

논문 2010-47SP-6-1

# 적응적인 블록 모드를 이용한 집적 영상의 효율적인 압축 방법

## ( An Efficient Compression Method of Integral Images Using Adaptive Block Modes )

전 주 일\*, 강 현 수\*

( Ju-Il Jeon and Hyun-Soo Kang )

### 요 약

본 논문에서는 집적 영상을 효율적으로 압축하는 방법을 제안한다. 집적 영상은 3차원 영상을 기록하고 표현하는 대표적인 기술이다. 제안된 방법은 3D 이산 코사인 변환(3D-DCT)를 기반으로 하고 있다. 집적 영상 부호화에서 3D-DCT는 인접 요소 영상간의 중복성을 줄이는 효율적인 압축 방법으로 알려져 있다. 제안방법은 3D-DCT에 기반 하면서 적응적인 블록 모드를 이용한 효과적인 집적영상 압축 방법으로서, 영상의 특성에 따라 적응적으로 3D-DCT 블록을 구성함으로써 가변블록 크기 3D-DCT 블록을 생성하고, 이 가변블록 크기의 3D 블록에 대해 3D-DCT를 수행한다. 실험 결과를 보면 제안방법이 기존 방법과 비교해서 더욱 높은 압축 효율을 보여주었다. 특히 영상이 복잡하여 많은 비트량이 발생하는 영상이 오버헤드 비트의 영향이 적기 때문에 더 좋은 압축 효율을 보여주었다.

### Abstract

In this paper, we propose an efficient compression method of the integral images. The integral imaging is a well-known technique to represent and record three-dimensional images. The proposed method is based on three dimensional discrete cosine transform (3D-DCT). The 3D-DCT techniques for the integral images have been reported as an efficient coding method for the integral images which reduces the redundancies between adjacent elemental images. The proposed method is a compression method efficient to integral images using adaptive block mode(ABM), based on the 3D-DCT technique. In the ABM, 3D-DCT blocks adaptive to the characteristics of integral images are constructed, which produces variable block size 3D-DCT blocks, and then 3D-DCTs for the 3D blocks are performed. Experimental results show that the proposed method gives significant improvement in coding efficiency. Especially, at the high bit-rates, the proposed method is more excellent since the overhead incurred by the proposed method take less part of the total bits.

**Keywords :** 영상압축, 집적영상, 3D-DCT

## I. 서 론

3차원 영상 기술의 발전은 과학과 엔터테인먼트 산업 양방향에서 끊임없이 추구되었다. 3차원 디스플레이와 영상통신 시스템은 많은 응용분야에서 존재하며, 최근 크게 대두되는 것이 3차원 텔레비전이다. 최근 차세대 오디오비주얼 시스템의 가중 중점이 되는 연구는

다수의 카메라로 촬영되고 다수의 디스플레이로 출력되는 3D 입체 영상이다.

집적영상(integral image)은 3D 영상을 기록하거나 출력하는 테크닉 중의 하나로 한 개의 오브젝트를 다른 시점을 가지는 여러 개의 요소 영상(elemental image)으로 이루어진 영상이다. 집적 영상을 이용하여 각 요소 영상의 시점 차이 등의 데이터들을 이용하여 3차원 입체 영상으로 만들 수 있다<sup>[1~10]</sup>. 집적영상은 렌즈 어레이를 이용하여 촬영된다. 2차원으로 기록된 집적 영상을 3D 입체 영상으로 구현하면 완전한 3D 입체 모델 홀로그래피와 유사하다. 하지만 집적 영상은

\* 정회원, 충북대학교

(Chungbuk National University)

※ 이 논문은 2009년도 충북대학교 학술연구지원사업의 연구비 지원에 의하여 연구되었음

접수일자: 2010년3월10일. 수정완료일: 2010년7월18일

홀로그래피와 다르게 간접적인 빛 없이, 더욱 편리하게 촬영하고 출력할 수 있다. 하지만 실시간 처리, 고해상도 디스플레이 등 몇 가지의 기술적으로 극복해야 할 문제가 있다.

집적 영상은 앞에서 말한바와 같이 여러 개의 요소 영상으로 이루어져 있으며 이 요소영상들의 해상도는 3차원 복원의 정확도에 매우 중요함에 따라 고품질의 영상이 필요하다. 따라서 메모리에 저장하거나 실시간 전송을 위해서는 효율적인 압축 알고리즘이 필요하다.

이런 집적영상에서 인접한 각 요소 영상 간의 상관도가 매우 높기 때문에 각 인접 영상간의 중복성을 제거하여 압축할 수 있는 여러 가지 알고리즘을 적용할 수 있다. 이와 관련하여 기존에는 다음과 같은 알고리즘들이 있다.

먼저 집적 영상을 동영상 압축 방법인 MPEG-2에 적용하여 압축하는 알고리즘이 있다<sup>[11~12]</sup>. 집적 영상의 요소 영상들이 마치 동영상의 프레임처럼 각각 시간적 상관도가 있다고 보고, 요소 영상들을 1차원으로 나열하여 영상 스트림을 구성한 후, 동영상 압축 알고리즘인 MPEG-2에 적용하게 된다. 기존 논문들에서는 집적 영상의 요소영상들을 평행형(parallel), 나선형(spiral), 힐버트형(Hilbert) 등의 다양한 동영상 스트림 구성방법을 제안하고 있다.

