

# DCT와 계층 분할 벡터 양자화를 이용한 3차원 영상 부호화<sup>☆</sup>

## 3D Image Coding Using DCT and Hierarchical Segmentation Vector Quantization

조 성 환\*  
Cho Seong Hwan

김 응 성\*\*  
Kim Eung Sung

### 요 약

본 논문에서는 3차원 영상의 압축 전송을 위하여 3차원 영상에 대해 3차원 DCT를 수행하고 원 영상과의 비교에 따라 영상의 3차원 블록들을 계층적으로 분할하여 각 블록에 대해 유한상태 벡터 양자화를 수행하는 알고리즘을 제안한다. 3차원 DCT의 계수 특징을 이용하여 영상에 크기가 큰 배경 블록과 크기가 작은 윤곽선 블록으로 계층적으로 분할하고, 블록 계층 분할 정보를 전송한다. 각 계층에 속한 블록들에 대해 따로 부호화를 설계하고 부호 비트 수를 줄이기 위해 유한상태 벡터 양자화를 이용하여 부호단어의 인덱스를 계층 분할 정보와 함께 전송한다. Small Lobster와 Head 영상에 대하여 본 알고리즘으로 부호화 했을 때 기존의 HFSVQ를 이용한 알고리즘보다는 각각 1.91 dB과 1.47 dB만큼 더 좋은 영상의 화질을 얻을 수 있었다.

### Abstract

In this paper, for compression and transmission of 3D image, we propose an algorithm which executes 3D discrete cosine transform(DCT) for 3D images, hierarchically segments 3D blocks of an image in comparison with the original image and executes finite-state vector quantization(FSVQ) for each 3D block. Using 3D DCT coefficient feature, a 3D image is segmented hierarchically into large smooth blocks and small edge blocks, then the block hierarchy informations are transmitted. The codebooks are constructed for each hierarchical blocks respectively, the encoder transmits codeword index using FSVQ for reducing encoded bit with hierarchical segmentation information. The new algorithm suggested in this paper shows that the quality of Small Lobster and Head image increased by 1.91 dB and 1.47 dB respectively compared with those of HFSVQ.

☞ Keyword : 3D DCT, FSVQ, HFSVQ

## 1. 서 론

최근의 인터넷을 통한 정보 전달에서 디지털 영상정보를 인터넷 전송망을 통해 전송하는 경우, 영상 데이터의 양이 방대하기 때문에 현재의 인터넷 전송로 하에서 만족할 만한 영상 정보 전달을

위해서는 데이터의 압축이 반드시 필요하다. 특히 3차원 영상은 2차원 평면 영상에 비해 차원이 하나 더 늘어남으로써 언제나 많은 저장 공간이 필요하며 전송량 또한 많게 된다. 대용량의 데이터를 가능한 빠르게 전달하기 위한 시간과 큰 메모리 공간의 필요성은 그런 데이터들의 표현에서도 매우 중요한 과제가 된다.

일반적으로 3차원 영상을 표현하는 방법은 크게 두 가지로 나뉜다. 첫 번째 방법은 데이터의 렌더링을 중시한 표면 표현 방식으로 이는 특정 모양의 기본 조각을 이용하여 3차원 물체를 구성하는 방법이다. 두 번째 방법은 오브젝트가 있는

\* 정 회 원 : 금강대학교 컴퓨터전공 부교수  
shcho@ggu.ac.kr(제 1저자)

\*\* 정 회 원 : 경기공업대학 컴퓨터정보시스템과 조교수  
imagecap@kinst.ac.kr(공동저자)

☆ 이 논문은 2003년도 금강대학교 교내연구비의 지원에 의해서 이루어졌음.

[2004/07/16 투고 - 2004/08/16, 09/17 심사 - 2004/10/11 심사완료]

데이터 전체를 3차원 좌표계에서의 각 점으로 간주하고 특정 값으로 나타냄으로써 데이터 처리의 정확도를 얻고자 하는 스칼라 데이터 표현 방법이다. 이들 방법 중에서 데이터의 정확성을 요구하는 의학 영상 등에 많이 사용되는 스칼라 데이터 표현 방식을 본 논문에서 적용대상으로 선택하였다. 3차원 영상 데이터는 복셀(voxel)이라는 단위로 이루어졌으며, 이 복셀 값을 이용해 3차원 정보로 저장한다. 하지만 이 데이터 표현방식은 그 정확도에 비해 방대한 저장용량을 요구함으로써 데이터의 전송 및 저장에서 문제가 발생할 수 있다. 따라서 데이터의 크기를 줄이는 압축이 요구되었으며, 3차원 영상 데이터 압축에 관한 여러 가지 방법이 제시되었다.

기존 방법 중에는 3차원 DCT(Discrete Cosine Transform)를 이용하는 방법[1]이 있는데, DCT를 이용하는 방법은 2차원 영상의 JPEG 압축 방법을 3차원으로 확장한 형태로서,  $8 \times 8 \times 8$  블록으로 영상을 나누고 3D DCT, 지그재그 스캔, 허프만 부호화 방법을 사용한다.

