

일반논문-11-16-4-03

GPU를 이용한 실시간 양안식 영상 생성 방법

신인용^{a)}, 호요성^{a)‡}

Real-time Stereo Video Generation using Graphics Processing Unit

In-Yong Shin^{a)} and Yo-Sung Ho^{a)‡}

요약

양안식 3차원 방송의 경우 좌우 두 시점에 해당하는 영상을 동시에 전송해야 하기 때문에 전송 대역폭의 부담이 매우 크다. 이러한 부담을 줄이기 위해 좌우 시점의 두 영상을 전송하는 대신에 좌영상과 이에 해당하는 깊이맵을 부호화하여 전송하는 방법이 있다. 이러한 3차원 방송 시스템의 수신단에서는 좌영상과 깊이맵을 복호한 뒤에 우영상을 만들어 좌우 영상을 실시간으로 출력한다. 본 논문에서는 좌영상과 깊이맵을 이용하여 가상시점 영상을 생성할 때 생기는 빈 공간을 효율적으로 채우는 기법을 제안하고, 전 과정의 실시간 처리를 위해 이를 GPU상에서 병렬로 처리되도록 구현했다. 그 결과 효과적으로 홀 채움을 수행하면서 CPU 대비 15배 이상 빠르게 양안식 영상을 생성할 수 있었다.

Abstract

In this paper, we propose a fast depth-image-based rendering method to generate a virtual view image in real-time using a graphic processor unit (GPU) for a 3D broadcasting system. Before the transmission, we encode the input 2D+depth video using the H.264 coding standard. At the receiver, we decode the received bitstream and generate a stereo video using a GPU which can compute in parallel. In this paper, we apply a simple and efficient hole filling method to reduce the decoder complexity and reduce hole filling errors. Besides, we design a vertical parallel structure for a forward mapping process to take advantage of the single instruction multiple thread structure of GPU. We also utilize high speed GPU memories to boost the computation speed. As a result, we can generate virtual view images 15 times faster than the case of CPU-based processing.

Key Words: CUDA, DIBR, GPU programming, Stereoscopic video codec, Parallel processing

1. 서론

최근 3차원 멀티미디어 서비스에 대한 관심이 증대되면서 3차원 방송에 대한 다양한 연구가 진행되고 있다. 3차원 방송은 인간의 좌우 시각 차이에 기인하는 양안 깊이 단서

를 이용하기 때문에 기존의 2차원 방송에 비해 높은 입체감과 몰입감을 제공한다^[1]. 3차원 콘텐츠의 대표적인 예는 양안식 영상으로 3차원 입체감을 제공하기 위해 사람의 양쪽 눈에 해당하는 두 가지 시점을 제공한다. 이 방법은 전송 측면을 고려했을 때, 기존의 2차원 방송에 비해 송신 채널 대역폭이 2배 정도 증가하게 된다^[2]. 이를 해결하기 위해 좌우 영상 모두를 보내는 대신에 좌영상과 압축률이 좋은 깊이 정보를 송신하고, 수신부에서 깊이 영상기반 렌더링 (depth image based rendering, DIBR) 기술로 우영상을 생

a) 광주과학기술원 정보통신공학과
Gwangju Institute of Science and Technology
‡ 교신저자 : 호요성 (hoyo@gist.ac.kr)
· 접수일(2011년3월16일), 수정일(2011년6월10일), 게재확정일(2011년6월10일)

성한다³⁾. 이 방법의 경우, 수신부의 복잡한 DIBR의 복잡도와 홀채움의 성능이 실시간 송수신과 품질에 영향을 미치게 된다. 홀채움의 경우 빠른 처리속도를 위해 단순한 방향보간 홀채움이 사용되곤 하였는데 홀이 상대적으로 많이 발생하는 단일시점의 컬러영상과 깊이영상을 이용한 합성에서는 잘못된 홀채움결과를 일으키기 쉬워진다.

본 논문에서는 방향보간 홀채움 방법의 단점을 해소하면서 빠르고 좀 더 효율적인 홀채움 방법을 제안하고, DIBR의 복잡도를 낮추기 위해서 병렬처리가 가능한 GPU를 이용하여 수신된 영상을 복호화하고 우영상을 생성하는 과정을 실시간으로 수행하는 방법을 제안한다.