또 다른 방법으로 요소 영상간의 교차 상관도를 고려하여 다양한 변환 방법을 이용한 방법들이 있다. 그 중 한가지로 DWT (Discrete Wavelet Transform)와 DCT (Discrete Cosine Transform)를 접목시킨 하이브리드 사용된 하이브리드 압축 방법이 있다. 집적 영상의 각 요소영상을 DWT한 후에 각 요소 영상의 같은 주파수 대역의 픽셀들을 모아 DCT블록을 구성하여, DCT를 한번 더 수행하는 방법이 있다<sup>[13]</sup>. 또 다른 한가지로 각 요소 영상의 2D-DCT 블록을 Z축으로 모아서 3D-DCT 블록을 구성하여 3D-DCT를 수행하는 방법이 있다. 앞에서 언급한 것처럼 각 요소 영상간의 상관도가 높기 때문에 인접한 각 요소 영상의 같은 위치에 있는 2D-DCT 블록을 Z축으로 모아서 3D-DCT를 수행하면 DCT의 효율을 더 향상시킬 수 있다. 집적영상에서 인접한 8개의 요소 영상들의 8x8 DCT블록을 모아서, 8x8x8 크기의 3D-DCT 블록을 구성하여 3D-DCT를 수행하는 방법이 있다<sup>[14]</sup>.

본 논문에서는 이러한 집적 영상을 효율적으로 압축하는 방법에 대하여 제안한다. 집적 영상 역시 촬영한

물체나 촬영 방법에 따라 다양한 특징을 가진다. 제안 방법은 다양한 특징을 가지는 집적 영상에 따라 각 요소영상을 이용하여 적응적으로 3D-DCT블록을 구성하고, 각 3D-DCT블록의 특성에 따라서 가변 블록 크기로 3D-DCT를 수행하는 방법이다.

본 논문은 다음과 같이 구성되어 있다. II장에서는 집적 영상에 대하여 간단히 설명한다. III장에서는 3D-DCT를 이용한 기존의 집적 영상 압축 알고리즘에 대하여 살펴보고 IV장에서 제안한 방법에 대하여 살펴본다. V장에서는 실험을 통하여 제안 알고리즘의 성능을 측정하고 VI장에서 결론을 맺는다.

## II. 집적 영상(Integral Image)

### 1. 집적 영상의 촬영 방법

집적영상(integral image)은 3D 영상을 기록하거나 출력하는 기술 중의 하나로 한 개의 오브젝트를 다른 시점을 가지는 여러 개의 요소 영상(elemental image)으로 구성된 영상이다. 집적 영상을 이용하여 각 요소 영상의 시점 차이 등의 데이터들을 이용하여 3차원 입체 영상으로 만들 수 있다. 집적영상은 그림 1과 같이 렌즈 어레이를 이용하여 촬영된다. 그림 1과 같이 촬영된 영상은 각각의 요소 영상이 한 물체에 대하여 다른 시점의 영상을 가지며, 인접 요소 영상간의 상관도가 매우 높게 나타나게 된다.

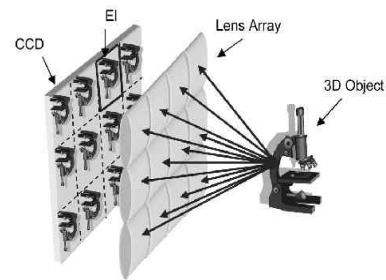


그림 1. 렌즈 어레이를 이용한 집적영상 촬영  
Fig. 1. A shooting integral images using a lens-array.

### 2. 집적 영상의 특징

그림 2를 통해서 실제로 촬영된 집적 영상이 어떤 모습을 가지는지 확인할 수 있다. 집적 영상은 각 요소 영상이 약간의 시차를 가지는 유사한 영상들의 모임임을 확인할 수 있다. 따라서 인접한 각 요소 영상 간의 상관



그림 2. 컵 물체의 집적영상  
Fig. 2. An integral image of a cup object.

도가 매우 높기 때문에 각 인접 영상간의 공간적 중복성을 제거하여 압축할 수 있는 여러 가지 압축 알고리즘을 적용할 수 있다. 대표적으로 요소 영상을 여러 가지 방법으로 나열하여, 동영상 압축 알고리즘을 적용하는 방법이나, 3D-DCT를 이용하는 방법 등이 있다.

### III. 3D-DCT를 이용한 기존 집적영상 압축 방법

#### 1. 3D-DCT (Discrete Cosine Transform)

DCT는 신호의 에너지를 저주파 대역에 모으는데 유용한 도구로서 2D-DCT의 경우 JPEG, MPEG, H.264 등의 많은 표준 영상 부호화 방법에서 널리 사용되고 있다. 3D-DCT는 공간적인 2차원 DCT 블록을 여러 개 모아 시간축을 포함한 3차원 DCT 블록을 만들어 3D-DCT를 수행하는 방법이다<sup>[15]</sup>.  $N_x \times N_y$  블록에 대한 2D-DCT 표현식에 시간축  $N_z$ 를 추가하여,  $N_x \times N_y \times N_z$  입체 블록에 대한 3D-DCT와 3D-IDCT의 표현식을 다음의 식 (1)과 (2)처럼 정의할 수 있다.

