

일반논문-09-14-5-02

비트평면 기반 무손실 깊이정보 맵 부호화 방법

김 경 용^{a)}, 박 광 훈^{a)†}, 서 덕 영^{a)}

Bit-plane based Lossless Depth Map Coding Method

Kyung Yong Kim^{a)}, Gwang Hoon Park^{a)†}, and Doug Young Suh^{a)}

요 약

본 논문에서는 MPEG 3차원 비디오 표준 깊이정보 맵에 대한 효율적인 무손실 압축 방법을 제안한다. 일반적으로 깊이정보 맵을 부호화할 때 자연영상에 적용되는 H.264 등의 동영상 부호화 방법을 그대로 사용하고 있는데, 이러한 부호화 방법은 깊이정보 맵의 영상특성을 고려하지 않은 방법이다. 본 논문에서는 깊이정보 맵의 무손실 압축 방법으로 MPEG-4 Part-2 Visual의 이진형상 부호화를 이용한 비트평면 부호화 방법을 제안하였다. 실험결과로서 제안하는 방법이 28.91:1의 압축률을 실현하였고, 화면 내 예측만을 수행한 경우에 JPEG-LS보다 24.84%, JPEG-2000보다 39.35%, H.264 (CAVLC 적용)보다 30.30% 그리고 H.264 (CABAC 적용)보다 16.65% 정도의 비트량 절감을 실현하였고, 또한 화면 내 예측뿐만 아니라 화면 간 예측을 모두 수행한 경우에 H.264 (CAVLC 적용)보다 36.22% 그리고 H.264 (CABAC 적용)보다 23.71% 정도의 비트량 절감을 실현할 수 있었다.

Abstract

This paper proposes a method for efficient lossless depth map coding for MPEG 3D-Video coding. In general, the conventional video coding method such as H.264 has been used for depth map coding. However, the conventional video coding methods do not consider the image characteristics of the depth map. Therefore, as a lossless depth map coding method, this paper proposes a bit-plane based lossless depth map coding method by using the MPEG-4 Part 2 shape coding scheme. Simulation results show that the proposed method achieves the compression ratios of 28.91:1. In intra-only coding, proposed method reduces the bitrate by 24.84% in comparison with the JPEG-LS scheme, by 39.35% in comparison with the JPEG-2000 scheme, by 30.30% in comparison with the H.264(CAVLC mode) scheme, and by 16.65% in comparison with the H.264(CABAC mode) scheme. In addition, in intra and inter coding the proposed method reduces the bitrate by 36.22% in comparison with the H.264(CAVLC mode) scheme, and by 23.71% in comparison with the H.264(CABAC mode) scheme.

Keyword : 3D-Video Coding, Depth Map Coding, FTV (Free view-point TV), MVC (Multi-view Video Coding)

1. 서 론

방송 통신 융합에 따라 디지털 미디어 콘텐츠의 부호화 기술이 급격한 발전을 이루었고, 이로 인해 사용자에게 높은 품질을 가진 디지털 미디어 콘텐츠의 제공이 가능해졌다. 더 나아가 사용자는 현장감과 사실감을 제공해주는 3차원 콘텐츠에 대한 서비스를 요구하고 있다. 이에 비디오 표

a) 경희대학교 전자정보대학

College of Electronics and Information, Kyung Hee University

† 교신저자 : 박광훈(ghpark@khu.ac.kr)

※ 이 논문은 교육과학기술부 및 한국과학재단의 국가지정연구실 지원사업(R0A-2005-000-10061-0)과 지식경제부 및 정보통신연구진흥원의 대학 IT연구센터 지원 사업(ITA-2009-C1090-0903-0011)의 연구 결과로 수행되었음.

· 접수일(2009년4월24일), 수정일(1차:2009년7월13일, 2차:8월18일), 게재확정일(2009년8월18일)

준화 단체인 ISO/IEC의 MPEG(Moving Picture Experts Group)과 ITU-T의 VCEG(Video Coding Experts Group)의 공동 표준화 그룹인 JVT(Joint Video Team)에서는 두 개 시점 이상의 다시점 카메라로부터 입력된 다시점 영상을 효율적으로 부호화하기 위한 다시점 비디오 부호화 (Multi-view Video Coding, H.264 Amd. 4 MVC) 국제표준^[1]을 완료하였다. 하지만 보다 높은 현장감을 제공하기 위해서는 상당히 많은 수의 시점(View) 영상 데이터가 필요하기 때문에 전달되어야 영상 데이터의 양이 커지게 되며, 이는 전달 미디어의 전송 대역폭과 저장 용량의 한계로 인해 큰 어려움이 존재한다. 이러한 문제점을 보완할 수 있는 방법으로 비디오 표준화 단체인 ISO/IEC의 MPEG에서 3차원 비디오 표준^[2]을 시작하려고 하고 있다. 3차원 비디오 표준은 다수의 카메라로부터 입력 받은 영상 이외에도 사용자가 원하는 다양한 가상 시점들을 깊이정보 맵을 이용한 시점 보간(view interpolation) 방법을 통해 무한대까지 생성할 수 있도록 지원한다. 따라서 3차원 비디오 표준에서는 소수의 시점(View) 영상 데이터와 그것의 깊이정보 맵만을 전송하면 되기 때문에, 대역폭과 저장 공간을 절약할 수 있다는 장점을 가진다.

3차원 비디오 표준은 3차원 디스플레이의 다양성을 제공해 주기 위한 표준으로 대표적인 응용 시스템으로 일본 나고야(Nagoya) 대학에서 제안한 자유 시점 TV(FTV(Free view-point TV)) 시스템^[3]이 있다. 자유 시점 TV는 사용자가 원하는 위치와 보는 방향을 자유롭게 선택하여 서로 다른 시점에서 실제 3차원 공간을 보이도록 하는 새롭고 매우

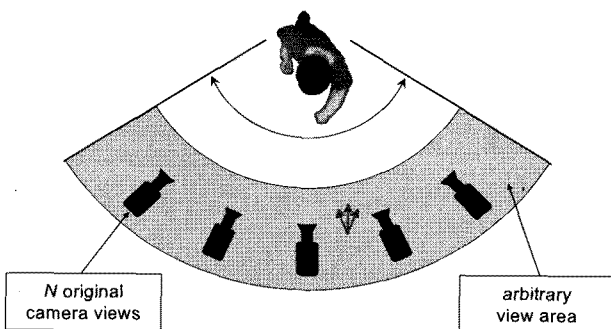


그림 1. FTV 시스템에서의 렌더링 예
Fig. 1. Example of rendering of the FTV system

광범위한 형태의 오디오 비디오 시스템이다. 그림 1은 자유 시점 TV 시스템에서의 렌더링 예^[4]이며, $N(N=5)$ 개의 카메라에서 입력된 영상(검정색 카메라)을 통해 가상 시점의 영상들을 생성함으로써, 다양한 가상 시점(부채꼴 모양의 회색영역)의 영상을 무한대로 생성할 수 있게 한다.

