

영상기반 재조명을 위한 GPU 기반 라디오시티 구현

Implementation of Progressive Radiosity on GPU for Image based Relighting

김준환, Jun Hwan Kim, 황용호, Yong Ho Hwang, 홍현기, Hyun Ki Hong

중앙대학교 첨단영상대학원 영상공학과

요약 전역조명기법(global illumination)중에서 난반사(diffuse reflection) 객체들 사이의 관계를 효과적으로 표현하는 라디오시티(radiosity)방법은 객체들 사이의 에너지 교환에 에너지 평형 상태를 모델링 한다. 그러나 라디오시티는 많은 계산량으로 인해 실시간 활용에는 적합하지 않았다. 최근 장면생성에 걸리는 소요시간을 크게 단축시킬 수 있는 비용대비 고성능의 그래픽스 하드웨어(GPU)를 이용한 방법들이 제안되고 있다. 객체들 사이에서 교환되는 에너지는 라디언스(radiance)로 표현이 가능하며, 이러한 라디언스는 대상 장면에서 취득한 HDR(High Dynamic Range) 영상으로부터 라디언스 맵을 구성해서 얻을 수 있다. 이를 기반으로 대상장면의 조명환경을 구성하면 대상장면의 복잡도와는 별개로 빠르고 사실적인 합성장면을 생성할 수 있다. 본 논문에서는 G. Coombe 등이 제안한 점진적 세분(progressive refinement) 알고리즘을 수정하여 라디언스 맵을 이용할 수 있도록 하였으며, 각 텍셀(texel)설정 및 보간(interpolation) 적용 등에 따른 실험 결과를 얻고 분석하였다. 구현된 방법은 이후 영상기반 재조명과 그래픽스 하드웨어를 이용한 영상합성 기술로 영화, 애니메이션, 가상현실, 게임 등에 다양하게 활용될 예정이다.

핵심어: Global Illumination, Radiosity, Relighting, GPU, HDR Radiance Map

1. 서론

컴퓨터 그래픽스(computer Graphics) 분야에서 사실적인(photo realistic) 장면의 표현은 매우 중요한 연구 주제 중 하나이다. 3차원 공간의 현실감을 극대화 할 수 있는 처리 방법 중 전역조명(global illumination)모델은 객체 간에 발생하는 빛의 다양한 상호작용을 모델링 할 수 있으며, 이를 통해 매우 사실적인 장면을 표현할 수 있다[1]. 그러나 많은 데이터와 처리시간을 필요로 하기 때문에 전역조명 모델은 실시간 렌더링(rendering)에 활용되지 않았다. 현재 이러한 제약을 해결하기 위한 방법들이 활발히 연구 되고 있다.

최근 몇 년간 컴퓨터 산업의 성장에 따라 그래픽스 하드웨어(GPU)는 중앙처리장치(CPU)보다 높은 부동소수점 연산 능력과 단일 명령 복수 데이터 연산 방식(SIMD)구조로 물리적인 현상에 기반한 전역 조명기법의 많은 데이터의 긴 처리시간을 적은 비용으로 보다 빠른 처리가 가능해졌다[2]. 또한 영상기반(image based) 그래픽스 기술은 영상을 입력으로 하기 때문에 대상 장면의 복잡도와 별개로 적은 비용으로 빠르고 사실성 높은 장면을 표현할 수 있다. 이러한 영상기반 그래픽스 기술은 실제 영상으로부터 조명 정보를 추출하여 이를 가상 객체에 적용할 수 있

으며 실제영상에 새로운 조명을 표현 할 수 있다[3].

본 논문에서는 기존의 G. Coombe[4] 등이 제안한 그래픽스 하드웨어에 기반한 점진적 세분 라디오시티(progressive refinement radiosity) 방법을 수정 및 보완하여 HDR(High Dynamic Range) 라디언스 맵을 기반으로 한 라디오시티 방법을 구현하였다. 기존의 중앙처리장치에 기반한 점진적 세분 방법은 난반사 객체(diffuse reflector)로 구성된 환경에서 전역조명 효과를 이전의 라디오시티 방법에 비해 비교적 빠른 시간에 표현할 수 있는 방법이다. 그래픽스 하드웨어를 적극적으로 사용함으로써 중앙처리 장치를 이용하는 기존의 방법보다 비교적 빠른 렌더링 결과를 얻을 수 있다. 또한 HDR 라디언스 정보를 이용하여 임의의 물체를 합성하기 위해 필요한 전체 렌더링 시간이 크게 단축된다.

