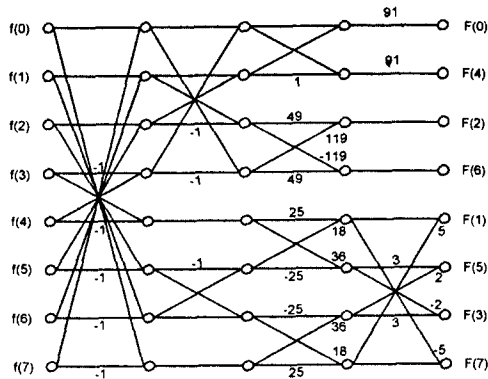


그림 2. OICT에 대한 고속 계산 알고리즘의 신호 흐름도(N=8)  
 Fig. 2. Fast computational algorithm Signal flow graph for Forward OICT(N=8).

### III. 성능 비교

본 논문에서 제시한 OICT의 성능 비교를 위해 입력 신호가 1차 정상(first order stationary) Markov process라 가정하고 일반적으로 많이 사용하는 변환의 효율성, 최대감소비트, 윈너 필터에 의한 평균자승오차를 구하여 다른 변환인 DCT, OIT와 비교한다.

그리고 DCT 계수를 OICT와 같은 7비트로 구성 하고 실제 영상인 "GIRL", "LENA", "COUPLE", "CRONKITE" 등에 적용하여 복원 영상에 대한 평균 자승오차를 구한다.

여기에서 OICT의 성능은 정규화 계수(k)를 10 비트로 양자화하여 계산한 결과이다.

#### 1. 변환의 효율성<sup>[6] [8] [9]</sup>

변환의 효율성(transform efficiency)은 변환 영역에서의 비상관 관계(decorrelation)의 정도를 정의하는 것으로, 최적 변환인 KLT는 모든 상관 계수  $\rho$ 에

대하여 100%의 효율성을 보여 준다.

입력 신호를  $f$ , 변환 행렬을  $\phi$ , 변환 벡터를 F라 할 때 자기공분산행렬  $T'$ 는 식(15)와 같다.

$$T' = E \{ FF^T \} = \phi \{ ff^T \} \phi^T = \phi U \phi^T \quad (15)$$

여기에서 U는  $(i, j)$  번째 요소가 상관계수  $\rho$  ( $0 \leq \rho \leq 1$ )에 대하여  $\rho^{|i-j|}$ 의 값을 갖는 공분산행렬이다.

식(15)를 이용하여 변환의 효율성  $\eta$ 는 식 (16)과 같이 정의 된다.

$$\eta = \frac{\sum_{i=1}^N T'(i, i)}{\sum_{i=1}^N \sum_{j=1}^N T'(i, j)} \times 100(\%) \quad (16)$$

여기에서  $T'(i, j)$ 는 식(15)의 자기공분산행렬이며,  $T'(i, i)$ 는 자기공분산행렬의 대각 행렬이다.

상관계수  $\rho$  ( $0 < \rho < 1$ )에 대하여 OICT와 DCT, OIT의 변환의 효율성을 N=8일 때 표 1에 나타내었다. 표 1에 나타난 바와 같이 N=8일 때 OICT의 효율성은 DCT, OIT 보다 성능이 우수하였다.

표 1. 변환의 효율성(N=8,  $\rho$  ( $0 < \rho < 1$ ))  
 Table 1. Transform efficiency(N=8,  $\rho$  ( $0 < \rho < 1$ )).

변환 $\rho$	OICT	DCT	OIT
0.1	94.57	94.54	94.66
0.2	90.40	90.33	90.58
0.3	87.22	87.12	87.51
0.4	84.88	84.74	85.26
0.5	83.34	83.15	83.82
0.6	82.68	82.44	83.25
0.7	83.15	82.87	83.78
0.8	85.26	84.97	85.83
0.9	90.07	89.83	90.15
0.91	90.77	90.56	90.81
0.92	91.53	91.32	91.53
0.93	92.34	92.15	92.31
0.94	93.20	93.04	93.15
0.95	94.12	94.00	94.03
0.96	95.10	95.01	94.96
0.97	96.18	96.13	96.01
0.98	97.34	97.33	97.19
0.99	98.60	98.61	98.51