-3D-DCT :

$$S(w,v,u) = a_{3D}(w,v,u) \sum_{z=0}^{N_z-1} \sum_{y=0}^{N_y-1} \sum_{x=0}^{N_x-1} s(z,y,x) \cdot \cos(t_1) \cos(t_2) \cos(t_3) \quad (1)$$

-3D-IDCT :

$$s(z,y,x) = \sum_{w=0}^{N_x-1} \sum_{v=0}^{N_y-1} \sum_{u=0}^{N_z-1} a_{3D}(w,v,u) S(z,y,x) \cdot \cos(t_1) \cos(t_2) \cos(t_3) \quad (2)$$

where

$$t_1 = \frac{(2x+1)u\pi}{2N_x}, t_2 = \frac{(2y+1)v\pi}{2N_y}, t_3 = \frac{(2z+1)w\pi}{2N_z},$$

$$a_{3D}(w,v,u) = \sqrt{\frac{2}{N_x} \frac{2}{N_y} \frac{2}{N_z}} C(w) C(v) C(u),$$

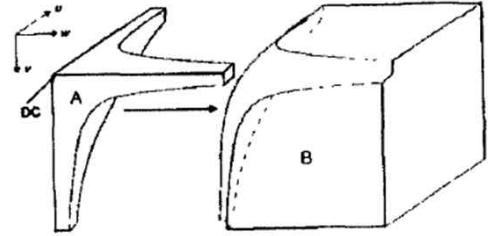


그림 3. 3D-DCT 블록의 입체도  
Fig. 3. A solid of the 3D-DCT.

$$C(k) = \begin{cases} \frac{1}{\sqrt{2}}, & k=0 \\ 1 & \text{otherwise} \end{cases}$$

$s(z,y,x)$ : 픽셀 데이터

$S(w,v,u)$ : DCT계수

$w,v,u$ : 시간과 수평/수직 주파수 변환영역에 대한인덱스

3D-DCT는 그림 3과 같이 저주파 대역으로 에너지를 모음으로써 효율적인 압축을 얻어낼 수 있다.

#### 2. 집적 영상에서의 3D-DCT

앞에서 언급한 바와 같이 집적 영상에서는 각 요소 영상 간 상관도가 높으므로 그림 4에 나타난 바와 같이 인접한 요소 영상들을 모아서 3D-DCT를 수행한다면 저주파 대역에 높은 에너지 집중 효과를 얻을 수 있다.

참고문헌 [16]에서 3D-DCT를 이용한 두 가지의 집적영상 압축 방법을 제안하고 있다. 그 두 가지 방법은 그림 4와 같다. 즉 첫 번째 방법은 가로방향의 8개의 요소 영상들을 모아서 8x8x8 3D-DCT 블록을 구성하여 3D-DCT 압축을 수행한 것이다(3D-DCT1). 두 번째 방

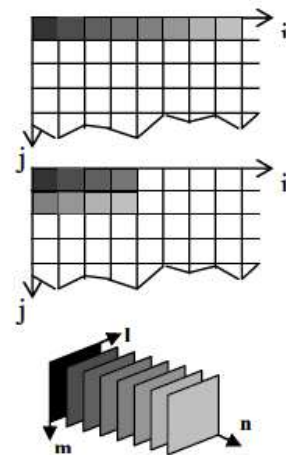


그림 4. 집적 영상에서의 두 가지 3D-DCT블록 구성방법  
Fig. 4. Two construction methods of the integral image.

법은 가로로 4개의 요소 영상과 바로 아래쪽 4개의 요소 영상을 모아서 8x8x8 3D-DCT 블록을 구성하여 3D-DCT 압축을 수행한 것이다(3D-DCT2). 두 가지 방법 모두 JPEG 기반의 방법에 비해 좋은 성능을 보여주었다. 본 논문에서 제안한 방법 역시 이 두 가지 방법과 비교를 통하여 성능을 평가하였다.

#### IV. 영상 특성에 따른 적응적 집적영상 압축 방법

일반적인 영상도 촬영 방식이나 촬영한 피사체에 따라 다양한 특성을 가진다. 집적 영상 역시 여러 가지 요인에 따라 다양한 특성을 가지게 된다. 물체를 근접해서 촬영한 경우 그림 5의 (a)처럼 요소 영상 하나가 대상 물체인 바나나의 일부분만을 포함하는 집적 영상이 되며, 물체를 원거리에서 촬영할 경우 그림 5의 (b)처럼 요소 영상 하나 하나가 각각 대상 물체인 바나나와 사과를 모두 포함하고 있는 집적 영상이 된다. 또한 촬영한 물체에 따라 다양한 특성을 가지는 집적영상이 되기도 한다. 하지만 기존의 방법들은 모든 집적 영상을 한 가지 방법으로 부호화 하고 있다. 본 논문에서는 이런 점을 보완하기 위해 집적 영상에 특성에 따라 적응적으로 3D-DCT블록을 구성하고 가변 블록 크기로 3D-DCT를 수행하여 압축하는 방법을 제안한다.