복원 영상의 화질을 좋게 하고 효과적인 압축 기법을 위해서 여러 기법 중 벡터 양자화(VQ)에 대한 알고리즘들이 개발되고 있는데[2], 이웃 블록 간의 상관관계를 이용하는 유한상태 벡터 양자화(Finite-State Vector Quantization : FSVQ)[3,4] 알고리즘과 이 기법에 계층적 할당 방법을 추가한 계층적 유한상태 벡터 양자화(Hierarchy FSVQ : HFSVQ) 알고리즘[5]이 있다. 이 알고리즘은 영상을 부블록(subblock)으로 나누어 각 블록의 최대 화소 값 차이를 계산하여 각각의 블록을 여러 계층으로 할당한 후, 각 계층에 대해 FSVQ를 적용한다. 그러나 이 알고리즘에서는 한 블록의 특성에 대한 특징 파라미터로서 화소 값의 차이만을 사용하므로 거의 배경에 가까운 블록도 윤곽선(edge) 영역으로 판별하여 전체 복원 영상의 화질을 열화시키는 문제가 발생한다.

본 논문에서는 3차원 영상의 특징 판별을 확실 하 하고 전송 비트율을 낮추기 위하여 3차원 DCT

를 사용하고 영상을 여러 블록 크기를 갖는 계층으로 분할하여 유한상태 벡터 양자화를 수행하는 알고리즘을 제안한다. 본 논문의 알고리즘에서 원 영상은 3차원 DCT를 이용한 블록 분할 작업에 의해 각각 다른 크기를 갖는 블록들로 분해하고, 서로 다른 계층으로 할당한다. 영상의 배경 영역에 해당하는 블록은 크기가 더 크며, 높은 계층으로 할당한다. 화소 값의 변화가 큰 윤곽선 영역에 위치한 블록들에 대해서는 그에 해당하는 주 부호책(master codebook)의 모든 부호단어들이 대표 벡터 선택을 위해 사용된다. 제안 알고리즘을 이용한 3차원 영상 부호화의 부호는 두 부분으로 구성되는데, 영상 블록의 계층 할당에 대한 정보를 제공하는 계층 분할 정보 부호와 부호책에서의 부호단어 인덱스 부호로 구성된다. 본 논문의 제안 알고리즘은 3차원 DCT를 사용하여 각 블록의 특징을 추출함으로써 영상의 블록 계층 분할을 확실히 할 수 있고, FSVQ를 사용함으로써 벡터 양자화의 부호단어 인덱스 전송 비트 수를 적게 유지시킬 수 있다. 시뮬레이션 결과 본 논문에서 제안한 방법으로 영상 부호화를 했을 경우 기존의 HFSVQ[5]를 이용한 방법과 비교하여 더 좋은 화질의 영상을 얻을 수 있었다.

## 2. 3차원 DCT

$J \times K \times L$  크기의 3차원 블록  $f(j,k,l)$ 에 대한 3차원 DCT는 다음과 같이 정의된다[1].

$$F(u,v,w) = \sum_{j=0}^{J-1} \sum_{k=0}^{K-1} \sum_{l=0}^{L-1} f(j,k,l) \cdot A(u,v,w;j,k,l), \quad (1)$$

$$A(u,v,w;j,k,l) = C(u,j) \cdot C(v,k) \cdot C(w,l), \quad (2)$$

$$C(u,j) = \frac{\sqrt{2}}{J} \quad u = 0 \quad (3)$$

$$C(u,j) = \frac{2}{J} \cos \left\{ \frac{(2j+1) \cdot u\pi}{2J} \right\} \quad u > 0$$

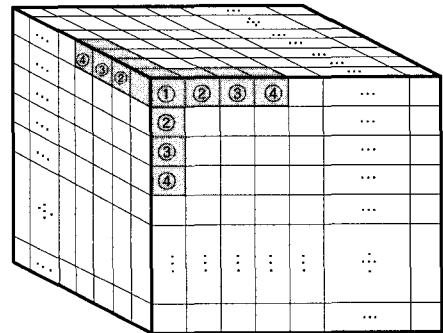
여기서  $A(u,v,w;j,k,l)$ 은 3차원 DCT 커널이고, DCT는 가분리 특성이 있기 때문에 3차원 DCT 커널은 식(2)와 같이 1차원 DCT 커널의 곱으로 나타낼 수 있다. 즉, 3차원 DCT 변환은 1차원 DCT 변환을  $x, y, z$ 축으로 행함으로써 가능하다.

### 3. 3차원 DCT를 이용한 블록 분할

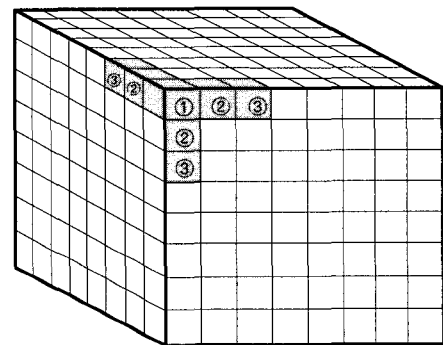
본 논문에서는 벡터 양자화 기법에서 발생하는 오류를 최소화하기 위해 블록들을 계층적으로 분할하고, 각 계층에 속하는 블록들에 따로 벡터 양자화를 수행하는 알고리즘을 제안한다.