2장에서는 기존의 라디오시티 방법에 대해서 간단히 소개하고 그래픽스 하드웨어에서의 라디오시티와 영상기반 재조명(relighting)에 관한 연구를 소개하며, 3장에서는 HDR 라디언스 맵과 GPU를 이용한 점진적 세분 라디오 시티 방법의 구현에 대해서 설명하고 4장에서는 구현된 알고리즘으로부터 얻어진 시뮬레이션 결과를 분석하고 5장에서 결론을 맺는다.

2. 기존연구

2.1 라디오시티 [5]

정적인 환경에서 에너지 복사로 인하여 환경안의 에너지 상태는 결국 에너지 평형 상태에 이른다. 열복사는 가시광선 영역을 포함하고 있으며 이때 단위 시간당, 단위 면적당 임의의 한 면에서 방출되는 복사 에너지를 라디오 시티라 한다.

라디오시티 방법은 에너지를 방사하는 모든 다른 면으로부터 각 면에 도달하는 에너지의 관계를 고려하여 전역 조명을 모델링하며, 색의 섞임과 부드러운 그림자 등을 표현할 수 있는 특징을 가진다. 한 면에서 나오는 빛은 방출하는 빛과 다른 객체로부터 들어오는 빛으로 구성되고 다음 수식 (1)과 같이 표현된다.

$$B_i A_i = E_i A_i + \rho_i \sum_{j=1}^n B_j A_j F_{ji} \quad (1)$$

여기서,

B_i = 면 i 의 라디오시티

A_i = 면 i 의 면적

E_i = 면 i 의 단위면적당 방출하는 에너지

ρ_i = 면 i 의 단위면적당 방출하는 에너지

ρ_i = 면 i 의 반사도

B_j = 면 j 의 면적

F_{ji} = 면 j 에서 면 i 로의 폼팩터(form-factor)

폼팩터는 면 j 를 떠나 면 i 에 도달하는 에너지의 비율을 나타낸다. 식(1)의 방정식은 장면의 모든 면에 대해 존재하고 이는 행렬의 형태로 표현된다.

기존의 라디오시티 방법은 행렬형태의 연립 방정식을 풀어 각 면의 라디오시티를 구했지만, 이러한 과정은 많은 메모리와 시간이 소요된다. 이러한 단점을 보완하기 위해 점진적 세분 방법은 에너지가 높은 면으로부터 그 다음 순서대로 다른 면이 방출하는 에너지를 고려하는 방식으로 라디오시티 방정식을 풀어 렌더링 시간을 단축시켰다.

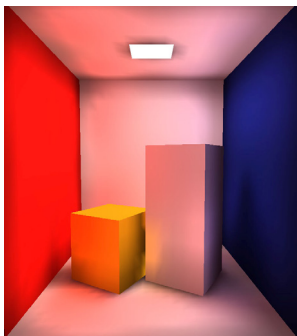


그림 1. CPU를 이용한 점진적 세분 라디오시티 방법

2.2 그래픽스 하드웨어를 이용한 라디오시티[4]

앞 절에서 언급했던 바와 같이 기존의 중앙처리장치를 이용한 점진적 세분 방법은 유한한 대상장면을 일정한 면들로 분할하고 다른 면들과의 관계를 바탕으로 해당 면에서의 라디오시티를 계산한다. 그래픽스 하드웨어를 이용한 점진적 세분 방법은 기존의 방법과는 달리 각 면에 해당하는 라디오시티 텍스처(texture)와 잔여(residual)에너지 텍스처를 생성하여 각 텍스처를 구성하는 단위인 텍셀(texel)을 기본요소로 라디오시티를 계산한다.

그래픽스 하드웨어를 이용한 점진적 세분 방법에서는 에너지 방사 면으로부터 에너지를 받는 텍셀간의 폼팩터 결정 과정에 많은 계산량을 필요로 한다. 이는 그래픽스 하드웨어의 단일 명령 복사 데이터 연산 방식(SIMD)구조에 의하여 병렬 처리되기 때문에 중앙처리장치를 이용하는 것보다 렌더링 속도가 크게 단축된다. 표1은 그래픽스 하드웨어를 이용한 점진적 세분 라디오 시티의 의사 코드이다.