논문의 구성은 다음과 같다. 2장에서는 깊이 영상과 컬러 영상을 활용한 양안식 방송 시스템에 대해 설명한다. 3장에서는 제안하는 홀채움 방법과 GPU를 통한 구현 방법을 설명한다. 4장에서는 구현 결과를 보이고 기존 연구와 비교한다. 마지막으로 5장에서는 결론을 맺는다.

II. 양안식 방송 시스템

일반적으로 양안식 영상 디스플레이 장치는 동일한 장면에 대해 서로 다른 좌우 시점으로 촬영된 두 개의 영상을 필요로 한다. 그러므로 그림 1과 같이 전송 대역폭이 두 배로 증가되는 문제가 발생된다.

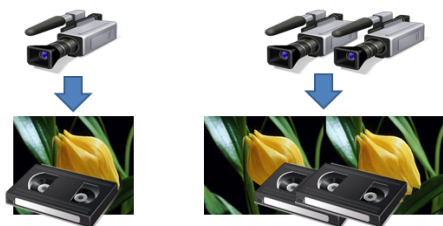


그림 1. 시점 개수에 따른 전송 대역폭 비교
Fig. 1. Bandwidth comparison of mono and stereo video

이러한 대역폭 증가 문제를 해결하기 위해 두 시점을 보내는 대신 하나의 시점과 그에 상응하는 깊이맵을 부호화하여 송신하는 방법이 사용된다. 이 방법은 깊이맵의 부호화 효율이 좋아서, 그림 2에 보인 것처럼, 필요한 전송 대역

폭이 줄어드는 장점이 있다. 하지만 수신부에서 그림 3과 같은 DIBR 과정이 스테레오 영상 합성을 위해 필요하다⁴⁾.

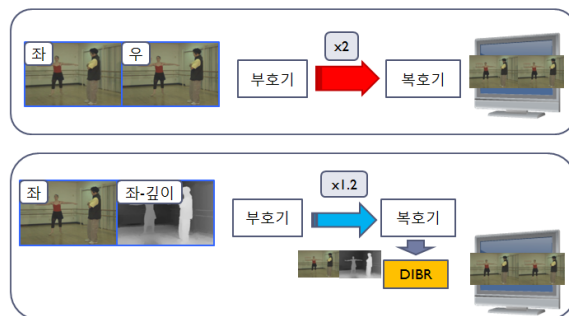


그림 2. 양안식 영상 전송 방식 비교
Fig. 2. Comparison of stereoscopic video broadcasting systems

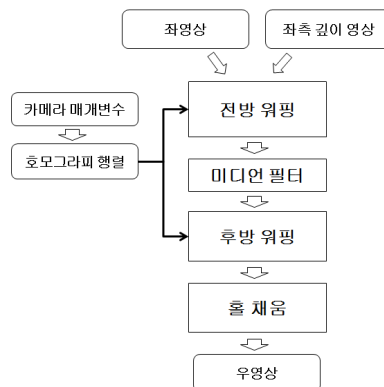


그림 3. DIBR의 순서도
Fig. 3. Block diagram of depth image based rendering algorithm

1. 3차원 워핑 과정

3차원 워핑 과정은 전방(forward) 워핑, 미디언 필터링, 후방(backward) 워핑으로 구성된다^{4,5)}. 전방 워핑에서는 좌시점 깊이맵을 우시점 깊이맵으로 변환하여 그림 4(a)와 같은 결과를 얻는다. 전방워핑 결과를 미디언 필터링으로 채우고, 마지막 후방 워핑에서는 우시점 깊이맵을 이용하여 좌시점 색상 영상으로부터 우시점 색상 영상을 생성한다. 3차원 워핑 이후 그림 4(b)와 같은 영상을 얻을 수 있다. 그림 4(b)는 좌영상에 존재하지 않던 새로운 영역으로 아직 화소값이 채워지지 못한 홀(hole) 영역이 존재한다.

