

논문 2009-46SP-1-6

그래픽스 응용을 위한 다시점 텍스처 및 깊이 정보의 근접 무손실 부호화

(Near-lossless Coding of Multiview Texture and Depth Information for Graphics Applications)

윤 승 옥*, 호 요 성**

(Seung-Uk Yoon and Yo-Sung Ho)

요 약

본 논문에서는 복잡한 삼차원 장면을 다시점 텍스처와 깊이맵을 이용해 표현하고 이를 부호화하는 기법을 제안한다. 이를 위해 입력 다시점 데이터를 압축된 텍스처 및 깊이맵쌍 포맷으로 변환하고 제안한 코덱인 X-Codec을 통해 손실을 최소화하면서 부호화 효율을 향상시킨다. 우선 제안한 부호화 시스템은 다시점 색상과 깊이영상을 입력으로 사용하므로 기존의 기하학 기반 렌더링 기법과 달리 렌더링 속도가 장면의 복잡도가 아닌 입력 영상의 해상도에 따라 좌우된다. 또한, 영상을 블록 단위로 나누어 부호화와 복호화를 수행하기 때문에 임의 영역 접근이 용이하고, 그래픽스 프로세싱 유닛을 이용한 실시간 렌더링이 가능하다. 실험결과 제안한 방법은 다시점 데이터를 이용한 다시점 렌더링, 선택적 복호화, 깊이정보를 이용한 삼차원 장면의 조작 등 다양한 기능을 동시에 제공하므로 대용량 데이터로부터 삼차원 장면을 탐색하는 그래픽스 응용에 유용하다.

Abstract

This paper introduces representation and coding schemes of multiview texture and depth data for complex three-dimensional scenes. We represent input color and depth images using compressed texture and depth map pairs. The proposed X-codec encodes them further to increase compression ratio in a near-lossless way. Our system resolves two problems. First, rendering time and output visual quality depend on input image resolutions rather than scene complexity since a depth image-based rendering techniques is used. Second, the random access problem of conventional image-based rendering could be effectively solved using our image block-based compression schemes. From experimental results, the proposed approach is useful to graphics applications because it provides multiview rendering, selective decoding, and scene manipulation functionalities.

Keywords: 다시점 데이터 부호화, 삼차원 장면 표현, 영상 기반 렌더링

I. 서 론

최근 가속화되고 있는 컴퓨터 그래픽스 기술의 발달

로 게임, 영화, 가상현실 등의 다양한 멀티미디어 응용 분야에서 사용자가 실사와 합성영상의 차이를 구별하기가 어려워지고 있다. 다각형 메쉬는 이러한 응용에서 삼차원 장면이나 객체를 나타내기 위해 가장 많이 사용되는 표현 중의 하나이다.

하지만 실사와 같은 장면을 다각형 메쉬로 모델링하고 렌더링하는 일은 많은 노력과 시간을 요구한다. 이를 위해서는 장면의 복잡도에 따라 수백에서 일억 개 이상의 다각형 메쉬가 필요하다^[1]. 예를 들면, 그래픽스 기술로 합성한 한 장의 영상이 테라바이트의 용량을 차지하기도 하며, 영화 한 편의 총 데이터양이 페타바이트에

* 학생회원, ** 평생회원, 광주과학기술원 정보통신공학과 (Gwangju Institute of Science and Technology, Department of Information and Communications)

※ 본 연구는 광주과학기술원(GIST) 실감방송연구센터(RBRC)를 통한 정보통신대학 IT연구센터(ITRC)의 지원에 의한 것입니다. 또한, 본 논문에서 사용된 그래픽스 테스트 데이터는 Microsoft Research Asia에서 제공되었습니다.

접수일자: 2008년7월2일, 수정완료일: 2009년1월7일

이르기도 한다^[2]. 심지어 렌더팜을 사용하는 경우에도 한 장면의 렌더링 시간이 많게는 며칠씩 걸리기도 한다.

영상 기반 렌더링(image-based rendering, IBR)은 여러 시점의 이차원 영상을 이용하여 삼차원 공간의 임의 시점을 생성한다. 이러한 접근 방식은 렌더링 시간이 생략하려는 화면의 복잡도와 무관하고, 많은 경우 원하는 장면을 생성하기 위해 복잡한 삼차원 모델링을 할 필요가 없다. 하지만 상호작용이 어렵고 고화질 입력을 사용할 경우 데이터양이 늘어난다. 그리고 화소나 선을 기본으로 표현하므로 임의 영역 접근에 오랜 시간이 걸려 실시간 응용에 적용하기 어렵다.