3차원 비디오 표준에서 기본적으로 고려하고 있는 3차원 비디오 시스템의 구성은 우선 스테레오 카메라, 깊이정보 카메라 다시점 카메라, 2차원 영상을 3차원 영상으로 변환 등을 이용하여 획득한 3차원 영상 콘텐츠를 부호화하여 네트워크를 통해 단말에 전송하거나 CD/DVD와 같은 저장 매체에 의해 저장한다. 단말에서는 전송받은 비트스트림을 복호화하여 3차원 영상으로 복원한 후 깊이정보 맵을 이용한 보간(Interpolation) 방법을 통해 가상 시점의 영상들을 생성한다. 생성된 가상 시점의 영상들은 다양한 디스플레이 장치에 맞게 렌더링 된다.

3차원 비디오 표준에서는 입력 받은 시점 영상을 부호화 및 복호화하는 과정 이외에 깊이정보 맵을 생성하는 과정, 깊이정보 맵을 부호화 및 복호화하는 과정, 깊이정보 맵을 이용하여 가상 시점의 영상을 생성하는 방법이 추가적으로 필요하다^[5]. 2009년 현재 3차원 비디오 표준 관련 MPEG 표준화 회의에서는 주로 깊이정보 맵을 생성하는 방법과 깊이정보 맵을 이용하여 가상의 시점 영상을 생성하는 방법에 대한 연구를 수행하고 있다. 깊이정보 맵 부호화와 관련된 국제표준은 아직 존재하지 않으며, 단지 MPEG-C Part 3^[6]에서 깊이정보 맵을 이용하여 3 차원 영상을 렌더링 하기 위해 필요한 변수(화면 내에서 실세계의 카메라와 가장 가까이 있는 객체의 평면 공간의 위치 정보와 가장 멀리 있는 객체의 평면 공간의 위치 정보) 정보에 대한 부호화 방법만 정의하고 있을 뿐이다. 실제로 3 차원 콘텐츠를 이용한 응용 분야에서는 깊이정보 맵을 부호화하기 위한 방법으로, 일반적으로 사용되는 동영상 압축 방법을 그대로 사용하고 있는 실정이다^[7]. 따라서 실제 영상이 아닌 깊이정보 맵의 부호화 방법에 대한 연구가 반드시 필요하다고 할 수 있다.

본 논문의 구성은 다음과 같다. II장에서는 깊이정보 맵과 특성에 대하여 자세하게 분석하고, III장에서는 II에서 분석한 내용을 바탕으로 깊이정보 맵의 특성에 맞게 부호화

하는 방법에 대하여 설명하고, IV장에서 제안하는 방법에 대한 성능을 평가한다. 마지막으로 V장에서는 결론을 기술한다.

II. 깊이정보 맵

깊이정보 맵(Depth Map)이란, 현재 시점에서 카메라와 실제 사물(object)과의 거리를 일정한 비트수로 표현한 것이다. 깊이정보 맵의 예를 들면, 그림 2는 MPEG의 다시점 비디오 부호화의 실험 영상인 "Breakdancers"의 실사 영상(그림 2(a))과 그것의 깊이정보 맵(그림 2(b))을 보여주고 있다. 실제 그림 2의 깊이정보 맵은 각 화소에 대응하는 화면에 보이는 깊이 정보를 8비트로 표현한 것으로, 카메라와 가까울수록 큰 값(밝은 값)으로 표현된다.

그림 2의 깊이정보 맵을 이용하여 각 화소에서 실세계와의 거리(Z)를 구하는 방법은 식 (1)과 같다.

$$Z = Z_{far} + \nu \cdot \frac{Z_{near} - Z_{far}}{255} \quad \text{with } \nu \in [0, \dots, 255] \quad (1)$$

ν 는 그림 2의 깊이정보 맵에서 실제 표현되는 깊이정보 값이고, Z_{far} 와 Z_{near} 는 실제 MPEG-C Part 3에서 정의하는 변수로써 그림 2의 실세계의 관측 공간에서 가장 먼 부분(Z_{far})과 가까운 부분(Z_{near})의 실제 위치를 나타낸다. 따라서 깊이정보 맵에 표현되는 깊이 정보는 영상에서 실세계에서의 가장 먼 부분과 가까운 부분을 2ⁿ (n: 깊이정보 맵을 표현하는 비트수, 그림 2의 깊이정보 맵과 상기 수식에서는 n=8) 등분한 것을 표현한 것이다.

깊이정보 맵은 일반적인 실사 영상과는 다르게 상당히 완만한 특성을 갖는데, 그림 3을 통해서 쉽게 그 특성을 알 수 있다. 그림 3은 그림 2의 실사 영상과 깊이정보 맵의 각 화소의 레벨(실사 영상에서는 휘도(luminance) 성분의 수준 즉 밝기를 뜻하고, 깊이정보 맵에서는 깊이의 수준을 뜻한다)을 표현한 3차원 그래프인데, 실제 영상의 3차원 그래프(그림 3(a))에서는 화소 간 변화가 심한 형태를 보여주고 있음을 확인할 수 있고, 반면에 깊이정보 맵의 3차원(그림 3(b))는 화소 간 변화가 상당히 완만한 형태를 보여주고 있음을 확인할 수 있다.

