

전역 조명 알고리즘에서의 조화 평균 거리 계산의 분석

(Analysis of Harmonic Mean Distance Calculation
in Global Illumination Algorithms)

차득현[†] 임인성^{††}

(Deukhyun Cha) (Insung Ihm)

요약 전역 조명(global illumination) 효과를 사실적으로 렌더링하기 위해서는 복잡한 경로를 통해 입사하는 빛의 정보 해당하는 직접 조명 및 간접 조명을 정확하게 계산해주어야 한다. 이 과정에서 주어진 물체 표면 지점에 대해 계산되는 간접 조명 정보는 주변 환경의 기하적인 형태에 큰 영향을 받게 된다. 조화 평균 거리(harmonic mean distance)는 3차원 공간상에서 주어진 한 지점에서 보이는 물체들과의 거리를 나타내는 척도로 많이 사용되는 수학적 도구로서, 광휘 캐시(irradiance/radiance cache)나 환경 페색(ambient occlusion) 등의 렌더링 효과를 생성하는데 주요하게 사용된다. 본 논문에서는 대표적인 고품질 전역 조명 렌더링 알고리즘인 최종 수집(final gathering) 방법 및 포톤 맵(photon mapping) 기법을 통해 다양한 환경에서 계산되는 조화 평균 거리에 대한 근사값의 정확성에 대해 분석한다. 이러한 분석 결과를 기반으로 효과적인 조화 평균 거리 계산의 근사화 기법 개발에 있어서 고려해야 할 점들과 방향을 제시한다.

키워드 : 전역 조명, 조화 평균 거리, 최종 수집, 큐브 맵, 포톤 맵

Abstract In order to render global illumination realistically, we need to accurately compute the direct and indirect illumination that represents the light information incoming through complex light paths. In this process, the indirect illumination at given point is greatly affected by surrounding geometries. Harmonic mean distance is a mathematical tool which is often used as a metric indicating the distance from a surface point to its visible objects in 3D space, and plays a key role in such advanced global illumination algorithms as irradiance/radiance caching and ambient occlusion. In this paper, we analyze the accuracy of harmonic mean distance estimated against various environments in the final gathering and photon mapping methods. Based on the experimental results, we discuss our experiences and future directions that may help develop an effective harmonic mean distance computation method in the future.

Key words : global illumination, harmonic mean distance, final gathering, cube-map, photon mapping

1. 서론

· 이 논문은 2008년도 정부(교육기술과학부)의 재원으로 한국학술진흥재단의 지원을 받아 수행된 연구임(과제번호: KRF-2008-313-D00920)

† 학생회원 : 서강대학교 컴퓨터공학과

seaboy7@sogang.ac.kr

†† 종신회원 : 서강대학교 컴퓨터공학과 교수

ihm@sogang.ac.kr

논문접수 : 2009년 11월 25일

심사완료 : 2009년 12월 22일

Copyright©2010 한국정보과학회: 개인 목적이나 교육 목적인 경우, 이 저작물의 전체 또는 일부에 대한 복사본 혹은 디지털 사본의 제작을 허가합니다. 이 때, 사본은 상업적 수단으로 사용할 수 없으며 첫 페이지에 본 문구와 출처를 반드시 명시해야 합니다. 이 외의 목적으로 복제, 배포, 출판, 전송 등 모든 유형의 사용행위를 하는 경우에 대하여는 사전에 허가를 얻고 비용을 지불해야 합니다.

정보과학회논문지: 컴퓨팅의 실제 및 레터 제16권 제2호(2010.2)

사실적인 전역 조명 효과의 렌더링은 3차원 공간의 한 지점에서 복잡한 경로를 통해 입사하는 빛의 정보를 계산하여야 하는 문제로 1986년 Kajiya에 의해 발표된 식 (1)의 렌더링 방정식에 잘 나타나 있다[1]. 입사하는 빛의 정보는 광원의 위치 및 밝기, 광원의 개수, 주변 물체들의 위치 및 물체 표면의 성질 등 장면을 구성하는 여러 요소들에 영향을 받아 결정되게 된다. 특히 현재 전역 조명을 계산하고자 하는 지점 주변에 물체들이 어떻게 분포되어 있는가에 따라 빛의 입사하는 정도나 성질이 달라지는데 이러한 분포를 표현해 줄 수 있는 방법 중 하나가 반구 방향으로 보이는 물체들과의 거리

의 평균을 계산하는 조화 평균 거리이다.

$$L_o(x, \vec{\omega}_o) = L_e(x, \vec{\omega}_o) + \int_{\Omega} f_r(x, \vec{\omega}_i, \vec{\omega}_o) L_i(x, \vec{\omega}_i) (\vec{\omega}_i \cdot \vec{n}) d\omega_i \quad (1)$$