이러한 문제점을 해결할 수 있는 대안으로 깊이정보를 이용하는 방법들이 제안되고 있다^[4~6]. 한 장의 깊이맵은 2.5차원의 장면 기하(geometry)정보를 제공할 수 있으며, 다시점 깊이맵은 삼차원 장면을 복원하는데 유용하다. 최근 전통적인 스테레오 정합방법 외에 깊이정보를 레이저가 물체에 반사되어 오는 시간차를 측정하는 센서나 구조광(structured light) 등의 하드웨어장치로부터 직접 얻을 수 있는 방법들이 제안되고 있다^[7]. 또한, 카메라의 가격이 떨어지고 컴퓨터 비전 기술이 발전하면서 다시점 깊이정보의 활용가능성이 높아지고 있다. 따라서 다시점 영상과 깊이정보를 사용하는 기법이 전통적인 다각형 메쉬 기반 렌더링과 영상 기반 렌더링 사이를 연결하는 현실적인 해결책이 될 수 있다.

한편, 가상공간을 탐색하는 등의 실시간 응용에서는 입력 데이터의 빠른 로딩과 부분적인 복호화, 실시간 렌더링이 중요한 문제이다. 비록 영상 기반 렌더링이 실사와 같은 렌더링 결과를 얻는 데 유용하지만, 이를 이용한 실시간 렌더링 파이프라인은 거의 개발되지 않았다. 반면 그래픽스 프로세서 기술의 발달로 최근에는 그래픽스 프로세싱 유닛(graphics processing unit, GPU)이 실시간 렌더링에 필수적이다. 하지만 GPU의 입력은 영상이 아닌 삼차원 모델의 기하정보이다. 따라서 다시점 색상 및 깊이정보를 이용하는 경우 GPU로 렌더링을 수행하기 위해 깊이영상에서 기하정보를 복원해야 한다. 이때, 텍스처나 깊이영상을 일반적인 영상 부호화 방법으로 압축하면 GPU에서 곧바로 사용할 수 없고, 적합한 포맷으로 변환시 시간이 오래 걸리는 단점이 있다.

본 논문에서는 전통적인 메쉬 기반 표현과 영상 기반 표현의 단점을 보완하기 위해 깊이영상 기반 표현을 사용한다. 이를 통해 삼차원 기하정보를 복원해 GPU를 이용한 실시간 렌더링에 이용할 수 있다. 동시에 입력

텍스처와 깊이영상을 GPU에서 직접 소모될 수 있는 포맷으로 압축하는 근접 무손실 부호화 기법을 제안한다.

본 논문의 II장에서는 다각형 메쉬 기반 표현, 깊이정보를 이용한 삼차원 장면 표현, 영상을 이용한 실시간 장면 탐색 기법에 대해 설명한다. III장에서는 제안한 텍스처 및 깊이맵 쌍의 표현과 부호화 방법에 대해 설명한다. IV장에서는 제안한 방법에 대한 실험결과를 보여주고, V장에서 본 논문의 결론을 맺는다.

II. 기존의 삼차원 장면 표현 기법

1. 삼차원 객체의 다각형 메쉬 표현

삼차원 객체를 표현하는 방법에는 다각형 메쉬, 꼭지점군(point clouds), 스플라인 곡선(spline curves), 복셀(voxels) 등이 있지만 멀티미디어 응용에서 가장 많이 사용되는 것은 다각형 메쉬이다. 왜냐하면 메쉬에 포함된 기하, 연결성(connectivity), 광학(photometry)정보를 이용해 사용자가 다양한 상호작용을 할 수 있기 때문이다. 하지만 복잡한 모델링과 오랜 시간이 걸리는 렌더링, 서로 다른 세 가지 데이터 성분을 부호화해야 하므로 부호화기의 복잡도가 높은 문제가 있다.

2. 깊이정보를 이용한 삼차원 장면 표현

영상기반 렌더링 기법들은 기하정보의 사용량에 따라 기하정보 없이 순수하게 영상만을 이용하는 경우, 화소간의 연관성을 이용하는 경우, 깊이정보와 같은 확실한 기하정보를 사용하는 경우로 분류할 수 있다^[3].

Kanade^[4]는 51대의 카메라로 구성된 반구형 돔에서 삼차원 표면을 복원했다. 하지만 시점변화에 따른 조명 불일치로 인해 왜곡이 발생했다. Zitnick^[5]은 8대의 카메라로부터 획득한 동영상을 이용해 고화질의 중간 시점 생성 기법을 제안했다. 색상 및 깊이정보를 부호화하고 GPU를 사용해 15fps의 속도로 중간시점을 렌더링하는 방법도 함께 제안했다. 최근 Verlan^[6]은 GPU 기반 다시점 렌더링 기법을 개발해 Zitnick이 사용했던 같은 데이터에 대한 렌더링 속도를 50fps까지 향상시켰다.

