

일반논문-11-16-2-08

## 적응적 배타적 논리합을 이용한 깊이정보 맵 코딩 방법

김 경 용<sup>a)</sup>, 박 광 훈<sup>a)\*</sup>

### A Depth-map Coding Method using the Adaptive XOR Operation

Kyung Yong Kim<sup>a)</sup> and Gwang Hoon Park<sup>a)\*</sup>

#### 요 약

본 논문에서는 실제 영상과는 다른 특성을 지니는 깊이정보 맵의 효율적인 부호화 방법을 제안한다. 깊이정보 맵은 객체 내부 혹은 배경 부분에서 상당히 완만한 특성을 지니지만, 객체 경계 부분에서는 아주 날카로운 에지 성분이 존재한다는 특징이 있다. 그리고 깊이정보 맵을 비트평면 단위로 분리하였을 때, 비트평면 간 완전일치/반전일치되는 특성이 객체 경계 부분에서 자주 발생한다는 특징이 있다. 그래서 본 논문에서는 객체 경계 부분에서 비트평면의 이진 영상간 일치여부를 적절하게 이용하여 깊이정보 맵을 비트평면 단위로 분리하여 비트평면 간 적응적 XOR 연산을 이용한 블록 기반 비트평면 부호화 방법을 제안한다. 또한 비트평면 단위 영상 부호화 방법과 DCT 기반 동영상 압축 방법(H.264/AVC)의 장점을 적절하게 이용하여 블록 단위 비트평면 부호화 방법과 기존의 블록 단위 동영상 부호화 방법을 적응적으로 선택하여 부호화하였다. 실험 결과 제안하는 방법이 H.264/AVC보다 BD-PSNR이 0.9 dB ~ 1.5 dB 향상되었고 BD-rate가 11.8 % ~ 20.8 % 감소되었다. 또한 제안하는 방법이 블록 기반 적응적 깊이정보 맵 부호화 방법보다 BD-PSNR이 0.5 dB ~ 0.8 dB 향상되었고 BD-rate가 7.7 % ~ 12.2 % 감소되어 제안하는 방법의 우수함을 확인할 수 있었다. 또한 복원된 깊이정보 맵을 이용하여 생성된 가상 영상 간의 비교에서 제안하는 방법이 DCT 기반 동영상 압축 방법보다 주관적 화질이 향상된 것을 확인할 수 있었으며, 블록 기반 적응적 깊이정보 맵 부호화 방법과 비교하여 주관적 화질이 비슷하다는 것을 확인할 수 있었다.

#### Abstract

This paper proposes an efficient coding method of the depth-map which is different from the natural images. The depth-map are so smooth in both inner parts of the objects and background, but it has sharp edges on the object-boundaries like a cliff. In addition, when a depth-map block is decomposed into bit planes, the characteristic of perfect matching or inverted matching between bit planes often occurs on the object-boundaries. Therefore, the proposed depth-map coding scheme is designed to have the bit-plane unit coding method using the adaptive XOR method for efficiently coding the depth-map images on the object-boundary areas, as well as the conventional DCT-based coding scheme (for example, H.264/AVC) for efficiently coding the inside area images of the objects or the background depth-map images. The experimental results show that the proposed algorithm improves the average bit-rate savings as 11.8 % ~ 20.8% and the average PSNR (Peak Signal-to-Noise Ratio) gains as 0.9 dB ~ 1.5 dB in comparison with the H.264/AVC coding scheme. And the proposed algorithm improves the average bit-rate savings as 7.7 % ~ 12.2 % and the average PSNR gains as 0.5 dB ~ 0.8 dB in comparison with the adaptive block-based depth-map coding scheme. It can be confirmed that the proposed method improves the subjective quality of synthesized image using the decoded depth-map in comparison with the H.264/AVC coding scheme. And the subjective quality of the proposed method was similar to the subjective quality of the adaptive block-based depth-map coding scheme.

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

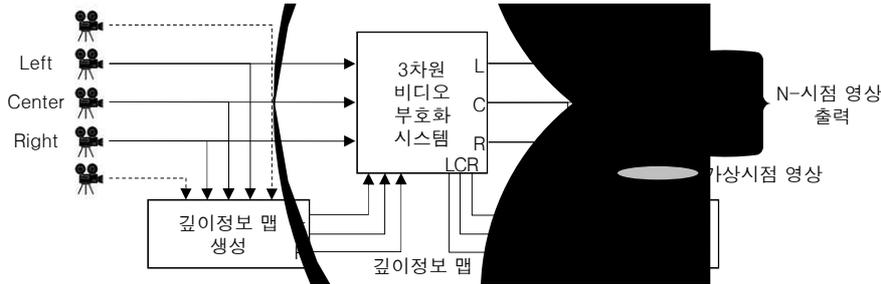


그림 1. 3개의 시점 영상으로 구성된 N-시점 영상 시스템  
 Fig. 1. N-view system with the 3-view configuration

## 1. 서론

최근 3차원 영화의 흥행으로 인해 3차원 콘텐츠에 대한 관심이 높아지고 있다. 그 결과 영화뿐만 아니라 방송, 의료, 광고 등과 같은 다양한 분야에서 3차원 콘텐츠를 활용한 응용 서비스에 대한 연구 개발 및 산업화가 급속도로 진행되고 있다. 특히 3차원 입체 디스플레이 장치의 발전으로 가정에서도 위성방송 혹은 저장매체를 통해 HD급 화질의 3차원 콘텐츠를 시청할 수 있게 되었다. 또한 국내에서는 지상파 3D HDTV 실험 방송이 추진 중이며, 기존의 지상파 방송 방식과의 호환성을 고려한 방송 서비스를 계획 중이다. 기존 지상파 방송 방식과의 호환성을 고려하고 동시에 고화질의 방송 서비스를 실현하기 위해서 가장 문제가 되는 것이 전송 대역폭이다. 지상파 방송의 경우에는 비트율이 19.4 Mbps로 제한되어 있다. 이러한 제한된 비트율 내에서 기존 방식과의 호환성을 가지는 좌영상과 우영상을 고화질로 압축해야 한다는 기술적인 문제점이 존재한다.

이러한 3D 방송 서비스의 전송 대역폭과 관련된 문제점을 해결하고 3차원 콘텐츠를 활용한 응용 서비스의 활성화를 위해서 국제 표준화 기구인 ISO/IEC의 MPEG(Moving Picture Experts Group)에서는 보다 진보된 3차원 데이터 형

식을 정의하였다. 또한, 정의된 3차원 데이터에 대한 처리 기술 및 부호화를 위해서 3차원 비디오 부호화 표준이 진행 중이다. 진보된 3차원 데이터 형식은 다시점 영상뿐만 아니라 깊이정보 맵(Depth-map) 그리고 카메라와 관련된 변수 등의 부가정보로 구성되어 있다. 깊이정보 맵은 실제 세계에서 카메라와 객체 간의 거리를 일정한 비트수로 표현한 영상으로서, 대체적으로 단조로운 영상 특성을 지니고 있기 때문에 실제 영상을 부호화할 때에 비해 높은 압축률을 나타낸다. 따라서 3D 방송 서비스의 전송 대역폭 문제를 해결할 수 있다는 장점을 갖는다. 또한, 이 깊이정보 맵을 이용하여 가상시점 영상을 무한대로 합성할 수 있기 때문에 3차원 응용 서비스에 다양하게 활용될 수 있다는 장점을 갖는다.

그림 1은 MPEG의 3차원 비디오 부호화 표준에서 정의한 N(N>3)-시점 영상 시스템<sup>[1]</sup>이다. 이 시스템에서는 좌, 우, 중앙의 시점에 대한 실제 영상과 그것의 주변 시점의 실제 영상을 입력받아서 좌, 우, 중앙의 시점에 대한 깊이정보 맵을 각각 생성한다. 그리고 좌, 우, 중앙 시점의 실제 영상과 그것의 깊이정보 맵들은 3차원 비디오 부호화 시스템을 통해 부호화 및 복호화 과정을 거치게 된다. 이는 기존의 다시점 영상 서비스를 위해 많은 시점에 대한 실제 영상을 부호화하여 전송할 때 발생하는 문제점인 전송 대역폭의 한계를 실제 영상에 비해 압축률이 높은 깊이정보 맵으로 해결한 것이다. 즉, 소수의 시점에 대한 영상과 그것의 깊이정보 맵만을 부호화함으로써 전송 비트율을 낮출 수 있다. 또한 전송되지 않은 시점 영상과 그 이외의 가상시점 영상들을 깊이정보 맵을 이용하여 무한대로 생성함으로써 다시점 영상 서비스를 제공할 수 있게 된다. 복원된 좌, 우, 중앙 시점의 영

a) 경희대학교 전자정보대학 컴퓨터공학과  
 Media Lab., College of Electronics and Information, Kyung Hee University

‡ 교신저자 : 박광훈 (ghpark@khu.ac.kr)  
 \* 이 논문은 지식경제부 및 정보통신산업진흥원의 대학 IT연구센터 지원 사업(NIPA-2011-(C1090-1111-0001))의 연구 결과로 수행되었음.  
 · 접수일(2010년9월7일), 수정일(1차:2011년1월11일,2차:3월7일), 게재확정일(2011년3월7일)

상들과 여러 개의 가상시점 영상들은 무안경식 다시점 디스플레이 장치에 의해 재생됨으로써 3차원 입체 영상 서비스에 활용될 수 있다. 또한 2차원 및 HM(Head Mount) 디스플레이 장치를 통해 사용자가 원하는 시점을 자유롭게 변경해 가면서 볼 수 있도록 지원하는 자유시점 TV(FTV; Free view-point TV)<sup>[2]</sup> 서비스에도 활용될 수 있다.

이러한 깊이정보 맵을 이용한 3차원 응용 서비스를 위해 MPEG의 3차원 비디오 부호화 표준에서는 깊이정보 맵의 생성 방법과 깊이정보 맵을 이용한 가상시점 영상의 합성 방법에 대한 연구를 주로 수행하였다. 계속된 연구로 인해 최근 몇몇 실험 영상에서 생성된 깊이정보 맵의 화질이 가상시점 영상을 생성하는데 만족할 만한 수준에 이르게 되었다. 그래서 해당 실험 영상을 통해 부호화 실험이 진행되었고, 3차원 응용 서비스를 위해 요구되는 적정 비트율에 대한 측정을 수행하였다. 이러한 실험 결과를 바탕으로, 앞으로는 실제 영상과 깊이정보 맵을 포함한 3차원 데이터의 효율적인 압축 방법에 대한 연구가 수행될 예정이다.