고품질 전역 조명 효과를 계산하기 위한 다양한 연구들이 진행되었는데 광선 추적법(ray tracing)[2], 분산 광선 추적법(distributed ray tracing)[3], 래디오시티(radiosity)[4,5], 포톤 매핑(photon mapping)[6,7], 최종 수집(final gathering)[8] 등의 대표적인 전역 조명 기법들이 제안되었다. 이를 기법들 중 특히 고품질 전역 조명 효과를 생성할 수 있는 방법으로 포톤 매핑 기법과 최종 수집 기법이 널리 사용되고 있는데 이 두 방법 모두 광선 추적법에 기반을 둔 방법이라는 것에서 같지만 포톤 매핑 기법은 광원으로부터 여러 번 반사되어 입사하는 빛의 정보를 확률적으로 시뮬레이션하고 이 정보를 밀도 추정 기법(density estimation)을 사용하여 근사화 한다는 점에서, 최종 수집 기법은 주어진 지점에 입사하기 직전의 빛의 정보를 샘플링된 광선의 추적을 통해 정확하게 계산하고자 하는 방법이라는 점에서 차이가 있다. 때문에 포톤 매핑 기법과 최종 수집 기법에 대한 결과를 분석한다는 것은 확률적인 계산 및 영역 정보의 근사화 결과와 직접적인 샘플링 계산을 분석하는 의미를 가지게 된다. 조화 평균 거리의 계산은 최종 수집 기법과 같이 직관적인 샘플링 기법을 통해서 계산될 수 있으며 이를 가속화하기 위해서는 샘플링 속도의 향상이나 샘플링 개수를 최소화 혹은 샘플링을 선택적인 지점들에 대해 수행하여 전체적인 샘플링 개수를 최소화 하는 방법 등을 사용할 수 있다. 또한 확률적인 방법 등을 통해 근사적인 정보를 빠르게 생성할 수 있다. 이러한 가속화된 기법을 개발하는데 있어서의 목표는 생성하고자 하는 정보의 품질을 유지하면서 최대의 성능을 도출할 수 있도록 하는 것이다.

본 논문에서는 널리 응용되는 전역 조명 기법으로서 직관적인 샘플링 기법인 최종 수집 기법과 확률적인 근사화 기법인 포톤 매핑 기법을 통해 기하 정보를 표현하는 중요한 척도로 사용되고 있는 조화 평균 거리를 계산하였다. 두 기법에서 성능과 정보의 정확성을 결정하는 요소들인 광선 샘플링 개수, 사용된 포톤의 개수, 밀도 추정 기법에서의 포톤 수집 개수 변화 등의 조건 값에 따른 결과를 생성하고 이를 분석함으로써 조화 평균 거리 계산이 필요하거나 새로운 조화 평균 거리 계산 기법을 개발하고자하는 경우에 주요하게 참조 될 수 있는 정보를 제공한다. 특히 CPU에서의 전통적인 광선 추적 기반 최종 수집 기법과 GPU를 통해 가속화 될 수 있는 큐브 맵 샘플링을 통한 최종 수집 기법을 분석 하며, 포톤 매핑 기법 기반의 조화 평균 거리 계산을

위해 사용 가능한 두가지 기법을 제시하고 결과를 분석 한다.

최근 활발하게 연구되고 있는 GPU 기반 전역 조명 기법들[9-12]에서도 조화 평균 거리 혹은 유사한 과정을 통해 계산 가능한 구체 조화(spherical harmonic) 값이 중요하게 사용되고 있으며 기존의 다른 기법들을 더욱 발전시키기 위해서도 중요하게 응용될 수 있는 정 보로 사용될 수 있다. 그러므로 널리 알려진 방법들을 사용하여 계산된 결과를 분석하는 것은 그 특성을 이해하고 최적화된 계산 기법을 설계하는 데에 있어서 의미 있는 작업이라고 할 수 있으며, 광휘 캐시[13,14]나 환경 폐색[15,16] 등의 알고리즘 성능을 향상하고 새로운 전역 조명 알고리즘을 개발하는 일에도 주요하다고 할 수 있다.

본 논문의 구성은 다음과 같다. 2장에서는 본 연구에서 사용하는 기법들에 대해 간략하게 설명하고 본 연구의 기여도를 보인다. 3장에서는 최종 수집 기법 및 포톤 매핑 기법을 통해 조화 평균 거리를 계산하는 방법을 설명하며 4장에서 이 방법들을 통해 계산된 결과들을 분석한다. 5장에서 결론 및 향후 연구 과제를 논의한다.

2. 연구 배경 및 기여도

2.1 조화 평균 거리

조화 평균은 소수의 큰 값에 의해 전체적인 평균값이 증가하는 것을 방지할 수 있는 수학적으로 정의된 평균 계산 방법의 하나로 다음의 식과 같이 계산된다(식 (2)).

$$H_{mean} = \frac{n}{\frac{1}{x_1} + \frac{1}{x_2} + \dots + \frac{1}{x_n}}, \quad x_i > 0 \text{ for all } i \quad (2)$$

조화 평균 거리는 3차원 공간의 한 지점에 대해 반구 위 모든 방향에 대한 주변 물체와의 거리를 계산하여 이를 조화 평균 한 값이다. 결국 현재 지점에 대하여 주변 물체들이 얼마나 가까이 있는지를 나타내 줄 수 있는 값이 되며 이는 간접 조명 계산에서 주변 물체들을 출발하는 광휘 정보에 얼마나 민감하게 반응할 것인지, 혹은 주변 물체에 의해 가려지는 빛이 얼마나 많을 것인지 등을 예측할 수 있는 척도로 기존의 전역 조명 기법들에서 빈번하게 사용되고 있다.