3. 영상 기반 가상공간 탐색

사실적인 가상공간은 복잡한 건물, 풍경, 그리고 넓은 지역을 포함하며 이에 수반하는 데이터양도 많기 때문에, 렌더링 화질과 데이터양 사이의 효과적인 절충이 필요하다. 방대한 데이터 중의 일부분을 빠르게 로딩하고

접근하기 위해서는 선택적인 복호화가 필수적이다.

기존의 방법으로 QuickTime VR^[8]은 미리 획득한 파노라마 영상들을 이동하면서 사용자가 장면을 탐색할 수 있다. 제한사항으로는 고정된 장면에 대해서만 적용되었다는 점과 제한된 시야각을 들 수 있다. Shade^[9]는 탐험속도를 높이기 위해 영상을 계층적으로 분할하고 사용자의 이동에 따라 필요한 부분을 빠르게 가져오는 캐싱 기법을 제안했다. Uyttendaele^[10]은 상호작용이 가능한 동영상 탐색 시스템을 제안했다. 하지만 전체적인 장면을 육면체에 매핑하기 때문에 장면전환 시 왜곡이 나타나며, 미리 정해진 경로로만 탐색이 가능하다.

III. 다시점 텍스처 및 깊이의 근접 무손실 부호화

본 논문에서는 삼차원 장면을 다시점 텍스처 및 깊이 맵으로 구성된 데이터로 보고, 이 방대한 데이터를 최소한의 손실로 부호화하는 방법을 제안한다. 제안 방식은 깊이영상 기반 표현의 하나로 영상 블록을 기본 단위로 선택적 부호화가 가능하다. 또한, 다시점 깊이맵에서 삼차원 기하정보를 복원할 수 있어 GPU 기반의 실시간 렌더링이 가능하다. 즉, 영상 기반 렌더링 기법과 메쉬 기반 표현의 장점을 모두 취하는 효과적인 방법이다. 그림 1에 제안한 부호화 시스템의 흐름도를 나타냈다.

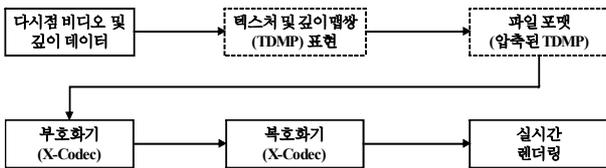


그림 1. 제안한 다시점 데이터 부호화 시스템의 흐름도
Fig. 1. The Overall Flow of the Proposed System.

1. 텍스처 및 깊이맵쌍 표현

삼차원 장면을 효과적으로 표현하기 위해 본 논문에서는 이를 텍스처 및 깊이맵 쌍(texture and depth map pair, TDMP)으로 나타낸다. 그림 2는 제안한 TDMP 표현을 나타낸다. 한 장의 텍스처와 이에 상응하는 깊이맵이 TDMP를 구성하며, 각 TDMP는 알파(A), R, G, B, 깊이(D) 성분 및 카메라 매개변수를 갖는다. 따라서 한 시점의 색상 및 깊이 동영상은 TDMP 시퀀스로, 다시점 데이터는 다중 TDMP 시퀀스로 간주할 수 있다.

그림 2에서 각 성분은 화소당 8비트를 가지며 256 단

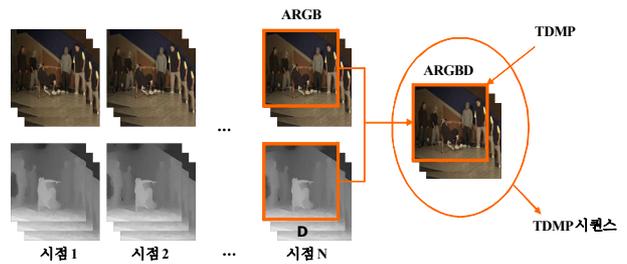


그림 2. 텍스처 및 깊이맵쌍 (TDMP) 표현
Fig. 2. Texture and Depth Map Pair (TDMP).

계의 차이를 표현한다. 색상에는 24비트가 할당되며, 알파와 깊이에는 8비트씩이 할당된다. 비록 256 단계를 표현하는 깊이맵이 모든 응용에 최적인 것은 아니나, Zitnick^[5]은 8비트 깊이맵과 실사영상을 이용해 고화질의 중간 시점을 생성할 수 있음을 검증했다. 그래픽스 영상으로부터 계층적 깊이영상을 생성하는 경우는 20비트의 깊이맵을 사용하기도 한다^[11]. 깊이 단계와 출력 영상의 화질과의 관계는 아직 명확히 규명되지 않아 추가적인 연구가 필요하다. 따라서 실사영상을 입력으로 사용하는 본 논문에서는 8비트 깊이맵을 사용했다.