일반적으로 깊이정보 맵을 부호화하기 위해서 사용되는

방법은 실제 영상을 부호화하는데 이용되는 DCT(Discrete Cosine Transform) 기반 부호화 방법을 동일하게 적용하고 있다<sup>[3]</sup>. 이는 깊이정보 맵의 영상 특성을 전혀 고려하지 않은 것이다. 따라서 실제 영상과 다른 특성을 지니고 있는 깊이정보 맵의 효율적인 부호화 방법에 대한 연구가 필요하다.

본 논문의 구성은 다음과 같다. II장에서는 깊이정보 맵의 기존 부호화 방법의 문제점과 적응적 블록 기반 깊이정보 맵 부호화 방법에 대하여 설명한다. III장에서는 깊이정보 맵의 비트평면 간 특성에 대하여 분석하고, IV장에서는 III에서 분석한 내용을 바탕으로 깊이정보 맵의 특성에 맞게 부호화하는 방법에 대하여 설명한다. V장에서 제안하는 방법에 대한 성능을 평가한다. 마지막으로 VI장에서는 결론을 기술한다.

## II. 깊이정보 맵의 기존 부호화 방법

깊이정보 맵은 실세계에서 카메라와 객체 간의 거리를



그림 2. "Breakdancers" 원본 깊이정보 맵(a)과 복원된 깊이정보 맵(e)을 이용해서 생성한 가상시점 영상 간 주관적 화질 비교  
 Fig. 2. Comparison of the subjective picture quality of the synthesized image generated by using the original depth map(a) and the decoded depth map(e) after encoding in the "Breakdancers" sequence

나타내는 영상이므로, 배경 및 객체 내부에서는 실제 영상에 비해 화소의 변화가 상당히 완만하다. 하지만 깊이정보 맵의 객체 경계 부분에서는 화소 간의 차이가 커서 아주 날카로운 에지(Edge) 성분이 나타난다는 특성이 존재한다. 따라서 기존의 DCT 기반 동영상 부호화 방법을 이용하여 깊이정보 맵을 부호화할 경우, 객체 경계 부분에서의 날카로운 에지 성분 때문에 고주파 성분이 현저하게 증가하게 된다. 이러한 고주파 성분의 증가는 양자화 단계에서의 손실로 인해 결과적으로 객체 경계 부분에서 심한 뭉개짐이 발생된다는 문제점이 존재한다. 그림 2는 MPEG의 다시점 비디오 부호화의 실험 영상인 “Breakdancers” 깊이정보 맵의 일부이다. 그림 2(a)는 “Breakdancers”의 원본 깊이정보 맵이고 그림 2(e)는 “Breakdancers”의 깊이정보 맵을 양자화 변수(Quantization Parameter) '37'로 H.264(또는 MPEG-4 part 10 AVC; Advanced Video Coding)<sup>[4]</sup>를 사용하여 압축 후 복원한 영상이다. 또한 그림 2(c)와 그림 2(g)는 각각 그림 2(a)와 그림 2(e)의 깊이정보 맵을 이용하여 생성한 가상시점 영상이다. 가상시점 영상은 MPEG의 3차원 비디오 부호화에서 개발된 가상시점 합성 참조 소프트웨어(VSRS, View Synthesis Reference Software)<sup>[5,6]</sup>를 사용하여 생성하였다. 예를 들어, 8개의 카메라를 이용하여 촬영한 다시점 영상에서 2번 시점(2번 카메라)에 대한 가상시점 영상을 생성하기 위해서는 2번 시점에 인접한 시점의 영상들(1번 시점과 3번 시점 영상)과 그것의 깊이정보 맵(1번 시점과 3번 시점 깊이정보 맵)을 이용하여 생성한다. 그림 2(e)의 깊이정보 맵 일부분을 확대한 영상(그림 2(f))을 확인해 보면, 객체 경계 부분에서 DCT 기반 동영상 부호화 방법의 문제점인 뭉개짐이 존재한다. 그로 인해 그림 2(g)의 가상시점 영상을 확대한 영상(그림 2(h))을 확인해 보면,

객체 경계 부분에서 왜곡이 심하게 발생된 것을 확인할 수 있다. 따라서 가상시점 영상을 효과적으로 생성하기 위해서는 깊이정보 맵을 부호화할 때 객체 경계 부분을 객체와 배경이 명확히 구분되도록 부호화하여야 한다.

이러한 DCT 기반 동영상 부호화 방법의 문제점인 객체 경계 부분의 뭉개짐 현상을 해결하기 위해서 블록기반 비트평면 단위의 깊이정보 맵 부호화 방법<sup>[7,8]</sup>이 제안되었다. 이 방법은 기존의 DCT 기반 동영상 부호화 방법(그림 3(A))과 비트평면 단위의 부호화 방법(그림 3(B))을 H.264/AVC의 율-왜곡 최적화 방법(Rate-distortion Optimization)<sup>[9]</sup>을 이용하여 블록기반 적응적으로 부호화를 수행한다. 비트평면 단위의 부호화 방법은 깊이정보 맵의 객체 경계 부분에서 비트평면의 이진 영상들이 서로 일치하거나 반전일치하는 특성이 자주 발생된다는 점을 이용한다. 비트평면 단위의 이진 부호화 방법은 깊이정보 맵의 객체 경계 부분과 같이 아주 날카로운 에지 성분에서 기존의 DCT 기반 동영상 부호화 방법과 비교해서 높은 부호화 효율을 나타낸다 하지만, 객체 내부 혹은 배경 부분과 같이 평활한 영상에서는 기존의 DCT 기반 동영상 부호화 방법의 부호화 효율이 더 높게 나타났다. 따라서 현재 부호화할 블록이 객체 내부 혹은 배경 부분과 같이 평활한 영상인지 혹은 객체 경계 부분과 같이 날카로운 에지 성분이 있는 영상인지에 따라 적응적으로 부호화되도록 그림 3과 같이 설계되었다. 그림 3(A)의 DCT 기반 동영상 부호화 방법은 평활한 영상에서 수행되며, H.264/AVC를 이용하여 부호화된다. 그림 3(B)의 비트평면 단위의 부호화 방법은 날카로운 에지 성분이 있는 영상에서 수행되며, 그레이코드 변환, 비트평면 분리, 비트평면 부호화의 순서로 부호화된다. 우선 비트평면 부호화단계에서 이진 부호화 효율을 높이기 위해

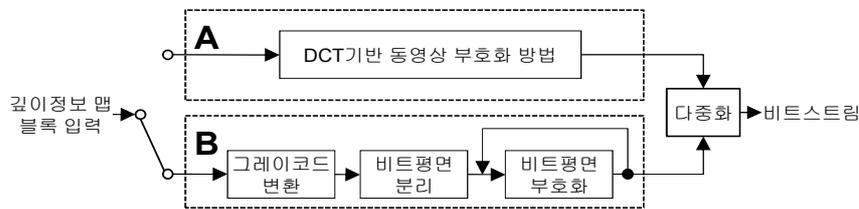


그림 3. 적응적 블록기반 깊이정보 맵 부호화 방법<sup>[7]</sup>  
 Fig. 3. Adaptive block-based depth-map coding method

깊이정보 맵을 그레이코드<sup>[10]</sup>로 변환한다. 이러한 그레이코드 변환 과정은 각 비트평면 내에서 주변 비트값의 유사도를 증가시키는 그레이코드의 특성과 깊이정보 맵 블록의 비트평면 간 일치 혹은 반전일치되는 특성으로 인해 비트평면을 좀 더 단조롭게 변환시킨다. 비트평면 분리 과정에서는 그레이코드로 변환된 깊이정보 맵을 비트평면 단위로 분리한다. 비트평면 부호화 과정에서는 분리된 비트평면들을 최상위 비트평면(MSBP; Most significant bit-plane)부터 최하위 비트평면(LSBP; Least significant bit-plane)까지 MPEG-4 Part-2 Visual(ISO/IEC 14496-2)<sup>[11]</sup>의 이진 형상 부호화(binary shape coding)<sup>[12]</sup>를 이용하여 이진 부호화를 수행한다. 비트평면 부호화 과정의 이진 형상 부호화에서는 최종적으로 이진 형상 부호화에서 사용하는 문맥기반 산술 부호화(CAE; Context-based Arithmetic Encoding)<sup>[13]</sup>를 이용하여 각각의 이진 화소를 부호화한다.

### III. 깊이정보 맵 분석

비트평면 단위의 부호화 방법<sup>[7,8]</sup>에서는 깊이정보 맵의

비트평면을 좀 더 단조롭게 변환하기 위해서 그레이코드로 변환한 후 각각의 비트평면을 이진 부호화한다. 일반적으로는 영상에 그레이 코드를 적용하게 되면 비트평면의 이진 영상이 더욱 단조로운 형태가 된다. 하지만, 이와는 반대로 깊이정보 맵의 블록에 그레이 코드를 적용함으로써 일부 깊이정보 맵 블록의 비트평면이 더욱 복잡해져서 이진 부호화의 효율을 감소시키는 경우가 발생하기도 한다.

예를 들어, 그림 4는 “Breakdancers” 영상의 깊이정보 맵에서 객체 경계 부분의 블록(16x16 화소)들을 비트평면 단위로 분리한 후 각각의 비트평면에 속한 이진 영상을 나타낸다. 그리고 그림 5는 깊이정보 맵 블록에서 그레이코드로의 변환으로 인해 비트평면이 더욱 복잡해지는 경우를 보여준다. 그레이코드로 변환하기 전의 이진 영상들(그림 5(a)와 그림 5(b)와 그림 5(c))과 그레이코드로 변환 후의 이진 영상들(그림 5(d)와 그림 5(e)와 그림 5(f))을 비교해보면, 그레이코드로 변환된 후의 이진 영상들이 더욱 복잡해진 것을 확인할 수 있다.