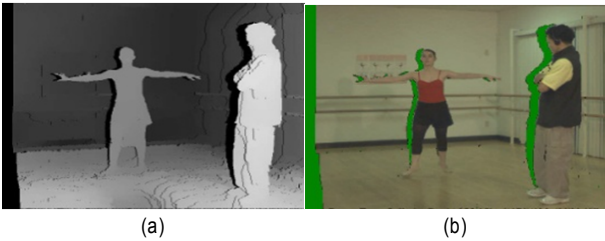


그림 4. (a) 전방 워핑 결과 (b) 후방 워핑 결과
 Fig. 4. (a) Forward warping result (b) Backward warping result

2. 홀 채움 과정

참조시점에 영상정보가 존재하지 않는 홀 영역은 홀 채움 과정을 통하여 채운다. 홀 영역은 좌시점에서 촬영되지 않은 새로운 부분이기 때문에 정확한 값을 얻기 어렵지만, 홀 영역 주변에 있는 화소들의 정보를 이용하여 예측한다.

일반적으로 홀 영역은 시점 이동에 따라 전경과 배경의 경계가 분리되면서 전경에 의해 가려졌던 배경이 새롭게 나타나는 부분이다. 따라서 홀은 배경 영역과 유사성이 높은 특성을 갖는다^[6]. 그림 5에 표현된 방향 보간 홀 채움 방법은 홀 영역을 배경 화소의 값으로 단순하게 베이스라인 방향으로 채워나가는 방식이다. 이 방법은 간단하게 홀을 채울 수 있어 고속 홀 채움 방식에 사용되었다^[7].



그림 5. 방향 보간 홀 채움 방법
 Fig. 5. (a) Warped color image with hole region (b) Result of traditional directional interpolation method

III. 제안하는 알고리즘

많은 경우에 방향 보간 방법은 빠르면서도 적절한 홀 채움 결과를 생성한다. 하지만, 그림 5와 같은 경우와 같이, 팔 안쪽 홀 영역의 경우 홀 영역이 모두 전경에 둘러싸여

홀이 잘못 채워지는 경우가 발생하는 것을 알 수 있다. 또한 홀 채움에 필요한 정보를 이전 프레임의 정보를 활용하는 시간적 상관도를 고려할 필요가 있었다. 따라서 제안하는 알고리즘은 홀 영역의 공간적 상관도와 시간적 상관도를 모두 고려하는 방법을 제안하고 모든 과정은 실시간 처리를 위해 GPU 병렬처리를 수행하였다.



그림 6. (a) 홀을 갖는 워핑된 색상 영상 (b) 방향 보간 방법의 잘못된 홀 채움
 Fig. 6. (a) Warped color image (b) Erroneous hole filling result

1. 공간적 상관도를 이용한 홀 채움 방법

일반적으로 홀 영역의 폭은 홀 영역의 양쪽 끝부분을 이루는 두 화소가 갖는 변이(Disparity)값의 차이에 의해 결정된다. 보통 홀의 양쪽 끝부분 화소는 좌영상에서는 서로 인접한 화소이다. 그러므로 일반적인 경우 방향 보간 방법은 홀 영역을 적절하게 채운다. 하지만 그림 6와 같은 경우 홀 영역의 양쪽 끝부분 화소들이 좌영상에서 서로 인접하지 않기 때문에 잘못된 홀 채움 결과를 만든다. 이러한 문제를 해결하기 위해, 홀을 채우기 전에 홀 영역의 양쪽 끝부분에 붙어있는 화소들에 대하여 두 화소가 좌영상에서 서로 인접한 화소인지 아닌지를 판단하는 과정을 수행한다. 판단 방법은 F_disp 와 B_disp 가 전경과 배경의 변이 값이고 th 는 양수의 문턱값(threshold value)이라고 할 때, 식 (1)의 값이 음수인 경우 좌영상에서 서로 인접하다고 판단하고, 나머지 경우에는 좌영상에서 서로 떨어져 존재하는 화소라고 판단한다.

$$A = \text{hole width} - ((F_disp - B_disp) + th) \quad (1)$$

식 (1)이 음수인 홀 영역은 방향 보간 방법으로 잘 채워지기 때문에 방향 보간 방법을 수행하게 되고, 그림 7과 같이

접선으로 표시된 영역은 식 (1)의 값이 음수가 아닌 값이다. 이러한 영역은 홀을 채울 때 홀의 우측 끝 화소의 위치를 좌영상에서 찾고, 좌영상에서 좌측 방향으로 존재하는 화소값들을 복사하여 홀을 채운다.