2. 압축된 텍스처 및 깊이맵 쌍 포맷

삼차원 장면의 효율적인 표현은 효과적인 파일 포맷과 밀접한 연관이 있다. 그래픽스 분야에서는 텍스처와 법선맵을 효과적으로 표현하기 위한 많은 연구를 수행해 왔다. 그 중 DirectX texture compression (DXTC, DXTn)와 ATI 3Dc^[12]가 대표적인 압축 포맷으로 널리 사용되고 있다. 압축하기 전의 텍스처를 처리하려면 많은 양의 메모리와 저장 공간이 요구되므로 실제적인 응용을 위해서는 이의 압축이 필수적이다. 이러한 포맷들에 기반해 본 논문에서는 TDMP의 압축된 포맷

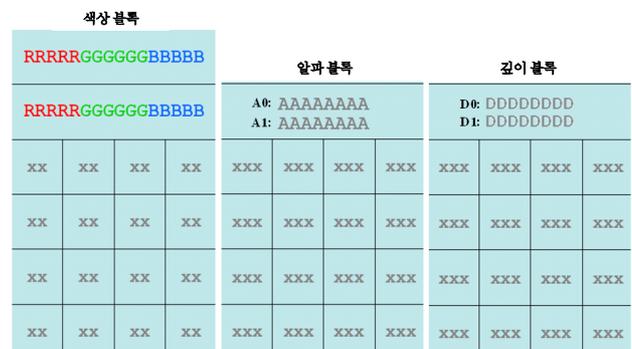


그림 3. 압축된 텍스처 및 깊이맵쌍 (C-TDMP) 포맷
Fig. 3. Compressed TDMP (C-TDMP) Format.

(compressed TDMP, C-TDMP)을 제안한다. 그림 3은 제안한 C-TDMP 파일 포맷을 나타낸다.

C-TDMP는 색상, 알파, 깊이의 세 가지 성분으로 구성된다. 각 성분을 압축하기 위해 우선 두 개의 대표값을 선택한다. 예를 들어, 색상 블록의 경우에 두 개의 16비트 대표 색상 $C0$ 와 $C1$ 을 선택하고, 이를 이용해 나머지 값들을 보간한다. 본 논문에서는 대표값으로 블록의 최소값과 최대값을 사용했다. 보간된 화소값들은 2비트 인덱스를 이용해 지정된다.

압축 과정은 모든 블록에 반복적으로 적용되며, 본 논문에서는 블록의 크기를 4x4로 고정했다. 이는 4x4 크기의 블록에서는 색상, 알파, 깊이값이 차이가 매우 작아 왜곡이 발생해도 이를 확인하기가 어렵기 때문이다. 따라서 16개의 화소를 64비트로 부호화하므로 유효 비트율(effective bitrate)은 화소당 4비트가 된다.

색상 블록의 경우 각 화소의 값은 $C0$ 와 $C1$ 으로부터 보간된 화소값, $color_0, color_1, color_2, color_3$, 중 하나로 선택된다. 이들은 식(1)에 의해 선형 보간된다.

$$\begin{aligned} color_0 &= C0, color_1 = C1, \\ color_2 &= \lfloor (2 \times C0 + C1 + 1)/3 \rfloor, \\ color_3 &= \lfloor (C0 + 2 \times C1 + 1)/3 \rfloor. \end{aligned} \quad (1)$$

깊이맵은 화소당 8비트로 표현되므로, 깊이 블록에 대해서는 두 개의 8비트 대표값이 선택되고 나머지 값들은 3비트 인덱스로 표시된다. 대표값 $D0$ 와 $D1$ 은 블록의 최소값과 최대값이며, 블록의 각 화소값은 $D0$ 와 $D1$ 에 의해 보간된 $depth_0$ 와 $depth_7$ 에서 선택된다. 여섯 개의 중간값은 식(2)에 의해 보간된다. 알파 블록의 부호화도 깊이 블록과 동일한 과정을 거쳐 수행된다.

$$\begin{aligned} depth_0 &= D0, depth_1 = D1, \\ depth_2 &= \lfloor (6 \times D0 + 1 \times D1 + 3)/7 \rfloor, \\ depth_3 &= \lfloor (5 \times D0 + 2 \times D1 + 3)/7 \rfloor, \\ depth_4 &= \lfloor (4 \times D0 + 3 \times D1 + 3)/7 \rfloor, \\ depth_5 &= \lfloor (3 \times D0 + 4 \times D1 + 3)/7 \rfloor, \\ depth_6 &= \lfloor (2 \times D0 + 5 \times D1 + 3)/7 \rfloor, \\ depth_7 &= \lfloor (1 \times D0 + 6 \times D1 + 3)/7 \rfloor. \end{aligned} \quad (2)$$