따라서 본 논문에서는 비트평면 부호화 시 비트평면의 이진영상 간 일치 여부를 적절하게 이용하기 위한 방법으로, 그림 6과 같이 깊이정보 맵을 비트평면 단위로 분리하

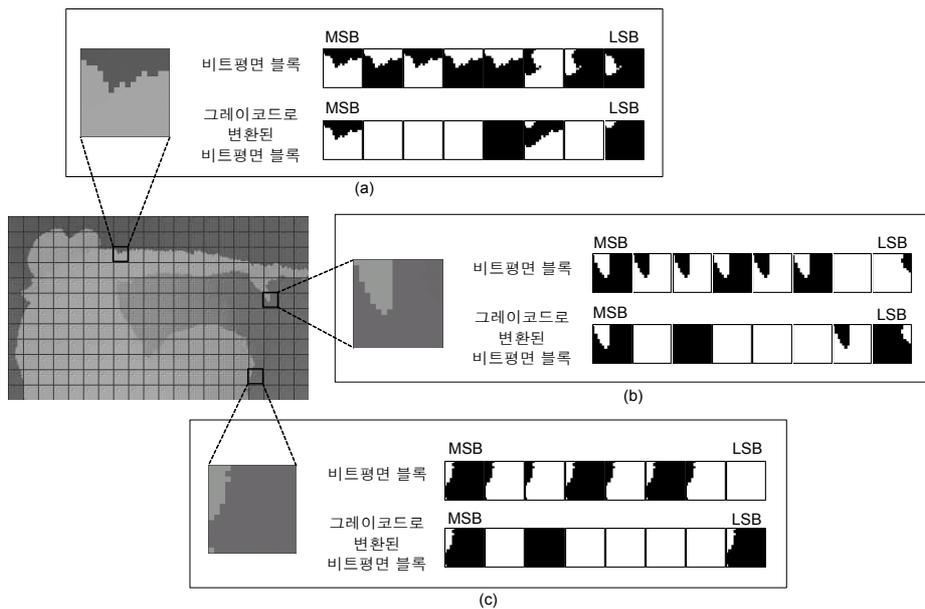


그림 4. “Breakdancers” 영상의 깊이정보 맵에서 객체 경계 부분 블록의 비트평면 분석  
 Fig. 4. Bit-plane analysis of the object boundary block with the depth-map of “Breakdancers” sequence

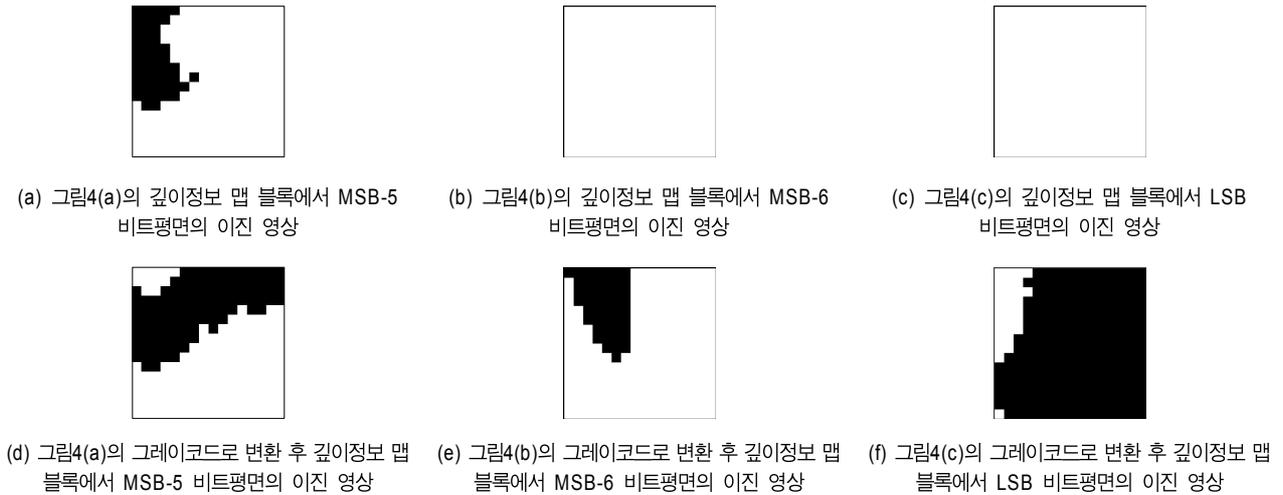


그림 5. 그레이코드로의 변환 전과 변환 후의 비트평면의 이진 영상 간 비교  
 Fig. 5. Comparison of the bit-plane's binary image of before and after the gray coding

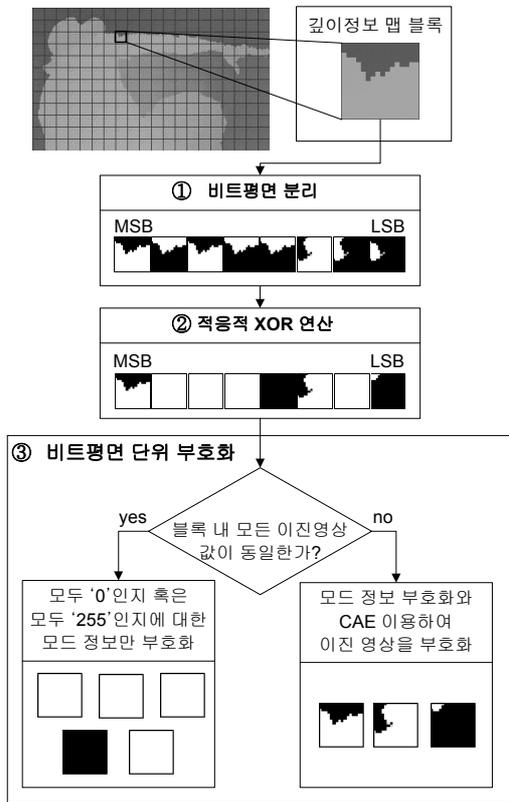


그림 6. 적응적 XOR 연산을 이용한 비트평면 단위 부호화 방법  
 Fig. 6. bit-plane unit coding method using the adaptive XOR method

여 비트평면 간 적응적 XOR(배타적 논리합) 연산을 이용한 블록 기반 비트평면 부호화 방법을 제안하며, 다음과 같은 단계로 부호화를 수행할 수 있다.

- 1 단계. 입력된 깊이정보 맵 블록을 비트평면 단위로 분리한다.
- 2 단계. 각각의 비트평면 블록에 적응적 XOR 연산을 수행한다. 예를 들어, 부호화기에서 MSB-(i) 비트평면 부호화 시, 상위 비트평면인 MSB-(i-1) 비트평면과 XOR 연산을 수행한 후 부호화되어 발생된 비트양('A')을 측정한다. 그리고 MSB-(i) 비트평면 부호화 시, 상위 비트평면인 MSB-(i-1) 비트평면과 XOR 연산을 수행하지 않고 부호화되어 발생된 비트양('B')을 측정한다. 측정된 비트양 'A'와 'B'를 비교하여 비트양이 더 적게 발생된 것을 선택한다. 이러한 XOR 연산의 수행여부 정보는 비트스트림에 포함되어 복호화기로 전송된다.
- 3 단계. MSB 비트평면부터 LSB 비트평면 순으로 아래 그림 6과 같은 비트평면 부호화를 반복적으로 수행한다.
  - 3-1 단계. 블록내의 모든 이진영상 값이 동일한 경우, Step 3-2를 수행하며 다른 경우 Step 3-3을 수행한다.
  - 3-2 단계. 블록내의 모든 이진영상 값이 모두 '0(0)'인지 혹은 모두 '255(1)'인지에 대한 모드 정보만 부호화한다.
  - 3-3 단계. 각각의 비트평면 블록을 문맥기반 산술 부호화 (CAE) 방법으로 부호화한다.

실제 깊이정보 맵의 블록에 적응적 XOR 연산을 수행한 후 비트평면 부호화하는 방법은 객체 경계 부분을 부호화하고자 할 때 DCT 기반 동영상 압축 방법보다 좋은 압축 효율을 나타내지만, 객체 내부 또는 배경 부분에서는 DCT 기반 동영상 압축 방법이 더 좋은 압축 효율을 나타낸다. 그러므로 깊이정보 맵의 객체 경계 부분인지 혹은 깊이정보 맵의 객체 내부 그리고 배경 부분인지에 따라 적절한 부호화 방법이 적용되도록 DCT 기반 동영상 부호화 방법과 적응적 XOR 연산을 이용한 비트평면 부호화 방법을 적응적으로 수행하는 방법이 적용되어야 한다.

#### IV. 적응적 XOR 연산을 이용한 블록 기반 깊이정보 맵 부호화 방법

본 논문에서는, 깊이정보 맵을 비트평면 단위로 분리하

여 비트평면 간 적응적 XOR 연산을 이용한 블록 기반 비트 평면 부호화 방법을 제안한다. 또한, 이러한 비트평면 부호화 방법과 DCT 기반 동영상 부호화 방법을 적응적으로 선택하여 부호화한다. 제안하는 방법의 부호화 구조도는 그림 7과 같다.

그림 7의 부호화 구조에서는 입력된 깊이정보 맵을 블록 단위로 분리하여 부호화를 수행한다. 입력된 깊이정보 맵 블록은 DCT 기반 부호화(그림 7(A)) 모드 혹은 비트평면 부호화(그림 7(B)) 모드 중에 H.264/AVC에서 사용 중인 유효-왜곡 최적화 방법을 이용하여 부호화 효율이 높은 모드로 선택된다. 그리고 부호화된 결과가 부호화 모드 정보와 함께 비트스트림으로 출력된다. 제안된 방법은 DCT 기반 부호화 방법(그림 7(A))과 비트평면 부호화 방법(그림 7(B))을 모두 수행한 후 부호화 효율이 높은 모드를 선택하므로 부호화되는 시간이 기존 방법보다 증가된다는 단점이 존재한다. 하지만 복호화되는 시간은 크게 증가하지 않는다.