2.2 관련 전역 조명 렌더링 알고리즘

2.2.1 포톤 매핑 기법

포톤 매핑 기법은 난반사 표면에서의 간접 조명 효과를 효과적으로 표현할 수 있는 전역 조명 알고리즘으로서 포톤 맵을 생성하는 단계와 이를 가시화하는 단계로 이루어진다[6].

포톤 맵을 생성하는 단계에서는 광원을 샘플링하여 생성한 포톤(photon)을 추적하여 상호작용하는 난반사

물체 위 3차원 지점에서 포톤 정보를 저장한다. 이때 상호작용하는 포토니 물체 표면의 성질에 따라 여러 개로 나뉘어 추적되지 않고 확률적으로 하나의 성질을 선택하여 추적되는데 그 효율성이 있다.

가시화 단계에서는 직접 조명과 간접 정반사 조명을 효과적으로 생성해 주는 광선 추적 기법을 사용한다. 추적한 광선이 만나는 지점이 난반사 물체인 경우에 포톤 맵에 접근하여 해당 지점 주변의 가까운 포톤 정보를 검색하여 광휘를 계산함으로써 직접 조명과 간접 정반사 조명 그리고 간접 난반사 조명을 효과적으로 생성하게 된다. 그림 1은 포토니 추적되고 가시화되는 한 예를 보여주고 있다.

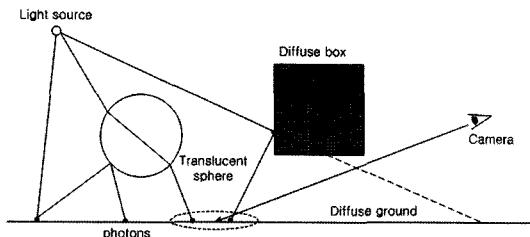


그림 1 포톤 매핑 알고리즘을 통한 전역 조명 렌더링 과정. 포토니를 추적하고 추적된 포토니들을 저장하여 포톤 맵을 생성하고 이를 가시화하는 과정에 대한 간략한 예를 보여주고 있다.

2.2.2 최종 수집 기법

최종 수집 기법은 전역 조명 효과를 계산하기 위해 반구위 방향으로부터 들어오는 빛을 추적하는 과정을 광선 추적 기법을 통하여 계산하는 직접적인 방법이다 [8]. 이때 정해진 개수의 수집 광선에 대하여 반구위 방향을 적절하게 샘플링하고 광선을 추적하게 되며, 추적된 광선이 물체와 만나는 지점의 광휘를 계산하여 해당 방향으로 입사하는 단일 반사(bounce) 광휘를 얻을 수

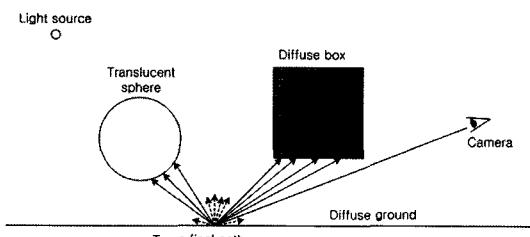


그림 2 최종 수집 알고리즘을 통한 전역 조명 렌더링 과정. 주어진 3차원 공간상의 한 지점에서 최종 수집 광선을 추적하여 반구위 방향으로 입사하는 광휘 정보를 계산하는 과정의 한 예를 보여주고 있다.

있다. 각 방향으로 추적한 광선을 통해 계산된 입사 광휘 정보가 시점 방향으로 어떻게 반사되는지를 계산하여 고품질 전역 조명 정보를 얻을 수 있다. 이때 추적된 광선이 물체와 만나는 지점에서의 광휘 정보가 포톤 매핑 기법 등을 통해 얻어진 다중 반사에 대한 것이라면 최종 수집 기법을 통해 고품질 다중 반사 전역 조명 (multiple bounce global illumination) 효과를 나타낼 수 있게 된다. 그럼 2에서는 주어진 3차원 지점에서의 최종 수집의 수행 예를 보여주고 있다.

2.2.3 GPU를 통한 최종 수집 기법 계산

최종 수집 기법은 광선의 추적이 아닌 3차원 물체를 2차원 평면에 투영하고 이를 샘플링 하는 기법을 통해 가속될 수 있는데 특히 큐브 맵의 다섯 면에 투영 이미지를 생성하고 이 큐브 맵을 샘플링 함으로써 빠르고 정확하게 최종 수집 기법을 근사화 할 수 있다[5]. 최근 빠르게 발전하고 있는 프로그래밍 가능한 그래픽스 하드웨어의 레스터화 연산을 사용하여 큐브 맵(cube-map)을 생성하고 이를 역시 하드웨어적으로 제공하는 큐브 맵 샘플링 연산을 사용하여 보다 빠르게 계산할 수 있다. 그림 3에서는 주어진 3차원 지점에서 물체의 투영을 통해 큐브 맵을 생성하고 이를 샘플링하는 과정을 도식화 한 예를 보여주고 있다.