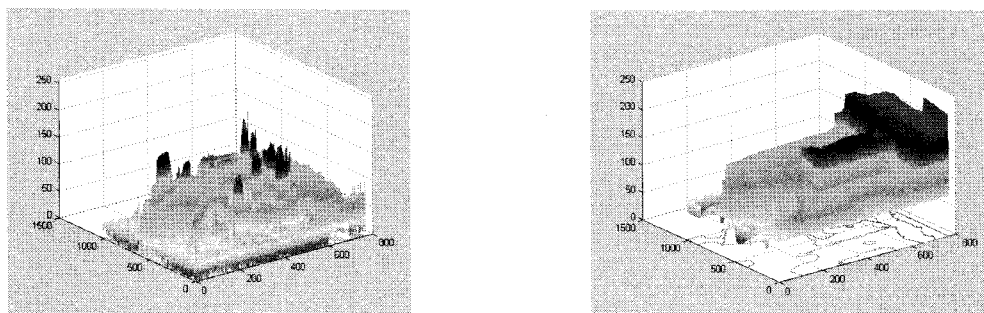
그림 4는 다시점 비디오 영상인 "Ballet"의 깊이정보 맵의 블록(16x16 화소)을 비트평면 단위로 분리한 후 각각의 비트평면에 속한 이진영상(Binary image)의 일부 예를 도시하였다. 비트평면의 이진 영상들을 살펴보면, 어떤 블록은 비트평면의 이진영상 간 완전히 일치하거나 또는 반전



(a) 실사 영상 (b) 깊이정보 맵

그림 2. "Breakdancers"의 실사 영상과 깊이정보 맵 영상 : (a) 실사 영상, (b) 깊이정보 맵

Fig. 2. Real image and depth map image in the "Breakdancers" sequence : (a) real Image, (b) depth map



(a) 실사 영상의 휘도 성분의 3차원 그래프 (b) 깊이정보 맵의 3차원 그래프

그림 3. 그림 2의 실사 영상과 깊이정보 맵의 각 화소 수준을 표현한 3차원 그래프 : (a) 실사 영상, (b) 깊이정보 맵

Fig. 3. 3-D graph represented with each pixel-level of the real image and depth map in Fig. 2 : (a) real image, (b) depth map

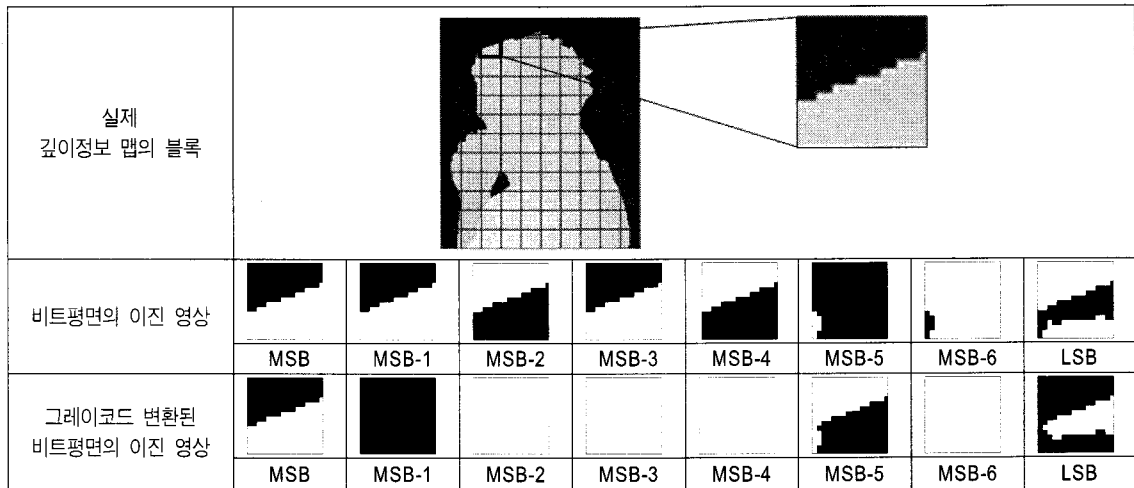


그림 4. "Ballet" 영상의 깊이정보 맵에서 객체 경계 부분 블록의 비트평면 분석
 Fig. 4. Bit-plane analysis of an object boundary block with depth map of in the "Ballet" sequence

되게 일치되는 영상 특성이 자주 발생됨을 발견할 수 있었다. 또한 그림 4의 깊이정보 맵 블록을 그레이코드로 변환한 후 비트평면으로 분리했을 경우, 깊이정보 맵의 화소 값 간의 변화가 상당히 완만하고 그레이코드에 의해 각 비트평면 단위에서 화소 값 간의 유사도가 높아짐으로 인해 분리된 비트평면의 이진 영상들 중에서 블록 내 모든 화소 값이 동일한 경우가 자주 발생하였다. 본 논문에서는 이러한 특성을 이용한 깊이정보 맵의 효율적인 부호화 방법을 제안하고자 한다.

III. 제안하는 깊이정보 맵 부호화 방법

II 장에서 언급된 것처럼 깊이정보 맵의 특성을 살펴보면, 깊이정보 맵은 실사 영상에 비해 화소 간 변화가 상당히 완만하다. 그리고 깊이정보 맵을 비트평면 단위로 분리하여 각각의 분리된 비트평면의 영상을 실사 영상의 비트평면 영상과 비교해보면, 실사 영상의 비트평면 영상에 비해 상당히 단순한 비트평면 영상 형태를 보여준다. 또한 깊이정보 맵을 블록으로 나누고 그 블록을 비트평면 단위로 나누었을 때, 비트평면의 이진 영상 간 완전히 일치되거나

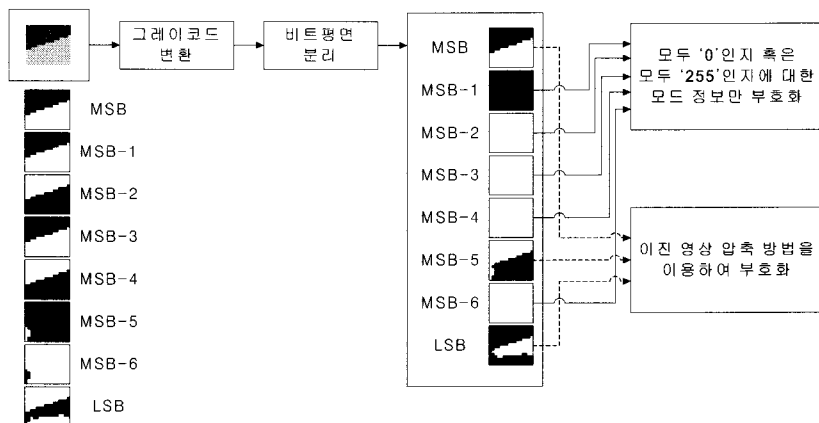


그림 5. 깊이정보 맵 블록을 그레이코드 변환한 후 비트평면 부호화하는 방법
 Fig. 5. Bit-plane encoding method after transforming a depth map block into a gray code

혹은 반전되게 일치하는 경우가 자주 발생하였다.

