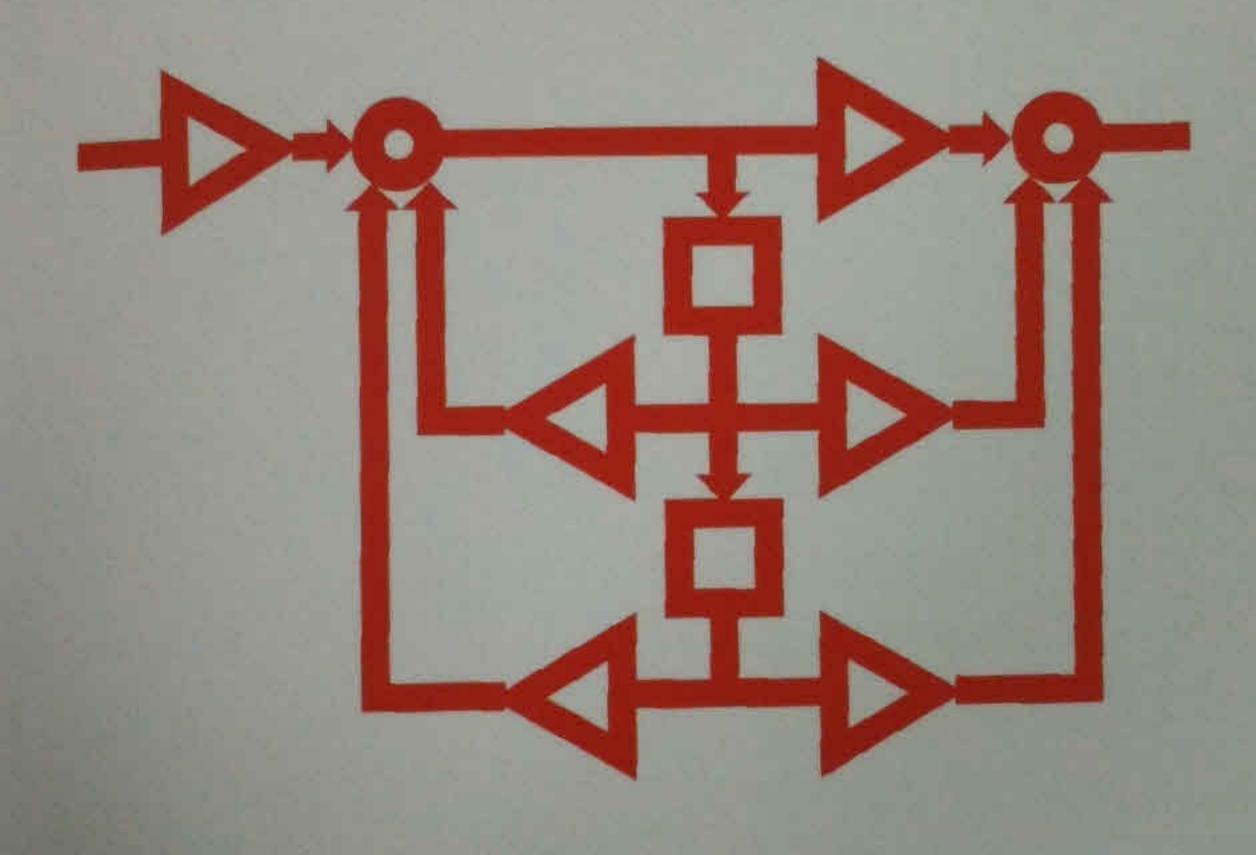
2010년도

제23회 신호처리합동학술대회 논문집

Proceeding of the 2010 Korea Signal Processing Conference

일 시: 2010년 10월 2일 (토)

장소: 중앙대학교



● 주최: 한국방송공학회 3DTV 연구회

대한전자공학회 음향 및 신호처리연구회, 회상처리 및 텔레비전연구회

한국통신학회 신호처리연구회, 영상통신연구회

한국음향학회 신호처리연구회, 음성처리연구회

● 주관 : Alegger 한국방송공학회

● 후원: 중앙대학교 (문화콘텐츠기술연구원, BK21 한국형디지털영상사업단, 서울미래형 콘텐츠컨버전스클러스터, 미래형영상콘텐츠기술연구단, 시각 및 지능시스템연구실), IEEE Korea SP Chapter, IEEE Korea CE Chapter

● 협찬: (주)네오시드넷, (주)스타넥스, (주)에디텍, 인피니티북스, (주)하이트론시스템즈

특별 세션 발표 (3)