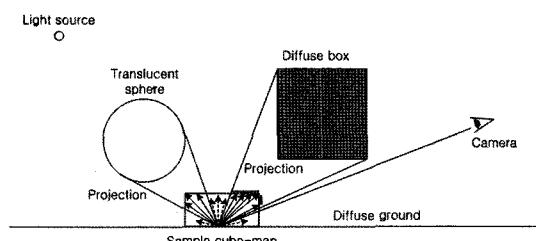


그림 3 큐브 맵 투영 및 샘플링을 통한 최종 수집 가속화 기법. 3차원 장면의 투영을 통해 큐브 맵을 생성하고 최종 수집 광선에 대해 큐브 맵을 샘플링함으로서 성능을 향상할 수 있으며 최근의 GPU의 하드웨어 연산을 통해 더욱 빠르게 계산 할 수 있다.

2.3 본 논문의 기여도

본 논문에서는 전역 조명 계산에서 매우 중요하게 사용될 수 있는 조화 평균 거리를 앞서 설명한 세 가지 전역 조명 계산 알고리즘을 사용하여 계산한 결과를 분석함으로써 다음의 사항들을 제시한다.

- 1) 최종 수집 계산을 통한 조화 평균 거리 계산에서 샘플링 개수에 따른 정확도 비교
- 2) CPU에서의 계산 대비 GPU에서 큐브 맵 투영을 통한 조화 평균 거리 계산의 샘플링 개수에 따른 정확도 비교

- 3) 포톤 매핑 기법을 통한 조화 평균 거리 계산 방법 및 다양한 조건에서의 계산 결과 비교
- 4) 포톤 매핑 기법을 통한 향상된 조화 평균 거리 계산 방법 및 다양한 조건에서의 계산 결과 비교

본 논문에서 제시하는 위의 비교 분석 결과를 통해 조화 평균 거리를 계산할 수 있는 가장 직접적인 방법인 CPU 및 GPU에서의 최종 수집 기법의 정확도를 보이고 가시적으로 충분한 샘플링 개수에 대한 근거를 제시한다. 특히 최종 수집 기법에 비해 빠르게 전역 조명 효과를 근사화할 수 있는 포톤 매핑 기법을 사용하여 조화 평균 거리를 근사화할 수 있는 가능한 한 방법을 제시하고 포톤의 개수, 포톤 수집 개수, 포톤 최대 반사수 등의 여러 조건의 변화에 따른 결과를 보임으로서 환경 요인의 영향 및 문제점을 분석한다. 또한 포톤 매핑 기법을 사용한 앞의 방법에 비해 향상된 조화 평균 거리를 근사화할 수 있는 수정된 기법을 제시하고 역시 다양한 조건하에서의 결과를 분석한다.

이러한 결과를 바탕으로 광휘 캐시나 환경 폐색 등의 응용에서 중요한 요소로 사용될 수 있는 조화 평균 거리 계산의 근사화 기법 개발에 있어서 해결해야 할 문제점 및 방향을 제시하도록 한다.

3. 전역 조명 렌더링 알고리즘을 통한 조화 평균 거리 계산

3.1 최종 수집 기법을 통한 조화 평균 거리 계산

최종 수집 기법에서 반구위 방향을 샘플링한 광선들을 추적하는 과정에서 각 광선이 물체와 충돌하였을 때 현재 시점이 바라보고 있는 지점과 광선이 충돌한 물체 지점과의 거리를 계산할 수 있게 된다. 이 거리들을 조화 평균하여 결과 값은 얻을 수 있다.

3.2 GPU에서의 최종 수집 기법을 통한 조화 평균 거리 계산

그래픽스 하드웨어에서 제공하는 래스터화 연산을 통해 큐브 맵을 생성하는 과정에서 각 물체 지점의 눈 좌표계에서의 깊이 값은 얻어 올 수 있다. 그림 4의 2차원 예에서 보는 것과 같이 생성된 큐브 맵에 대해 샘플링하는 벡터의 깊이 값 t 를 얻어 올 수 있고 이 값에 대해서 원하는 거리값 d 를 $t/\cos\theta$ 의 식으로 계산할 수 있게 된다. 이를 3차원 공간으로 확장하면 $t/f(\theta, \phi)$ 의 계산을 통해 실제 거리 값을 추정할 수 있는데 이때 θ, ϕ 의 범위에 따라 큐브 맵의 각 면이 결정되므로 각도를 각 경우에 맞게 조절해 주는 것이 필요하다. 즉 $0 < \theta < \pi/4$ 인 경우는 $f(\theta, \phi) = \cos(\theta)$ 가 되고 그 외 $\pi/4 < \theta < \pi/2$ 경우는 $f(\theta, \phi) = \cos(\phi)\sin(\theta)$ 이다. 앞의 두 경우는 각각 큐브 맵의 윗면인 경우와 4개 옆면인

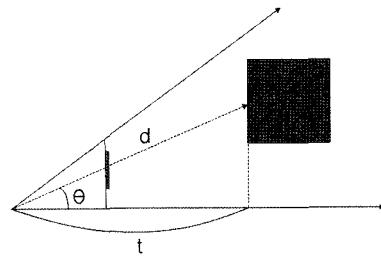


그림 4 투영 공간에서의 거리 관계에 대한 2차원 예

경우를 의미한다. ϕ 은 ϕ 값에 따라 큐브 맵의 4개 옆면 중 한 면이 결정되었을 때, 해당 면을 지나는 축을 기준으로 한 각도로 변환해 준 값이다.