3. 압축된 텍스처 및 깊이맵쌍의 부호화

일반적으로 입력 영상의 해상도가 클수록 주변장치에서 메모리로 영상이 로딩되는 속도가 GPU에서 데이터

가 소모되는 속도보다 훨씬 느려진다. 하지만 높은 압축률을 갖는 전통적인 영상 압축 기법은 임의 접근이 용이하지 않아 GPU에서 직접 사용되기 힘들다. 예를 들어 JPEG2000으로 압축된 데이터를 GPU에서 직접 소모되며 임의 접근이 용이한 DXTC 포맷으로 바꾸는 데는 많은 시간이 걸려 이를 실시간으로 변환하기가 어렵다.

이에 비해 제안한 C-TDMP 포맷은 DXTC에 기반하므로 임의 접근성이 좋고 GPU에서 직접 소모될 수 있다. 하지만 고정 길이 부호화를 사용하므로 압축률이 낮다. 따라서 본 논문에서는 C-TDMP의 압축률을 높이기 위해 새로운 코덱을 제안해 부호화 효율을 향상시킨다. X-Encoder는 C-TDMP 데이터를 오프라인 상에서 부호화하고, X-Decoder가 압축된 비트열을 복호한다. 영상 블록 단위로 처리되므로 선택적 복호화가 가능하고, 원하는 블록만을 로딩하므로 로딩 시간을 줄일 수 있다.

본 논문에서 목표로 하는 응용인 가상공간에서 삼차원 장면 탐색이나 고해상도 비디오 게임 등에서는 작은 해상도의 영상을 확대하는 것 보다는 왜곡이 적은 큰 해상도의 영상을 요구한다. 즉, 손실 부호화 보다는 무손실 부호화 방법을 사용해 영상을 압축하는 방식이 적합하다. 따라서 본 논문에서는 기존의 손실 압축 기법인 JPEG이나 H.264/AVC 등은 논의에서 제외하고 무손실 압축에 초점을 맞춘다. 전체적으로는 TDMP 압축 시 화소값을 보간함에 따라 약간의 왜곡이 포함되므로 근접 무손실 부호화 기법으로 명명하였다.

가. X-Codec의 프레임워크

그림 4는 제안한 X-Codec의 프레임워크를 나타낸다. C-TDMP에는 색상, 알파, 깊이 성분이 있으므로 성분별로 대표값과 인덱스를 부호화한다.

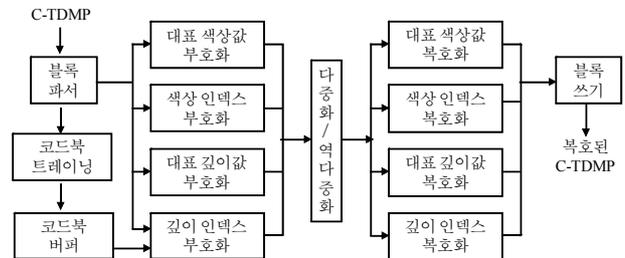


그림 4. X-Codec의 프레임워크

Fig. 4. The Framework of X-Codec.

나. 코드북 트레이닝

각 4x4 인덱스 블록은 16개의 인덱스로 구성되고, 각

인덱스는 코드워드를 나타내므로 이를 16차원 코드워드가 벡터로 생각할 수 있다. 본 논문에서는 전통적인 LBG 벡터 양자화 기법^[13]을 사용하였다. 코드북 트레이닝을 위해 두 대표값이 서로 같지 않은 인덱스 블록을 선택해 이를 트레이닝 표본으로 사용한다.

다. 대표값 부호화

색상 블록에서 두 대표값 $C0$ 와 $C1$ 은 블록내에서 가장 멀리 떨어진 값이다. 블록 크기가 4×4 로 매우 작기 때문에 대부분의 경우 두 값의 차이도 작다. 이런 상관관계를 이용하기 위해 본 논문에서는 $C0$ 와 $C1$ 을 식(3)의 Harr 변환을 통해 CL 과 CH 로 변환한다.

$$CH = C0 - C1, CL = C1 + \lfloor (CH + 1) / 2 \rfloor \quad (3)$$

여기서 CL 과 CH 는 저주파와 고주파 신호성분이다. $C0$ 와 $C1$ 이 다르면 현재 블록의 왼쪽 혹은 위쪽 블록의 CL 을 통해 현재 블록의 CL 을 예측하고 예측오류를 문맥 기반 산술 부호화(context-based arithmetic coding)한다. 색상에는 R, G, B 채널이 있으므로 우선 R 채널에 대해 이 과정을 수행한 후, R 채널에서의 결과를 문맥으로 처리해 산술 부호화한다. 깊이와 알파 성분은 하나의 채널에 대해서만 예측 부호화 과정을 수행한다.