이러한 현상을 바탕으로 깊이정보 맵을 블록으로 나눈 후 블록의 각 화소를 그레이코드로 변환하고 비트평면 분리하게 되면, 그림 5의 예와 같이 MSB-1, MSB-2, MSB-3, MSB-4, MSB-6 비트평면 블록은 블록 내의 모든 이진 영상 값이 모두 '0(0)'인지 혹은 모두 '255(1)'인지에 대한 모드 정보만 부호화하면 되기 때문에 적은 비트로 부호화가 가능하다. 그리고 MSB, MSB-5, LSB 비트평면 블록은 이진영상 안에서 주변 화소 값 간의 유사도가 높기 때문에 이진영상 압축 방법으로 부호화할 경우에 높은 압축 효율로 부호화가 가능하다.

그림 6은 본 논문에서 제안하는 방법으로, 그레이 부호화를 이용한 깊이정보 맵의 비트평면 단위 부호화 방법에서 부호화 블록도를 보여준다. 부호화 블록도에서 입력되는 데이터는 깊이정보 맵(영상 포맷 4:0:0)이고, 데이터를 처리하는 단위는 가로 세로 16x16화소 크기의 블록이다. 그림 6의 깊이정보 맵의 비트평면 부호화 방법은 다음과 같은 순서로 진행된다.

- 1 단계. 그레이코드 변환 과정에서는 깊이정보 맵 블록을 입력 받아서 각각의 화소를 그레이코드로의 변환을 수행한다. 실제 m -비트 깊이 정보 화소의 이진값 $(a_{m-1} \dots a_2 a_1 a_0)_2$ 를 그레이코드 $(g_{m-1} \dots g_2 g_1 g_0)_2$ 로 변경하는 방법은 식 (2)와 같다.
- 2 단계. 비트평면 분리 과정에서는 n -비트로 표현되는 깊이 정보 맵 블록을 입력 받아서 비트평면 부호화를 위해 n 개의 비트평면 블록으로 분리한다.
- 3 단계. 분리한 n 개의 비트평면 블록은 각각 비트평면 부호화 단계로 MSB 비트평면부터 MSB-1, MSB-2, MSB-3, MSB-4, MSB-5, MSB-6, LSB 비트평면 순으로 입력되어 비트평면 수(n)만큼 비트평면 부호화를 반복해서 수행하며, 비트평면 부호화를 수행한 결과로 비트스트림을 출력한다. 이때 각각의 비트평면을 부호화하는 방법은 그림 5의 예와 같이 비트평면 블록 내의 화소 값이 모두 동일할 경우에는 그 블록이 모두 '0(0)'인지 혹은 모두 '255(1)'인지에 대한 정보만 부호화하고, 비트평면 블록 내의 화소 값이 다를 경우에는 이진 영상 압축 방법을 이용하여 부호화를 수행한다.

$$g_i = a_i \oplus a_{i+1} \quad 0 \leq i \leq m-2 \quad (2)$$

$$g_{m-1} = a_{m-1}$$

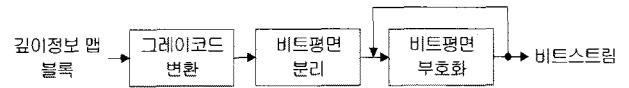


그림 6. 제안하는 방법의 부호화 블록도
Fig. 6. Encoder block diagram of the proposed method

깊이정보 맵의 비트평면 복호화 방법은 그림 7에 도시한 바와 같고, 다음과 같은 순서로 진행된다.

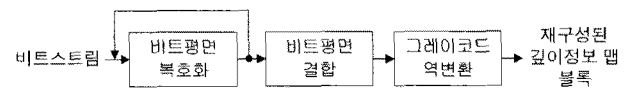


그림 7. 제안하는 방법의 복호화 블록도
Fig. 7. Decoder block diagram of the proposed method

- 1 단계. 비트스트림을 입력 받아 깊이정보 맵의 비트평면 수(n)만큼 반복해서 비트평면 복호화 과정이 수행되며, 입력된 비트스트림을 각 비트평면 별로 복호화하여 n 개의 비트평면 영상을 출력한다.
- 2 단계. 비트평면 결합 과정에서는 출력된 각각의 비트평면 영상을 결합하여 n -비트로 표현되는 영상을 출력한다.
- 3 단계. 역 그레이코드 변환 과정에서는 그레이코드로 표현된 깊이정보 맵을 원래의 깊이정보 맵 형태로 복원을 수행하며, 실제 n -비트 깊이 정보 화소의 그레이코드 $(g_{m-1} \dots g_2 g_1 g_0)_2$ 를 이진값 $(a_{m-1} \dots a_2 a_1 a_0)_2$ 로 변경하는 방법은 식 (3)과 같다.

$$a_{m-1} = g_{m-1} \quad 0 \leq i \leq m-2 \quad (3)$$

$$a_i = a_{i+1} \oplus g_i$$

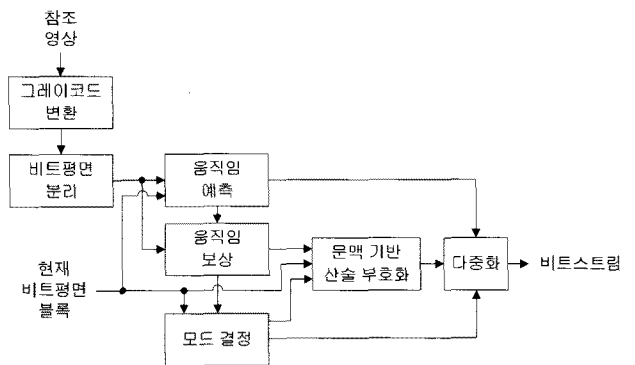


그림 8. 비트평면 부호화 방법의 블록도
Fig. 8. Block diagram of the bit-plane encoding method

그림 6에서 비트평면 부호화를 수행하는 방법은 국제 동 영상 표준인 MPEG-4 Part-2 Visual(ISO/IEC 14496-2)[8]의 이진 형상 부호화(binary shape coding)^[9] 방법을 이용하며, 비트평면 부호화 방법의 블록도는 그림 8과 같다.