3.3 포톤 매핑 기법을 통한 조화 평균 거리 계산 방법 I

포톤 매핑 기법에서 각 포톤들이 충돌하는 지점에 대해 빛의 색깔 및 세기 정보를 저장하는 대신 포톤이 이동한 거리를 저장하고 가시화 단계에서 이 거리 정보를 수집하여 조화 평균 거리를 근사화할 수 있다. R. Herzog 등의 논문에서는 입사하는 포톤의 거리를 저장하여 조화 평균 거리를 계산하였다[17]. 본 논문에서는 입사하는 포톤이 이동한 거리를 저장하지 않고 반사된 포톤이 이동한 거리를 포톤의 출발 지점에 저장하도록 하여 시점에서 출발하는 포톤들에 대한 계산까지 재귀적으로 일관성 있게 수행되도록 하였다. 또한 반구 방향으로 보이는 물체들과의 거리를 계산하고자 하는 것이므로 모든 물체 표면의 성질을 난반사 표면으로 가정하였다.

이렇게 저장된 포톤을 kd-tree 형태의 공간 자료구조로 재구성하고 이를 통해 주변에 가장 가까운 N개의 포톤을 빠르게 검색하여 각 포톤이 가지고 있는 거리 정보를 사용하여 조화 평균 거리를 계산한다.

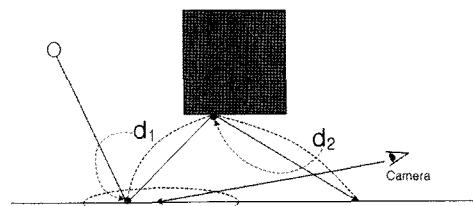


그림 5 포톤 매핑 기법을 통한 조화 평균 거리 근사화 방법 I. 광원에서 추적되는 각 포톤들이 다음 상호 작용 지점과의 거리를 저장하게 되며 가시화 과정에서 이 거리 정보를 수집하여 근사화된 조화 평균 거리를 계산할 수 있다.

3.4 포톤 매핑 기법을 통한 조화 평균 거리 계산 방법 II

3.3절에서 설명하는 기법을 사용하는 경우의 문제점은 하나의 포톤이 가지고 있는 거리 값이 미치는 영향이

포톤의 수집 반경 내 모든 지점에 적용되므로 오차가 확산될 수 있는 있다는 것과 수집 반경이 커질수록 계산되는 거리 정보가 부정확해진다는 것이다. 이는 일반적인 포톤 매핑 기법에서도 동일하게 나타나는 문제이나, 조화 평균 거리 계산의 경우에는 세밀한 형태를 가지는 지점이나 코너 지점에서 매우 작은 값을 가지는 거리가 계산되었을 때 이 값이 영향을 미치는 수집 반경 내 지점들에서의 조화 평균 값이 매우 작아지게 되는 특징이 있다.

그림 6은 3.3절의 기법을 사용하였을 때의 결과 예이다. 충분하지 못한 포톤을 사용하였기 때문에 전체적으로 계산된 거리 값이 오차를 가지고 있는 것을 확인할 수 있으며 특히 세밀한 물체 부분이나 코너 주변에 매우 작게 계산된 거리 값으로 인하여 검은 디스크 형태의 오차 지점들이 생성되고 이 지점들은 포톤 수집 개수가 증가하여 그 반경이 커질수록 뚜렷하게 확대됨을 확인할 수 있다.

이와 같은 오차를 줄이기 위해 본 논문에서는 수정된 두 번째 방법을 제안한다. 포톤에 다음 충돌 지점과의 거리뿐만 아니라 방향을 함께 저장하여 거리를 직접 조화 평균 계산에 사용하는 대신 현재 포톤의 위치, 다음 충돌 지점과의 거리 및 방향을 통해 다음 충돌 지점의 위치를 계산하고 이 위치와 현재 거리 계산이 이루어지

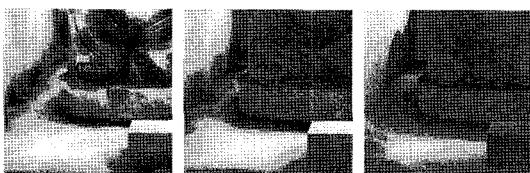
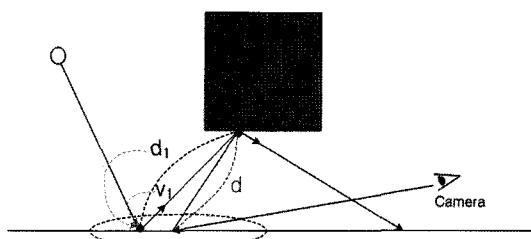
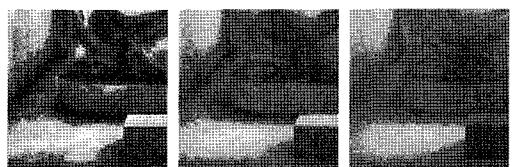


그림 6 포톤 매핑 기법을 통한 조화 평균 거리 계산 방법 I에서 포톤의 수집 개수 변화에 따른 조화 평균 거리 오차. 약 130,000개의 포톤에 대하여 왼쪽에서부터 각각 64개, 256개, 1024개의 포톤을 수집한 경우에 조화 평균 거리 값을 그레이스케일로 표현한 결과이다.