표 1. 점진적 세분 라디오시티 on GPU 의사코드

```

Initialize shooter residual E
While not converged{
  Render scene from POV of shooter
  For each receiving polygon{
    For all texel{
      If texel is visible {
        Compute form factor FF
        E = q * FF * E
        Add E to residual Texture' texel
        Add E to Radiosity Texture's texel
      }
    }
  }
}
Shooter's residual E = 0;
  
```

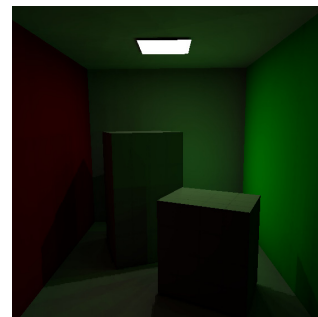


그림 2. GPU를 이용한 점진적 세분 라디오시티 방법

2.3 영상기반 재조명

가상 물체와 실제 영상을 효과적으로 합성하기 위해 Fournier 등은 대상 장면에 대한 3차원 모델과 라디오시티 렌더링을 이용하는 방법을 제안하였다[6]. 이 방법은 물체 주변에 있는 간접조명의 효과를 표현할 수 있는 장점이 있다.

Drettakis 등은 영상에서 다면체를 이루는 정점(vertex) 등을 사용자가 선택하고 컴퓨터 비전 기술로 3차원 공간을 재구성하는 방법을 제안하였다[7]. 또한 모자이크(mosaic)

기술로 파노라마 영상을 만들어서 입력영상의 시야를 넓혀 조명의 위치를 입력하는데 소요되는 시간을 줄였다.

Debevec 등은 노출이 다른 여러 장의 사진으로부터 이미징 시스템의 응답함수를 복원하고 대상 공간의 라디언스 맵을 구성한 후 실제 세계의 조명 환경을 정확하게 분석할 수 있는 방법을 제안하였다. 그리고 측정된 라디언스 값으로부터 전역 조명 모델을 사용하여 대상 영상에 가상 물체를 합성하였다[3]. 이 방법은 고정된 위치에서 금속 구(mirror ball)상에 반사된 조명 정보를 임의의 3차원 공간에 매핑하고, 조명기반으로 모델링 된 공간을 구현하였다. 대상 장면을 먼(distant) 장면, 지역(local) 장면, 합성 물체 등으로 구분해서 전역 조명을 적용함으로써 사실적인 렌더링 영상을 표현하였다.

3. 구현된 라디오시티

본 논문에서 구현에 사용된 라디언스 맵은 비어있는 코넬상자(cornell box)를 그래픽스 하드웨어를 이용한 일반적인 점진적 세분 방법을 이용하여 렌더링한 다음, 각 면을 HDR 라디언스 맵으로 변환하여 취득하였다. 그림 3의 라디언스맵으로 환경을 구성한다

2장에서 언급한 바와 같이 라디언스는 단위 시간당, 단위 미소 입체각(solid angle)당, 그 면의 단위 법선당 방출되는 에너지의 방향적 분포를 의미한다. 구성된 공간을 구성하는 면들은 전부 난 반사체(diffuse)라고 가정하기 때문에 라디언스는 라디오시티와 같다. HDR 라디언스 이미지로 텍스처 맵핑된 임의의 공간 상에서 가상의 CG(computer graphics)객체를 합성하기 위해서는 다음과 같은 라디오시티의 재분배과정이 필요하다.