일반적인 벡터 양자화 기법은 날카로운 윤곽선 부분의 주파수 오류로 인해 윤곽선을 무디게 만들며 복원 영상의 화질에 큰 영향을 주고, 배경 영역에서의 왜곡(distortion)은 블록화 현상(blocking effect)이 없다면 인간의 눈에 명백하게 관찰되지 않는다. 영상의 각 3차원 블록에 대해 그 블록이 어느 계층에 속하는지 여부를 결정하기 위하여 3차원 DCT를 계산한다. 3차원 DCT를 이용한 각 블록의 계층 분할은 다음과 같은 3 단계로 수행한다.

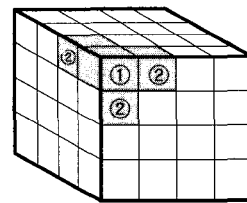
[단계 1] 원 3차원 영상 육면체 중에서 상단의 한쪽 모서리를 원점으로 잡아 그 원점으로부터  $x, y, z$  축을 따라  $16 \times 16 \times 16$ 의 크기를 갖는 부블록(subblock)으로 나누어 3차원  $16 \times 16 \times 16$  DCT를 계산한다. 계산 결과에 대해 그림 1 (a)와 같이 3차원 DCT 계수 값들 중에서 현재 블록의 DC 평균값인 원점 위치의 ①의 값과 저주파수의 AC 값인 각 축의 ②,③,④의 9개 값을 포함한 10개항의 값은 그대로 유지하고 나머지 값은 0으로 만들어 3차원 DCT의 역변환을 계산한다. 그 결과 값들과 원 영상의 해당 블록과의 평균 제곱 오차(Mean Square Error : MSE)를 계산하여 그 값이 문턱치(threshold)  $T_1$ 보다 작으면 현재  $16 \times 16 \times 16$  블록을 계층  $L_1$ 에 할당하고,  $T_1$ 보다 크거나 같으면 현재 블록을 8개의  $8 \times 8 \times 8$  블록으로 분할한다.



(a)



(b)



(c)

(그림 1) 각 블록에서의 3차원 DCT 계수 값 선택  
(a)  $16 \times 16 \times 16$  블록 (b)  $8 \times 8 \times 8$  블록 (c)  $4 \times 4 \times 4$  블록

[단계 2]  $16 \times 16 \times 16$  블록에서 분할된 각각의 8개의  $8 \times 8 \times 8$  블록에 대하여 3차원  $8 \times 8 \times 8$  DCT를 계산한다. 계산 결과에 대해 그림 1 (b)와 같이 ①, ②, ③ 위치의 DCT 계수 값들은 그대로 유지하고 나머지 값은 0으로 만들어 3차원 DCT의 역변환을 취한다. 그 결과 값들과 원 영상의 해당 블록과의 평균 제곱 오차를 계산하여 그 값

이 단계 1과 같은 문턱치  $T_2$ 보다 작으면 현재  $8 \times 8 \times 8$  블록을 계층  $L_2$ 에 할당하고,  $T_2$ 보다 크거나 같으면 8개의  $4 \times 4 \times 4$  블록으로 분할한다.

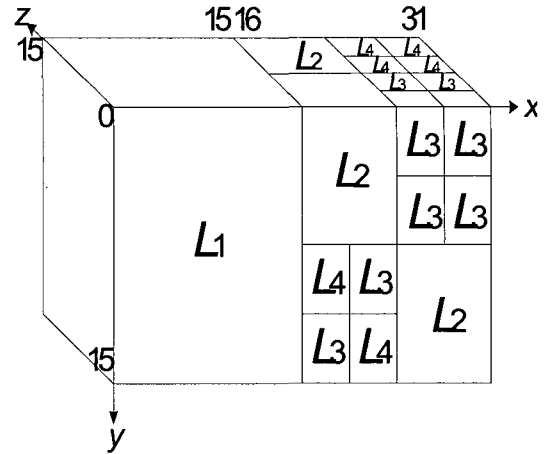
[단계 3] 분할된 8개의  $4 \times 4 \times 4$  블록 각각에 대하여 3차원  $4 \times 4 \times 4$  DCT를 계산한다. 그 결과 그림 1 (c)와 같이 ①, ② 위치의 DCT 계수 값들은 그대로 유지하고 나머지 값은 0으로 만들어 3차원 DCT의 역변환을 취해 원 영상의 해당 블록과의 평균 제곱 오차를 계산하여 그 값이 문턱치  $T_3$ 보다 작으면 현재  $4 \times 4 \times 4$  블록을 계층  $L_3$ 에 할당하고,  $T_3$ 보다 크거나 같으면 현재  $4 \times 4 \times 4$  블록을 계층  $L_4$ 에 할당한다.

#### 4. 계층 분할 정보 전송

3차원 DCT를 이용하여 각 3차원 블록을  $L_1, L_2, L_3, L_4$ 의 4개의 계층으로 분할하면, 각 블록은 4개의 계층  $L_1, L_2, L_3, L_4$ 중 하나에 속하고, 각 계층에 속하는 블록들은  $L_3$ 와  $L_4$ 를 제외하고 서로 크기가 다르다. 그 결과,  $L_1$  계층으로 갈수록 배경 영역에 가까운 블록들을,  $L_4$  계층으로 갈수록 윤곽선과 같은 비교적 블록내의 화소 값 변화가 큰 블록들을 포함하게 된다. 보다 더 정확한 윤곽선 영역의 부호화를 위해 같은  $4 \times 4 \times 4$  블록에 대해서는  $L_3$ 와  $L_4$ 로 분할하는데, 계층  $L_3$ 와  $L_4$ 는  $T_1$ 이나  $T_2$ 보다 더 큰 값을 가지는 문턱치  $T_3$ 를 사용하여 큰 화소 값 변화를 가지는 블록들을 할당한다.