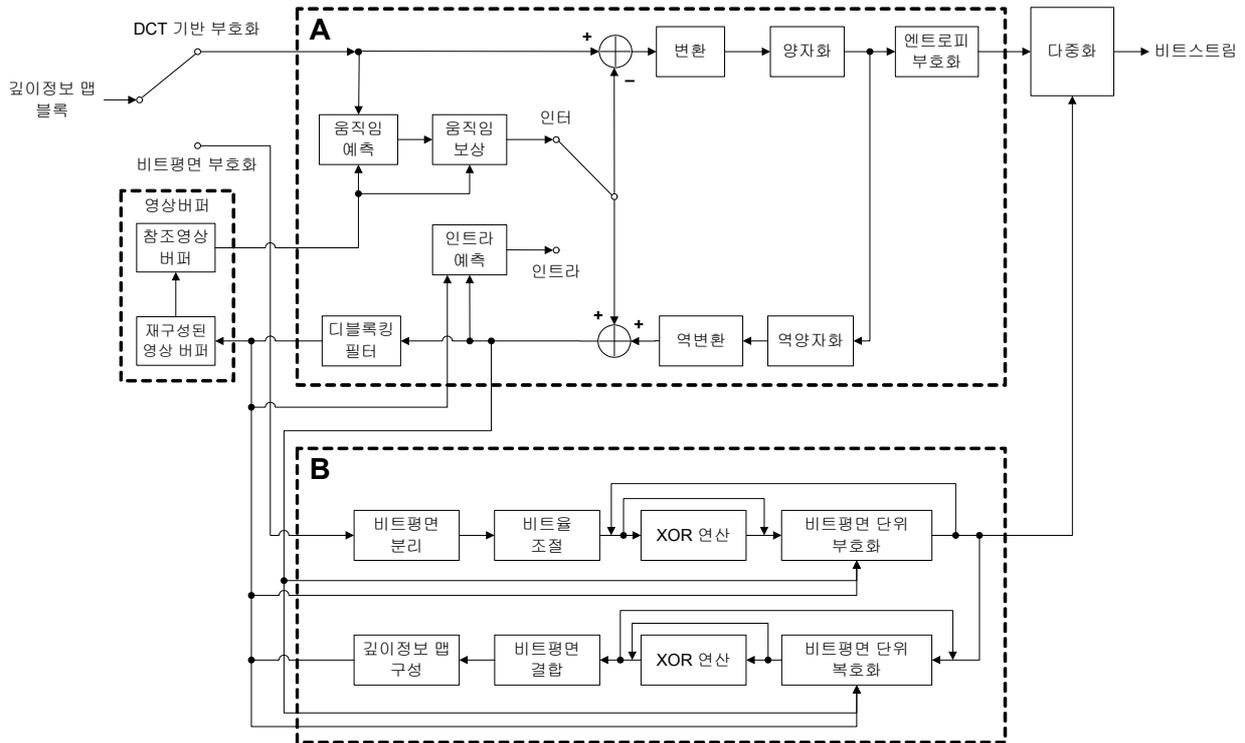


그림 7. 제안하는 방법의 부호화 블록도  
Fig. 7. Encoder block diagram of the proposed method

그림 7(A)에서 DCT 기반 부호화는 H.264/AVC 부호화 방법을 나타내며, 가로 세로 16x16화소 크기의 블록 단위로 입력되어 부호화가 수행된다. 부호화는 입력된 현재 블록에 대한 예측 블록을 생성한 후, 현재 블록과 예측 블록 간의 차분 부호화로 수행된다. 이때 예측 블록의 생성은 화면 내 상관성을 이용한 인트라(Intra) 방법과 화면 간 상관성을 이용한 인터(Inter) 방법을 통해 생성된다. DCT 기반 부호화의 자세한 설명은 다음과 같다.

- (1) 예측 블록은 현재 블록을 인트라로 부호화할지 혹은 인터로 부호화할지에 따라 다르게 수행된다. 인트라 방법으로 부호화될 경우에는 그림 7(A)의 내부 스위치가 인트라로 전환되고, 인트라 예측 과정에서 현재 블록의 이미 부호화된 주변 화소 값들을 이용하여 현재 블록에 대한 예측 블록을 생성한다. 인터 방법으로 부호화될 경우에는 그림 7(A)의 내부 스위치가 인터로 전환되고, 움직임 예측 과정과 움직임 보상 과정을 통해 현재 블록에 대한 예측 블록을 생성한다. 움직임 예측 과정에서는 참조 영상 버퍼(시간적으로 이전에 부호화된 후 복호화된 영상이 저장되어 있는 버퍼)에 저장되어 있는 참조 영상에서 현재 입력된 블록과 가장 비슷한 영역을 찾아 움직임 벡터(Motion Vector)를 구한다. 움직임 보상 과정에서는 찾은 움직임 벡터를 이용하여 참조 영상 버퍼에 저장되어 있는 참조 영상에서 예측 블록을 가져온다.
- (2) 다음으로, 생성된 예측 블록과 현재 입력된 블록과의 차분을 구하여 잔여(오차) 블록(Residual Block)을 생성한 후, 이 잔여 블록에 대한 부호화를 수행한다. 잔여 블록에 대한 부호화는 변환, 양자화, 재배열, 엔트로피 부호화의 순서로 수행되며, 상세한 부호화 과정은 다음과 같다.
  - (2-1) 변환 과정에서는 기본적으로 입력된 잔여 블록에 DCT 변환을 수행하여 변환계수(Transform Coefficient)를 출력한다. 이러한 변환은 공간 영역(Spatial Domain)의 화소 성분을 주파수 영역(Frequency Domain)으로 변환시킴으로써, 저주파 성분을 한 곳에 집중시킬 수 있고 그 결과 부호화를 효과적으로 수행할 수 있다는 장점을 갖는다.
  - (2-2) 양자화 과정에서는 원하는 비트율을 얻기 위해 입력된 변환계수를 양자화 파라미터에 따라 미리 정해놓은 값들로 나누는 연산을 수행하여 양자화된 계수(Quantized Coefficient)를 출력한다.

- (2-3) 재배열 과정에서는 양자화된 계수를 DC계수를 시작으로 저주파 성분부터 고주파 성분까지 지그재그(Zigzag) 순서로 읽어서 일련의 심볼로 구성한다.
- (2-4) 엔트로피 부호화 과정에서는 입력된 일련의 심볼을 확률 분포에 따른 엔트로피(Entropy) 부호화를 수행하여 비트스트림을 출력한다. 이때 문맥기반 적응적 가변 길이 부호화(CAVLC; Context-adaptive Variable Length Coding) 혹은 문맥기반 적응적 이진 산술 부호화(CABAC; Context-based Adaptive Binary Arithmetic Coding<sup>[15]</sup>)를 사용하여 부호화한다.
- (3) DCT 기반 부호화가 수행된 결과는 다중화 과정으로 입력되어 통합된 비트스트림으로 출력된다.
- (4) 일반적인 동영상 부호화는 영상 간(Inter-frame) 예측 방법을 이용하여 부호화를 수행한다. 그렇기 때문에 현재 부호화된 영상을 이후에 입력된 영상의 참조 영상으로 사용하고 또한 부호화 및 복호화에서 예측기의 동일성을 유지시키기 위해 현재 부호화된 영상을 복호화하여 저장할 필요가 있다. 따라서 양자화된 계수를 역양자화 과정과 역변환 과정을 수행하여 잔여 블록을 생성하고, 생성된 잔여 블록과 예측 영상을 더하여 복원된 블록을 생성한다. 복원된 블록은 인트라 예측 혹은 비트평면 단위 부호화/복호화 과정에서 참조 블록으로 사용된다. 마지막으로, 복원된 블록은 디블록킹 필터를 통해 부호화 과정에서 발생한 블록킹 현상(Blocking Artifact)을 제거한 다음에 재구성된 영상 버퍼(현재 부호화하고 있는 영상이 저장되어 있는 버퍼)에 저장한다.
- (5) 블록 단위로 현재 영상에 대한 부호화가 모두 완료되면, 재구성된 영상 버퍼에 있는 재구성된 영상을 참조 영상 버퍼로 이동한다.

그림 7(B)에서 비트평면 부호화는 적응적 XOR 연산을 이용한 비트평면 단위 부호화 방법을 나타낸다. 가로 세로 16x16화소 크기의 블록 단위로 입력되어 화면 내 부호화 방법만 사용하여 부호화된다. 그림 7(B)에 도시한 비트평면 부호화는 비트평면 분리, 비트율 조절, 적응적 XOR 연산, 비트평면 단위 부호화 순서로 진행되며, 자세한 설명은 다음과 같다.

- (1) 비트평면 분리 과정에서는 비트평면 단위 부호화를 수행하기 위해서 N-비트로 표현되는 깊이정보 맵 블록을 N개의 비트평면 블록으로 분리한다.
- (2) 비트율 조절 과정은 원하는 비트율을 얻기 위해 분리된 비트평면들 중에서 몇 개의 비트평면을 부호화할지를 결정한다. 그리고 하위 비트평면보다 상위 비트평면의 중요도가 높으므로, MSB 비트평면부터 LSB 비트평면 순으로 부호화할 비트평면의 수가 결정된다. 여기서 부호화할 비트평면의 수는 H.264/AVC의 율-왜곡 최적화 방법을 이용하여 가장 적합한 수를 결정하게 된다.
- (3) 분리한 n개의 비트평면 블록은 각각 배타적 논리합 연산과 비트평면 부호화 단계로 MSB 비트평면부터 MSB-1, MSB-2, MSB-3, MSB-4, MSB-5, MSB-6, LSB 비트평면 순으로 반복하여 입력되며, 비트평면 부호화를 수행한 결과로 비트스트림을 출력한다. 이때 비트율 조절 과정에서 결정된 비트평면들만 부호화를 수행한다.
- (3-1) XOR 연산 과정에서는 현재 부호화할 비트평면 블록과 상위 비트평면 블록(이전에 부호화된 비트평면 블록) 간 각각의 화소마다 XOR 연산을 수행한다. 이러한 XOR 연산 과정은 현재 비트평면 블록과 상위 비트평면 블록(이전에 부호화된 비트평면 블록) 간의 상관성에 따라 적응적으로 수행한다. XOR 연산의 수행여부를 결정하는 방법은 그림 8에 도시한 것과 같이 부호화된 비트량을 통해 결정된다. 우선 현재 입력된 비트평면 블록을 비트평면 부호화한 결과의 비트양('A')을 측정한다. 또한, 현재 입력된 비트평면 블록과 이전에 입력된 비트평면 블록 간의 XOR 연산을 수행한 비트평면 블록을 비트평면 부호화한 결과의 비트양('B')을 측정한다. 그리고 측정된 비트양 'A'와 'B'를 비교하여 'A'가 'B'보다 크다면 현재 입력된 비트평면 블록에 XOR 연산을 수행한다. 이때 XOR 연산의 수행여부에 대한 정보는 XOR 수행 플래그(1 비트)를 비트스트림에 포함시켜 전송하며, XOR 수행 플래그가 '1'이면 XOR 연산을 수행하고 '0'이면 XOR 연산을 수행하지 않는다.
- (3-2) 비트평면 단위 부호화 과정에서는 각각의 비트평면을 MPEG-4 Part-2 Visual(ISO/IEC 14496-2)<sup>[11]</sup>의 이진 영상 부호화<sup>[12]</sup>를 이용하여 이진 부호화하여 비트스트림을 출력한다. 비트평면 단위 부호화 과정의 이진 영상 부호화에서는 최종적으로 문맥기반 산술 부호화(CAE; Context-based Arithmetic Encoding)<sup>[13]</sup>를 이용하여 각각의 이진 화소를 부호화한다.