라. 인덱스 부호화

두 대표값이 다른 경우는 화소마다 다른 인덱스를 갖게 되므로 이를 부호화해야 한다. 이를 위해 코드북에서 현재 인덱스 블록과 정합이 되는 코드워드를 검색한다. 만약 검색에 성공하면 그 코드워드를 직접 산술부호화하고, 아닌 경우는 각 인덱스를 산술 부호화한다. 이 때, 좌상단, 상단, 좌측의 인덱스들이 현재 화소의 인덱스를 부호화하는데 문맥으로 사용된다.

4. 실시간 다시점 렌더링

다시점 렌더링의 개념은 새로운 시점을 렌더링하는데 다시점 영상들 사이의 공간적인 상관관계를 이용하는 것이다^[5~6,14]. 기존의 방법으로 Zitnick^[5]은 주 계층과 경계선 계층을 분리하여 처리함으로써 경계면에 나타나는 왜곡을 줄이는 방법을 사용하였다. Verlan^[6]은 시점 변화에 의존적인 화소기반 블렌딩 (blending) 기법을 이용하여 50fps의 속도로 새로운 시점을 렌더링 하였다. 본 논문에서는 Verlan^[6]의 방법을 채택했으나 화소가 아닌 블록단위로 다시점 렌더링을 수행하였다.

IV. 실험 결과

본 논문에서는 세 종류의 다시점 테스트 시퀀스를 사용해 제안한 방법을 검증했다. 실사 테스트 데이터로 “Breakdancers”와 “Ballet”^[16]을 사용했으며, 정확한 깊이정보를 이용한 렌더링 기법의 검증을 위해 그래픽스 데이터인 “Room”을 이용했다. 그림 5에 실험에 사용된 테스트 데이터 표본을 나타냈다.



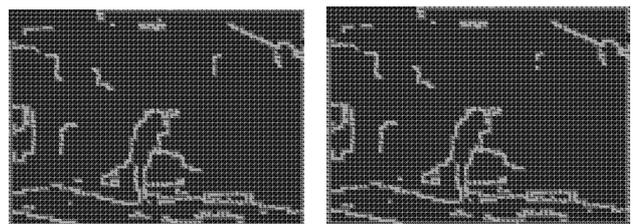
(a) “Breakdancers” (b) “Ballet” (c) “Room”

그림 5. 다시점 테스트 데이터의 텍스처 및 깊이맵
Fig. 5. Texture and depth maps of the test sequences.

1. 복원된 기하정보의 정확성 평가

제안한 C-TDMP와 X-Codec을 통해 “Breakdancers”의 깊이맵을 압축하고 이를 복호한 후 기하정보를 복원해 삼차원 장면을 생성한 결과를 그림 6에 나타냈다.

12:1의 압축률에도 불구하고 주관적인 화질 평가를 통해 원본 깊이맵과 복원한 깊이맵으로부터 생성한 삼차원 장면 사이의 차이를 구별하기가 어렵다. 0번 카메라의 10장의 프레임에 대해 원본과 복원된 깊이맵의 평균 PSNR을 계산하면 41.98 dB이다. 객관적 평가를 위해 “Breakdancers” 데이터에 대해 대표적인 무손실 압축 방식인 ZIP과 JPEG-LS의 경우와 제안한 방식의 압축률을 비교하였다. 테스트 데이터의 해상도는 1024 x 768이다. 무손실 압축이므로 주관적 화질평가로는 서로



(a) 원본 깊이맵 (b) 복원한 깊이맵

그림 6. 깊이맵으로부터 생성된 삼차원 장면
Fig. 6. Recovered Scene Geometry.

의 차이를 구분하기 어려우며, 100 프레임 영상에 대한 평균 압축률은 ZIP의 경우 1.69:1, JPEG-LS의 경우 3.87:1, 제안방식은 12:1이다.

삼차원 장면 탐색과 같은 그래픽스 응용에서는 충돌 검출이나 장면 조작 등의 상호작용이 요구되므로, 정확한 기하정보의 복원이 중요하다. 이런 측면에서 높은 압축률에도 원본과의 차이가 적은 삼차원 장면을 생성할 수 있는 제안 방식의 유용성을 확인할 수 있다.