영상의 각 블록에 대해 계층 분할을 한 다음, 각 블록에 대한 계층 할당 정보를 전송하는데, 이는 유한상태 벡터 양자화와 동시에 수행하는 것으로 블록 계층 분할에 대한 오버헤드 정보이다.

그림 2에는 두 개의  $16 \times 16 \times 16$  블록에 대한 계층 분할된 블록들의 예가 나타나 있다. 그림과 같이 계층 할당된 각 블록에 대한 정보는 블록의 벡터 양자화 부호단어 인덱스 전에 전송한다. 그

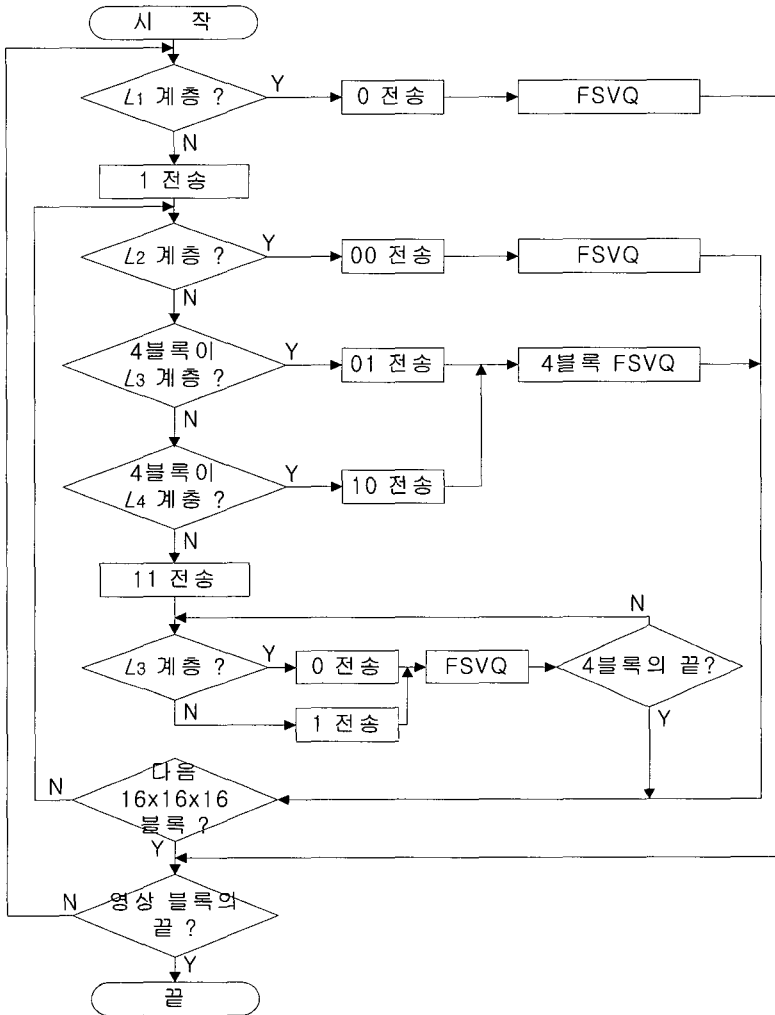


〈그림 2〉 계층 분할된 3차원 블록의 예

림 3은 계층 분할 정보를 전송하는 알고리즘이다.

그림 4는 그림 3의 알고리즘을 사용하여 그림 2의 블록 계층 분할의 예에 대한 계층 분할 정보를 전송한 결과이다.

그림 4에서 계층 정보 순서는  $x, y, z$  축의 순으로 정렬을 하였으며, 그림 2에서 보이지 않는 블록들에 대한 계층 할당 정보는 예로서 밑줄로 표시해 두었다. 그림 4에서 □는 계층 분할 정보를 전송한 후에 해당 블록에 대한 유한상태 벡터 양자화를 수행한 후 전송하는 부호단어에 대한 인덱스 정보를 의미한다. 그림에서  $16 \times 16 \times 16$  크기의 두 3차원 블록에 대해 부가되는 계층 분할 정보량이 크게 증가하지 않는다는 것을 알 수 있다. 기존의 벡터 양자화에 비해서는 이 정보가 오버헤드 정보로서 부가되지만 블록을 계층적으로 분할하였고, 전송 비트 수를 줄이기 위해 벡터 양자화에는 유한상태 벡터 양자화를 수행하기 때문에 전체 전송 비트율을 낮게 유지할 수 있다. 또한 그림 4에서 알 수 있듯이 낮은 계층에 해당하는 블록에 대해서는 더 많은 부호가 할당됨을 알 수 있다. 일반적으로 영상에서 높은 계층에 할당되는 배경 부분에 가까운 블록들의 수가 더 많기 때문에 계층 정보 전송량은 크게 증가하지 않는다.