그림 7. 제안하는 보간 방법
Fig. 7. Proposed hole filling method

2. 시간적 상관도를 이용한 홀 채움 방법

앞서 설명한 홀 채움 과정에서는 공간적 상관도만을 고려했다. 고정된 카메라에서 촬영된 영상의 경우 일반적으로 배경 영상이 지속적으로 동일하게 존재하는 특성이 있다. 따라서 일정시간동안 배경영역이 고정된 경우 깊이 영상의 값들을 분석하여 전경이 일부 제거된 배경영상을 얻는다.

이렇게 얻어진 배경 영상은 전경이 일부 제거가 되었기 때문에 홀을 채우는데 매우 효과적이다. 식 (2)에서 I 와 BG_I 는 각각 홀 채움 이전의 우영상과 우시점의 배경 영상을 나타내고, x, y 는 화소의 좌표를 나타내고, t 는 시간을 나타낸다. 이때 배경 영상의 변화 또한 고려해야 한다. 따라서 t 의 경우 과거 모든 시간에 대해서가 아닌 현재 시간으로부터 제한된 범위의 시간을 사용한다.

$$BG_I(x,y) = I(x,y,t') \quad (2)$$

여기서

$$t' = \arg \min_t Depth(x,y,t) \quad (3)$$

식 (2)에서 I 의 홀 영역은 색상값이 없기 때문에 배경 영상을 획득할 수 없다. 결국 배경 영상 전체가 유효한 색상값을 갖기 힘들다. 이런 이유로 홀 영역을 채우는 과정에서

배경 영상이 존재하는 부분에 한정하여 배경 영상을 사용한다. 배경 영상을 사용하지 못하는 부분에 대해서는 공간적 홀 채움 방법을 사용하여 홀을 채우게 된다.

3. GPU를 이용한 병렬처리 기법

DIBR과정의 실시간 처리를 위해 병렬처리를 사용하였다. GPU를 사용한 병렬화 기법인 CUDA는 병렬화를 위한 단일 명령 복수 데이터(single instruction multiple data) 구조를 가지고 있어 동일한 명령을 다수의 데이터에 동시에 적용시키는 것이 가능하다^[8]. 이 구조를 활용하기 위해서는 데이터 사이에 종속성이 없어야 하는데, 3차원 전방 워핑을 병렬화하는 경우 둘 이상의 화소값이 워핑 후 동일한 위치로 투영되어 동시에 처리할 데이터가 겹치는 문제가 생길 수 있다.

이를 해결하기 위해, 투영 위치가 서로 겹칠 수 있는 화소들의 분포 특성을 고려하였다. 투영 위치가 겹칠 수 있는 화소들은 원 시점과 가상시점이 수평 이동된 구조를 갖기 때문에 시점 이동 방향과 동일하게 수평적으로 분포한다. 이 특성을 이용하여 병렬화 단위를 그림 8 과 같이 세로축으로 묶어 투영 위치가 겹치지 않도록 처리했다.

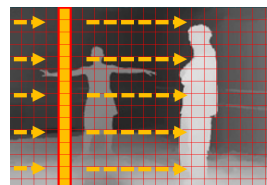


그림 8. 3차원 워핑 병렬화 구조
Fig. 8. Grid Structure for forward warping

IV. 실험 결과

이 논문에서 제안한 방법의 성능을 평가하려고 MPEG에서 제공하는 테스트 영상들을 가지고 실험하였다. 실시간 구현을 위해 각 영상들의 크기는 SD급으로 수정하여 사용하였다. 그림 9은 기존의 실시간 홀 채움에 사용된 방법과 새롭게 제안하는 홀 채움의 결과를 비교한 것이다. 그림



그림 9. 제안하는 홀 채움 결과
Fig. 9. Results of proposed method

9(a)는 방향 보간 홀 채움의 결과 영상으로 굽어진 손의 안쪽 부분과 같이 복잡한 전경에 의해 발생하는 잘못된 홀 채움 결과를 보였다.