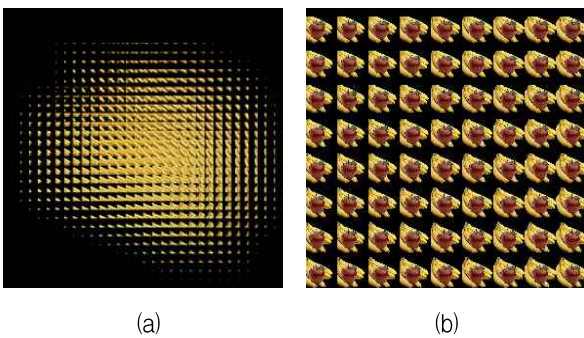


그림 5. 다른 특성을 가진 집적 영상 (a) 근거리로 촬영한 집적영상 (b) 원거리로 촬영한 집적영상

Fig. 5. Two integral images having different character (a) A shoted integral image at a close range (b) A shoted integral image at a long range.

##### 1. 적응적 3D-DCT 블록 구성방법

본 논문에서 제안하는 적응적 3D-DCT 블록 구성 방법은 그림 6 (a)에 나타난 바와 같이 4x4로 이루어진 16개의 요소영상을 한 집합으로 구성된다. 이렇게 한 그룹으로 요소 영상이 집합을 이룰 때 이런 집합을 이

루는 각 그룹마다 다른 특성을 보일 수 있다. 예를 들어 그림 5 (a)의 영상을 보면 상단 좌측의 요소 영상 그룹은 배경이 많고 가운데 부분의 요소 영상 그룹은 대부분이 물체로 이루어져 있다. 또 다른 경우로 각 그룹을 이루는 요소 영상들이 각 그룹마다 다른 경계면을 가지는 물체의 일부분을 포함할 수도 있다. 이런 특징들로 인해 3D-DCT블록을 구성할 때 그림 6의 (b)와 (c)의 두 가지 방법 중에 한 가지를 적응적으로 선택하게 된다. 즉, 두 가지 방법으로 모두 부호화 한 후에 더 적은 비트량이 발생하는 방법으로 부호화 한다. 디코더에는 어떤 방법으로 3D-DCT 블록을 구성했는지 flag 비트 1비트를 추가로 전송한다. 비록 flag비트로 인해 그룹별로 1비트의 오버헤드가 발생하지만 제안 방법을 적용함으로써 얻는 이득에 비해서 큰 오버헤드는 아니다. 본 논문에서는 32x32픽셀 크기의 요소 영상이 32x32개로 이루어진 집적 영상을 사용하므로 총 8x8개의 그룹이 나오게 되기 때문에 총 64비트의 오버헤드가 발생한다. 한 그룹이 8x8x16 블록 크기의 3D-DCT블록 16개로 구성된다.

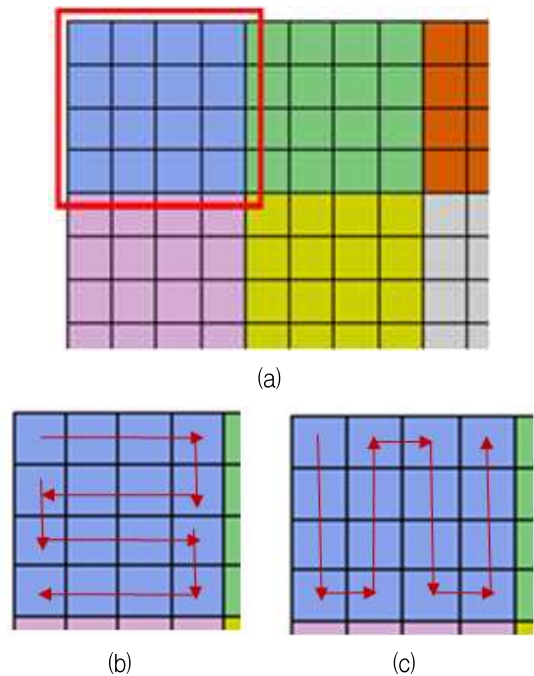


그림 6. 집적 영상의 그룹화 방법 (a) 한 그룹으로 구성되는 요소영상 (b) scan mode0(Horizontal) (c) scan mode1(Vertical)

Fig. 6. A grouping method of the integral images. (a) The elemental images are constructed a one group (b) scan mode0(Horizontal) (c) scan mode1(Vertical).