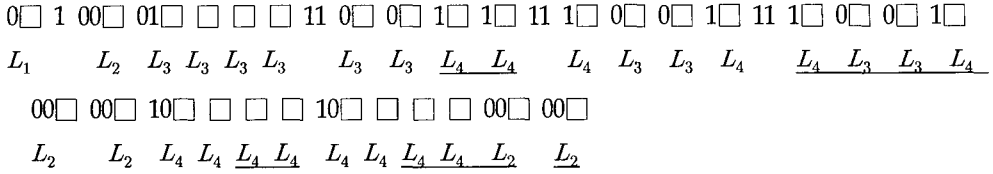
〈그림 3〉 계층 분할 정보 전송 알고리즘

### 5. 계층적 유한상태 부호화

그림 4에서와 같이 각 블록에 대한 계층 분할 정보 전송 후에는 벡터 양자화의 결과인 부호단어 인덱스를 전송해야 한다. 본 논문에서는 각 블록의 부호화에 유한상태 벡터 양자화기(FSVQ)를 사용하는데, 이 FSVQ는 이웃 블록들 사이의 상관관계를 이용하기 때문에 복원 영상의 화질을 개선하고 벡터 양자화기의 전송 비트율을 감소시킬 수 있으며, 이전 블록의 상태를 기억하여 부호화

를 수행한다[6].

유한상태 벡터 양자화기는 일반적인 VQ의 유한한 집합, 즉 부호화기 상태에 따라서 자신의 부호책을 가지는 벡터 양자화기로 생각할 수 있다. 현재 입력 벡터의 상태는 이전에 부호화된 벡터에 의해 정의되며, 각각의 입력 벡터를 현재 부호화기 상태에 의하여 결정된 부호책을 사용하여 부호화한다. 유한상태 벡터 양자화기에는 주 부호책(master codebook)과 상태 부호책(state codebook)이 존재하는데,  $S = \{s_i; i = 1, \dots, M\}$ 를



〈그림 4〉 그림 2의 계층 분할 정보

상태 공간이라 할 때, 각 상태  $s_i$ 에 대하여  $s_i$ 의 상태 부호책으로서 주 부호책  $Y = \{y_i: i = 1, \dots, N_m\}$ 으로부터  $N_f$  개의 부호단어를 선택한다. 이전에 부호화된 벡터의 부호단어를  $p$ 라 할 때, 상태 함수  $f(p)$ 가 현재 입력 벡터  $x$ 의 상태를 결정하기 위하여 사용된다. 입력 벡터  $x$ 를 부호화하기 위하여 부호화기는 현재 상태  $s = f(p)$ 를 찾고 그에 해당하는 부호단어를 찾기 위하여 주 부호책  $Y$ 가 아닌 상태 부호책  $SC_s$ 를 탐색한다. 일반적으로 상태 부호책의 부호단어 수인  $N_f$ 는 주 부호책의 부호단어 수  $N_m$ 보다 훨씬 적다. 현재 부호화하려는 블록에 대해 상태 부호책을 선택하고, 그 상태 부호책으로부터 가장 작은 왜곡을 가지는 부호 단어를 선택하여 그 인덱스를 복호화기로 전송한다. 본 논문에서 부호책은 계층별로 작성하기 때문에 각 계층은 자신의 부호책을 따로 가지고 있다. 각각의 주 부호책 작성 후에 상태 부호책의 설계는 영상의 이웃 블록들 사이의 상관관계를 이용하여 작성한다.

그림 5와 같이 현재 부호화되는 입력 벡터, 즉  $x$

에 대한 이전에 부호화된 상위 블록과 좌측 블록과의 왜곡(distortion)을 계산하여 상태 부호책을 작성한다. 현재 입력 벡터  $x$ 의 상태 부호책  $SC_s$  선택은 각 상태 부호책의 대표 벡터  $p$ 와 현재 입력 벡터의 각각 이웃 열과 이웃 행을 따라서  $x$ 의 상위 블록  $u$ 와 좌측 블록  $l$ 에 가장 잘 부합되는 상태 부호책으로 한다. 즉, 크기  $m \times n$ 의 블록을 정의했을 때, 상태 부호책의 대표 벡터와 현재 입력 벡터의 상위 블록과의 수평 방향 왜곡  $hd(p, u)$ 를

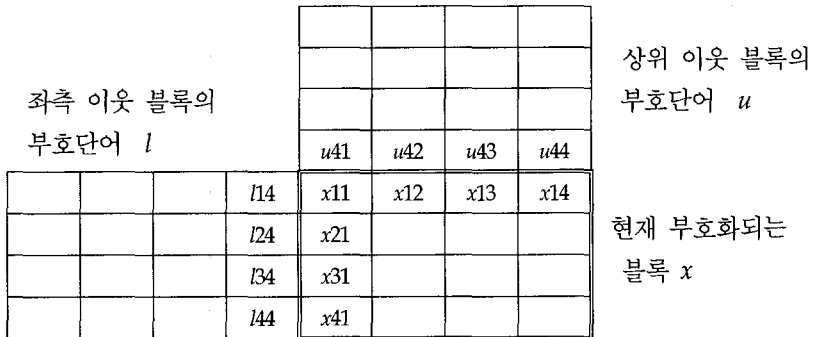
$$hd(p, u) = \sum_{j=1}^n (p_{1j} - u_{mj})^2 \tag{4}$$