2. 시점 생성을 통한 주관적 화질 평가

본 논문에서 제안한 부호화 과정을 거친 후 다시점 영상이 제대로 생성되는 지를 확인하기 위해서 복호된 다중 TDMP로부터 새로운 시점의 영상을 생성하였다. 우선 한 장의 TDMP로부터 다시점 영상을 생성한 결과를 그림 7에 나타냈다. 앞서 언급한 바와 같이, 한 장의 깊이맵이 2.5차원의 기하정보를 제공할 수 있기 때문에 일정한 시야각 내에서는 다시점을 복원할 수 있다. 하지만 시야각과 시점 변화가 커짐에 따라 결과 영상에 두드러진 왜곡이 발생하는 것을 확인할 수 있다.

이러한 문제점을 해결하기 위해서는 다시점 색상 및 깊이정보가 필요하다. 본 논문에서 사용된 다시점 렌더링 기법을 통해 다수의 TDMP로부터 생성한 다시점 영상을 그림 8에 나타냈다. 그림 7과 달리 시점 변화가 뚜렷하지만, 눈에 띄는 왜곡이 발생하지 않았다.

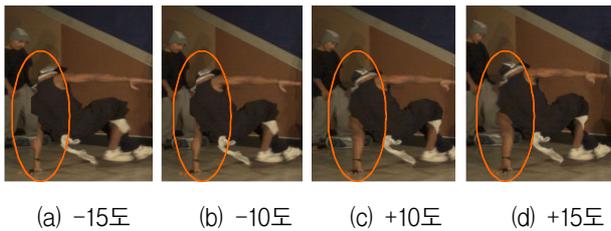


그림 7. 하나의 TDMP로부터 생성된 다시점 영상의 왜곡
Fig. 7. Generated distortions from a single TDMP.



그림 8. 다수의 TDMP로부터 생성된 다시점 영상
Fig. 8. Multiview Rendering Results from Multiple TDMPs.

3. 선택적 복호화

넓은 지역이나 장면을 탐색하려면 방대한 데이터 중

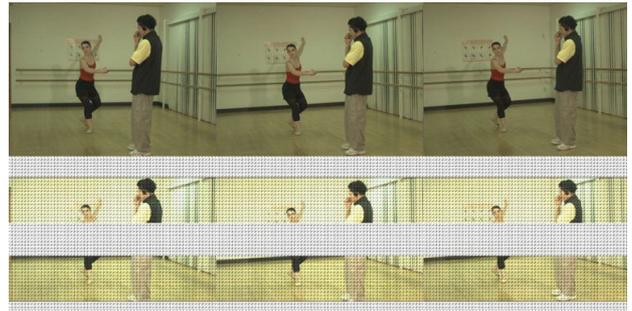


그림 9. 선택적 복호화 결과
Fig. 9. Selective Decoding Result.

일부만을 복호화해서 렌더링하는 기능이 필요하다. 제안한 부호화 방법은 영상의 블록을 기본 단위로 사용하므로 임의 영역을 복호화 할 수 있다. 그림 9는 부분적으로 복호된 다시점 영상의 예를 나타낸다.

4. 삼차원 장면 조작

제안한 방법은 깊이맵으로부터 삼차원 정보를 추출하므로 순수한 영상 기반 렌더링 기법으로는 하기 어려운 삼차원 장면 조작이 가능하다. 복원 기하정보를 통해 장면의 삼차원 위치를 계산할 수 있다. 그림 10은 삼차원 장면의 회전, 확대, 축소의 상호작용 결과를 보여준다.

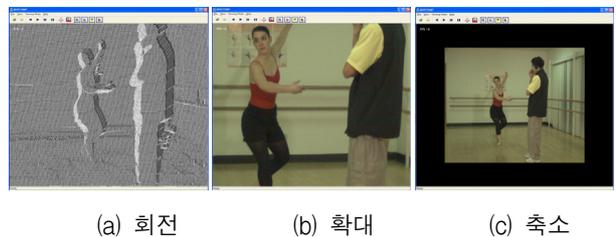


그림 10. 삼차원 장면의 조작
Fig. 10. Manipulation of the 3-D Scene.

V. 결론

본 논문에서는 텍스처와 깊이맵을 쌍으로 표현하고 이를 압축함으로써 복잡한 삼차원 장면에 대한 다시점 데이터를 최소한의 손실로 부호화하는 기법을 제안했다. 제안한 부호화 시스템은 영상 블록 단위로 부호화와 복호화를 수행하므로 임의 영역에 접근이 용이하고, GPU를 이용한 실시간 렌더링이 가능하다. 또한, 다시점 렌더링, 선택적 복호화, 삼차원 장면 조작 등의 기능을 제공한다. 따라서 대용량 데이터로부터 삼차원 장면을 탐색하는 등의 그래픽스 응용에 매우 유용하다.