(a) 포톤 매핑 기법을 통한 조화 평균 거리 근사화 방법 II를 통한 거리 계산 수행 예. 수집되는 포톤에 대하여 주어진 지점을 기준으로 새롭게 계산된 거리 d 를 사용한다.



(b) 포톤 수집 개수 변화에 따른 조화 평균 거리 계산 결과. 약 130,000개의 포톤에 대하여 왼쪽에서부터 각각 64개, 256개, 1024개의 포톤을 수집한 경우이다.

그림 7 포톤 매핑 기법을 통한 조화 평균 거리 근사화 방법 II를 통한 결과 예. 방법 I에 비하여 작은 거리 값이 미치는 영향이 감소된 것을 볼 수 있으며 수집 개수가 커질수록 누적되는 오차가 커지는 현상이 크게 감소하고 있음을 확인할 수 있다.

는 지점과의 거리를 다시 계산하여 조화 평균 계산에 사용하는 것이다. 그림 7은 (a) 이러한 계산 과정에 대한 도식화 예와 (b) 그림 6과 동일한 환경에서 수정된 방법 II를 사용하였을 때의 작은 거리 값에 의한 오차가 감소하고 수집 반경이 커질수록 거리값이 부정확해지는 문제에 대하여 향상된 결과를 보여준다.

4. 실험 결과 및 분석

본 논문에서 전역 조명 알고리즘에서의 조화 평균 거리 계산 분석을 위해 다음 두 장면을 대상으로 실험하였다(그림 8). 장면 A는 정육면체 안에 세밀함을 충



그림 8 본 실험에서 사용된 장면. 왼쪽에서부터 장면 A, 장면 B, 본 실험에서 사용된 시점에서의 장면 B에 대한 렌더링 이미지이다. 장면 A는 76,787개 삼각형, 장면 B는 612,613개 삼각형으로 이루어져 있다. 장면 A는 정육면체 상단 중앙에 스포트 광원을 하나 가지고 있으며 장면 B는 실외로부터 안을 비추고 있는 스포트 광원을 가지고 있다.

히 가지고 있는 물체들을 포함하고 있어 조화 평균 거리의 정확도를 확인하는 주요 장면으로 사용되었으며, 장면 B는 비교적 넓은 공간 안에 보다 복잡하게 물체들이 구성되어 있어 광원과 카메라에서 포톤을 추적하는 경우의 차이를 비교하는 데 사용되었다.

4.1 최종 수집 기법을 통한 조화 평균 거리 계산 분석

먼저 최종 수집 기법을 통한 조화 평균 거리 계산을 수행할 때 가능한 환경 설정에 대한 결과를 분석할 수 있다. 본 논문에서는 앞서 설명한 CPU에서의 광선 추적 기반 최종 수집 기법과 GPU에서의 큐브 맵 기반 최종 수집 기법에 대해 최종 수집 광선 개수의 변화에 따른 각 방법의 결과를 분석하였다.

표 1은 장면 A에서 CPU와 GPU의 각 방법에서 주어진 수집 광선 개수에 대한 결과(d^i)와 65,536(256×256)개의 수집 광선을 사용한 결과(d_{ref}^i)에 대하여 각 픽셀이 가지고 있는 거리의 차이 값의 평균을 장면의 바운딩 박스(bounding box)의 대각 길이($l_{diagonal}$)로 나눈 값에 100을 곱한 다음의 식 (3)을 통해 계산한 값이다. 이렇게 함으로써 다양한 크기의 장면에 대하여 정규화 효과를 가지는 차이 값의 평균을 비교할 수 있다.

$$f_{diff} = \frac{\sum_{i=0}^{N_{refs}} |d_{ref}^i - d^i|}{N_{pixels}} \times \frac{100}{l_{diagonal}} \quad (3)$$

표 1의 결과로부터 최종 수집 광선의 개수가 증가할 수록 차이 값이 점차 작아지는 것을 볼 수 있으며 1,024 개의 수집 광선을 사용한 경우부터 차이 값이 진동하면서 서서히 감소하는 경향을 보이고 있는 것을 확인할 수 있다. 그림 12와 그림 13에서 보는 것과 같이 실제로 1,024개 이상의 수집 광선을 사용한 경우부터는 GPU와 CPU의 경우 모두 계산된 조화 평균을 나타낸 이미지를에 대한 시각적인 차이를 느끼기 어렵다. 그러나 그림 12에서 보다 복잡하게 구성된 장면 B의 경우가 장면 A

표 1 CPU와 GPU의 방법에서 주어진 수집 광선 개수에 따른 정확도 측정. 65,536개의 최종 수집 광선을 사용한 결과와 각 픽셀간 거리 값의 차이에 대한 평균을 통해 각 수집 광선 개수를 사용한 결과의 정확성을 나타내었다.