발표장소 : 제2공학관 219호

| ISP Techniques for Digital Camera | |
|--|-------------------------------|
| | • 과장: 신정호 (한경대) |
| | • 시간 : 10:00 ~ 11:30 |
| [S3-01] 영상 향상을 위한 립러 보정 방법 비교 | 91 |
| *************************************** | ···· 양세정, 집윤아, 이병욱 (이화여대) |
| [S3-02] Mobile camera의 3A 시스템과 기술 발전 방향 | 94 |
| | 선규, 이상화, 김덕수, 침대한 (삼성전자) |
| [S3-03] Mobile camera if 위한 auto white balance 방법 | 97 |
| | |
| [53-04] 비원주사 보색필터 CCD에서의 라인단차 보상 방법 | 100 |
| | , 이정수, 정회인, 김효주 ((주) 넥스트집) |
| [S3-05] Rolling Shutter 환경에서의 모션벡터 예측에 대한 연구 | 103 |
| *************************************** | 조재신, 이성권 (정류네) |
| | |
| Computer Vision Application | • 좌장 : 윤국진 (GIST) |
| | • 시간 : 13:30 ~ 15:00 |
| and the second second second and second second | 107 |
| [S3-06] 동적 장면에 대한 파노라마 생성기법 연구 | 신용호, 박민규, 윤국진 (GIST) |
| [83-07] 적외선 소행표적 탐지를 위한 오정보 감소 기법 동향 및 전 | |
| [S3-07] 직의전 오정보기 된 기회 기진 고 3고 1 전 호*. 양유경** | 이주형** (*영남대, **국방과학연구소) |
| [53-08] 비등방성 정규화器 이용한 단일 영상의 모션 디볼터링 | 114 |
| 153-081 41-5-8-8 8 11-4-12 -1-0-12 -1-0-12 | 흥한유, 유진우, 박인규 (인하대) |
| [S3-09] IPTV를 위한 HMM기반 손제소처 인식 시스템 | 118 |
| [S3-09] IPT V는 위안 IRMINITE 도 기가 다 이 찬수*, 천성용*, | 집상호*, 이장우** (*영남대, **DGIST) |
| The state of the s | |
| | |
| | |
| on III 1 Decement | |
| 3D Visual Processing | • 좌장 : 실재엥 (울산파기대) |
| | • 시간 : 15:30 ~ 17:00 |
| [S3-10] KL변환을 이용한 3차원 메쉬의 기하정보 압축 | 121 |
| [S3-10] KLEES OF SHE SAFE WATER THE SHEET OF | 이대연, 완재균, 김창수 (고려대) |
| 그 그 그 그 그 그 그 그 그 그 그 그 그 그 그 그 그 그 그 | 1263 |
| · · · · · · · · · · · · · · · · · · · | **, 변원민*, 박성기* (*KIST, **서울대) |
| | 서의 압축 |
| [S3-12] 간설때턴간의 상관관계를 이용한 PSD 너지를 받고 기상수 *** 이상 |) 숙* (*서울대, **울산과기대, ***고려대) |
| 3 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 | |
| [S3-13] 호파적인 홈 제통을 이용한 원이션 경험이 - | |
| | |
| [S3-14] 公司司文 3D TV에서 Crosstate 3 - 1 | 아마다 이수연, 작영신 (출산과기대) |
| | |

효과적인 홀 채움을 이용한 실시간 양안식 영상 생성 방법

신 인 용, 호 요 성 광주과학기술원 정보통신공학과

Efficient Hole Filling Method for Real-time Stereo Video Generation

In-Yong Shin, Yo-Sung Ho
Gwangju Institute of Science and Technology (GIST)
{siy0808, hoyo}@gist.ac.kr

Abstract: 양안식 3차원 방송의 경우 좌우 두 시절에 해당하는 영상을 동시에 전송해야 하기 때문에 전송 대역폭 부담이 매우 크다. 이러한 부담을 줄이기 위해 좌우 시절의 두 영상을 전송하는 대신에 좌영상과 이에 해당하는 깊이맵을 부호화하여 전송하는 방법이 있다. 이러한 3차원 방송 시스템의 수산단에서는 좌영상과 깊이맵을 복호한 뒤에 우영상을 만들어 좌우 영상을 실시간으로 출력한다. 본 논문에서는 좌영상과 깊이맵을 이용하여 가장시점 영상을 생성할 때 챙기는 빈 공간을 효율적으로 채우는 기법을 제안하고, 전 과정의 실시간 처리를 위해 알고리즘을 GPU상에서 병렬로 처리되도록 구현했다. 그 결과 효과적으로 홀 채움을 수행하면서 CPU 대비 15배 이상 빠르게 양안식 영상을 생성할수 있었다.

Keywords: DIBR, 병렬처리, CUDA

I. 서론

최근 3차원 멀티미디어 서비스에 대한 관심이 증대되면서 3차원 방송에 대한 다양한 연구가 진행되고 있다. 3차원 방송은 인간의 좌우 시각 차이에 기언하는 양안 깊이 단 서를 이용하기 때문에 기존의 2차원 방송에 비해 높은 입 체감과 물업감을 제공할 수 있다 [1].

3차원 콘텐츠의 대표적인 에는 양안식 영상이다. 양안 식 영상은 인간이 3차원 임체감을 느끼도록 사람의 양쪽 눈에 서로 다른 시점의 좌영상과 우영상을 제공한다. 이 방법은 전송 측면을 고려했을 때, 기존의 2차원 방송에 비해 송신 채널 대역폭이 2배 정도 증가한다.

이러한 대역폭 증가의 문제점을 해결하기 위해 다음과 같은 방법을 사용할 수 있다. 첫째로, 시점간의 유사성을 활용하여 데이터양을 줄이는 다시점 비디오 부호화 방법 이 있다. 둘째로, 좌우 영상 모두를 보내는 대신에 좌영상 과 그에 상용하는 깊이 정보를 송신하고, 수신부에서 깊 이영상기반 렌더링(depth image based rendering, DIBR) 기술로 우영상을 생성하여 양안식 영상을 즐릭하는 방식 이 있다 [2]. 두 번째 방법의 경우, 우영상 대신 부호를 효율이 높은 좌영상의 깊어맵을 압축하여 전송하기 때문에 대역폭을 상당히 줄일 수 있다. 하지만, 수신부의 복을 한 DIBR 과정 때문에 양안식 영상을 출력하는데 오랜시간이 소요되어 실시간 처리에 문제가 생길 수 있다.