- (4) 비트평면 부호화가 수행된 결과는 다중화 과정으로 입력되어 통합된 비트스트림으로 출력된다.
- (5) 비트평면 단위의 부호화를 수행한 후, 향후에 입력되는 깊이정보 맵의 부호화를 위해 현재 부호화한 블록을 다시 복호화하여 재구성된 블록을 생성하여야 한다. 따라서 비트스트림을 입력 받아 부호화한 비트평면 수(m개: 부호화 시 원하는 비트율을 얻기 위해 실제 부호화된 비트평면의 수)만큼 반복해서 비트평면 단위 복호화 과정이 수행되며, 복호화된 결과로 m개의 비트평면 영상을 출력한다.
- (6) XOR 연산 과정은 복호화된 XOR 수행 여부 정보에 따라 선택적으로 해당 비트평면에 수행된다. XOR 연산에서는 이전에 복호화된 비트평면 블록과 현재 복호화한 비트평면 블록 간 각각의 화소마다 XOR 연산을 수행한다.
- (7) 비트평면 결합 과정에서는 출력된 m개의 비트평면을 결합하여 M-비트로 표현되는 영상을 출력한다.
- (8) 깊이정보 맵 구성 과정에서는 M-비트로 표현되는 영상을 실제 깊이정보 맵의 비트수(N-비트)에 맞게 구성한다. N-비트의 깊이정보 맵을 출력하기 위하여 남아있는 비트평면들(MSB+m+1 비트평면부터 LSB 비트평면까지)을 모두 '0(0)'으로 대입한다.
- (9) 재구성된 깊이정보 맵은 그림 7(A)의 DCT 기반 부호화에서 인트라 예측 과정에 이용되거나 혹은 그림 7(B)의 비트평면 부호화/복호화 과정에서 참조 블록으로 사용되며, 최종적으로 재구성된 영상 버퍼에 저장된다.

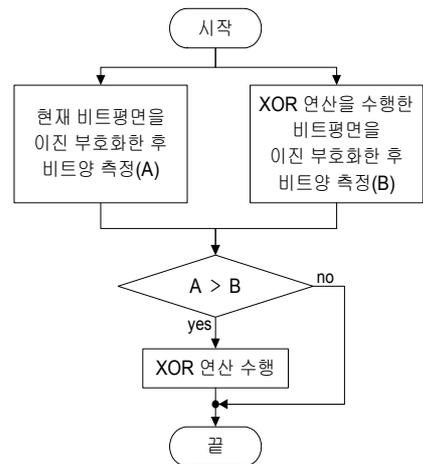


그림 8. XOR 연산의 수행 여부를 결정하는 방법  
 Fig. 8. Method deciding whether or not the execution of XOR operation

s그림 7에서 비트평면 부호화를 수행하는 방법은 국제 동영상 표준인 MPEG-4 Part-2 Visual(ISO/IEC 14496-2)<sup>[11]</sup>의 이진 형상 부호화(binary shape coding)<sup>[12]</sup> 방법을 응용하여 사용하며, 제안하는 비트평면 부호화의 구조도는 그림 9와 같다.

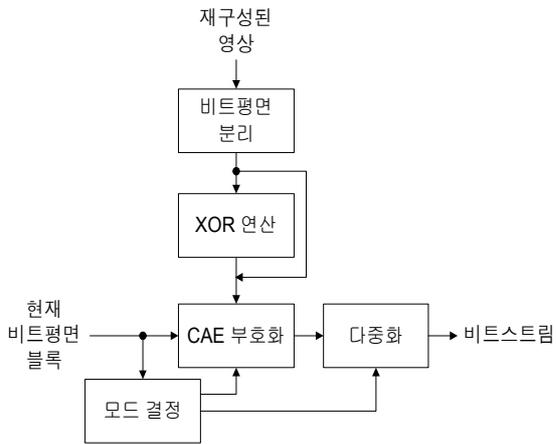


그림 9. 비트평면 부호화 방법의 블록도  
Fig. 9. Block diagram of the bit-plane encoding method

그림 9의 비트평면 부호화 방법은 다음과 같은 순서로 진행된다.

- (1) 모드 결정 과정은 현재 비트평면 블록의 부호화 모드를 결정하는 부분으로써, 현재 비트평면 블록의 모든 이진영상 값이 동일한 값이라면, 부호화 모드 정보만 비트스트림에 포함된다. 이때 현재 비트평면 블록 내의 이진 영상 값이 모두 '255(1)'일 경우에는 'all\_1' 모드로 부호화되며, 모두 '0(0)'일 경우에는 'all\_0' 모드로 부호화된다. 현재 비트평면 블록의 모든 이진영상 값이 동일한 값이 아니라면, 'intraCAE' 모드로 설정되고 부호화 모드 정보와 함께 현재 이진 영상을 CAE 방법<sup>[13]</sup>을 사용하여 부호화한 정보가 비트스트림에 포함된다.
- (2) CAE 부호화 과정<sup>[13]</sup>은 우선 현재 부호화할 화소에 인접한 주변 화소들을 이용하여 컨텍스트를 생성한다. 그리고 생성된 컨텍스트와 맞는 확률을 미리 정의된 확률 표에서 찾은 후, 그 확률을 기반으로 이진 산술 부호화(Binary Arithmetic Coding)를 수행한다.
- (3) 다중화에서는 현재 비트평면 블록의 부호화 모드, XOR 연산의 수행 여부 정보, CAE 부호화한 결과를 하나의 비트스트림으로 생성한다.

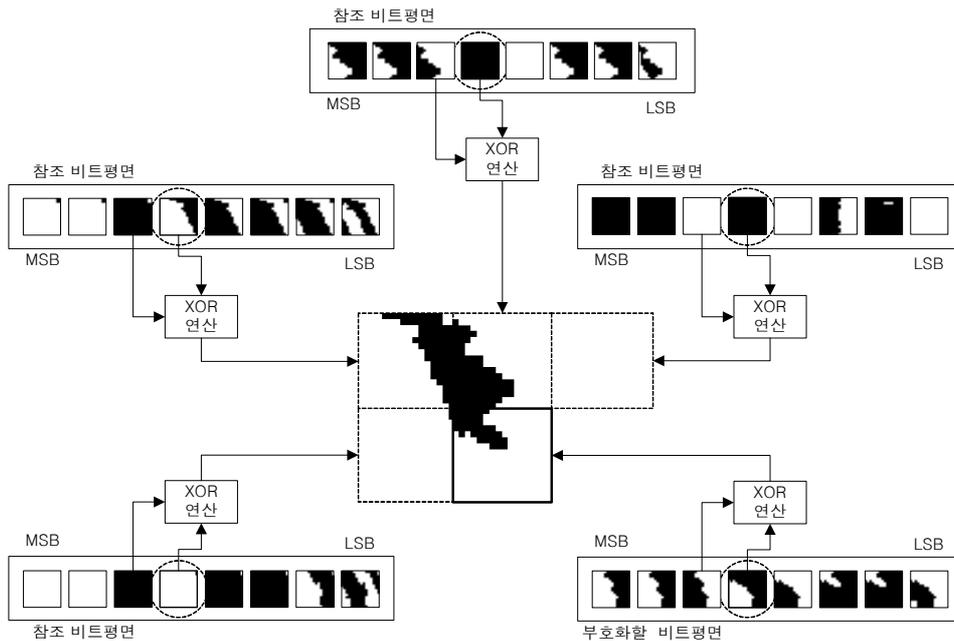


그림 10. 참조 비트평면의 구성 방법  
Fig. 10. Construction method of the reference bit-plane

그림 9에서 입력된 현재 비트평면 블록은 현재 부호화하려고 하는 깊이정보 맵 블록에서 부호화를 수행할 비트평면 블록을 의미한다. 재구성된 영상은 DCT 기반 부호화(그림 7(A))를 통해 복호화되어 디블록킹 필터를 거치지 않은 영상이거나 혹은 비트평면 부호화(그림 7(B))를 통해 복호화된 영상을 의미한다. 입력된 재구성된 영상은 비트평면 분리 과정을 거쳐 각각의 비트평면으로 분리된다. 분리된 비트평면 영상 중에서 현재 부호화할 비트평면 블록과 동일한 비트평면 영상은 현재 부호화할 비트평면 블록의 XOR 수행 여부에 따라 XOR 연산 과정이 수행된다.

예를 들어 그림 10은 현재 부호화할 비트평면(MSB-3) 블록에 XOR 연산을 수행하여 CAE 부호화 할 경우, 참조 비트평면의 구성 방법을 도시한 그림이다. MSB-3 비트평면 블록에 CAE 부호화를 수행하기 위해서는 MSB-3 비트평면 블록과 인접한 비트평면 블록의 화소 정보가 필요하다. 따라서 MSB-3 비트평면 블록과 인접한 참조 비트평면을 구성하기 위해서, 주변 깊이정보 맵 블록들을 비트평면 분리하고 분리된 비트평면 블록들 중에서 MSB-3 비트평면 블록을 이용하여 참조 비트평면을 구성하여야 한다. 이때 현재 부호화할 MSB-3 비트평면 블록이 XOR 연산을 수행하였기 때문에, 참조 비트평면들도 XOR 연산을 수행하여야 한다.

그림 7에서 비트평면 복호화를 수행하는 방법은 국제 동영상 표준인 MPEG-4 Part-2 Visual(ISO/IEC 14496-2)<sup>[11]</sup>의 이진 형상 부호화(binary shape coding)<sup>[12]</sup> 방법을 응용하여 사용하며, 제안하는 비트평면 복호화의 구조도는 그림 11과 같다.