그림 8에서 데이터를 처리하는 단위는 가로 세로 16x16 화소 크기로 이진 형태의 비트평면 블록이다. 비트평면 부호화 방법은 다음과 같은 순서로 진행된다.

- 1 단계. 먼저, 움직임 예측 과정은 시간 방향 예측을 사용하여 현재 비트평면을 부호화하고자 할 경우에 수행되는 것으로서 현재 부호화하려는 비트평면 블록과 가장 유사한 부분을 참조 비트평면에서 검색하여 가장 잘 매칭이 되는 영역과의 움직임 변위(움직임 벡터)를 계산하여 출력한다. 이때 참조 비트평면은 시간 방향으로 이전에 부호화된 비트평면 영상으로써 참조 영상을 그레이 코드로 변환한 후 비트평면 분리하여 현재 부호화하려는 비트평면과 동일한 수준의 비트평면을 가져옴으로써 얻어진다.
- 2 단계. 움직임 보상 과정은 시간 방향 예측을 사용하여 현재 비트평면을 부호화하고자 할 경우에 수행되는 것으로, 움직임 예측 과정에서 생성된 움직임 벡터를 이용하여 참조 비트평면으로부터 움직임 보상된 블록을 출력한다.
- 3 단계. 모드 결정 과정에서는 현재 블록과 움직임 보상된 블록을 통해 현재 비트평면 블록의 모드를 결정한다. 비트평면 블록의 모드는 시간 방향 예측을 사용하지 않는 인트라(Intra) 모드와 시간 방향 예측을 사용하는 인터(Inter) 모드로 나뉘어진다. 인트라 모드로 부호화할 경우에 만약 현재 비트평면 블록 내의 화소 값이 모두 동일하다면, 블록 모드 정보('all_0'(비트평면 블록 내의 화소 값이 모두 '0(0)' 또는 'all_1'(비트평면 블록 내의 화소 값이 모두 '255(1)'))만 부호화된다. 그렇지 않고 현재 비트평면 블록 내의 화소 값들이 다르다면, 블록 모드 정보('CAE')뿐만 아니라 현재 비트평면 블록을 문맥기반 산술 부호화(CAE(Context-based Arithmetic Encoding))^[10] 방법으로 이진영상 부호화한다. 인터 모드로 부호화할 경우에 만약 현재 비트평면 블록 내의 화소 값이 모두 동일하거나 참조 비트평면에서 현재 비트평면 블록과 동일한 위치의 참조 블록, 또는 움직임 보상을 수행한 참조 블록이 현재 비트평면 블록과 일치한다면, 블록 모드 정보('all_0'(비트평면 블록 내의 화소 값이 모두 '0(0)' 또는 'all_1'(비트평면 블록 내의 화소 값이 모두 '255(1)')) 또는

'No_residual'(참조 블록이 현재 비트평면 블록과 일치))만 부호화된다. 그렇지 않고 현재 비트평면 블록 내의 화소 값들이 다르거나 움직임 보상을 수행한 참조 블록이 현재 비트평면 블록과 일치하지 않는다면, 블록 모드 정보('CAE')뿐만 아니라 현재 비트평면 블록을 문맥기반 산술 부호화 방법으로 이진영상 부호화를 수행한다.

- 4 단계. 문맥기반 산술 부호화 과정은 현재 부호화할 화소의 주변 화소 값들로 문맥 템플릿을 구성한 후, 구성된 문맥 템플릿을 인덱스(index)로 하는 확률 표를 참조하여 이진 산술 부호화를 수행한다. 이때 현재 비트평면 블록의 모드가 인트라 모드일 경우에는 각 화소의 주변의 화소 정보를 기반으로 문맥 템플릿을 구성하고, 인터 모드일 경우에는 현재 화소의 주변 화소 정보와 현재 화소에 대응하는 참조 비트평면의 화소 그리고 그 주변 화소 정보를 기반으로 문맥 템플릿을 구성한다.
- 5 단계. 다중화 과정에서는 현재 비트평면 블록의 움직임 벡터와 현재 비트평면 블록의 모드, 그리고 문맥기반 산술 부호화를 수행한 결과를 조합하여 비트스트림을 생성한다.

그림 7에서 비트평면 복호화를 수행하는 방법은 국제 동 영상 표준인 MPEG-4 Part-2 Visual^[8]의 이진 형상 부호화^[9] 방법을 이용하며, 그것의 블록도는 그림 9와 같다.

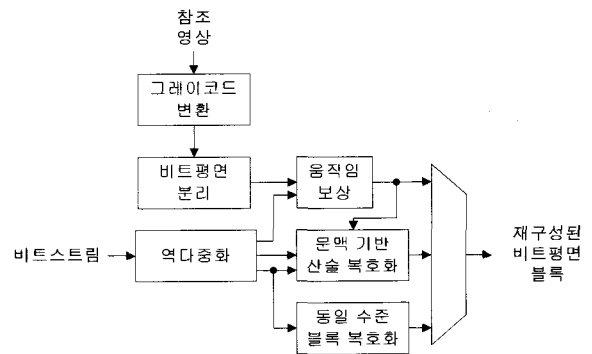


그림 9. 비트평면 복호화 방법의 블록도
Fig. 9. Block diagram of the bit-plane decoding method

그림 9는 비트평면 복호화를 수행하는 구조도를 나타내며, 비트스트림을 입력 받아 비트평면 복호화를 수행하여 재구성된 비트평면 블록을 출력하며, 다음과 같은 순서로 진행된다.