본 논문에서는 좌우 영상을 모두 보내는 대신에 최형 상과 그에 대응하는 깊이맵을 압축 부호화하여 보내고, 수신부에서 DIBR 방법으로 양안식 영상을 생성하는 방법 을 다룬다. 우영상을 만들 때 기존의 방향 보간 혼 제품 에서 발생할 수 있는 문제점을 해결하면서 GPU를 이용 하여 실시간으로 DIBR을 수행하는 방법을 제안한다

Ⅱ. 양안식 방송 시스템

일반적으로 양안식 영상 디스플레이 장치는 동일한 장면에 대해 서로 다른 좌우 시점으로 촬영된 두 개의 영생을 필요로 한다. 그러므로 양안식 영상 방송을 위해서는 좌우 시점 두 영상을 통신 채널로 보내야 한다. 하지만 두 시점을 동시에 보내는 경우, 그림 1과 같이, 건송 대역폭이 두 배로 증가되는 문제가 발생된다.

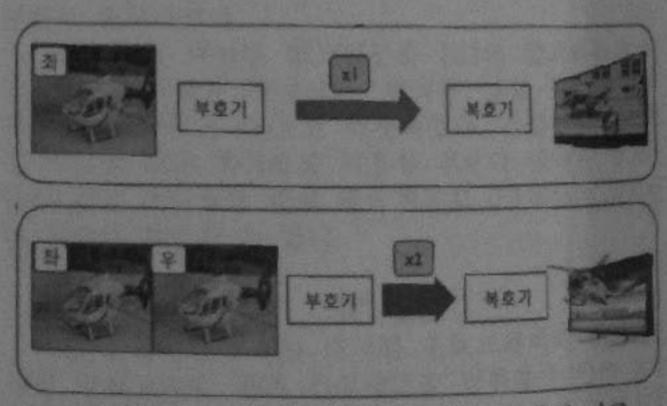


그림 1. 시점 개수에 따른 전송 대역폭 비교

이러한 대역폭 증가 문제를 해결하기 위해 및 가지방 법들이 제안되었는데, 본 논문에서는 두 시점을 보내는 대신 하나의 시점과 그에 상응하는 깊이맵을 부호화하여

송신하는 방법을 사용했다. 이 방법은 깊이맵의 부호화 효율이 좋아서, 그림 2에 보인 것처럼, 필요한 전송 대역 목이 2배에서 1.2배로 줄어드는 장점이 있다. 하지만 수신 부에서 받아들인 영상은 양안식 영상이 아니고, 좌영상과 깊이뱀이다. 따라서 이것을 양안식 영상으로 만들기 위해 DIBR 과정을 실시간으로 처리해야 한다. 수신부에 포함 되는 DIBR 과정은 그림 3과 같은 순서로 진행된다 [3].

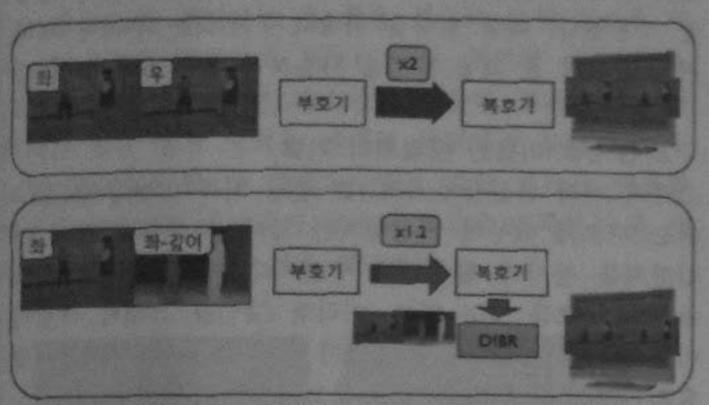


그림 2 좌우 영상을 보내는 방식과 좌영상과 깊이맵을 보내는 방식의 전송 대역폭 비교

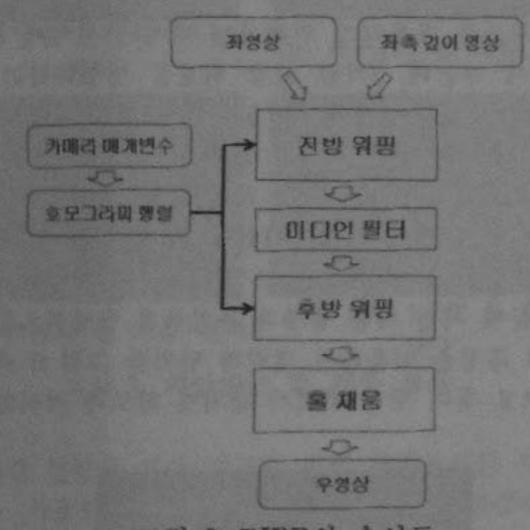


그림 3. DIBR의 순서도

1. 3차원 워핑 과정

3차원 워핑 과정은 전방(forward) 워핑, 미디언 필터링, 후방(backward) 위핑으로 구성된다 [4]. 전방 워핑에서는 좌시점 깊이땝을 우시점 깊이맵으로 변환하여 그림 4(a) 와 같은 결과를 얻는다. 전방 워핑된 깊이맵에 존재하는 균열처럼 보이는 작은 빈 곳을 주변 깊이값의 미디언 값 으로 새운다. 마지막 후방 워핑에서는 우시점 깊이맵을 이용하여 좌시점 색상 영상으로부터 우시점 색상 영상을 생성한다. 이때 모든 워핑 과정은 호모그라피 행렬을 이 용한다. 호모그라피 행렬은 깊이값에 따라 화소들이 옮겨 절 위치를 결정한다. 3차원 위핑 과정이 끝나면 그림 4(b)와 같은 영상을 얻을 수 있다. 그림 4(b)는 좌영상에 존재하지 않던 새로운 영역으로 아직 화소값이 채워지지 옷한 홀(hole) 영역이 존재한다.