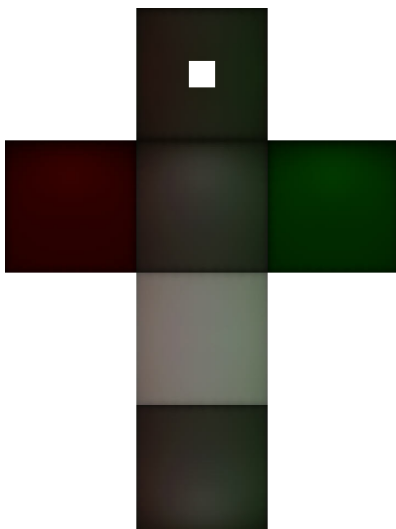


그림 3. HDR 라디언스 맵

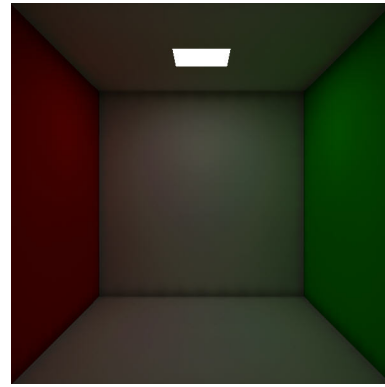


그림 4. HDR 라디언스 맵으로 구성된 환경

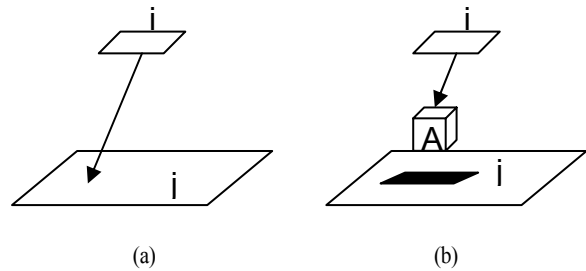


그림 5. 인접한 두 면간의 방사관계

그림 5(a)에서 객체가 없을 경우 라디오시티는 각면의 에너지 방사과정을 통해 에너지 평형을 이루게 된다. 그림 5(b)경우와 같이 가상의 객체가 추가되면 에너지 방사과정에서 라디오시티는 가상 객체의 표면과의 관계의해 계산된다. 이 경우 동일한 양만큼의 라디오시티를 면 i에서 가상 객체 A에 의해 가려진 텍셀들의 라디오시티에서 감소시키면 에너지 평형상태에 도달한다. HDR 라디언스 맵을 사용할 수 있도록 G. Coombe[4]가 제안한 라디오시티 방법을 HDR 라디언스 맵을 사용할 수 있도록 수정하였으며 의사 코드를 표 2에 나타내었다. 알고리즘을 단계별로 살펴보면 다음 표 2와 같다.

표 2. 수정된 점진적 세분 라디오시티 알고리즘

```

Search the max energy polygon
Render scene from POV of shooter
For each receiving polygon{
  For all texel{
    If texel is radiance map's{
      If texel is not visible {
        Compute form factor FF
        E = q * FF * E
        Opp E to Radiosity Texture's texel
      }
    }
    Else texel is CG Object's map's{
      If texel is visible {
        Compute form factor FF
        E = q * FF * E
        Add E to residual Texture' texel
        Add E to Radiosity Texture's texel
      }
    }
  }
}
Shooter's residual E = 0;

```

3.1 에너지 방출 및 가시성(visibility) 결정

점진적 래디오시티 방법은 높은 에너지를 가지는 면부터 에너지를 방출하며, 각 표면에서 공간 내 에너지를 받을 가능성이 있는 모든 면과의 관계를 구한다. 에너지를 방출하는 면과 에너지를 받는 면 사이의 관계는 Wallace[8]가 제안한 수식 (2)를 사용한다.

$$F_{i \rightarrow \Delta j} = \Delta j \frac{\cos \theta_i \cos \theta_j}{\pi r^2 + \Delta j} \quad (2)$$

장면을 구성하는 텍셀의 가시성 여부를 결정하기 위해 반 육면체(hemicube)사영이 아닌 반구(hemisphere)사영[9]을 이용한다. 반 육면체의 경우 5면에 대하여 고려해야 하지만 반구사영 방법은 한번의 고려로 장면을 구성하는 모든 표면의 가시성을 결정할 수 있다.

표 3. 반구 사영을 위한 의사 코드

```

Out HemiShpereproj(v = vertex, M = modelview,
                  n = near, f = far)
    v = M * v;
    vn = normalize(v);
    out.x = vn.x;
    out.y = vn.y;
    out.z = (-2*v.z*f-n)/(f-n);
    
```