- 1 단계. 역다중화 과정에서는 비트스트림을 입력으로 받아 현재 비트평면 블록의 움직임 벡터와 현재 비트평면 블록의 모드, 그리고 문맥기반 산술 복호화 과정에서 사용될 비트스트림을 출력한다.
- 2 단계. 움직임 보상 과정은 현재 복호화를 수행할 비트평면 블록이 시간 방향 예측을 사용했을 경우에 수행되는 것으로, 복호화된 움직임 벡터를 이용하여 참조 비트평면으로부터 움직임 보상된 비트평면 블록을 출력한다. 이때 참조 비트평면은 시간 방향으로 이전에 복호화된 비트평면 영상으로, 참조 영상을 그레이 코드로 변환한 후 비트평면 분리하여 현재 부호화하려는 비트평면과 동일한 수준의 비트평면을 가져옴으로써 얻어진다.
- 3 단계. 동일 레벨 블록 복호화 과정은 복호화된 비트평면 블록의 모드가 'all_0' 혹은 'all_1' 일 경우에 수행된다. 만약 비트평면 블록의 모드가 'all_0' 라면 비트평면 블록의 모든 화소 값에 '0(0)'을 대입하고, 만약 비트평면 블록의 모드가 'all_1' 이라면 비트평면 블록의 모든 화소 값에 '255(1)'을 대입한다.
- 4 단계. 문맥기반 산술 복호화^[10] 과정은 복호화된 비트평면 모드가 'all_0' 또는 'all_1' 이 아니거나, 혹은

참조 비트평면에서 현재 비트평면블록과 동일한 위치의 참조 블록, 또는 움직임 보상을 수행한 참조 블록이 현재 비트평면 블록과 일치하지 않을 경우에 수행된다. 문맥기반 산술 복호화는 현재 복호화할 화소에 대한 문맥 템플릿을 구성한 후, 구성된 문맥 템플릿을 인덱스로 하는 확률 표를 참조하여 이진 산술 복호화를 수행한다.

IV. 제안하는 방법의 성능 평가

본 논문에서 제안하는 방법의 성능을 평가하기 위하여, MPEG-4 참조 소프트웨어인 Microsoft-FDAM1-2.3-001213^[11]을 기반으로 본 논문에서 제안된 방법을 구현하여 실험을 수행하였고, 또한 정지영상 무손실 압축 방법인 JPEG-LS^[12]의 참조 소프트웨어와 JPEG-2000^[13]의 참조 소프트웨어인 jasper 1.900.0^[14] 그리고 동영상 압축 국제 표준인 H.264 참조 소프트웨어인 JM(Joint Model) 13.2^[15]를 이용하여 깊이정보 맵에 대한 무손실 압축 부호화 성능에 대한 비교를 수행하였다. 실험에 사용한 영상은 그림 10에 도시한 바와 같이, Microsoft에서 제공한 "Ballet", "Breakdancers" 영상의 깊이정보 맵과 Philips에서 제공한

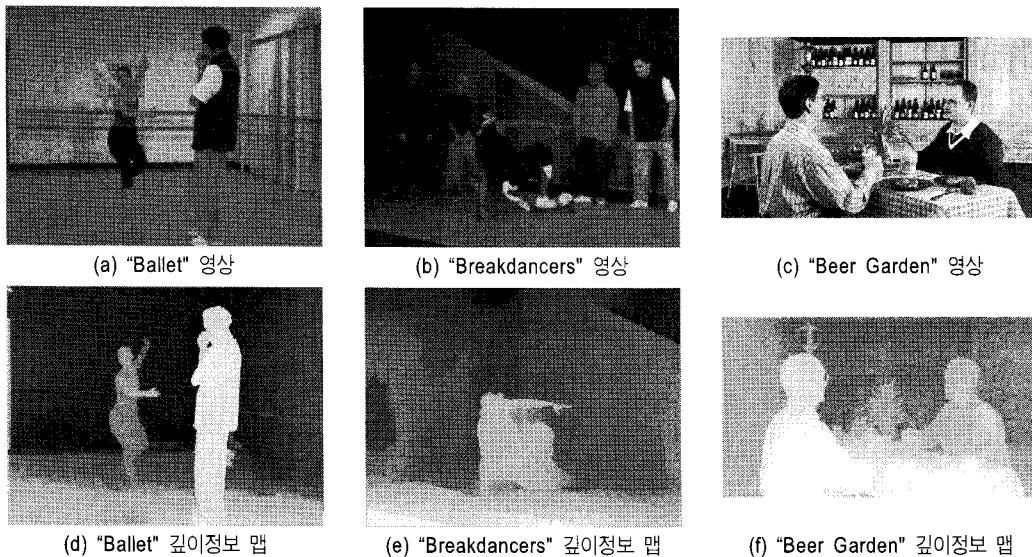


그림 10. 실험 3차원 비디오 영상과 해당 깊이정보 맵
 Fig. 10. Test 3D video sequences and their depth-maps.

“Beer Garden” 영상의 깊이정보 맵이며, 상세한 실험 조건은 표 1과 같다.

“Ballet”과 “Breakdancers” 영상은 실사 영상이며 그것의 깊이정보 맵은 실사 영상을 기반으로 구성되었다. 그리고 “Beer Garden” 영상은 실사 영상이 아닌 컴퓨터 그래픽으로 배경 영상을 구성하고 마스크를 이용한 실사 영상과 조합으로 영상이 구성되었고, 깊이정보 맵 또한 컴퓨터 그래픽 영상의 깊이정보 맵과 실사영상의 깊이정보 맵 간의 조합으로 구성된다.

표 1. 실험 조건
Table 1. Test condition

해상도	1024x768 (Ballet, Breakdancers), 1920x1080 (Beer Garden)
프레임 수	100 Frames(Ballet, Breakdancers), 150 Frames (Beer Garden)
영상 형식	YUV 4:0:0
예측 구조	Intra only (JPEG-LS, JPEG-2000), Intra only (H.264 CAVLC, H.264 CABAC, 제안하는 방법), I-P-P-P- (H.264 CAVLC, H.264 CABAC, 제안하는 방법)
부호화 방법	JPEG-LS, JPEG-2000, H.264(CAVLC, CABAC), 제안하는 방법(CAE)

표 2는 본 논문에서 제안하는 방법과 JPEG-LS, JPEG-2000, 그리고 H.264를 이용하여 깊이정보 맵의 무손실 압축 성능을 비교한 결과를 보여주고 있다. 일반적으로 실사 영상의 무손실 압축에서 JPEG-LS의 부호화 효율이 H.264(CABAC 적용)보다 높는데^[16], 깊이정보 맵의 무손

실 압축에서는 H.264(CABAC 적용)의 부호화 효율이 JPEG-LS보다 높았다.