2. 가변 블록크기 3D-DCT

앞에서 제안한 방법처럼 적응적인 구성 방법 외에도 3D-DCT블록 크기도 압축 성능에 큰 영향을 끼친다. 그 외에도 3D-DCT블록의 크기도 영상 압축 효율에 큰 영향을 미친다. 복잡한 오브젝트의 경우 작은 블록 크기로 변환 및 부호화를 하는 것이 더 효율적이고 배경 같이 복잡하지 않은 부분의 경우 큰 블록 크기로 부호화 하는 것이 효율적이다. 표 1의 실험결과를 그림 9 (a)의 영상1을 3D-DCT블록 크기 별로 압축 성능을 비교한 것으로, 전체적으로 4x4x16으로 부호화한 것이 성능이 높게 나왔다. 그 이유를 살펴보면 그림 7에서 볼 수 있듯이 4x4블록 크기의 경우 모두 배경으로 모든 값이 0값을 가지므로 최소의 비트량이 발생하지만 8x8 크기나 16x16 크기로 블록 크기가 커지면 데이터 값을 포함하게 된다. 그래서 작은 크기로 변환 하는 것이 효율

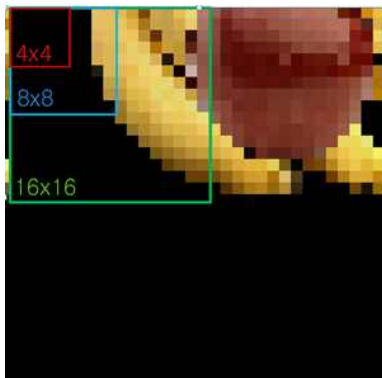


그림 7. 한 요소 영상의 블록 크기  
Fig. 7. A block size of the one elemental image

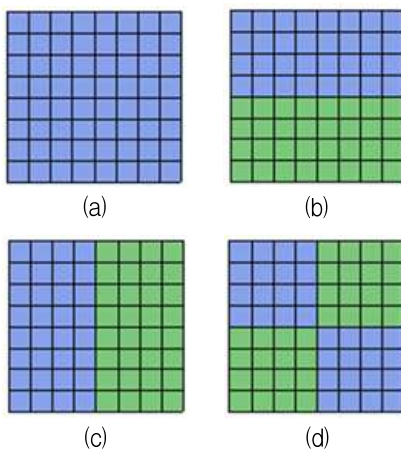


그림 8. 가변 블록크기 부호화의 4가지 모드 (a)Mode0(8x8) (b)Mode0(8x4) (c)Mode0(4x8) (d)Mode0(4x4)  
Fig. 8. The four modes of a variable block size coding (a) Mode0(8x8) (b) Mode0(8x4) (c) Mode0(4x8) (d) Mode0(4x4)

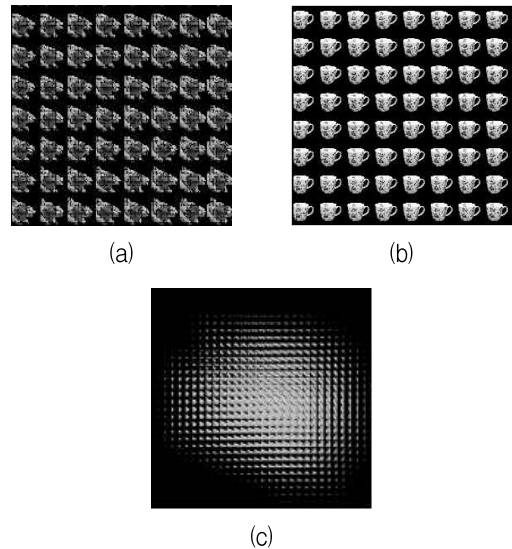


그림 9. 실험에 사용된 집적 영상 (a) 영상1(바나나와 사과) (b) 영상2(컵) (c) 영상3(banana)  
Fig. 9. The integral images using the experiment. (a) image1 (banana and apple) (b) image2 (cup) (c) image3 (banana)

적이지만 16x16 크기의 블록으로 변환해도 모두 0값이 나오는 배경의 경우 16x16 크기의 블록 한 개로 변환하는 것이 8x8 크기의 블록 4개로 부호화하는 것보다 효율이 더 좋게 나오기 때문에 영상의 부분별로 적응적으로 부호화 하는 것이 효율적이다.

본 논문에서는 이런 결과를 바탕으로 가변블록 크기 부호화를 제안한다. 앞에서 구성한 8x8x16 크기의 3D-DCT 블록을 그림 8에서 볼 수 있는 4가지의 블록 크기 모드로 부호화를 모두 수행한 후 각각의 모드별 발생 비트량을 체크하여 더 적은 비트가 발생하는 모드로 부호화 하였다. 가변 블록 크기 부호화 역시 디코더에 어떠한 모드를 사용하였는지 flag비트를 이용하여 전송해줘야 한다. 본 논문에서는 모두 4가지의 모드를 이용하여 부호화하였기 때문에 각 8x8x16 3D-DCT 블록 별로 2비트의 추가적인 오버헤드 비트가 발생한다. 본 논문에서 사용한 집적 영상에서 총 오버헤드 비트는 32x32픽셀 크기의 요소 영상이 32x32개로 이루어진 집적 영상으로 총 8x8개의 그룹이 나오며 한 그룹이 8x8x16 블록 크기의 3D-DCT블록 16개로 구성되므로 8x8x16x4=4096 비트가 발생한다.

이렇게 가변 블록크기 3D-DCT를 이용하여 변환된 계수는 선형 양자화를 통하여 양자화 되고, 지수-골롬(Exponential Golomb) 부호를 이용하여 가변길이부호화(run-length coding)하였다.