#### 2. 최대감소비트<sup>[6] [8] [9]</sup>

최대감소비트(Maximum Reducible Bit : MRB)는 이 값이 클수록 정보 전송에 대하여 요구되는 비트율이 감소 된다는 의미에서 변환의 성능을 측정하는 방법으로 사용되는데 이는 다음 식(17)과 같다.

$$MRB = -\frac{1}{2N} \sum_{i=1}^N \log_2 T'(i, i) \quad (17)$$

여기에서  $T'(i, i)$ 는 식(15)에 나타난 자기공분산행렬의 대각행렬이다.

상관계수  $\rho(0 < \rho < 1)$ 에 대하여 OICT와 DCT, OIT의 최대감소비트를  $N=8$ 일 때 표 2에 나타내었다.

표 2에 나타난 바와 같이  $N=8$ 일 때 OICT는 상관계수  $\rho(0 < \rho < 1)$ 에서도 DCT, OIT 보다 최대감소비트가 크다는 것을 알 수 있다.

표 2. 최대감소비트( $N=8, \rho(0 < \rho < 1)$ )  
Table 2. Maximum reducible bit( $N=8, \rho(0 < \rho < 1)$ )

변환 $\rho$	OICT	DCT	OIT
0.1	0.0058369	0.0058287	0.0058581
0.2	0.0240093	0.0239685	0.0240878
0.3	0.0560870	0.0560060	0.0562733
0.4	0.1047855	0.1046395	0.1051045
0.5	0.1746121	0.1743869	0.1750781
0.6	0.2734258	0.2731188	0.2740217
0.7	0.4162103	0.4158406	0.4168621
0.8	0.6367531	0.6363798	0.6372930
0.9	1.0426930	1.0424410	1.0428060
0.91	1.106273	1.106043	1.106318
0.92	1.177742	1.177536	1.177712
0.93	1.259198	1.259019	1.259089
0.94	1.353717	1.353568	1.353523
0.95	1.466067	1.465951	1.465782
0.96	1.604236	1.604155	1.603853
0.97	1.783193	1.783150	1.782705
0.98	2.036547	2.036547	0.035948
0.99	2.471535	2.471548	2.470817

$$MSE = 1 - \frac{1}{N} \sum_{i=1}^N \frac{\phi_x^2(i, i)}{\phi_x^2(i, i) + \phi_n^2(i, i)} \quad (18)$$

여기서  $\phi_x(i, i)$ 와  $\phi_n(i, i)$ 는 각각 입력 신호와 잡음에 대한 변환 영역에서의 공분산행렬의 분산 값이다. 상관계수  $\rho(0 < \rho < 1)$ 에 대하여 표 3에 OICT와 DCT, OIT의 평균자승오차를 나타내었다.

표 3에 나타난 바와 같이  $N=8$ 일 때 OICT는 DCT, OIT 보다는 평균자승오차가 작음을 알 수 있다.

표 3. 평균자승오차  $N=8, (0 < \rho < 1, SNR=1)$   
Table 3. Mean Square error for Wiener filter ( $N=8, \rho(0 < \rho < 1), SNR=1$ )

변환 $\rho$	OICT	DCT	OIT
0.1	0.4979783	0.4979889	0.4979782
0.2	0.4917733	0.4918017	0.4917585
0.3	0.4810533	0.4811078	0.4810112
0.4	0.4651318	0.4654003	0.4652357
0.5	0.4439788	0.4439077	0.4436729
0.6	0.4152492	0.4154204	0.4151348
0.7	0.3776988	0.3779082	0.3776226
0.8	0.3272534	0.3274923	0.3272875
0.9	0.2543137	0.2545826	0.2545257
0.91	0.2448852	0.2451594	0.2451164
0.92	0.2348632	0.2351415	0.2351115
0.93	0.2241498	0.2244347	0.224416
0.94	0.2126248	0.2129162	0.2129071
0.95	0.2001244	0.2004235	0.2004218
0.96	0.1864304	0.186738	0.1867411
0.97	0.1712325	0.1715491	0.1715542
0.98	0.1540808	0.1544083	0.1544127
0.99	0.1342919	0.1346299	0.1346318

3. 윈너 필터에 의한 평균자승오차<sup>[6][8][9]</sup>

그림 3은 신호처리 평가를 위한 일반화된 1차 윈너 필터(Wiener filter)의 블록 선도이다.