라 하고, 수직 방향의 왜곡  $vd(p, l)$ 를

$$vd(p, l) = \sum_{i=1}^m (p_{i1} - l_{in})^2 \tag{5}$$

라 할 때, 입력 벡터  $x$ 의 왜곡  $dist(x)$ 는 다음과 같이 정의한다.

$$dist(x) = hd(p, u) + vd(p, l) \tag{6}$$



〈그림 5〉 유한상태 벡터 양자화기의 부호화. (블록크기 4×4)

여러 상태 부호책  $SC_s$ 들의 대표 벡터  $p$ 에 대하여 가장 작은 왜곡  $dist$ 를 가지는  $SC_s$ 를 현재 입력 벡터의 상태 부호책으로 선택한다. 선택된 상태 부호책의 각 부호단어들과 현재 블록의 화소 값들과의 MSE를 계산하여 그 값이 가장 작은 부호단어의 인덱스를 부호화하여 복호화기로 전송한다. 이와 같이 현재 입력 벡터에 대한 상태 부호책의 선택에 이전 블록의 정보를 이용하기 때문에 복호화기에서 상태 부호책의 선택을 위한 부가 정보를 부호화기에서 전송할 필요가 없다. 이는 결국 큰 주 부호책에서 부호 단어를 선택하여 그에 대한 인덱스를 전송하는 것에 비하여 작은 크기의 상태 부호책의 인덱스만을 전송하여 비트율을 줄이는 효과가 있다.

그림 6은 그림 5의 2차원적 유한상태 벡터 양자화를 3차원으로 확장한 것이다. 그림 5에서는 현재 부호화되는 블록의 좌측과 상단 8개 값과의 비교를 통해 부호화를 수행하지만, 3차원 영상인 경우는 그림 6과 같이 현재 부호화되는 블록(각 그림 하단의 우측 안쪽 블록)의 좌측, 상위, 앞쪽 3개의 면과 이웃하는 블록 면과의 비교를 통하여 부호화를 수행한다.

만일 현재 부호화 블록  $x$ 의 좌측 블록  $l$ 과 상위 블록  $u$ , 앞쪽 블록  $f$ 가 같은 계층에 속하지 않는다면 블록의 크기도 다르다. 이 경우에 각 블록  $l$ 과  $u$ ,  $f$ 는 현재 부호화 블록  $x$ 의 크기로 조정되어야 한다. 예를 들어  $x$ 가 계층  $L_1$ 에 속하지

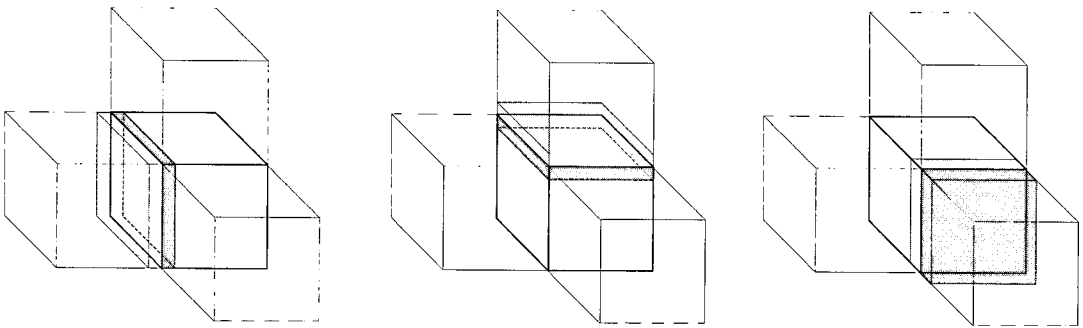
만 그 이웃 블록들이 다른 계층에 속한다면,  $u$ 는  $x$ 의 부호화 계산을 위한 면의 화소 개수와 같도록  $x$  위쪽의 몇몇 블록들을 조합하고, 마찬가지로  $l$ 과  $f$ 도  $x$  좌측과 앞쪽의 블록들을 조합해야 한다. 그러나  $u$ 와  $l$ 과  $f$ 는 현재 블록  $x$ 에 대해 이전에 이미 부호화된 블록들이기 때문에  $x$ 와의 부호화 계산을 위한 면의 화소를 구성할 수 있다. 한편  $x$ 가 낮은 계층에 속한다면 이웃 블록들은 더 높은 계층에 속하게 된다. 즉, 이 경우  $x$ 는  $u$ 와  $l$ ,  $f$ 의 크기보다 작은 블록이 되므로 현재 블록에 대한 상태 부호책 선택 계산을 수행할 수 있다.