참 고 문 헌

- [1] M. Levoy, K. Pulli, B. Curless, S. Rusinkiewicz, D. Koller, L. Pereira, M. Ginzton, S. Anderson, J. Davis, J. Ginsberg, J. Shade, and D. Fulk, "The digital Michelangelo project: 3D scanning of large statues," in *Proc. Annu. Conf. Computer Graphics (SIGGRAPH'00)*, pp. 131-144, July 2000.
- [2] "Weta: rendering Lord of the Rings," DIGIT, <http://www.digitmag.co.uk>, 2004.
- [3] H. Shum, S. Kang, and S. Chan, "Survey of image-based representations and compression techniques," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 13, no. 11, pp. 1020-1037, Nov. 2003.
- [4] T. Kanade, P. Rander, and P. Narayanan, "Virtualized reality: constructing virtual worlds from real scenes," *IEEE Multimedia Mag.*, vol. 1, no. 1, pp. 34-47, 1997.
- [5] C. Zitnick, S. Kang, M. Uyttendaele, S. Winder, and R. Szeliski, "High-quality video view interpolation using a layered representation," *ACM Trans. Graphics*, vol. 23, no. 3, pp. 600-608, Aug. 2004.
- [6] P. Verlani, A. Goswami, P. Narayanan, S. Dwivedi, and S. Penta, "Depth images: representation and real-time rendering," in *Proc. Int. Symp. 3D Data Processing, Visualization and Transmission*, June 2006.
- [7] P. Benzie, N. Grammalidis, S. Malassiotis, J. Ostermann, S. Piekh, V. Sainow, C. Theobalt, T. Thevar, and X. Zabulis, "3-D time-varying scene capture technologies—survey," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 17, no. 11, pp. 1568-1586, Nov. 2007.
- [8] S. Chen, "QuickTime VR: an image-based approach to virtual environment navigation," in *Proc. Annu. Conf. Computer Graphics (SIGGRAPH'95)*, pp. 29-38, Aug. 1995.
- [9] J. Shade, D. Lischinski, D. Salesin, T. DeRose, and J. Snyder, "Hierarchical image caching for accelerated walkthroughs of complex environments," in *Proc. Annu. Conf. Computer Graphics (SIGGRAPH'96)*, pp. 75-82, Aug. 1996.
- [10] M. Uyttendaele, A. Criminisi, S. Kang, S. Winder, R. Szeliski, and R. Hartley, "Image-based interactive exploration of real-world environments," *IEEE Computer Graphics and Applications*, vol. 24, no. 3, pp. 52-63, May/June 2004.
- [11] J. Shade, S. Gortler, L. He, and R. Szeliski, "Layered depth images," in *Proc. Annu. Conf. Computer Graphics (SIGGRAPH'98)*, pp. 231-242, Jul. 1998.
- [12] Akozak, "3Dc white paper," ATI Technologies, INC., May 2004.
- [13] Y. Linde, A. Buzo, and R. Gray, "An algorithm for vector quantizer design," *IEEE Trans. Communications*, vol. COM-28, no. 1, pp. 84-95, Jan. 1980.
- [14] M. Halle, "Multiple viewpoint rendering," in *Proc. Annu. Conf. Computer Graphics (SIGGRAPH'98)*, pp. 243-254, July 1998.
- [15] Interactive visual media group at Microsoft Research, <http://research.microsoft.com/vision/InteractiveVisualMediaGroup/3DVideoDownload>, 2004.

저 자 소 개



윤 승 욱(학생회원)
 2001년 2월 서강대학교
 전자공학과 졸업(학사).
 2002년 8월 광주과학기술원 정보
 통신공학과 졸업(석사).
 2008년 8월 광주과학기술원 정보
 통신공학과 졸업(박사).

<주관심분야: 다시점 비디오 부호화, 계층적 깊이영상 부호화 및 영상 기반 렌더링, 실감 미디어 처리 및 실감방송>



호 요 성(평생회원)
 1981년 2월 서울대학교
 전자공학과 졸업(학사).
 1983년 2월 서울대학교 대학원
 전자공학과 졸업(석사).
 1983년 3월~1995년 9월
 한국전자통신연구원
 선임연구원.

1989년 12월 미국 University of California, Santa Barbara Department of Electrical and Computer Engineering(박사),

1990년 1월~1993년 5월 미국 Philips 연구소 Senior Research Member.

1995년 9월~현재 광주과학기술원 정보통신 공학과 교수.

<주관심분야: 디지털 신호처리, 영상신호 처리 및 압축, 초저속 영상통신, 디지털 TV와 고선명 TV, MPEG 표준, 다시점 비디오 부호화, 실감방송>