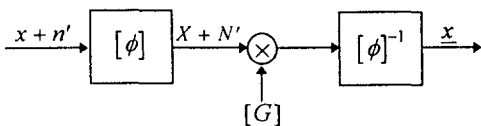


그림 3. 1차원 윈너 필터의 블록 선도  
Fig. 3. Block diagram of 1-D Wiener filter.

입력 신호가 1차 Markov process라 가정하고, 입력 신호와 복원된 신호와의 평균자승오차를 이용하여 변환의 성능을 평가하는데 이 윈너 필터에 의한 평균자승오차는 식(18)과 같다.

4. 복원 영상의 평균자승오차

변환 계수가 7비트로 구성되는 OICT의 성능 측정 방법으로 한 화소당 8 비트의 해상도를 갖고  $256 \times 256$  개의 화소로 구성된 실제 영상인 "GIRL", "LENA", "COUPLE", "CRONKITE" 등의 영상을  $8 \times 8$  블록으로 나누어 정방향 변환을 수행한 다음 역방향 변환을 수행하여 복원 영상에 대한 오차를 비교한다. 이때 DCT의 변환 계수는 OICT의 변환 계수와 같은 크기인 7비트로 구성한다. 따라서 DCT는 변환 계수 값의 절삭에 의한 오차가 발생하게 된다.

복원 영상에 대한 평가 방법은 식(19)의 평균자승오차를 사용하였으며, 그 결과를 표 4에 나타내었다.

$$e(M) = (x - \bar{x})^2 / 256^2 \quad (19)$$

여기서  $x$ 는 원 영상이고  $\bar{x}$ 는 복원 영상이다.

표 4에 나타난 바와 같이 OICT는 복원 영상의 평균 자승오차가 0 값을 가지나 DCT는 평균자승오차가 0 값을 갖지 않는다는 것을 알 수 있다. 이 결과로 볼 때 DCT 계수를 OICT와 같은 7 비트로 구성하여 하드웨어 구현을 할 경우 직교 변환의 성질이 상실되어 복원 영상의 화질에 치명적인 영향을 미치게 된다. 따라서 같은 비트 수로 구현할 경우 OICT는 DCT 보다 성능이 우수하다고 할 수 있다.

표 4. 복원 영상의 평균자승오차(N=8)  
Table 4. Mean square error for reconstructed images(N=8).

영상 \ 변환	OICT	DCT	OIT
GIRL	0	1.466	0
LENA	0	1.085	0
COUPLE	0	2.513	0
CRONKITE	0	1.435	0

#### IV. 결 론

본 논문에서는 고성능 2차원 DCT 프로세서 구현을 위해 변환 계수가 정수 값으로 구성되고, 정방향과 역방향 변환 모두 정수형 연산을 수행하는 최적화 정수형 역변환(OICT)과 이의 고속 계산 알고리즘을 제시하였다.

OICT는 OIT 구현 방법을 이용하여 정방향과 역방향 변환 후 정규화 계수를 처리할 수 있도록 정규화 계수로 인한 오차를 무시할 수 있는 범위에서 단일한 정규화 계수로 근사화 시키고, 그 중에서 Hilbert-Schmidt norm 값이 적으면서 고속 알고리즘 구현이 가능한 정수 값을 선정하여 구현하였다. 또한 고속 알고리즘은 행렬 분해를 이용하여 정수형 연산을 수행하는 20개의 곱셈 수를 갖도록 하였다.

본 논문에서 제시한 N=8일 때 OICT의 성능 비교를 위해 변환의 효율성, 최대 감소비트, 윈너 필터에 의한 평균자승오차를 구하여 DCT, OIT와 비교한 결과 OICT가 DCT나 OIT보다 우수 하였다. 뿐만 아니라 DCT 계수를 OICT와 같은 7비트로 구성하여 복원 영상의 평균자승오차를 비교한 결과 OICT는 0이었으나 DCT는 직교 변환의 성질이 상실되어 복원 영상의 화질에 치명적인 영향을 미치게 되었다. OICT가 DCT 보다 성능이 우수한 것은 Hilbert-Schmidt norm이

작을수록 변환의 성능이 최적 변환인 KLT에 더 근접한 성능을 가지므로 OICT의 계수 값을 구할 때 DCT 보다 Hilbert-Schmidt norm이 작은 정수 값을 구하였기 때문이다.