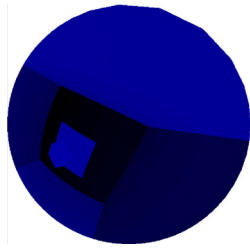


그림 6. 반구사형

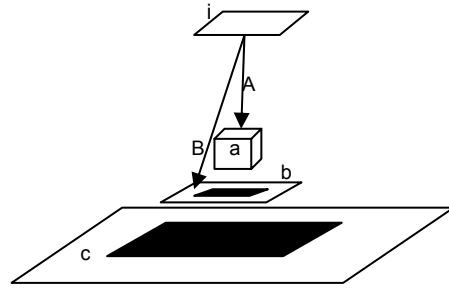
에너지를 방출하는 면에서 바라본 대상 장면을 표 3의 의사코드를 이용하여 렌더링한다. 이때 각 면은 고유의 r,g,b로 구성된 ID를 가지며, 이 ID는 각 면에 색상으로 표현된다. 이러한 과정은 정점 프로그램(vertex program)에서 구현되고 그래픽스 하드웨어의 정점처리장치에서 처리된다. 그림 6과 같이 렌더링된 장면은 아이템 버퍼(item buffer) 텍스처에 저장된다.

에너지를 방출할 때 에너지를 받을 수 있는 후보 면의 텍셀을 반구로 역사영(back projection)하여 텍셀을 포함하는 후보 면의 ID와 역사영 된 위치의 색 비교를 통하여 해당 텍셀의 가시성을 결정한다.

3.2 그림자 생성

그림자를 생성하기 위해서는 임의의 면으로부터 방출

되는 에너지가 가려진다고 판단된 텍셀에 대해 해당면이 래디언스 맵인지 CG객체의 텍스처 맵인지를 구분해야 한다.



i	에너지 방출 면
A,B	면에서 방출된 에너지
a,b	CG객체 텍스처 맵
c	HDR 래디언스 맵

그림 7. HDR 래디언스 맵과 가상 CG객체의 텍스처맵과의 관계

그림 7에서 A의 경우 CG 객체에 가려져 b영역에 에너지가 도달하지 못하며, 이때 가려진 b는 영상합성을 위한 CG 객체의 텍스처이므로 에너지를 전달하지 않는다. B도 CG 객체에 가려져 c에 에너지가 도달하지 못하지만 A의 경우와 다르게 c는 래디언스 맵이다. 이 경우에는 이미 래디오시티가 결정된 이미지이므로 면 i에서 면 c의 가려진 텍셀 간의 관계를 계산하여 취득한 래디오시티 값만큼 텍셀의 래디오시티 값을 감소시키면 에너지 평형상태가 유지되고 자연스러운 그림자를 얻을 수 있다.

3.3 이후 방출 면 결정

에너지가 가장 큰 다음의 방출 면을 결정하기 위해 장면내의 모든 각 면에 해당하는 잔여 에너지의 mip맵(Mipmap) 생성하고 깊이 버퍼(depth buffer)를 비교한다.

먼저, 한 픽셀(pixel)의 프레임 버퍼(frame buffer)를 생성하고 모든 각 면의 잔여 에너지 텍스처의 최상위 mip맵을 생성한다. 최상위 mip맵은 1X1 텍셀이고 이 색상 값은 텍스처의 평균 색상 값이다. 이는 한 면의 전체 텍셀들의 잔여 에너지 평균이다. 한 면에 존재하는 래디오시티의 합은 잔여에너지 평균에 해당 텍스처의 해상도를 곱하여 어더지며, 총 래디오시티는 그 면의 넓이를 곱하여 계산된다.

각 면에 해당하는 총 잔여 래디오시티의 역수를 한 픽셀의 깊이값으로 지정하고 픽셀의 색상 값은 각 면의 ID로 할당되어 한 픽셀 프레임버퍼에 렌더링된다. 이때 픽셀의 색상값으로 지정된 ID가 일치하면 해당 ID 면이 에너지 방출면으로 고려되어 래디오시티 계산이 진행되며, 더 이상 참조되지 않을 경우에는 에너지 평형상태에 도달한 것으로 결정한다.

4. 실험결과 및 분석

본 논문에서는 일정한 정육면체에 HDR 라디언스 이미지를 텍스처 맵핑하여 임의의 공간을 구성했다. Pentium zeon 3.2GHz와 Nvidia Quadro 4500, 4GB, Window XP 환경에서 Cg쉐이딩(shading) 언어[10]를 사용하여 실험이 진행되었다.

총 면수는 168개이며 CG객체의 면수는 12개를 이용했다. 또한 면당 텍스처의 해상도는 16X16, 32X32이며, 아이템 텍스처의 해상도는 256X256, 512X512, 1024 X1024로 다양하게 실험하였다.