그림 4. (a) 전방 워핑 결과 (b) 후방 워핑 결과

2 홈 채움 과정

3차원 위핑 과정을 수행하면 홀 영역을 포함하는 우영상 이 만들어진다. 이때 워핑되지 못한 화소들인 홀 영역은 홀 채움 과정을 통하여 채운다. 홀 영역은 좌시점에서 촬 영되지 않은 새로운 부분이기 때문에 정확한 값을 얻기 어렵지만, 홑 영역 주변에 있는 화소들의 정보를 이용하 여 적용적으로 예측할 수 있다.

일반적으로 홀 영역은 시점 이동에 따라 전경과 배경 의 경계가 분리되면서 전경에 의해 가려졌던 배경이 새롭 게 나타나는 부분이다. 따라서 흩은 배경 영역과 유사성 이 높다 [5]. 그림 5에 표현된 방향 보간 홀 채움 방법은 홀 영역을 배경 화소의 값으로 채워나가는 방식이다. 이 방법은 간단하게 홀을 채울 수 있어 고속 홀 채움 방식에 사용되었다 [6]



그림 5. 방향 보간 홀 채움 방법

Ⅲ. 제안하는 알고리즘

많은 경우에 있어서 방향 보간 방법은 적절한 홀 채움 결 과를 생성한다. 하지만, 그림 6과 같은 경우를 보면, 팔 안쪽 홀 영역의 경우 홀 영역이 모두 전경에 둘러싸여 홀 이 잘못 채워지는 경우가 발생하는 것을 알 수 있다.



그림 6. (a) 홀을 갖는 워핑된 색상 영상 (b) 방향 보간 방법을 이용한 홀 채움 결과

1. 공간적 상관도를 이용한 홀 채움 방법 일반적으로 홀 영역의 폭은 흩 영역의 양쪽 끝부분을 이 루는 두 화소가 갖는 변이(Disparity)값의 차이에 의해 결 정된다. 보통 홀의 양쪽 끝부분 화소는 좌영상에서는 서 로 인접한 화소이다. 그러므로 일반적인 경우 방향 보간 방법은 홀 영역을 적절하게 채운다. 하지만 그림 6과 같 은 경우 홀 영역의 양쪽 끝부분 화소들이 좌영상에서 서 로 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_disp와 B_disp 가 전경과 배경의 변이 값이고 th는 양수의 문턱값이라고 할 때, 식(1)의 값이 음수인 경우 좌영상에서 서로 인접하다고 판단하고 나머지 경우에는 좌영상에서 서로 떨어져 존재하는 화소라고 판단한다.

$$A = hole_width - ((F_disp - B_disp) + th)$$
 (1)

식 (1)이 음수인 흩 영역은 방향 보간 방법으로 잘 채워지기 때문에 방향 보간 방법을 수행하게 되고. 그림 7과 같이 점선으로 표시된 영역은 식 (1)의 값이 음수가 아닌 값이다. 이러한 영역은 흩을 채울 때 홀의 우측 끝화소의 위치를 좌영상에서 찾고, 좌영상에서 좌즉 방향으로 존재하는 화소값들을 복사하여 홀을 채운다.



그림 7. 제안하는 보간 방법

2 시간적 상관도를 이용한 홀 채움 방법 앞서 설명한 홀 채움 과정에서는 공간적 상관도만을 고려 했다. 고정된 카메라에서 촬영된 영상의 경우 일반적으로 배경 영상이 지속적으로 동일하게 존재하는 특성이 있다. 따라서 일정시간동안 배경영역이 고정된 경우 깊이영상의 값들을 분석하여 전경이 일부 제거된 배경영상을 얻는다.

이렇게 얻어진 배경 영상은 전경이 일부 제거가 되었기 때문에 홀을 채우는데 매우 효과적이다. 식 (2)에서 I와 BG J는 각각 홀 채움 이전의 우영상과 우시점의 배경 영상을 나타내고, x, y는 화소의 좌표를 나타내고, t는 시간을 나타낸다. 이때 매경 영상의 변화 또한 고려해야 한다. 따라서 t의 경우 과거 모든 시간에 대해서가 야닌 현재시간으로부터 제한된 범위의 시간을 사용한다.

$$BG_{-}I(x,y) = I(x,y,t')$$
(2)

여기서

$$t' = \arg\min_{t} Depth(x, y, t)$$
 (3)

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

그래픽 처리 연산장치(GPU)는 중앙 처리 연산장치(CPU) 와는 다르게 내부에 연상장치가 많다. 따라서 각각의 연산장치를 동시에 활용하여 병렬처리를 이용한 고속화에 유리한 구조를 갖고 있다. 이러한 GPU를 그래픽 작업이 아닌 일반적인 용도에 사용하기 위해 CUDA를 통하여 병렬화된 가상시점 영상 생성 알고리즘을 구현했다.