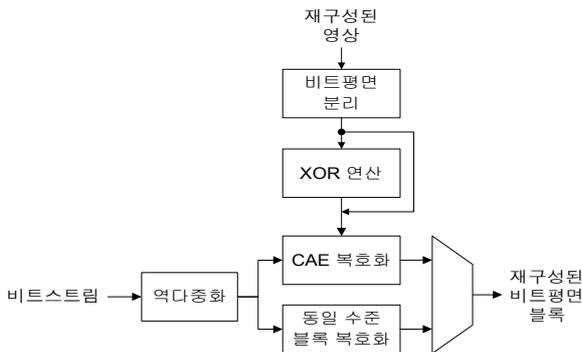


그림 11. 비트평면 복호화 방법의 복호화 블록도  
Fig. 11. Decoder block diagram of the bit-plane decoding method

그림 11의 복호화 블록도에서는 비트스트림을 입력 받아서 비트평면 복호화를 수행하여 재구성된 비트평면 블록을 출력하며, 다음과 같은 순서로 진행된다.

- (1) 역다중화 과정에서는 비트스트림을 입력받아 현재 비트평면 블록의 부호화 모드, XOR 연산의 수행 여부 정보, 그리고 CAE 복호화 과정에서 사용될 비트스트림을 출력한다.
- (2) 만약 복호화된 비트평면 블록의 부호화 모드가 'all\_0' 혹은 'all\_1'이라면, 동일 수준 블록 복호화 과정이 수행이 된다. 만약 비트평면 블록의 부호화 모드가 'intra-CAE'라면, CAE 복호화<sup>[13]</sup>가 수행된다.
- (3) 동일 수준 블록 복호화 과정에서는 비트평면 블록의 부호화 모드가 'all\_0'라면, 블록 내의 모든 이진영상 값을 '0(0)'으로 설정하여 재구성된 비트평면 블록을 출력한다. 만약 비트평면 블록의 부호화 모드가 'all\_1'이라면, 블록 내의 모든 이진영상 값을 '255(1)'로 설정하여 재구성된 비트평면 블록을 출력한다.
- (3) CAE 복호화 과정에서는 현재 비트평면 블록에서 복호화를 수행할 각 화소에 인접한 주변 화소들을 이용하여 콘텍스트를 생성한다. 생성된 콘텍스트와 맞는 확률을 미리 정의된 확률 표에서 찾은 후, 그 확률을 기반으로 이진산술 복호화를 수행하여 재구성된 비트평면 블록을 출력한다. 콘텍스트 생성에 사용되는 주변 화소들은 재구성된 영상 버퍼에 저장되어 있는 재구성된 영상의 비트평면을 이용한다. 이때 복호화할 현재 비트평면 블록의 XOR 연산 수행 정보에 따라 재구성된 영상의 해당 참조 비트평면 블록에 XOR 연산을 수행하여 콘텍스트를 생성하는데 사용된다.

본 논문에서 제안하는 깊이정보 맵 부호화 방법의 복호화 구조도는 그림 12와 같다.

그림 12의 역다중화에서는 비트스트림을 입력받아 부호화 모드 정보를 복호화하고 해당 부호화 모드에 따라 비트스트림을 분배한다. 비트평면 복호화 모드가 선택되었을 경우에는 그림 12(C)가 수행되며, DCT 기반 복호화 모드가 선택되었을 경우에는 그림 12(D)가 수행되어 재구성된 깊이정보 맵 블록을 출력한다.

그림 12(C)는 적응적 XOR을 이용한 비트평면 단위 복호화 방법의 구조도를 나타낸다. 비트평면 복호화 과정에서

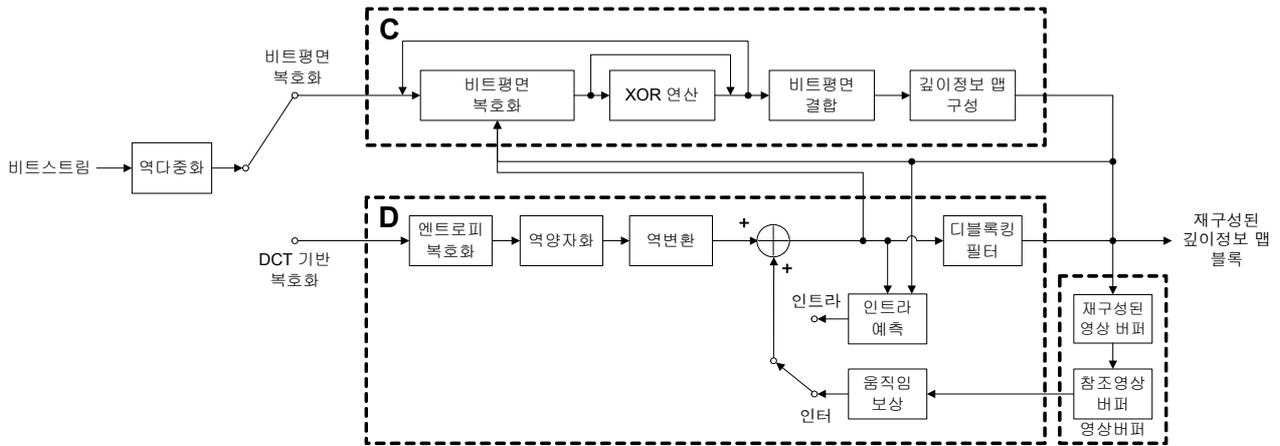


그림 12. 제안하는 방법의 복호화 블록도  
 Fig. 12. Decoder block diagram of the proposed method

는 비트스트림을 입력받아 부호화된 비트평면 수( $m$  개: 부호화 시 원하는 비트율을 얻기 위해 실제 부호화된 비트평면의 수)만큼 반복해서 비트평면 복호화가 수행된다. 입력된 비트스트림을 각 비트평면 별로 복호화하여  $m$ 개의 비트평면들을 출력한다. 이때 비트평면 복호화를 수행하는 방법은 그림 11의 비트평면 복호화 방법을 이용한다. 복호화된  $m$ 개의 비트평면들은 XOR 연산 과정에서 각 비트평면의 XOR 수행 여부 정보에 따라 이전에 복호화된 상위 비트평면 블록과 현재 복호화한 비트평면 블록 간 각각의 화소마다 XOR 연산을 수행한다. 그리고 비트평면 결합 과정에서 출력된 각각의 비트평면 영상을 결합하여  $M$ -비트로 표현되는 영상을 출력한다. 마지막으로 깊이정보 맵 구성 과정에서는  $M$ -비트로 표현되는 영상을 실제 깊이정보 맵의 비트수( $N$ -비트)에 맞게 구성하기 위하여 남아있는 비트평면들(MSB+ $m$ +1 비트평면부터 LSB 비트평면까지)을 모두 '0(0)'으로 대입하여  $N$ -비트의 복원된 깊이정보 맵을 출력한다. 복원된 깊이정보 맵은 그림 12(D)의 인트라 예측 과정 혹은 그림 12(C)의 비트평면 복호화 과정에서 참조블록으로 사용된다. 최종적으로 재구성된 영상 버퍼에 저장되거나 재구성된 깊이정보 맵으로 출력된다.

그림 12(D)는 DCT 기반 복호화 방법(H.264/AVC)의 구조도를 나타내며, 복호화 과정의 주요 흐름은 먼저 예측 블록을 생성한 후, 입력 받은 비트스트림을 복호화한 결과 블

록과 예측 블록을 더하여 재구성된 깊이정보 맵 블록을 생성하는 것이다. 예측 블록은 현재 블록이 인트라로 부호화되었는지 혹은 인터로 부호화되었는지에 따라 다르게 복호화된다. 먼저 인트라 방법으로 복호화할 경우에는 인트라 예측 과정에서 현재 블록의 이미 부호화된 주변 화소값을 이용하여 공간적 예측을 수행하여 예측 블록을 생성한다. 인터 모드일 경우에는 복호화된 움직임 벡터 정보를 이용하여 참조 영상 버퍼에 저장되어 있는 참조 영상에서 해당 영역을 찾아 움직임 보상을 수행함으로써 예측 블록을 생성한다. 그리고 엔트로피 복호화 과정에서 입력된 비트스트림을 화률 분포에 따라 엔트로피 복호화하여 양자화된 계수를 출력한다. 출력된 양자화된 계수에 역양자화와 역변환을 수행한 후 예측 영상과 더하여 복원된 블록을 생성한다. 복원된 블록은 비트평면 복호화 과정에서 참조로 사용되거나, 디블록킹 필터를 통해 블록킹 현상을 제거한 후 재구성된 영상 버퍼에 저장되거나 재구성된 깊이정보 맵으로 출력된다.

### V. 제안하는 방법의 성능 평가

본 논문에서 제안하는 방법의 우수성을 검증하기 위해 H.264/AVC의 참조 소프트웨어인 JM(Joint Model) 13.2<sup>[14]</sup>



그림 13. 실제 영상과 깊이정보 맵 영상  
 Fig. 13. Real image and depth-map image

에 실제 구현하여 DCT 기반 부호화 방법(H.264/AVC)과 적응적 블록기반 깊이정보 맵 부호화 방법<sup>[7]</sup> 그리고 제안하는 방법의 비교를 수행하였고, 실험 영상은 그림 13과 같다.

실험 조건은 표 1에 따라 수행하였으며, I-P-P-P 예측구조에서는 CAVLC와 CAE를 사용하여 부호화하였으며, Hierarchical B 예측구조에서는 CABAC(Context-based Adaptive Binary Arithmetic Coding)<sup>[15]</sup>를 사용하여 부호화를 수행하였다.

표 1. 실험 조건  
 Table 1. Test Condition

해상도 및 프레임률	Ballet, Breakdancers (1024x768, 15Hz), Champagne Tower (1280x960, 30Hz)
프레임수	100 Frames
영상포맷	YUV 4:0:0
양자화 파라미터	22, 27, 32, 37
예측구조	I-P-P-P-, Hierarchical B
엔트로피 부호화 방법	CAVLC, CABAC, CAE

그림 14와 그림 15 그리고 그림 16은 DCT 기반 부호화

방법(H.264/AVC)과 적응적 블록 기반 깊이정보 맵 부호화 방법<sup>[7]</sup>, 그리고 본 논문에서 제안하는 부호화 방법으로 깊이정보 맵을 부호화한 결과를 비트율 대비 PSNR(Peak Signal-to-Noise Ratio)의 RD(Rate-distortion)-Curve로 나타낸 것이다. 표 2는 각 부호화 방법에 대한 부호화 효율을 나타낸다. 그림 14와 그림 15, 그림 16, 그리고 표 2를 통해 확인할 수 있듯이, 제안하는 방법의 결과가 기존 방법보다 부호화 효율이 월등하게 향상된 것을 확인할 수 있다. 특히 고비트율에서의 부호화 효율 향상이 월등하다는 것을 확인할 수 있다.