이상의 결과로 볼 때, 정방향 변환은 물론 역방향 변환에서도 정수형 연산을 수행하는 변환으로써 OICT는 DCT보다 우수한 성능을 가지므로 DCT를 대신하여 사용할 수 있을 것으로 기대된다.

#### 참 고 문 헌

- [1] 신경욱, 전홍우, 강용섭, "Vector-radix 2차원 고속 DCT의 VLSI 구현을 위한 효율적인 어레이 알고리즘," 한국통신학회 논문지, vol. 18, no.12, pp. 1970-1981, Dec. 1993
- [2] C.K. Thompson, "Fourier transforms in VLSI," IEEE Trans. on Computers, vol. C-32, no.11, Nov. 1983.
- [3] 김준현, 고종석, 김성대, 김재균, "고속 DCT 변환 방식의 정수형 연산에 관한 연구," 한국통신학회 논문지, vol. 12, no.4, pp. 309-323, Aug. 1987
- [4] P1180/D2: IEEE Standard Board Draft Standard Specification for the Implementation of 8x8 Inverse Discrete Cosine Transform, Jul.1992.
- [5] A. Artierietal, "A one chip VLSI for real time two-dimensional discrete cosine transform," 1988 IEEE International Symposium on Circuits and Systems (ISCAS'88), pp.701-704, 1988.
- [6] K. R. Rao, P. Yip, "Discrete cosine Transform Algorithms, Advantages Application." Academic press, 1990.
- [7] W. K. Cham, "Development of integer cosine transforms by the principle of dyadic symmetry," IEE Proc. Inst. Elect. Eng. Part I, pp.276-282, Aug. 1989.
- [8] W.K.Cham and Y.T.Chan, "An order-16 integer cosine transform," IEEE Trans. on Signal Processing, vol.39, no.5, May 1991.
- [9] 이종하, 광훈성, "직교 정수형 변환" 대한전자공학회 논문지, vol.31-B, no.1, pp.64-71.

Jan. 1994

- [10] 이종하, 박훈성, "VLSI 구현을 위한 정수화 DCT 개발," 한국통신학회 논문지, vol.18, no.12, pp.1928-1933, Dec. 1993
- [11] M. Hamidi, J.Pearl, "Comparison of the cosine and Fourier transform of Markov -1 signals," IEEE Trans. Acoust., Spee

ch, Signal Process., vol. ASSP-24, pp. 428-429, Oct. 1976 [12] W. H. Chen, C. H. Smith, and S. C. Fralic," A Fast Computational algorithm for the Discrete Cosine Transform," IEEE Trans. on Commun., vol. COM-25, no. 9, pp. 1004-1009, Sep. 1977.

— 저 자 소 개 —

李宗河(正會員) 第 31卷 B 編 第 1 號 參照

현재 전주공업전문대학 전자과 조교수



金惠淑(正會員)

1956년 9월 9일생. 1979년 2월 숭실대학교 전자계산학과 졸업. 1986년 8월 성균관대학교 경영대학원 EDPS졸업(경영학석사) 1994년 8월 전북대학교 대학원 전자공학과 졸업(공학석사) 1988년 9월

~ 현재 숭실대학교 전자계산원 전임교수 주관심분야는 영상처리, HDTV, 멀티미디어 등임



宋寅俊(正會員)

1961년 2월 7일생. 1983년 전북대학교 전기공학과 공학사. 1985년 전북대학교 전기공학과 공학석사. 1988년~ 현재 한국전력공사 근무. 1993년~ 현재 전북대학교 컴퓨터공학과 박사과

정 재학중 주관심분야는 디지털신호 및 영상처리, 컴퓨터비전 등임

郭勳星(正會員) 第 31卷 B 編 第 1 號 參照

현재 전북대학교 컴퓨터공학과 교수