CUDA는 병렬화를 위한 단일 명령 복수 데이터(single instruction multiple data) 구조를 가지고 있어 동일한 명령을 다수의 데이터에 동시에 적용시키는 것이 가능하다[7]. 이 구조를 활용하기 위해서는 데이터 사이에 중속생이 없어야 하는데, 3차원 전방 워핑을 병렬화하는 경우들 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 새로축으로 묶어 투영 위치가 겹치지 않도록 처리했다.

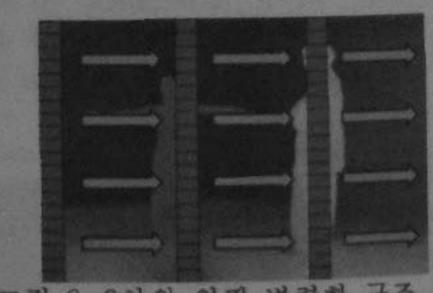


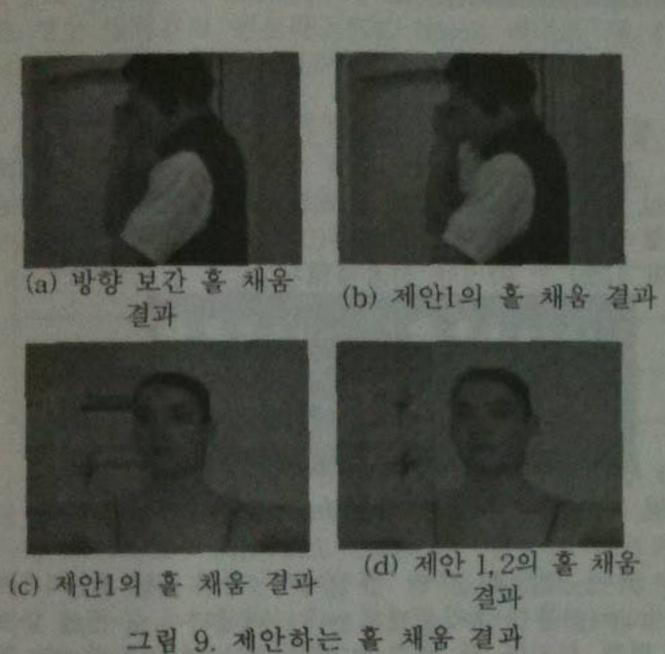
그림 8. 3차원 워핑 병렬화 구조

뿐만 아니라, 실시간 시스템의 GPU 성능을 최석화하기 위해서는 용도에 맞는 메모리의 사용이 메우 중요하다. GPU에는 전역 메모리, 공유 메모리, 레지스터 메모리, 텍스쳐 메모리 등 몇 가지 다른 특성을 갖는 메모리가 있는 데, 최적화된 속도를 내기 위해서 자주 참조되는 영상은 일기전용인 테스처 메모리에 위치시켜 사용했다. 텍스처 메모리의 경우 부분적인 수정은 불가능하다. 하지만 값을 참조하는 속도가 다른 메모리에 비하여 빠른 장점이 있어 참조하는 속도가 다른 메모리에 비하여 빠른 장점이 있어 자주 사용하는 호모그라피 행렬과 참조만 하는 영상을 텍스처 메모리에 위치시켜 고속화했다.

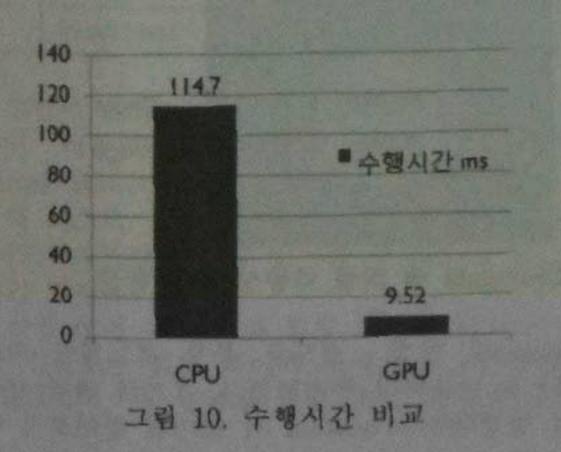
Ⅳ. 실험 결과

이 논문에서 제안한 방법의 성능을 평가하려고 Microsoft Research에서 제공하는 720x480 크기의 Ballet 영상을 사용하였다. 그림 9는 기존의 실시간 흘 채움에 사용된 방법과 새롭게 제안하는 흘 채움의 결과를 비교한 것이다. 그림 9(a)는 방향 보간 흘 채움의 결과 영상으로 굽어진 손의 안쪽 부분에서 잘못된 흘 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 복잡한무늬가 있는 홀의 영역을 제대로 복원하지 못하게 된다. 그림 9(d)에서 볼 수 있듯이, 배경 영상 획득을 이용한방법을 이용하면 카메라가 정지된 영상에 대해서는 홀 채움의 오차가 줄어드는 것을 확인했다.



실시간 양안식 영상 생성을 위해 병렬처리를 지원하는 GPU를 사용했다. 그림 10은 동일한 알고리즘에 대해 CPU와 GPU에서의 수행 시간을 비교한 것인데, GPU를 사용하여 수행 속도를 15배 이상 빠르게 했다. 그 결과보다 정확하게 홀 채움을 수행하면서도 720x480 크기를 갖는 그림 11(a)와 해당 깊이맵을 이용하여 그림 11(b)를 초당 100장 정도를 실시간으로 생성할 수 있다.