“Breakdancers” 영상과 “Ballet” 영상 그리고 “Champagne\_tower” 영상의 깊이정보 맵에서 평균적인 PSNR 향상을 나타내는 BD-PSNR 방법<sup>[16]</sup>을 사용하여 성능을 비교하였을 때, 제안하는 방법이 DCT 기반 부호화 방법보다 I-P-P-P 예측구조에서 1.3 dB 향상되었으며, Hierarchical B 예측구조에서 1.0 dB 향상되었다. 또한 제안하는 방법이 적응적 블록 기반 깊이정보 맵 부호화 방법보다 I-P-P-P 예측구조에서 0.8 dB 향상되었으며, Hierarchical B 예측구조에서 0.6 dB 향상되었다. 그리고 평균적인 bit-rate 감소량을 나타내는 BD-bitrate 방법<sup>[16]</sup>을 사용하여 성능을 비교하였을

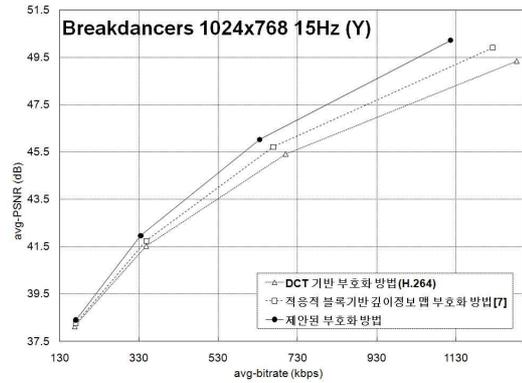
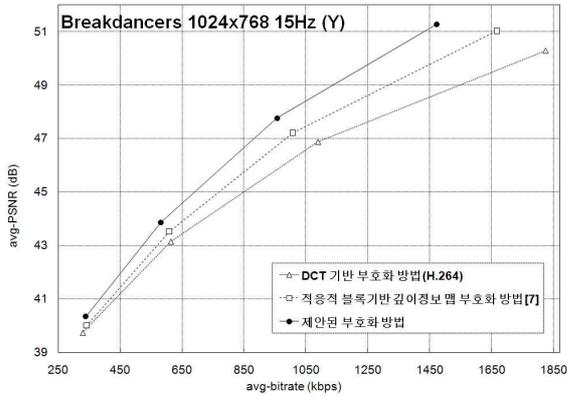


그림 14. Breakdancers 영상의 깊이정보 맵(1번 시점) 실험 결과 그래프: 좌(CAVLC), 우(CABAC)  
 Fig. 14. Comparison of PSNR results in depth-map(view #1) of Breakdancers sequence: left(CAVLC), right(CABAC)

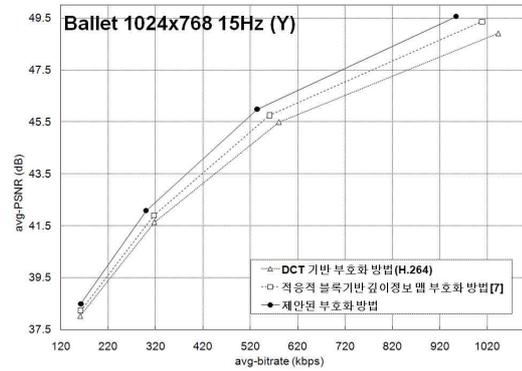
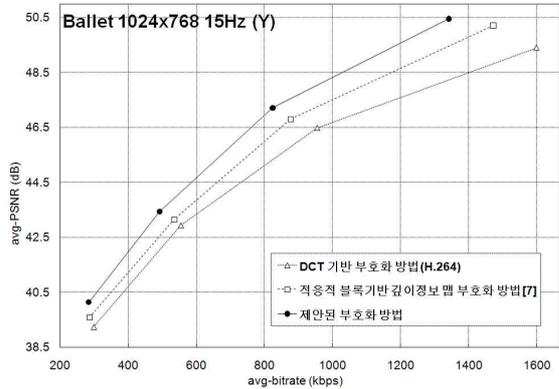


그림 15. Ballet 영상의 깊이정보 맵(1번 시점) 실험 결과 그래프: 좌(CAVLC), 우(CABAC)  
 Fig. 15. Comparison of PSNR results in depth-map(view #1) of Ballet sequence: left(CAVLC), right(CABAC)

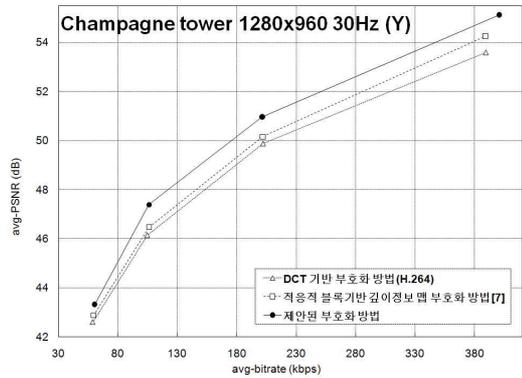
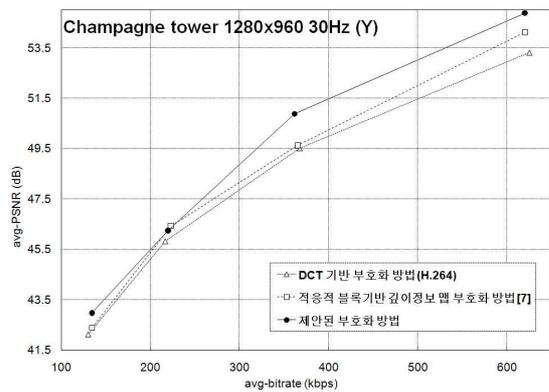


그림 16. Champagne\_tower 영상의 깊이정보 맵(39번 시점) 실험 결과 그래프: 좌(CAVLC), 우(CABAC)  
 Fig. 16. Comparison of the PSNR results in the depth-map(view #39) of Champagne\_tower sequence: left(CAVLC), right(CABAC)

표 2. 깊이정보 맵의 부호화 효율 비교

Table 2. Coding efficiency comparison of the depth-map coding methods

실험 영상	CAVLC, I-P-P-P				CABAC, Hierarchical B			
	제안방법과 DCT 기반 부호화 방법 (H.264/AVC) 간의 부호화 효율 비교		제안방법과 적응적 블록 기반 깊이정보 맵 부호화 방법 <sup>[7]</sup> 간의 부호화 효율 비교		제안방법과 DCT 기반 부호화 방법 (H.264/AVC) 간의 부호화 효율 비교		제안방법과 적응적 블록 기반 깊이정보 맵 부호화 방법 <sup>[7]</sup> 간의 부호화 효율 비교	
	BD-PSNR (dB)	BD-rate (%)	BD-PSNR (dB)	BD-rate (%)	BD-PSNR (dB)	BD-rate (%)	BD-PSNR (dB)	BD-rate (%)
ballet	1.5	-20.8	0.8	-11.7	0.9	-13.3	0.5	-7.7
breakdancers	1.3	-18.6	0.8	-10.3	0.9	-14.5	0.5	-8.5
champagne tower	1.0	-11.8	0.7	-8.5	1.1	-16.9	0.8	-12.2
Average	1.3	-17.1	0.8	-10.2	1.0	-14.9	0.6	-9.5

때, 제안하는 방법이 DCT 기반 부호화 방법보다 I-P-P-P 예측구조에서 17.1 %의 비트량이 감소하였으며, Hierarchical B 예측구조에서 14.9 %의 비트량이 감소하였다. 또한 제안하는 방법이 적응적 블록 기반 깊이정보 맵 부호화 방법보다 I-P-P-P 예측구조에서 10.2 %의 비트량이 감소하였으며, Hierarchical B 예측구조에서 9.5 %의 비트량이 감소하였다.

표 3은 부호화하지 않은 원본 깊이정보 맵을 이용하여

생성한 가상시점 영상의 PSNR 결과를 보여준다. 또한 표 3은 DCT 기반 부호화 방법과 적응적 블록 기반 깊이정보 맵 부호화 방법, 그리고 본 논문에서 제안하는 방법으로 압축하고 복원한 깊이정보 맵을 이용하여 생성한 가상시점 영상의 PSNR 결과를 보여준다. 가상시점 영상의 PSNR 비교 결과, 원본 깊이정보 맵을 이용하여 생성한 가상시점 영상의 PSNR 결과와 각 부호화 방법 간의 객관적인 화질이 1 dB 내의 정도로 차이가 크지 않다는 것을 알 수 있다.

표 3. 가상 영상들의 PSNR 결과 비교

Table 3. Comparison of the PSNR from the synthesized images.

	Ballet			Breakdancers			Champagne tower		
	PSNR (dB)			PSNR (dB)			PSNR (dB)		
	29.98			29.76			32.06		
	H.264/AVC	적응적 블록 기반 깊이정보 맵 부호화 방법 <sup>[7]</sup>	제안하는 방법	H.264/AVC	적응적 블록 기반 깊이정보 맵 부호화 방법 <sup>[7]</sup>	제안하는 방법	H.264/AVC	적응적 블록 기반 깊이정보 맵 부호화 방법 <sup>[7]</sup>	제안하는 방법
	PSNR (dB)	PSNR (dB)	PSNR (dB)	PSNR (dB)	PSNR (dB)	PSNR (dB)	PSNR (dB)	PSNR (dB)	PSNR (dB)
QP	CAVLC, I-P-P-P								
37	29.27	29.23	29.18	28.94	28.88	28.87	32.05	32.07	32.17
32	29.57	29.54	29.54	29.03	29.00	29.00	32.14	32.23	32.27
27	29.74	29.76	29.78	29.08	29.05	29.04	32.18	32.19	32.19
22	29.82	29.84	29.86	29.12	29.12	29.12	32.11	32.15	32.11
QP	CABAC, Hierarchical B								
37	29.38	29.53	29.30	28.89	28.88	28.86	32.23	32.26	32.24
32	29.58	29.66	29.64	29.03	29.03	28.99	32.28	32.34	32.37
27	29.76	29.78	29.79	29.09	29.10	29.06	32.28	32.28	32.28
22	29.81	29.87	29.86	29.12	29.14	29.11	32.24	32.21	32.16

일반적으로 가상시점 영상은 기하학적인 계산식과 카메라 위치 관련 정보를 바탕으로 깊이정보 맵을 이용한 3차원 워핑(warping) 방법을 이용하여 가상시점의 위치를 추정해서 생성한다. 하지만 실제 카메라에 대한 위치 정보에는 오차가 포함될 수 있기 때문에 이것을 이용해서 생성한 가상시점의 위치가 어긋날 수 있다. 더구나 “Ballet”과 “Break-dancers” 의 다시점 영상의 경우, 다시점 영상 촬영 시 카메라 배치가 원호(arc) 형태였기 때문에 카메라 위치 정보에 대한 오차는 더욱 클 수 있다.