## V. 실험

### 1. 실험 영상

실험에는 그림 9에서 나타나 있는 바나나와 사과를 32x32 렌즈어레이를 이용하여 촬영한 집적 영상과 컵을 32x32 렌즈어레이를 이용하여 촬영한 집적 영상 그리고 바나나를 근거리에서 32x32 렌즈어레이를 이용하여 촬영한 영상을 사용하였으며, 전체 영상의 크기는 1024x1024 픽셀 크기이다.

### 2. 실험 결과

앞에서 살펴본 세 가지의 집적 영상을 기존의 3D-DCT를 이용한 두 가지의 방법과 제안 방법을 실험을 통하여 그 성능을 비교하였다. 실험의 결과는 BD-BR과 BD-PSNR을 이용하여 평가하였다<sup>[17]</sup>. BD-BR과 BD-PSNR은 4가지의 QP로 이루어진 실험을 동일한 비트율이나 동일한 PSNR을 기준으로 상대적인 성능을 나타내는 평가 방법이다.

먼저, 표 1을 통해서 3D-DCT블록 크기별 부호화 효율을 살펴볼 수 있다. 영상 1과 영상 2에서 모두 작은 크기의 블록 모드가 큰 크기의 블록 모드 보다 부호화 효율이 좋았다. 앞에서 말했던 것처럼 복잡한 영상에서는 작은 크기의 블록모드가 더 효율적이기 때문이다. 이와 반대로 영상 3을 살펴보면 큰 크기의 블록모드가 부호화 효율이 더 좋았다. 영상 3의 경우 배경 부분만을 가지는 블록이 많고 단순한 블록이 비교적 많이 때

문이다.

실험 영상을 자세히 살펴보면 각 블록별로 배경 부분을 가지고 있는 블록도 있고, 복잡한 물체를 포함하고 있는 블록도 있다. 따라서 영상별로 가변 블록 크기로 부호화 할 경우 각 블록 그룹별로 부호화 모드를 살펴보면, 영상 1의 경우 총 1024개의 블록 그룹 중에 약 400개가 4x4x16블록 모드로 부호화 되었으며, 8x4x16블록 모드와 4x8x16블록 모드가 각각 약 150개 8x8x16블록 모드가 약 300개로 비교적 고른 모드 분포를 보여 주고 있다. 영상 3의 경우 총 1024개의 블록 그룹 중에 약 150개가 4x4x16블록 모드로 부호화 되었으며, 8x4x16블록 모드와 4x8x16블록 모드가 각각 약 120개 8x8x16블록 모드가 약 600개로 큰 블록 크기 모드가 많이 선택되었다. 앞에서 살펴본 바와 같이 영상 3에서 배경 부분만을 가지는 블록으로 인해 큰 크기의 블록모드가 효율적이기 때문이다.

그렇기 때문에 표 1의 결과처럼 각 블록 특성에 따라서 다른 크기의 블록 모드로 적응적으로 부호화 한다면 더 좋은 부호화 효율을 보인다. 하지만 이러한 가변 블록 크기 모드의 경우 모든 크기의 블록으로 부호화를 수행 한 후에 가장 성능이 좋은 블록 모드를 선택하기 때문에 한 개의 블록 모드만으로 부호화 하는 것에 비해 부호화 시간이 매우 오래 걸린다.

여기에 적응적 스캔모드를 추가한 제안 방법이 참고 문헌 [16]에서 제안한 3D-DCT 1과 3D-DCT 2에 비해 PSNR과 비트율 측면에서 뛰어난 성능을 보이고 있다.

표 1. 블록 크기별 부호화 성능 비교

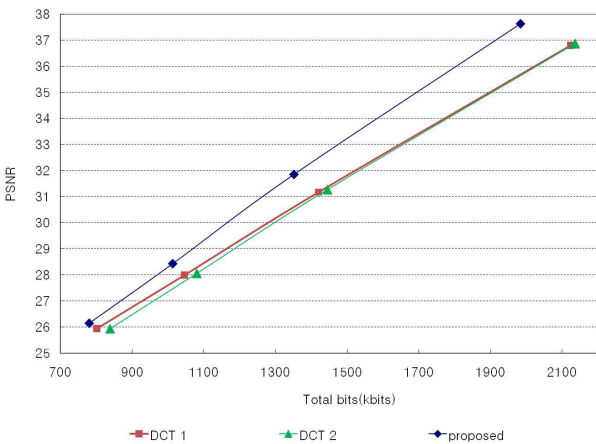
Tabel 1. The coding efficiency of the each block size.