(a) 좌영상 (b) 생성된 우영상 그림 11 입력 좌영상과 생성된 우영상

V. 결론

본 논문에서는 좌영상과 깊이맵을 이용하여 우시점 영상을 만들 때 효율적으로 홀을 채우는 기법과 GPU상에서 DIBR 전 과정을 실시간으로 처리하는 방법을 제안했다. 홀 영역 좌우 끝부분의 깊이값을 통해 이 영역이 좌영상에서 서로 인접하는지 판단했고, 확인된 인접유무 정보들이용하여 홀을 채우는 방법을 선택하였다. 3차원 워핑 및홀 채움의 모든 과정은 GPU상에서 병렬로 처리되도록구현하였다. 그 결과 효과적으로 홀 채움을 수행하면서 동일 알고리즘에 대해 CPU 기반의 처리 속도보다 15배이상 빠른 속도를 얻을 수 있었다.

감사의 글

본 연구는 지식경제부 및 정보통신산업진흥원의 대학 IT연구센터 지원 사업의 연구 결과로 수행되었습니다. (NIPA-2010- (C1090-1011-0003))

참고문헌

- [1] O. Schreer, P. Kauff, and T.Sikora, 3D Videocommunication, John Wiley&Sons, 2005.
- [2] L. McMilan, "A List-Priority Rendering Algorithm for Redisplaying Projected Surfaces," Technical Report TR95-005, Univ. of North Carolina, July 1995.
- [3] G. Wolberg, Digital Image Warping, IEEE Computer Society Press, March 1990.
- [4] ISO/IEC JTC1/SC29/WG11, Contribution for 3D Video Test Material of Outdoor Scene, M1537, July 2008.
- [5] 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.
- [6] 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.
- [7] nVIDIA Corporation, "CUDA 2.3 Programming Guide," in http://www.nvidia.com/object/cuda_ develop.html, 2009.

효과적인 홀 채움을 이용한 실시간 양안식 영상 생성 방법

신 인 용, 호 요 성 광주과학기술원 정보통신공학과

Efficient Hole Filling Method for Real-time Stereo Video Generation

In-Yong Shin, Yo-Sung Ho
Gwangju Institute of Science and Technology (GIST)
{siy0808, hoyo}@gist.ac.kr

Abstract: 양안식 3차원 방송의 경우 좌우 두 시점에 해당하는 영상을 동시에 전송해야 하기 때문에 전송 대역폭 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 실시간처리를 위해 알고리즘을 GPU상에서 병렬로 처리되도록 구현했다. 그 결과 효과적으로 홀 채움을 수행하면서 CPU 대비 15배 이상 빠르게 양안식 영상을 생성할수 있었다.

Keywords: DIBR, 병렬처리, CUDA

I. 서론

최근 3차원 멀티미디어 서비스에 대한 관심이 증대되면서 3차원 방송에 대한 다양한 연구가 진행되고 있다. 3차원 방송은 인간의 좌우 시각 차이에 기인하는 양안 깊이 단서를 이용하기 때문에 기존의 2차원 방송에 비해 높은 입체감과 몰입감을 제공할 수 있다 [1].

3차원 콘텐츠의 대표적인 예는 양안식 영상이다. 양안식 영상은 인간이 3차원 입체감을 느끼도록 사람의 양쪽눈에 서로 다른 시점의 좌영상과 우영상을 제공한다. 이방법은 전송 측면을 고려했을 때, 기존의 2차원 방송에비해 송신 채널 대역폭이 2배 정도 증가한다.

이러한 대역폭 증가의 문제점을 해결하기 위해 다음과 같은 방법을 사용할 수 있다. 첫째로, 시점간의 유사성을 활용하여 데이터양을 줄이는 다시점 비디오 부호화 방법 이 있다. 둘째로, 좌우 영상 모두를 보내는 대신에 좌영상 과 그에 상응하는 깊이 정보를 송신하고, 수신부에서 깊 이영상기반 렌더링(depth image based rendering, DIBR) 기술로 우영상을 생성하여 양안식 영상을 출력하는 방식 이 있다 [2]. 두 번째 방법의 경우, 우영상 대신 부호화 효율이 높은 좌영상의 깊이맵을 압축하여 전송하기 때문에 대역폭을 상당히 줄일 수 있다. 하지만, 수신부의 복잡한 DIBR 과정 때문에 양안식 영상을 출력하는데 오랜 시간이 소요되어 실시간 처리에 문제가 생길 수 있다.

본 논문에서는 좌우 영상을 모두 보내는 대신에 좌영 상과 그에 대응하는 깊이맵을 압축 부호화하여 보내고, 수신부에서 DIBR 방법으로 양안식 영상을 생성하는 방법 을 다룬다. 우영상을 만들 때 기존의 방향 보간 홀 채움 에서 발생할 수 있는 문제점을 해결하면서 GPU를 이용 하여 실시간으로 DIBR을 수행하는 방법을 제안한다.

Ⅱ. 양안식 방송 시스템

일반적으로 양안식 영상 디스플레이 장치는 동일한 장면에 대해 서로 다른 좌우 시점으로 촬영된 두 개의 영상을 필요로 한다. 그러므로 양안식 영상 방송을 위해서는 좌우 시점 두 영상을 통신 채널로 보내야 한다. 하지만두 시점을 동시에 보내는 경우, 그림 1과 같이, 전송 대역폭이 두 배로 증가되는 문제가 발생된다.