## 6. 실험 결과

제안 알고리즘의 평가는 하나의 복셀이 8비트로 구성된  $144 \times 144 \times 96$  크기의 Small Lobster와 Head 영상을 사용하였으며, 영상을 전송하는데 사용된 비트 수를 계산하는데 복셀 당 비트 수인  $bpv(\text{bit/voxel})$ 와 객관적 평가를 위해 복원 영상에 식 (7)의 PSNR을 계산하였다.

$$PSNR = 10 \log_{10} \frac{255^2}{MSE} [\text{dB}] \quad (7)$$

제안 알고리즘에서 계층  $L_1$ 과  $L_2$ 에 대한 주 부호책 크기는 1,024, 상태 부호책 크기는 128로 하였고,  $L_3$ 에 대한 주 부호책 크기는 8,192, 상태



〈그림 6〉 유한상태 벡터 양자화의 3차원 확장

부호책 크기는 2,048로 하였으며,  $L_4$ 에 대한 주 부호책 크기는 2,048로 하였다. 계층  $L_4$ 에 할당된 블록의 부호화에는 블록 경계에 확실한 윤곽선이 존재하는 경우이므로 복원 영상의 화질을 더 좋게 하기 위해 벡터 양자화에 상태 부호책이 아닌 주 부호책을 사용하였다. 블록의 계층 할당을 위한 문턱치  $T_1$ 은 30,  $T_2$ 는 100,  $T_3$ 는 200으로 하였고, 이 수치는 실험적으로 결정하였다. 각 알고리즘에서 주 부호책은 LBG 알고리즘[7]을 사용하여 설계하였다.

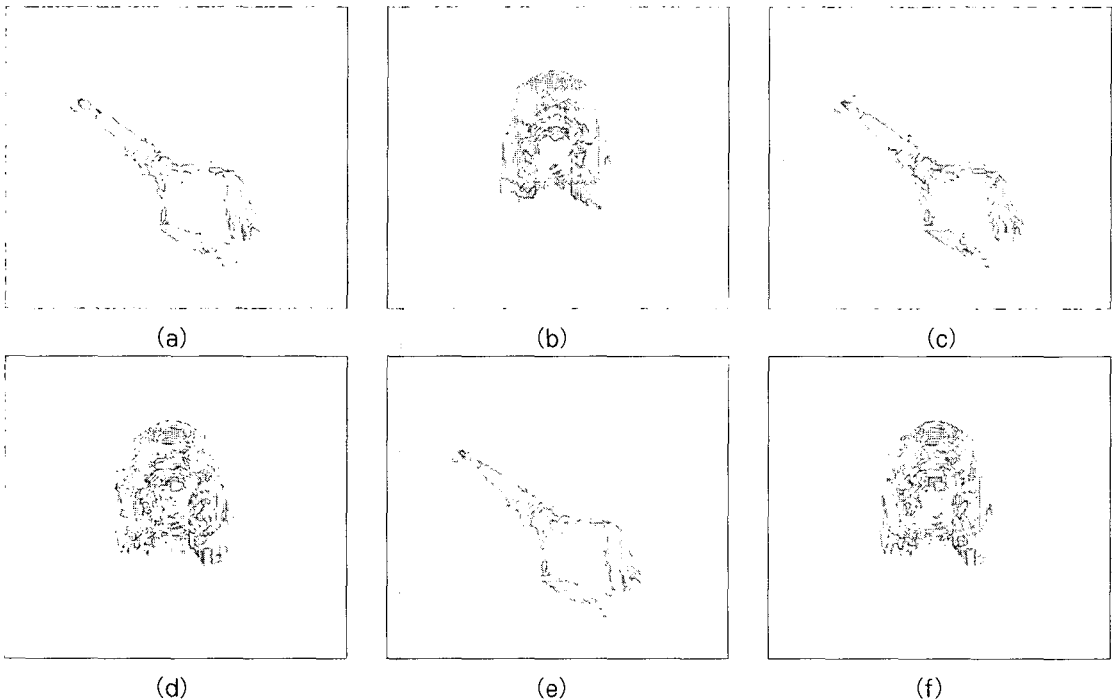
표 1은 기존 알고리즘인 HFSVQ와 제안 알고리즘의 두 영상에 대한 실험 결과를 식 (7)을 이용하여 PSNR 값을 계산한 것이다. 표에서 알 수 있듯이 Small Lobster 영상은 기존의 계층 유한 상태 벡터 양자화 기법인 HFSVQ에 비하여 1.91 dB만큼 화질이 더 좋았으며, Head 영상에 대해서는 HFSVQ에 비하여 1.47 dB만큼 복원 영상의 화질이 좋았다. 두 영상에 대해 HFSVQ보다 제안

〈표 1〉 HFSVQ와 제안 알고리즘의 결과 PSNR(dB) 비교

	HFSVQ	제안 알고리즘
Small Lobster 영상	34.17 (0.12 bpv)	36.08 (0.10 bpv)
Head 영상	33.06 (0.18 bpv)	34.53 (0.15 bpv)

알고리즘의 비트율이 낮았고, 전송 비트 수가 더 적은 상황에서 결과 PSNR이 제안 알고리즘이 더 좋았다. 이로써 본 논문에서 제안한 알고리즘이 더 좋은 성능을 나타냄을 알 수 있다.

그림 7에는  $x$ 축 상에서 본 Small Lobster와 Head의 원영상과 이에 대한 HFSVQ 알고리즘과 본 논문에서 제안한 알고리즘의 표 1에서의 시뮬레이션 결과 영상이 나타나 있다. 그림에서 각각의 HFSVQ 결과 영상과 제안 알고리즘의 결과 영상을 비교해 보면 HFSVQ 알고리즘보다 본 논문의 제안 알고리즘이 더 적게 왜곡이 일어난 것을 볼 수 있다.