위 실험환경에서 라디언스 이미지를 이용하지 않은 그래픽스 하드웨어에 기반한 점진적 세분 라디오시티방법을 사용하여 얻어진 렌더링 결과를 기준(reference)영상으로 하고, 라디언스 이미지를 이용한 기반 라디오시티 방법으로 렌더링된 결과를 비교한다. 또한 이미지를 이용하는 경우, 장면을 구성하는 표면당 텍스처의 해상도를 다양하게 설정하여 렌더링된 결과와 각 렌더링 시간을 표 4에 나타내었다.

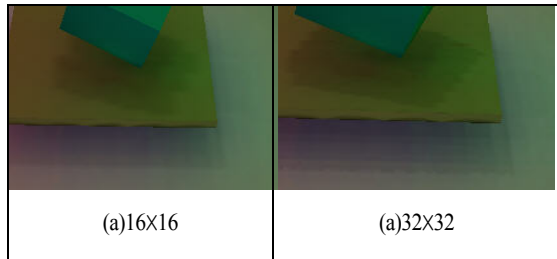


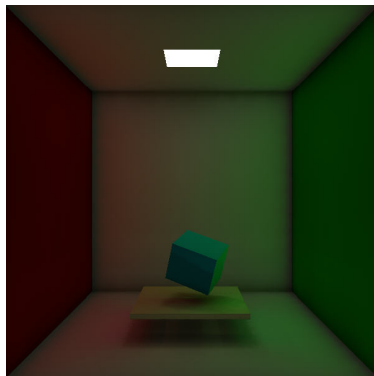
그림 8. 텍스처 해상도에 따른 그림자의 변화

HDR 라디언스 맵을 이용하여 렌더링했을 경우 그래픽스 하드웨어방법만을 이용한 결과보다 빠른 시간내에 최종 영상이 얻어지는 것을 표 4에서 확인하였다. 또한 텍스처의 해상도가 높아지면 에너지의 방사 횟수도 증가한다. 16X16, 32X32 해상도를 가진 텍스처들이 포함하고 있는 텍셀들은 폴리곤 계산에 직접 관여하기 때문에 텍셀수가 늘어날수록 렌더링에 소요되는 시간은 길어진다. 그러나 텍스처의 텍셀수가 증가할수록 최종 합성된 이미지의 질이 향상됨을 그림 8에서 확인하였다. 그리고 그림 9의 상위 3개의 렌더링된 영상에서 아이템 버퍼 텍스처가 작으면 각 면의 텍스처 주변부에 라디오시티가 계산되지 않음을 볼 수 있다.

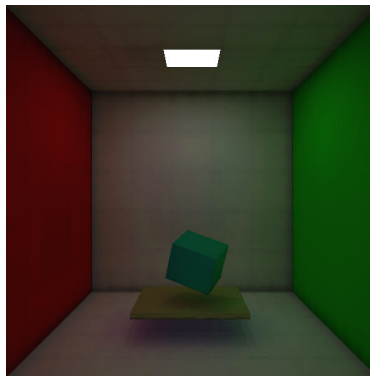
표 4. 텍스처 해상도에 따른 렌더링 과정 비교

	텍스처 해상도(텍셀)	렌더링 시간(초)	아이템버퍼 텍스처해상도(텍셀)	총 방사횟수
기준 영상	16X16	34	1024X1024	322
	32X32	41	1024X1024	334
합성 이미지	16X16	11	256X256	182
		13	512X512	182
	32X32	18	512X512	185
		24	1024X1024	185

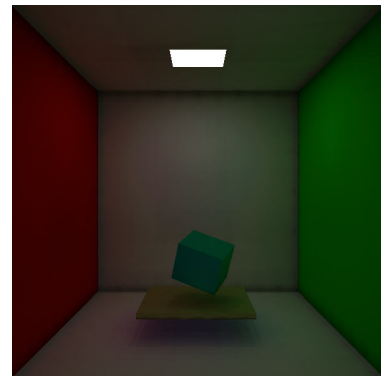
16X16텍스처



기준영상

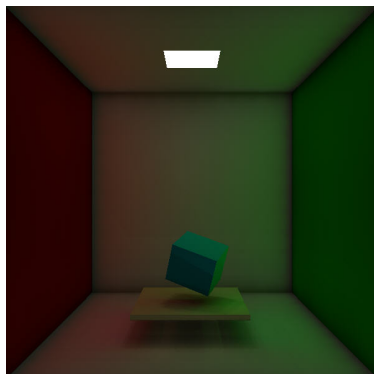


합성이미지(아이템버퍼 텍스처:256X256)