그림 1. 시점 개수에 따른 전송 대역폭 비교

이러한 대역폭 증가 문제를 해결하기 위해 몇 가지 방법들이 제안되었는데, 본 논문에서는 두 시점을 보내는 대신 하나의 시점과 그에 상응하는 깊이맵을 부호화하여 송신하는 방법을 사용했다. 이 방법은 깊이맵의 부호화효율이 좋아서, 그림 2에 보인 것처럼, 필요한 전송 대역폭이 2배에서 1.2배로 줄어드는 장점이 있다. 하지만 수신부에서 받아들인 영상은 양안식 영상이 아니고, 좌영상과깊이맵이다. 따라서 이것을 양안식 영상으로 만들기 위해 DIBR 과정을 실시간으로 처리해야 한다. 수신부에 포함되는 DIBR 과정은 그림 3과 같은 순서로 진행된다 [3].

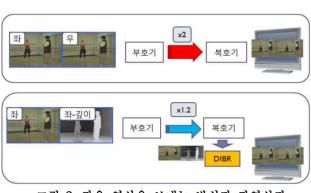


그림 2. 좌우 영상을 보내는 방식과 좌영상과 깊이맵을 보내는 방식의 전송 대역폭 비교

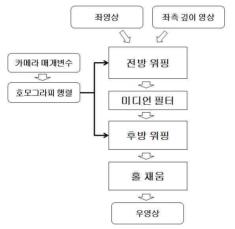


그림 3. DIBR의 순서도

1. 3차원 워핑 과정

3차원 워핑 과정은 전방(forward) 워핑, 미디언 필터링, 후방(backward) 워핑으로 구성된다 [4]. 전방 워핑에서는 좌시점 깊이맵을 우시점 깊이맵으로 변환하여 그림 4(a)와 같은 결과를 얻는다. 전방 워핑된 깊이맵에 존재하는 균열처럼 보이는 작은 빈 곳을 주변 깊이값의 미디언 값으로 채운다. 마지막 후방 워핑에서는 우시점 깊이맵을 이용하여 좌시점 색상 영상으로부터 우시점 색상 영상을 생성한다. 이때 모든 워핑 과정은 호모그라피 행렬을 이용한다. 호모그라피 행렬은 깊이값에 따라 화소들이 옮겨질 위치를 결정한다. 3차원 워핑 과정이 끝나면 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.

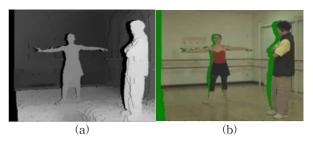


그림 4. (a) 전방 워핑 결과 (b) 후방 워핑 결과

2. 홀 채움 과정

3차원 워핑 과정을 수행하면 홀 영역을 포함하는 우영상이 만들어진다. 이때 워핑되지 못한 화소들인 홀 영역은 홀 채움 과정을 통하여 채운다. 홀 영역은 좌시점에서 촬영되지 않은 새로운 부분이기 때문에 정확한 값을 얻기어렵지만, 홀 영역 주변에 있는 화소들의 정보를 이용하여 적응적으로 예측할 수 있다.

일반적으로 홀 영역은 시점 이동에 따라 전경과 배경의 경계가 분리되면서 전경에 의해 가려졌던 배경이 새롭게 나타나는 부분이다. 따라서 홀은 배경 영역과 유사성이 높다 [5]. 그림 5에 표현된 방향 보간 홀 채움 방법은홀 영역을 배경 화소의 값으로 채워나가는 방식이다. 이방법은 간단하게홀을 채울 수 있어 고속홀 채움 방식에사용되었다[6].



그림 5. 방향 보간 홀 채움 방법

Ⅲ. 제안하는 알고리즘

많은 경우에 있어서 방향 보간 방법은 적절한 홀 채움 결과를 생성한다. 하지만, 그림 6과 같은 경우를 보면, 팔안쪽 홀 영역의 경우 홀 영역이 모두 전경에 둘러싸여 홀이 잘못 채워지는 경우가 발생하는 것을 알 수 있다.



그림 6. (a) 홀을 갖는 워핑된 색상 영상 (b) 방향 보간 방법을 이용한 홀 채움 결과

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 6과 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_disp와 B_disp 가 전경과 배경의 변이 값이고 th는 양수의 문턱값이라고 할 때, 식(1)의 값이 음수인 경우 좌영상에서 서로 인접하다고 판단하고 나머지 경우에는 좌영상에서 서로 떨어져 존재하는 화소라고 판단한다.

$$A = hole \ width - ((F \ disp - B \ disp) + th)$$
 (1)

식 (1)이 음수인 홀 영역은 방향 보간 방법으로 잘 채워지기 때문에 방향 보간 방법을 수행하게 되고. 그림 7과 같이 점선으로 표시된 영역은 식 (1)의 값이 음수가아닌 값이다. 이러한 영역은 홀을 채울 때 홀의 우측 끝화소의 위치를 좌영상에서 찾고, 좌영상에서 좌측 방향으로 존재하는 화소값들을 복사하여 홀을 채운다.



그림 7. 제안하는 보간 방법

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 깊이영상의 값들을 분석하여 전경이 일부 제거된 배경영상을 얻는다.