또한, 화면 내 예측만(Intra only)을 수행하여 부호화한 경우 “Ballet”과 “Breakdancers” 영상에서 제안하는 방법의 압축률이 각각 18.29:1, 16.88:1로 가장 높았다. 반면 “Beer Garden” 영상에서는 H.264(CABAC 적용) 방법의 압축률이 8.60:1로써 가장 높았고 제안하는 방법의 압축률은 7.69:1로써 H.264(CABAC 적용)와 JPEG-LS에 비해 낮았다. 영상특성 분석 결과, “Beer Garden” 실사 영상의 깊이정보 맵에서는 “Ballet”과 “Breakdancers” 영상에서 분석된 비트평면 간 완전일치/반전일치 하는 특성이 존재하였지만, 컴퓨터 그래픽을 이용해서 생성한 배경 영상의 깊이정보 맵은 실사 영상을 이용해서 생성한 깊이정보 맵의 특성(비트평면 간 완전일치 혹은 반전일치)이 잘 나타나지 않았다. 그 결과 본 논문에서 제안한 방법의 경우, 실사영상의 깊이정보 맵과 컴퓨터 그래픽으로 인위적으로 생성된 깊이정보 맵이 혼재한 “Beer Garden” 깊이정보 맵에서 그레이코드 변환을 이용한 비트평면 부호화 효율이 감소하게 된 것으로 분석된다.

하지만, 화면 내 예측뿐만 아니라 화면 간 예측을 모두 수행하여 부호화한 경우 (I-P-P-P-), “Ballet”과 “Breakdancers” 및 “Beer Garden” 깊이정보 맵을 대상으로 한 실험에서, 본 논문에서 제안하는 방법의 압축률이 각각 18.62:1, 16.54:1, 51.56:1로써 다른 부호화 방법들에 비해 월등히 높았다. 특히 “Beer Garden” 깊이정보 맵의 압축률이 화면 내 예측만을 수행한 방법보다 월등히 높게 나오게 된 것은 컴퓨터 그래픽으로 인위적으로 생성된 배경 영상

표 2. 제안하는 방법과 JPEG-LS, JPEG-2000 그리고 H.264와의 무손실 압축률 비교.
Table 2. The comparison of the compression ratios for the proposed method, JPEG-LS, JPEG-2000 and H.264 scheme.

영상	무손실 부호화 방법							
	Intra only					I-P-P-P-		
	JPEG-LS	JPEG-2000	H.264 (CAVLC)	H.264 (CABAC)	제안하는 방법	H.264 (CAVLC)	H.264 (CABAC)	제안하는 방법
Ballet	10.67	8.10	10.44	12.75	18.29	10.67	13.03	18.62
Breakdancers	10.46	7.98	9.64	11.56	16.88	9.59	11.63	16.54
Beer Garden	8.08	6.95	7.29	8.60	7.69	39.21	45.70	51.56
Average	9.74	7.68	9.13	10.97	14.29	19.82	23.45	28.91

표 3. JPEG-LS, JPEG-2000 그리고 H.264 방법과 비교한 제안하는 방법의 비트량 절감율
 Table 3. Bit-rate savings of the proposed scheme in comparison with the JPEG-LS, JPEG-2000 and H.264 scheme

영상	제안하는 방법과 비교한 비트량 절감율					
	Intra only				I-P-P-P-	
	JPEG-LS	JPEG-2000	H.264 (CAVLC)	H.264 (CABAC)	H.264 (CAVLC)	H.264 (CABAC)
Ballet	41.65 %	55.68 %	42.88 %	30.28 %	42.72 %	30.05 %
Breakdancers	38.01 %	52.74 %	42.88 %	31.49 %	41.99 %	29.70 %
Beer Garden	-5.13 %	9.62 %	5.15 %	-11.82 %	23.96 %	11.38 %
Average	24.84 %	39.35 %	30.30 %	16.65 %	36.22 %	23.71 %

의 깊이정보 맵이 시간 축 상에서 전반적으로 거의 변화가 없어서 화면 간 예측 시 부호화할 배경 영상의 잔여 신호에 대한 데이터의 양이 크게 줄어들었기 때문이다.

또한, "Ballet"과 "Breakdancers" 깊이정보 맵의 화면 내 예측뿐만 아니라 화면 간 예측을 모두 수행하여 부호화한 압축률이 화면 내 예측만을 수행하여 부호화한 압축률과 비교해서 비슷한 결과가 나오게 된 것은 해당 깊이정보 맵을 각 화면마다 독립적으로 생성하기 때문에 깊이정보 맵의 시간적 상관도가 많이 떨어지게 되어 화면 간 예측 효율이 크게 감소했기 때문이다. 이에 3차원 비디오 표준 회의에서는 깊이정보 맵의 시간적 상관도를 높이기 위한 연구도 수행중이다^[17].

그리고 표 3의 제안하는 방법과 비교한 비트량 절감율을 살펴보면, 화면 내 예측만을 수행하여 부호화한 경우 제안하는 방법이 JPEG-LS보다 24.84%, JPEG-2000보다 39.35%, H.264 (CAVLC 적용)보다 30.30% 그리고 H.264 (CABAC 적용)보다 16.65% 정도의 비트량을 절감을 실현할 수 있었다. 또한 화면 내 예측뿐만 아니라 화면 간 예측을 모두 수행하여 부호화한 경우 제안하는 방법이 H.264 (CAVLC 적용)보다 36.22% 그리고 H.264 (CABAC 적용)보다 23.71% 정도의 비트량 절감을 실현할 수 있었다.