그림 9(b)는 제안하는 홀 채움 결과로 그림 9(a)에 비해 홀 채움을 적절하게 채우는 것을 확인할 수 있었다. 그림 9(c)는 홀의 주변 화소만을 이용한 결과로 복잡한 무늬가 있는 홀의 영역을 제대로 복원하지 못하게 된다. 그림 9(d)에서 볼 수 있듯이, 배경 영상 획득을 이용한 방법을 이용하면 카메라가 정지된 영상에 대해서는 홀 채움의 오차가 줄어드는 것을 확인했고 PSNR 비교결과 약 1.2dB 향상되었음을 알 수 있었다.

복호화 및 영상합성과정이 모두 실시간으로 이루어져야 한다. 그림 10의 결과가 보여주는 CPU를 사용한 영상합성

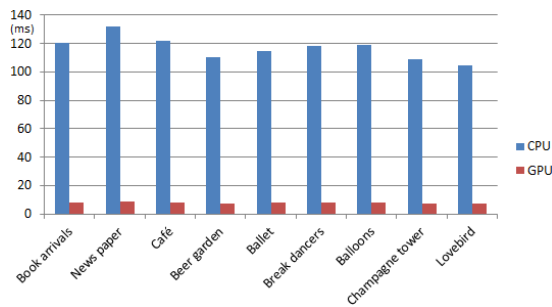


그림 10. 수행시간 비교
Fig. 10. Comparison of computation time

속도로는 복호화 및 DIBR과정의 실시간이 힘들다. 따라서 실시간 양안식 영상 생성을 위해 병렬처리를 지원하는 GPU를 사용하였다. 그림 10은 동일한 알고리즘에 대해 CPU와 GPU에서의 수행 시간을 비교한 것으로 영상합성에 대해서 GPU를 사용하여 수행 속도를 15배 이상 증가시킨 것을 보인다. 결과영상의 경우 CPU와 GPU 연산 모두 같은 영상을 생성하는 것으로 확인되었다.

수신단에서는 영상의 복호화 영상합성이 순차적으로 이루어진다. 복호화의 경우 그래픽카드에 내장된 칩셋을 사용하여 빠른 속도와 동시에 복호결과가 GPU에 저장되기 때문에 바로 영상합성에 활용할 수 있는 장점이 있다. 그림 11은 GPU에서 이루어지는 QP값에 따른 복호화 및 영상합성과정의 시간을 보인다. 영상의 종류에 따라 약간의 수행시간 차이는 있으나 QP값에 따른 수행시간 차이는 거의 없는 것으로 나타났다. 복호화 및 영상합성의 수행속도는 초당 60프레임을 넘어 실시간 처리를 하기에 충분하였다.