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

$$BG_{-}I(x,y) = I(x,y,t')$$
(2)

여기서

$$t' = \arg\min_{t} Depth(x, y, t)$$
 (3)

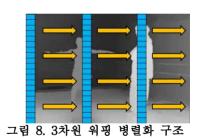
식 (2)에서 I의 홀 영역은 색상값이 없기 때문에 배경 영상을 획득할 수 없다. 결국 배경 영상 전체가 유효한 색상값을 갖기 힘들다. 이런 이유로 홀 영역을 채우는 과 정에서 배경 영상이 존재하는 부분에 한정하여 배경 영상 을 사용한다. 배경 영상을 사용하지 못하는 부분에 대해 서는 공간적 홀 채움 방법을 사용하여 홀을 채우게 된다.

3. GPU를 이용한 병렬처리 기법

그래픽 처리 연산장치(GPU)는 중앙 처리 연산장치(CPU) 와는 다르게 내부에 연상장치가 많다. 따라서 각각의 연산장치를 동시에 활용하여 병렬처리를 이용한 고속화에 유리한 구조를 갖고 있다. 이러한 GPU를 그래픽 작업이 아닌 일반적인 용도에 사용하기 위해 CUDA를 통하여 병렬화된 가상시점 영상 생성 알고리즘을 구현했다.

CUDA는 병렬화를 위한 단일 명령 복수 데이터(single instruction multiple data) 구조를 가지고 있어 동일한 명령을 다수의 데이터에 동시에 적용시키는 것이 가능하다[7]. 이 구조를 활용하기 위해서는 데이터 사이에 종속성이 없어야 하는데, 3차원 전방 워핑을 병렬화하는 경우둘 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.



뿐만 아니라, 실시간 시스템의 GPU 성능을 최적화하기 위해서는 용도에 맞는 메모리의 사용이 매우 중요하다. GPU에는 전역 메모리, 공유 메모리, 레지스터 메모리, 텍스쳐 메모리 등 몇 가지 다른 특성을 갖는 메모리가 있는데, 최적화된 속도를 내기 위해서 자주 참조되는 영상은 읽기전용인 텍스처 메모리에 위치시켜 사용했다. 텍스처

참조하는 속도가 다른 메모리에 비하여 빠른 장점이 있어 자주 사용하는 호모그라피 행렬과 참조만 하는 영상을 텍 스처 메모리에 위치시켜 고속화했다.

메모리의 경우 부분적인 수정은 불가능하다. 하지만 값을

Ⅳ. 실험 결과

이 논문에서 제안한 방법의 성능을 평가하려고 Microsoft Research에서 제공하는 720x480 크기의 Ballet 영상을 사용하였다. 그림 9는 기존의 실시간 홀 채움에 사용된 방법과 새롭게 제안하는 홀 채움의 결과를 비교한 것이다. 그림 9(a)는 방향 보간 홀 채움의 결과 영상으로 굽어진 손의 안쪽 부분에서 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 확인했다.





(a) 방향 보간 홀 채움 결과

(b) 제안1의 홀 채움 결과



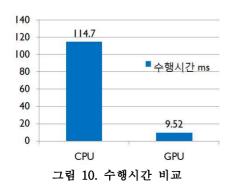


(c) 제안1의 홀 채움 결과

(d) 제안 1,2의 홀 채움 격과

그림 9. 제안하는 홀 채움 결과

실시간 양안식 영상 생성을 위해 병렬처리를 지원하는 GPU를 사용했다. 그림 10은 동일한 알고리즘에 대해 CPU와 GPU에서의 수행 시간을 비교한 것인데, GPU를 사용하여 수행 속도를 15배 이상 빠르게 했다. 그 결과보다 정확하게 홀 채움을 수행하면서도 720x480 크기를 갖는 그림 11(a)와 해당 깊이맵을 이용하여 그림 11(b)를 초당 100장 정도를 실시간으로 생성할 수 있다.





(a) 좌영상 (b) 생성된 우역 그림 11. 입력 좌영상과 생성된 우영상

Ⅴ. 결론

본 논문에서는 좌영상과 깊이맵을 이용하여 우시점 영상을 만들 때 효율적으로 홀을 채우는 기법과 GPU상에서 DIBR 전 과정을 실시간으로 처리하는 방법을 제안했다. 홀 영역 좌우 끝부분의 깊이값을 통해 이 영역이 좌영상에서 서로 인접하는지 판단했고, 확인된 인접유무 정보를 이용하여 홀을 채우는 방법을 선택하였다. 3차원 워핑 및홀 채움의 모든 과정은 GPU상에서 병렬로 처리되도록구현하였다. 그 결과 효과적으로 홀 채움을 수행하면서 동일 알고리즘에 대해 CPU 기반의 처리 속도보다 15배이상 빠른 속도를 얻을 수 있었다.

감사의 글

본 연구는 지식경제부 및 정보통신산업진흥원의 대학 IT연구센터 지원 사업의 연구 결과로 수행되었습니다. (NIPA-2010- (C1090-1011-0003))

참고문헌

- [1] O. Schreer, P. Kauff, and T.Sikora, *3D Videocommunication*, John Wiley&Sons, 2005.
- [2] L. McMilan, "A List-Priority Rendering Algorithm for Redisplaying Projected Surfaces," Technical Report TR95-005, Univ. of North Carolina, July 1995.
- [3] G. Wolberg, Digital Image Warping, IEEE Computer Society Press, March 1990.
- [4] ISO/IEC JTC1/SC29/WG11, Contribution for 3D Video Test Material of Outdoor Scene, M1537, July 2008.
- [5] 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.
- [6] 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.
- [7] nVIDIA Corporation, "CUDA 2.3 Programming Guide," in http://www.nvidia.com/object/cuda_develop.html, 2009.