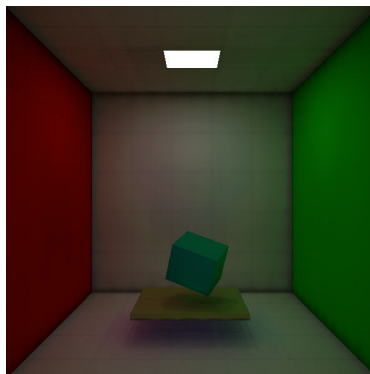


합성이미지(아이템버퍼 텍스처:512X512)

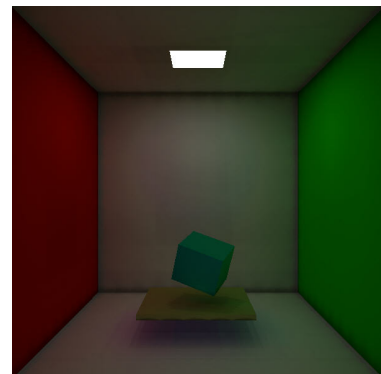
32X32텍스처



기준영상



합성이미지(아이템버퍼 텍스처:512X512)



합성이미지(아이템버퍼 텍스처:1024X1024)

그림 9. 결과영상



그림 10. 실제영상과 CG객체 합성

5. 결론

본 논문에서는 그래픽스 하드웨어와 HDR 레디언스 맵을 이용하여 취득된 영상과 CG객체의 합성을 구현하였으며, 대상 장면의 텍셀의 구성과 계산속도를 그래픽스 하드웨어를 이용한 래디언스 방법과 비교 및 분석하였다. 이후 연구에서는 적응적 분할방법을 구현하고 다양한 취득영상에서의 정반사 효과 등도 함께 고려할 예정이다.

감사의 글

본 연구는 서울시 산학연협력사업으로 구축된 서울 미래콘텐츠컨버전스 클러스터 지원으로 수행되었습니다.

참고문헌

- [1] M. F. Cohen, and J. R. Wallace, "Radiosity And Realistic Image Synthesis," Cambridge, MA, Academic Press, 1993.
- [2] M. Pharr and R. Fernando. GPU Gems 2: Programming Techniques for High-Performance Graphics and General Purpose Computation, Addison Wesley Professional, 2005.
- [3] P. Debevec, "Rendering Synthetic Objects into Real Scenes: Bridging Traditional and Image-Based Graphics with Global Illumination and High Dynamic Range Photography," Proceedings of SIGGRAPH pp. 189-198, 1998.
- [4] G. Coombe, M. J. Harris, and A. Lastra, "Radiosity on Graphics Hardware," Proceedings of Graphics Interface, pp.161-168, 2004.
- [5] M. F. Cohen, S. E. Chen, J. R. Wallace, and D. P. Greenberg, "A Progressive Refinement Approach to Fast Radiosity Image Generation," Proceedings of SIGGRAPH, pp.45-84, 1988.
- [6] A. Fournier, A. Gunawan, and C. Romanzin, "Common illumination between real and computer generated scenes," Proceedings of Graphics Interface, pp. 254-262, 1993.

- [7] G. Drettakis, L. Robert, and S. Bounoux, "Interactive common illumination for computer augmented reality," Proceedings of the 8th Eurographics Workshop on Rendering, pp. 45-57, 1997.
- [8] J. R. Wallace, K. A. Elmquist, and E. A. Haines. "A ray tracing algorithm for progressive radiosity," Proceedings of SIGGRAPH, 315-324, 1989.
- [9] W. Heidrich and H. P. Seidel. "View Independent Environment maps," In Workshop on Graphics Hardware, 1998.
- [10] W. R. Mark, R. S. Glanville, K. Akeley, and M. J. Kilgard, "Cg: A system for programming graphics hardware in a c-like language," Proceedings of SIGGRAPH, 2003.