sequence	Quant size	4x4x16		8x4x16		4x8x16		8x8x16		가변블록	
		Total Bits	PSNR	Total Bits	PSNR	Total Bits	PSNR	Total Bits	PSNR	Total Bits	PSNR
영상 1	16.0	1869048	37.48	1965130	37.32	2007242	37.25	2093690	37.10	1880794	37.62
	32.0	1260616	31.87	1320028	31.58	1346138	31.52	1399784	31.33	1259488	31.98
	48.0	949122	28.69	981710	28.37	1002384	28.32	1032910	28.09	943034	28.81
	64.0	745036	26.57	757698	26.25	775762	26.21	788932	25.99	735288	26.69
영상 2	16.0	1701732	38.95	1813518	38.62	1850854	38.56	1962958	38.24	1693742	39.12
	32.0	1196212	33.21	1255738	32.87	1278116	32.79	1334062	32.49	1188950	33.37
	48.0	935852	29.88	964200	29.58	984402	29.55	1013424	29.30	926114	30.07
	64.0	760212	27.63	775742	27.40	794460	27.37	808136	27.15	751320	27.84
영상 3	16.0	1309936	38.97	1300186	39.15	1298212	39.16	1279894	39.36	1243034	39.55
	32.0	836604	33.43	848670	33.54	848362	33.55	852884	33.69	815182	33.93
	48.0	603800	30.57	620734	30.57	623752	30.59	636280	30.64	593596	30.97
	64.0	465522	28.79	481744	28.68	487514	28.71	499054	28.67	459554	29.11

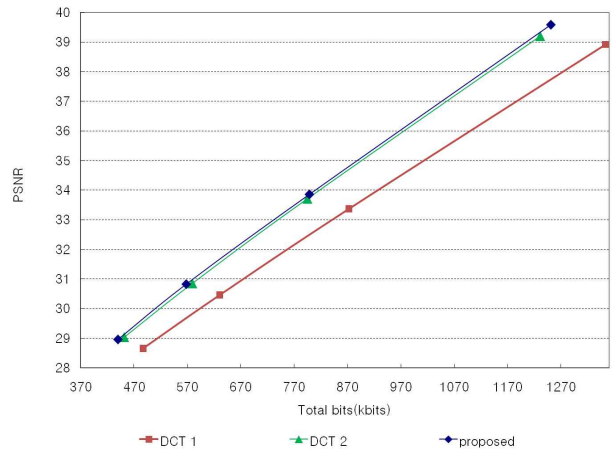
표 2. 기존 방법과 제안 방법의 부호화 성능 비교

Table 2. The comparison coding efficiency of the proposed method with the existing method.

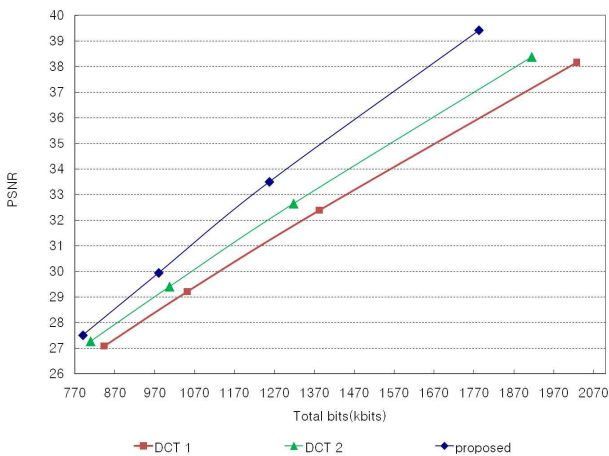
sequence	Quant size	DCT 1		DCT 2		제안방법					
		Total Bits	PSNR	Total Bits	PSNR	가변블록+ 적응적 SCAN		DCT1과 비교		DCT2과 비교	
						Total Bits	PSNR	BD-PSNR	BD-BR	BD-PSNR	BD-BR
영상 1	16	2123762	36.81	2136690	36.88	1983234	37.64	1.13	-9.35	1.33	-10.55
	32	1420412	31.18	1445186	31.28	1351414	31.86				
	48	1046130	28	1080462	28.06	1013126	28.44				
	64	801706	25.95	838424	25.94	780296	26.16				
영상 2	16	2027430	38.17	1914244	38.38	1782120	39.41	2.21	-15.31	1.35	-9.47
	32	1382506	32.39	1318006	32.65	1257058	33.49				
	48	1051468	29.21	1007260	29.41	979740	29.94				
	64	843394	27.08	809696	27.27	789978	27.51				
영상 3	16	1353082	38.92	1230630	39.2	1250848	39.58	1.37	-12.56	0.11	-0.99
	32	872586	33.37	794634	33.71	798486	33.86				
	48	630802	30.46	579186	30.85	567702	30.82				
	64	487174	28.66	451588	29.03	439810	28.96				



(a)



(c)



(b)

그림 10. 비트량에 따른 화질 (a) 집적 영상 1(사과와 바나나) 집적 영상 2(컵) (c) 집적 영상 3(바나나)

Fig. 10. Image quality depending on total bits (a) image1 (banana and apple) (b) image2 (cup) (c) image3 (banana).

즉, 각 그룹별로 적응적으로 3D-DCT블록을 구성하고, 각 3D-DCT 블록 별로 가변 블록 크기를 통해 변환을 수행한 것이 변환 효율이 향상되기 때문에 보다 우수한 부호화 성능을 보인 것으로 판단된다. 영상 1의 경우 제안 방법이 3-DCT1와 3D-DCT2와 비교해 동일 비트량에서 1.2dB의 PSNR 향상을 보여주었다. 영상 2의 경우에는 3D-DCT1에 비해 2.21dB, 3D-DCT2에 비해 1.35dB의 PSNR 향상을 보여주었으며, 영상 3에서

3D-DCT1에 비해 1.37dB, 3D-DCT2에 비해 0.11dB의 PSNR 향상을 보여주었다. 그림 10은 실험결과를 그래프로 나타낸 것이다.