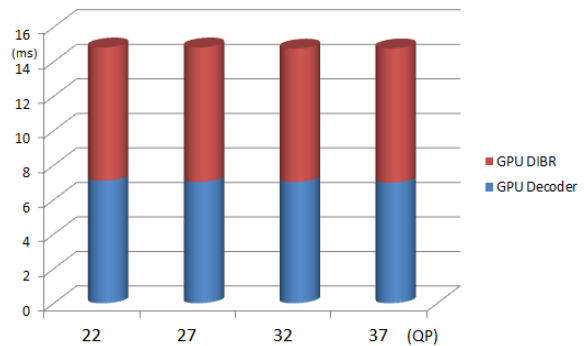


그림 11. QP값에 따른 디코더 및 영상합성 시간 비교
Fig. 11. Comparison of computation time

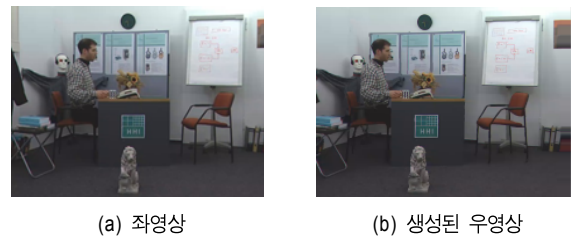


그림 12. 입력 좌영상과 생성된 우영상
Fig. 12. Original left image and synthesized right image

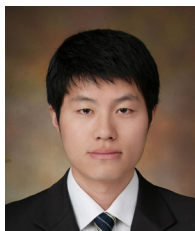
V. 결 론

본 논문에서는 좌영상과 깊이맵을 이용하여 우시점 영상을 만들 때 효율적으로 홀을 채우는 기법과 GPU상에서 DIBR 전 과정을 실시간으로 처리하는 방법을 제안했다. 좌우 양쪽의 참조영상을 사용하는 방법과 달리 좌시점만을 참조하기 때문에 발생할 수 있는 홀채움 오류가 나타난다. 홀 영역 좌우 끝부분의 깊이값을 통해 이 영역이 좌영상에서 서로 인접하는지 판단했고, 확인된 인접유무 정보를 이용하여 홀을 채우는 방법을 사용하여 이러한 오류를 줄일 수 있었다. 또한 배경영상을 획득하여 시간적 상관도를 고려한 홀채움을 수행하여 홀채움의 정확도를 높였다. 3차원 워핑 및 홀 채움의 모든 과정은 GPU상에서 병렬로 처리되도록 구현하였다. 그 결과 효과적으로 홀 채움을 수행하면서 동일 알고리즘에 대해 CPU 기반의 처리 속도보다 15배 이상 빠르게 처리하여 하여 복호화 및 영상합성 과정이 실시간으로 수행했다.

참 고 문 헌

- [1] O. Schreer, P. Kauff, and T. Sikora, 3D Videocommunication, John Wiley&Sons, 2005.
- [2] 최병호, 김용환, 김제우, 박지호, "스테레오 3D 방송을 위한 비디오 부호화 기술," 방송공학회 논문지, 제15권 제1호, pp. 24-36, 2010.
- [3] L. McMilan, "A List-Priority Rendering Algorithm for Redisplaying Projected Surfaces," Technical Report TR95-005, Univ. of North Carolina, July 1995.
- [4] G. Wolberg, Digital Image Warping, IEEE Computer Society Press, March 1990.
- [5] ISO/IEC JTC1/SC29/WG11, Contribution for 3D Video Test Material of Outdoor Scene, M1537, July 2008.
- [6] K. Oh, S. Yea, and Y. Ho, "Hole Filling Method using Depth Based In-painting for View Synthesis in Free Viewpoint Television and 3-D Video," Picture Coding Symposium, pp. 39-43, May 2009.
- [7] H. Shin, Y. Kim, H. Park, and J. Park, "Fast View Synthesis using GPU for 3D display," IEEE Transactions on Consumer Electronics, Vol. 54, No. 4, pp. 2068-2076, Nov. 2008.
- [8] nVIDIA Corporation, "CUDA 2.3 Programming Guide," in http://www.nvidia.com/object/cuda_develop.html, 2009.

저 자 소 개



신 인 용

- 2001년 ~ 2008년 : 인하대학교 전자공학과 학사
- 2009년 ~ 현재 : 광주과학기술원 정보기전공학부 정보통신공학과 석사과정
- 주관심분야 : 디지털 영상처리, H.264/AVC, 병렬처리



호 요 성

- 1977년 ~ 1981년 : 서울대학교 전자공학과 학사
- 1981년 ~ 1983년 : 서울대학교 전자공학과 석사
- 1983년 ~ 1995년 : 한국전지통신연구소 선임연구원
- 1985년 ~ 1989년 : University of California, 전기전산공학과 박사
- 1990년 ~ 1993년 : 미국 필립스 연구소 선임연구원
- 1995년 ~ 현재 : 광주과학기술원 정보통신공학부 교수
- 2003년 ~ 현재 : 광주과학기술원 실감방송연구센터 센터장
- 주관심분야 : 디지털 영상신호 처리 및 압축, 디지털 TV, MPEG 표준, 3차원 TV, 실감방송