예를 들어, 그림 17의 가상시점 영상(그림 17(b))은 MPEG의 3차원 비디오 부호화에서 개발된 가상시점 합성 참조 소프트웨어(VSRS, View Synthesis Reference Software)<sup>[5,6]</sup>를 사용하여 생성하였다. 생성된 영상은 2번 시점(2번 카메라)에 대한 가상시점 영상으로써, 2번 시점에 인접한 시점들(1번 시점과 3번 시점의 실제 원본 영상)과 그것의 깊이정보 맵(1번 시점과 3번 시점의 원본 깊이정보 맵)을 이용하여 생성하였다. 또한 그림 17(c)는 실제 영상(그림 17(a))과 가상시점 영상(그림 17(b)) 간의 오차 영상이다. 그림 17(a)와 그림 17(b)를 자세히 보면, 영상 내에서는 동일한 위치이지만 객체마다 약간의 위치 차이가 존재한다는 것을 오차 영상(그림 17(c))를 통해 확인할 수 있다. 이와 같은 오차는 가상시점 영상을 생성할 때 객체마다 가상시점의 위치를 정확히 구해내지 못하기 때문이다. 그래서 원본 깊이정보 맵을 이용하여 생성한 가상시점 영상의 객관적인 화질(PSNR)이 높게 나오지 않게 된 것이다.

또한 표 3에서 각 부호화 방법으로 생성한 가상시점 영상 간의 객관적인 화질이 거의 비슷하다는 것을 확인할 수 있다. 하지만 각 부호화 방법으로 복원된 깊이정보 맵을 이용하여 생성한 가상시점 영상의 주관적인 화질을 비교해 보면, 제안하는 방법이 DCT 기반 부호화 방법보다 우수하며, 적응적 블록 기반 깊이정보 맵 부호화 방법과 비슷하다는 것을 그림 18을 통해 확인할 수 있다. 그림 18의 가상시점 영상은 2번 시점에 대한 가상시점 영상으로써, 2번 시점에 인접한 시점들(1번 시점과 3번 시점의 실제 원본 영상)과 그것의 깊이정보 맵(1번 시점과 3번 시점을 각각의 부호화 방법으로 부호화한 후 복원한 깊이정보 맵)을 이용하여 생성하였다.

DCT 기반 부호화 방법으로 부호화한 깊이정보 맵을 이용하여 생성한 가상시점 영상들(그림 18(c), 그림 18(f), 그림 18(i))은 깊이정보 맵 부호화시 객체 경계 부분이 뭉개져서 객체와 배경을 명확히 구분하지 못한다. 그렇기 때문에 가상시점 영상의 객체 경계 부분에서 왜곡이 발생된 것을 확인할 수 있다. 반면 제안된 방법으로 생성한 가상시점 영상들(그림 18(e), 그림 18(h), 그림 18(k))은 깊이정보 맵 부호화시 객체 경계 부분을 비트평면 단위로 부호화함으로써 객체 경계 부분을 명확히 구분한다. 따라서 객체 경계 부분에서의 왜곡이 DCT 기반 부호화 방법보다 덜 발생된다는 것을 확인할 수 있다.

또한 제안된 방법으로 부호화한 깊이정보 맵을 이용하여 생성한 가상시점 영상들(그림 18(e), 그림 18(h), 그림 18



그림 17. 'Breakdancers' 영상에서 2번 시점의 실제 영상과 가상시점 영상 및 두 영상의 오차 영상 비교

Fig. 17. Comparison of the real image and the synthesized image and the subtracted image of the two images in the 'Breakdancers' sequence(view #2).



(a) 저비트율에서 DCT 기반 부호화 방법



(b) 저비트율에서 적응적 블록기반 깊이정보 맵 부호화 방법



(c) 저비트율에서 제안하는 방법



(d) 중비트율에서 DCT 기반 부호화 방법



(e) 중비트율에서 적응적 블록기반 깊이정보 맵 부호화 방법



(f) 중비트율에서 제안하는 방법



(g) 고비트율에서 DCT 기반 부호화 방법



(h) 고비트율에서 적응적 블록기반 깊이정보 맵 부호화 방법



(i) 고비트율에서 제안하는 방법

그림 18. 'Breakdancers' 영상에서 2번 시점의 가상시점 영상에 대한 주관적 화질 비교

Fig. 18. Comparison of the subjective quality of the synthesized image(view #2) in the 'Breakdancers' sequence

(k)이 적응적 블록 기반 깊이정보 맵 부호화 방법으로 부호화한 깊이정보 맵을 이용하여 생성한 가상시점 영상들(그림 18(d), 그림 18(g), 그림 18(j))과 비교해서 주관적 화질이 감소되지 않는 것을 확인할 수 있다.

## VI. 결론

본 논문에서는 3차원 비디오 부호화 표준의 일부분인 깊이정보 맵의 효율적인 부호화 방법에 대한 연구를 수행하

였다. 깊이정보 맵의 객체 경계 부분에서 비트평면 간 완전일치/반전일치의 특성을 적절하게 이용하기 위하여 비트평면 간 적응적 XOR 연산을 이용한 비트평면 부호화 방법을 제안하였다. 또한 비트평면 단위 이진 영상 부호화 방법과 DCT 기반 동영상 압축 방법의 장점을 적절하게 이용하기 위하여 제안된 블록 단위 비트평면 부호화 방법과 기존의 블록 단위 동영상 부호화 방법을 적응적으로 선택하여 부호화하였다. 실험결과, 제안하는 방법이 기존 방법보다 부호화 효율이 우수함을 확인할 수 있었으며, 저비트율보다는 고비트율에서의 성능개선이 더 우수함을 확인할 수 있었다. 또한 객체 경계 부분에 XOR 연산을 이용한 비트평면 부호화 방법이 적용됨으로써, 주관적 화질 측면에서 제안하는 방법이 DCT기반 부호화 방법보다 우수하였다. 그리고 제안된 방법이 적응적 블록 기반 깊이정보 맵 부호화 방법과 비교하여 주관적 화질이 떨어지지 않는다는 것을 확인할 수 있었다. 앞으로는 저비트율에서의 부호화 효율 개선 및 주관적 화질 향상을 위한 추가 연구가 필요하다.

### 참 고 문 헌

- [1] ISO/IEC JTC1/SC29/WG11, "Draft Report on Experimental Framework for 3D Video Coding," N11478, Geneva, Switzerland, July 2010.
- [2] M. Tanimoto, "Overview of Free Viewpoint Television," *Signal Processing: Image Communication*, Vol. 21, No. 6, pp.454-461, July 2006.
- [3] 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.
- [4] ITU-T Recommendation H.264 and ISO/IEC 14496-10 (MPEG-4 Part 10 AVC), "Advanced Video Coding for Generic Audiovisual Services," Version 1: March 2003, Version 2: May 2004, Version 3: March 2005, Version 4: September 2005, Version 5 and Version 6: June 2006, Version 7: April 2007, Version 8: July 2007.
- [5] Y. Mori, N. Fukushima, T. Yendo, T. Fujii, M. Tanimoto, "View generation with 3D warping using depth information for FTV," *Image Communication*, Vol. 24 No. 1-2, pp. 65-72, January 2009.
- [6] M. Tanimoto, T. Fujii, "View synthesis algorithm in view synthesis reference software 2.0 (VSRS 2.0)," ISO/IEC JTC 1/SC29/WG11 M16090, Lausanne, Switzerland, February 2009.
- [7] K.Y. Kim, G.H. Park, D.Y. Suh, "Adaptive Depth-map Coding for 3D-Video," *IEICE Trans. on INF. & SYST.*, Vol. E93-D, No. 8, pp. 2262-2272, August 2010.
- [8] K.Y. Kim, G.H. Park, D.Y. Suh, "Bitplane-based lossless depth-map coding," *SPIE Optical Engineering*, Vol. 49, No. 6, 067403, June 2010.
- [9] T. Wiegand, H. Schwarz, A. Joch, F. Kossentini, and G.J. Sullivan, "Rate-Constrained Coder Control and Comparison of Video Coding Standards," *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 13, No. 7, pp. 688-703, July 2003.
- [10] J. R. Bitner, G. Ehrlich, and E. M. Reingold, "Efficient generation of the binary reflected gray code and its applications," *Commun. ACM*, Vol. 19, No. 9, pp.517-521, September 1976.
- [11] ISO/IEC 14496-2 (MPEG-4 Visual), "Coding of Audio-Visual Objects - Part 2: Visual," Version 1: April 1999, Version 2: February 2000, Version 3: May 2004.
- [12] N. Brady, and F. Bossen, "Shape compression of moving objects using context-based arithmetic encoding," *Signal Processing: Image Communication*, Vol. 15, No. 7, pp. 601-617, May 2000.
- [13] 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.
- [14] F. Heinrich-Hertz-Institut, "H.264 Reference Software Version JM13.2," <http://iphome.hhi.de/suehring/tml>, May 2008.
- [15] D. Marpe, H. Schwarz, and T. Wiegand, "Context-Based Adaptive Binary Arithmetic Coding in the H.264/AVC Video Compression Standard," *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 13, No. 7, pp. 620-636, July 2003.
- [16] G. Bjontegaard, "Calculation of average PSNR differences between RD-curves," ITU-T SG16 Q.6, VCEG-M33, Texas, USA, April 2001.

---

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