## VI. 결 론

본 논문에서는 집적 영상을 효율적으로 압축하기 위한 3D-DCT 기반의 압축 방법을 제안하였다. 제안한 방법은 집적 영상이 촬영 물체나 촬영 방법에 따라 다양한 특성을 가지는 특성을 바탕으로, 적응적으로 3D-DCT블록을 구성하는 방법과 각 3D-DCT 블록별로 가변 블록 크기 3D-DCT를 수행하는 방법이다. 제안 방법은 영상 특성에 따라 최적의 3D-DCT를 수행하기 때문에 기존의 방법보다 뛰어난 성능을 보여준다. 제안 방법은 기존의 3D-DCT 방법과 비교해서 각 영상별로 동일 비트량에서 약 1dB에서 2dB의 PSNR 향상 효과를 보여주었다.

본 논문에서 제안된 방법은 여러 가지 블록 모드를 정의하고 그 중 최적의 모드를 선택하는 과정을 수행하므로 이에 따른 계산량의 증가를 초래한다. 블록 모드의 개수가 증가할수록 계산량의 그 개수에 정비례하여 증가하므로 이의 개선을 위한 연구가 추가적으로 필요하다. 또한 보다 효율적인 오버헤드 비트 부호화를 통해 추가적인 압축 효율 향상을 기대할 수 있다.

## 참 고 문 헌

- [1] Okoshi T, Three dimensional imaging techniques, Academic Press, 1976.
- [2] N. Davies and M. McCormick, "Holoscopic imaging with true 3D-content in full natural colour," J. Phot. Science, Vol. 40, pp. 46-49, 1992.
- [3] N. Davies, M. McCormick and M. Brewin, "Design and analysis of an image transfer system using microlens arrays," Opt. Eng., vol. 33, no. 11, pp. 3624-3633, 1994.
- [4] F. Okano, H. Hoshino, J. Arai, and I. Yuyama, "Real-time pickup method for a three-dimensional image based on integral photography," Appl. Opt. 36, 1598-1603, 1997.
- [5] F. Okano, H. Hoshino, J. Arai, and I. Yuyama, "Three-dimensional video system based on integral photography," Opt. Eng. 38, 1072-1077, 1997.
- [6] S. A. Benton, ed., Selected Papers on Three-Dimensional Displays, SPIE Optical Engineering Press, Bellingham, WA., 2001.
- [7] D. H. McMahon and H. J. Caulfield, "A technique for producing wide-angle holographic displays," Appl. Opt. 9, 91-96, 1970.
- [8] T. Okoshi, "Three-dimensional displays," in Proceedings of IEEE 68, 548-564, 1980.
- [9] G. Lippmann, "La photographic integral," C. R. Acad. Sci. 146, 446-451, 1908.
- [10] C. B. Burckhardt, "Optimum parameters and resolution limitation of integral photography," J. Opt. Soc. Am. 58, 71-76, 1968.
- [11] S. Yeom, A. Stern, and B. Javidi, "Compression of 3D color integral images," Opt. Express Vol. 12, pp. 1632 - 1642, 2004.
- [12] N. Sgouros, I. Kontaxakis, and M. Sangriotis, "Effect of different traversal schemes in integral image coding," Appl. Opt. 47, 28-37, 2008.
- [13] E. Elharar, A. Stern, and O. Hadar, "A Hybrid Compression Method for Integral Images Using Discrete Wavelet Transform and Discrete Cosine Transform," Journal of display technology, vol. 3, no. 3, 2007.
- [14] A. Aggoun and M. Tabit "Data compression of integral images for 3D TV," 3DTV Conference, pp. 1-4, 2007.
- [15] R.K.W. Chan, M.C. Lee, "3D-DCT quantization as a compression technique for video sequences," VSMM'97. Proceedings., International Conference, pp.188-196, Sept. 1997.
- [16] A. Aggoun and M. Tabit "Data compression of integral images for 3D TV," 3DTV Conference, pp. 1-4, 2007.
- [17] G. Bjontegaard, "Calculation of average PSNR differences between RD-curves," ITU-T Q-6/16, Doc.VCEG-M33, 2001.



— 저 자 소 개 —



전 주 일(정회원)  
2007년 충북대학교 전기전자컴퓨터공학부 학사 졸업  
2009년 충북대학교 정보통신공학과 석사 졸업.  
2010년 현재 충북대학교 정보통신공학과 박사과정.

<주관심분야 : 영상신호처리, 영상부호화>



강 현 수(정회원)-교신저자  
1991년 경북대학교 전자공학과 학사 졸업.  
1994년 KAIST 전기전자공학과 석사 졸업.  
1999년 KAIST 전기전자공학과 박사 졸업.

2010년 현재 충북대학교 정보통신공학부 부교수  
컴퓨터정보통신연구소 연구원

<주관심분야 : 영상신호처리, 영상부호화>