V. 결론

본 논문에서는 3차원 비디오 표준의 일부분인 깊이정보 맵의 무손실 부호화 방법에 대한 연구를 수행하였다. 깊이정보 맵은 일반적인 실제 영상과는 다르게 상당히 완만한

영상특성을 지니며, 그것을 블록으로 나누어 비트평면 분리를 수행하면 비트평면 간 이진영상이 완전 일치 또는 반전되게 일치하는 특성이 자주 발생하였다. 이러한 특성을 이용하여 본 논문에서는 블록으로 나누어진 깊이정보 맵을 그레이코드로 변환하고 비트평면 단위로 분리한 후 각각의 비트평면에 이진 형상 부호화를 이용한 비트평면 기반 무손실 압축 방법을 제안하였다. 부호화 성능 비교 평가에서, 제안된 방법은 기존의 부호화 방법들에 비해 월등히 높은 28.91:1의 압축률을 실현하였다. 따라서 깊이정보 맵에 대한 무손실 압축에서는 본 논문에서 제안하는 그레이코드 변환을 이용한 비트평면 단위 부호화 방법이 기존의 압축 방법들을 적용한 것보다 상당히 높은 압축률을 얻을 수 있음을 확인할 수 있었다. 앞으로는 깊이정보 맵의 특성을 고려한 손실 압축에 대한 연구가 향후 이루어져야 할 것이다.

참고 문헌

- [1] ISO/IEC JTC1/SC29/WG11, "Text of ISO/IEC 14496-10:200X/FDAM 1 Multi-view Video Coding," N9978, Hannover, Germany, July 2008.
- [2] ISO/IEC JTC1/SC29/WG11, "Introduction to 3D Video," N9784, Archamps, France, May 2008.
- [3] M. Tanimoto, "Overview of Free Viewpoint Television," Signal Processing: Image Communication, vol.21, no.6, pp.454-461, July 2006.
- [4] A. Smolic, K. Mueller, P. Merkle, N. Atzpadin, C. Fehn, M. Mueller, O. Schreer, R. Tanger, P. Kauff, T. Wiegand, "Multiview video plus depth (MVD) format for advanced 3D video systems," ISO/IEC JTC1/SC29/WG11 and ITU-T SG16 Q.6, JVT-W100, San Jose, USA, 21-27 April, 2007.
- [5] ISO/IEC JTC1/SC29/WG11, "Applications and Requirements on

- 3D Video Coding," N10358, Lausanne, Switzerland, February 2009.
- [6] ISO/IEC JTC1/SC29/WG11, "Text of ISO/IEC FDIS 23002-3 Representation of Auxiliary Video and Supplemental Information," N8768, Marrakech, Morocco, January 2007.
- [7] A. Smolic, K. Mueller, N. Stefanoski, J. Ostermann, A. Gotchev, G.B. Akar, G.A. Triantafyllidis and A.Koz: "Coding Algorithms for 3DTV - A Survey," IEEE Trans. on Circuits and Systems for Video Technology, Vol 7, Issue 11, pp. 1606-1621, November 2007.
- [8] ISO/IEC 14496-2, "Information technology - Coding of audio-visual objects - Part 2: Visual," Version 1: April 1999, Version 2: February 2000, Version 3: May 2004.
- [9] N. Brady, and F. Bossen, "Shape compression of moving objects using context-based arithmetic encoding," Signal Processing: Image Communication, Volume 15, pp. 601-617(17), Number 7, May 2000.
- [10] N. Brady, F. Bossen, and N. Murphy, "Context-based arithmetic encoding of 2D shape sequences," Special session on shape coding (ICIP '97), vol. 1, pp. 29-32, 1997.
- [11] ISO/IEC 14496-5/FDAM1, "MPEG-4 Video Reference Software," N3895, Pisa, Italy, January 2001.
- [12] M. J.Weinberger, "The LOCO-I lossless image compression algorithm: principles and standardization into JPEG-LS," IEEE Trans. Image Process., vol. 9, no. 8, pp. 1309 -1324, Aug. 2000.
- [13] C. Christopoulos, A. Skodras, and T. Ebrahimi, "The JPEG 2000 still image coding system: An overview," IEEE Transactions on Consumer Electronics, Vol. 46, pp. 1103-1127, 2000.
- [14] M.D. Adams, and R.K. Ward, "JasPer: a portable flexible open-source software tool kit for image coding/process," in IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP'04), pp. 241-244, May 2004.
- [15] F. Heinrich-Hertz-Institut, "H.264/AVC JM Reference Software, version JM 13.2," <http://iphome.hhi.de/suehring/tml/>, May 2008.
- [16] Y.L. Lee, K.H. Han, and G.J. Sullivan, "Improved Lossless Intra Coding for H.264/MPEG-4 AVC," IEEE Trans. Image Proc., vol. 15, no. 9, pp. 2610-2615, Sept. 2006.
- [17] 호요성, 이천, "3차원 비디오 부호화를 위한 국제 표준화 기술," 방송공학회지, 제14권 제2호, pp.31 ~ 44, June, 2009.

저 자 소 개



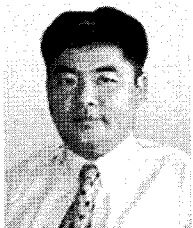
김 경 용

- 2007년 2월 : 경희대학교 컴퓨터공학과 학사
- 2009년 2월 : 경희대학교 컴퓨터공학과 석사
- 2009년 3월 ~ 현재 : 경희대학교 컴퓨터공학과 박사과정
- 주관심분야 : 비디오 신호처리, 멀티미디어, 영상처리



박 광 훈

- 1985년 2월 : 연세대학교 전자공학과 학사
- 1987년 7월 : 연세대학교 전자공학과 석사
- 1991년 1월 : Case Western Reserve University, Dept. of EEAP 석사
- 1995년 1월 : Case Western Reserve University, Dept. of EEAP 박사
- 1995년 3월 ~ 1997년 2월 : 현대전자 멀티미디어연구소 책임연구원
- 1997년 3월 ~ 2001년 2월 : 연세대학교 문리대학 전산학과 부교수
- 2001년 3월 ~ 현재 : 경희대학교 컴퓨터공학과 교수
- 주관심분야 : 멀티미디어, 비디오 신호처리, 패턴인식, 영상처리, 계산지능



서 덕 영

- 1980년 2월 : 서울대학교 원자핵공학과 학사
- 1985년 6월 : Georgia Tech 핵공학과 석사
- 1990년 6월 : Georgia Tech 전자공학과 박사
- 1990년 9월 ~ 1992년 3월 : 상공부 생산기술연구원 HDTV 연구개발단 선임연구원
- 2002년 2월 ~ 2003년 2월 : 미국 North Carolina State Univ. 방문교수
- 1992년 3월 ~ 현재 : 경희대학교 전자공학과 교수
- 주관심분야 : Networked Video