〈그림 7〉 “Small Lobster”와 “Head” 영상에 대한 HFSVQ와 제안 알고리즘의 비교 : (a) “Small Lobster” 원 영상 (b) “Head” 원 영상, (c),(d) HFSVQ 결과 영상, (e),(f) 제안 알고리즘의 결과 영상



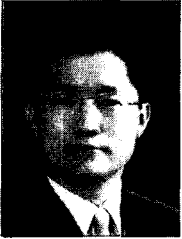
## 7. 결 론

본 논문에서는 3차원 영상을 인터넷 전송로를 통해 전송하는 경우 반드시 필요한 영상 압축 부호화를 위해서 3차원 DCT를 수행하고 그 결과에 대해 영상을 배경 영역과 윤곽선 영역으로 분할하여 각 영역에 대해 따로 벡터 양자화를 수행하는 알고리즘을 제안하였다. 영상을  $16 \times 16 \times 16$  크기의 블록으로 분할하여 3차원 DCT를 계산하고, 일정 영역의 계수 값들을 0으로 만든 다음, 다시 역 DCT를 계산해서 원 영상과 왜곡을 계산하여 문턱치에 따라 그 블록을 8 등분하여 하위 계층으로 분류한다. 계층으로의 분할은  $4 \times 4 \times 4$  크기가 될 때까지 실행하고, 분할된 각각의 블록에 대해 유한상태 벡터 양자화를 수행한다. 유한상태 벡터 양자화 수행 시에 각 블록에 대한 계층 분할 정보도 함께 전송하는데, 이는 실제 벡터 양자화의 부호단어 인덱스 정보가 아닌 오버헤드 정보이지만 유한상태 벡터 양자화를 사용함으로써 전송 비트 수를 많이 줄일 수 있었다. 시뮬레이션 결과 본 논문에서 제안한 방법으로 영상 부호화를 수행했을 경우 HFSVQ 알고리즘과 비교하여 더 좋은 화질의 영상을 얻을 수 있었다. 본 논문에서 제안한 3차원 영상에 대한 압축 알고리즘을 적용함으로써 인터넷을 통한 3차원 의료 영상의 전달이나 3차원 건축 설계 도면 등 크기가 큰 3차원 영상 데이터에 대해 좋은 화질을 갖는 압축된 용량의 영상을 전송할 수 있으며, 추후 연구 과제로는 블록 계층 분할에 사용하는 문턱치들을 실험적으로 결정하였는데, 이 값들을 영상에 따라 적응적으로 정해주는 알고리즘의 연구가 필요하다.

## 참 고 문 헌

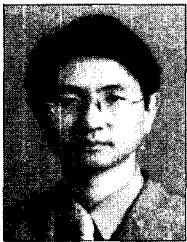
- [1] Boon-Lock Yeo and Bede Liu, "Volume Rendering of DCT-Based Compressed 3D Scalar Data," IEEE Transactions On Visualization and Computer Graphics, Vol. 1, No. March 1995.
- [2] N. M. Nasrabadi and R. B. King, "Image Coding Using Vector Quantization : a review," IEEE Trans. Commun., vol. COM-36, pp. 957-971, Aug. 1988.
- [3] J. Foster, R. M. Gray, and M. O. Dunham, "Finite State Vector Quantization for Waveform Coding," IEEE Trans. Inform. Theory, vol. IT-31, pp. 348-355, May 1985.
- [4] T. Kim, "New Finite State Vector Quantizers for Images," in Proc. ICASSP, 1988, pp. 1180-1183.
- [5] P. Yu and N. Venetsanopoulos, "Hierarchical Finite-State Vector Quantization for Image Coding," IEEE Trans. Commun., vol. 42, no. 11, pp. 3020-3026, Nov. 1994.
- [6] N. M. Nasrabadi and Y. Feng, "A Dynamic Finite- State Vector Quantization Scheme," in Proc. ICASSP, 1990, pp. 2261-2264.
- [7] Y. Linde, A. Buzo, and R. M. Gray, "An Algorithm for Vector Quantizer Design," IEEE Trans. Commun., vol. COM-28, pp. 84-95, Jan. 1980.

◎ 저자 소개 ◎



**조 성 환**

1980년 성균관대학교 전자공학과 졸업(학사)  
1982년 성균관대학교 대학원 전자공학과 졸업(공학석사)  
1991년 성균관대학교 대학원 전자공학과 졸업(공학박사)  
1982년~1985년 해군사관학교 전기 및 전자공학과 전임강사  
1997년 미국 Columbia 대학 CATT Visiting Scholar  
1985년~2002년 동서울대학 컴퓨터공학과 부교수  
2002년~현재 금강대학교 컴퓨터전공 부교수  
관심분야 : 영상처리, 신경회로망, 패턴인식, DRM 등  
E-mail : shcho@ggu.ac.kr



**김 응 성**

1989년 성균관대학교 전자공학과 졸업(학사)  
1992년 성균관대학교 대학원 전자공학과 졸업(공학석사)  
1998년 성균관대학교 대학원 전자공학과 졸업(공학박사)  
1995년~2000년 성균관대학교 과학기술연구소 연구전담요원  
2000년~현재 경기공업대학 컴퓨터정보시스템과 조교수  
관심분야 : 영상처리, 영상통신, 신경회로망 등  
E-mail : imagecap@kinst.ac.kr