	32	64	128	256	512
CPU	장면 A	0.5971	0.3121	0.1967	0.1113
	장면 B	0.8255	0.4799	0.3003	0.2181
GPU	장면 A	0.5538	0.3178	0.1921	0.1050
		1,024	2,048	4,096	8,192
CPU	장면 A	0.0357	0.0248	0.0351	0.0109
	장면 B	0.0994	0.0460	0.0454	0.0211
GPU	장면 A	0.0276	0.0155	0.0173	0.0080
					0.0211

에 비해 65,536개 광선을 사용한 결과에 늦게 가까워지는 것을 차이 이미지 변화를 통해 알 수 있다. 전역 조명 계산 과정에서는 각 방향에서 들어오는 빛의 색깔이 복잡하게 변할 수 있으므로 수집 광선의 개수가 더욱 많이 요구될 수 있으나 거리 값은 기하 특성에만 영향을 받게 되므로 주어진 장면에서 비교적 빨리 일정 수준 이상의 정확한 값을 얻을 수 있다고 할 수 있다. 본 실험에서 사용된 장면들에서는 1,024개 이상의 샘플링이 필요한 경우라고 말할 수 있다.

또한 CPU와 GPU를 통해 계산된 결과에 대하여 비교 할 수 있는데, GPU 계산에서 사용되는 큐브 맵이 각 사분면에 대하여 균일한 해상도를 가지므로 CPU와 샘플링 기법에서의 차이는 없다고 할 수 있다. 그러나 CPU는 직접적인 광선-물체 교차검사를 통해 연속 공간에서의 정보를 사용하지만 GPU는 큐브 맵에 투영되어 지정된 해상도로 이산화 된 물체의 정보를 사용하는 것에 차이가 있다. 또한 래스터화 연산을 통한 큐브 맵을 생성할 때 얻어진 깊이 정보를 사용하여 실제 광선에 대한 거리 정보로 변환한 값을 사용해야 하는 점도 거리 정보를 직접 계산할 수 있는 CPU와의 차이점이라고 할 수 있다.

그림 9에서 보는 것과 같이 CPU와 GPU에서 65,536 개의 수집 광선을 사용한 결과에 대한 평균 거리 차이는 0.029104로 CPU에서 계산된 값과 GPU에서 계산된 값이 비교적 유사하게 계산되고 있는 것을 확인할

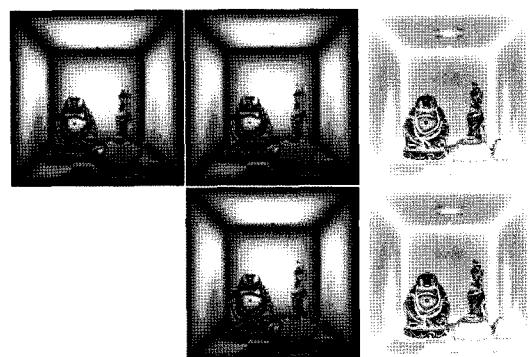


그림 9 65,536개의 수집 광선을 사용한 경우 조화 평균 거리 계산 결과. CPU(좌상), GPU 및 분면 당 64×64 해상도 큐브 맵을 사용(중상), GPU 및 분면 당 128×128 해상도 큐브 맵을 사용(중하) 하여 계산한 조화 평균값에 0.14배하여 이미지로 표현하였다. 오른쪽 상하의 두 이미지는 각각 CPU 계산 결과와 동일 행의 GPU 결과의 차이 값을 10배 한 결과이며, 차이 값이 0인 경우 흰색이며 차이가 클수록 검정색에 가까워진다.

수 있다. 즉 이산화된 정보 및 계계산 과정을 통해 얻어진 거리 값이 대부분의 지역에서 비교적 정확하다고 할 수 있는 반면, 기하 정보가 매우 세밀하게 변하는 지점에서는 다른 지점들에 비해 CPU와 비교적 큰 차이를 보이고 있는 것을 조화 평균 값 차이 이미지를 통해 확인할 수 있다. 이때 GPU 계산에서 사용된 큐브 맵은 해당 지점의 법선을 중심으로 한 좌표축의 각 분면에 대해 64×64 해상도를 가진다. 이를 128×128 해상도로 증가하였을 경우 같은 환경에서의 평균 거리 차이는 0.029091로 큰 차이를 보이지 않는 것을 그림 9의 해당 차이 이미지를 통해서도 확인할 수 있다. 그러므로 세밀한 지역에서의 차이는 큐브 맵의 해상도보다는 래스터화 연산을 통해 투영된 거리값이 매우 작아 수치적인 오차를 발생할 가능성이 커지기 때문이라고 할 수 있다. CPU에 비해 부정확한 부동 소수점 연산을 사용하는 GPU 계산의 정확도를 높이기 위해 최근의 GPU에서 제공하기 시작한 배정도수 부동 소수점 연산을 고려할 수 있으나 큐브 맵 기법이 기반으로 하는 그래픽스 파이프라인에서는 아직 지원되고 있지 않다.

4.2 포톤 매핑 기법을 통한 조화 평균 거리 계산 분석

본 논문에서는 포톤 매핑 기법을 사용하여 조화 평균 거리를 근사화하는데 인자로 사용될 수 있는 다음의 조건에 대한 결과를 제시한 방법 I과 II에 대해 수행하여 분석한다.

- 저장되는 포톤의 개수
 - 한 지점에서 수집되는 포톤의 최대 개수
 - 포톤을 샘플링하는 위치(광원 혹은 카메라)
 - 동일한 저장 포톤 개수에 대한 반사 횟수 변화
- 그림 14과 그림 15는 각각 방법 I과 II에 대해 저장되는 포톤 개수와 한 지점에서 수집되는 포톤 개수가 변화하는 경우에 대한 결과를 보여주고 있다. 일반적인 포톤 매핑 기법에서와 유사하게 포톤이 부족한 경우에는 수집 개수가 커질수록 계산 되는 거리가 부드럽게 변하게 되지만, 멀리서 측정된 거리 정보가 많이 포함되고 기하 정보가 세밀하게 변하는 지점에서는 포함되지 않아 할 거리 정보들이 평균 계산에 참여하므로 부정확한 결과를 생성하여 세밀함을 잃어버리게 된다. 그러나 저장되는 포톤 개수가 점차 커질수록 충분히 작은 수집 반경에서 원하는 개수의 포톤을 사용하여 거리 정보를 계산할 수 있으므로 보다 상세한 조화 평균 거리를 얻을 수 있다. 저장된 포톤의 개수가 적은 경우에 방법 II를 통해 오차를 효과적으로 줄일 수 있으나(그림 7), 저장된 포톤의 개수가 많은 경우에도 방법 II를 통해 계산한 결과가 거리 변화에 있어서 작은 거리를 가지는 포톤들이 미치는 영향이 일정 부분 교정되어 적용되므로 보다 거리 변화에 있어서 보다 정확하고 상세한 정보를

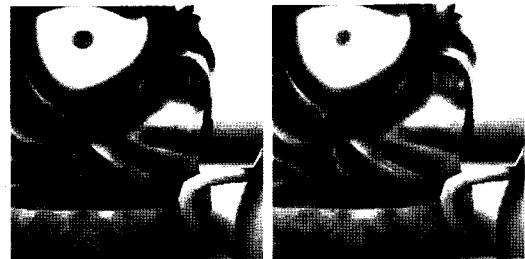


그림 10 약 8,530,000개의 많은 포톤에 대해 각 지점에서 65,536개의 포톤을 수집한 경우 방법 I과 II의 비교. 왼쪽부터 각각 방법 I과 방법 II에 의해 생성된 이미지의 일부이다.

생성하는 것을 그림 10을 통해 비교할 수 있는데, 특히 기하 정보가 세밀하게 변하는 손, 발 등의 부분에서 두드러진다.

한 가지 참고할 점은 포톤 자료구조와 이에 따른 포톤 저장시의 메모리 사용률이다. 포톤이 가지고 있어야 할 기본 정보는 저장 위치(12바이트), 방향(12바이트), 저장 위치에서의 물체 법선(12바이트), 포톤 파워(12바이트), 반사 횟수(2바이트), 평면 값(2바이트)가 필요하여 총 52바이트가 되며 여기에 포톤 파워 대신 거리 값을 저장하면 46바이트가 된다. 포톤의 방향과 물체의 법선은 포톤을 필터링 하는데 필요한 정보들이다. 또한 방법 II를 사용하기 위해 입사 방향이 아닌 이후 반사 방향을 함께 저장하면 총 58바이트의 메모리 공간이 하나의 포톤 당 필요하게 된다. 그래서 본 실험에서 약 8,530,000 개의 최대 포톤을 저장하였을 때 470MB 정도의 저장 공간이 필요하게 되며 전역 조명 효과를 포함하기 위해서 포톤 파워를 함께 저장하면 570MB 정도의 저장 공간이 필요하게 된다.

그림 16와 그림 17은 각각 장면 A와 장면 B에 대하여 같은 수의 포톤이 저장되었을 때 적은 반사 횟수를 가지는 포톤을 많이 샘플링 하는 경우와 많은 반사 횟수를 가지는 포톤을 적게 샘플링 하는 경우를 비교하기 위한 결과이다. 각각의 조건에서 포톤을 광원에서 샘플링 한 경우와 카메라에서 샘플링 한 경우, 포톤이 약 1,060,000개, 8,530,000개 저장된 경우, 또 포톤의 최대 수집 개수를 64, 256, 1024로 설정한 경우의 조합에 대하여 조화 평균 거리를 계산하였다. 모든 결과는 포톤 매핑 기반 방법 II를 사용한 결과이다.

그림 16의 장면 A는 비교적 제한적인 공간 안에 자체적인 세밀함은 가지고 있지만 전체적인 형태가 복잡하지는 않은 다섯 개의 물체들이 배치되어 있어 광원에서 출발한 포톤이 전체 장면에 균일하게 분포할 수 있는 환경이라고 할 수 있다. 정확한 조화 평균 거리를 계

산하는데 있어서는 주어진 지점으로 얼마나 다양한 방향에 대한 거리 정보를 얻을 수 있는가 하는 것이 중요한데 광원에서 출발한 포톤들이 확률적으로 모든 지점에 대해서 비교적 다양한 경로를 생성할 수 있는 환경인 것이다. 그러므로 장면 A에서는, 일반적으로 카메라에서 샘플링 된 포톤이 가시적인 지점에 대한 포톤의 밀도를 높여 보다 정확한 정보를 생성할 수 있으리라는 예상과 달리 광원에서 샘플링 된 포톤들이 보다 다양한 경로를 생성하여 세밀한 지점들에 대한 조화 평균 거리를 보다 정확하게 생성하고 있는 것을 확인할 수 있다. 또한 카메라에서 샘플링 된 경우와 광원에서 샘플링 된 경우 모두 최대 반사 횟수를 80번으로 크게 하고 적은 포톤을 추적한 경우에 대해서 보다 정확한 결과를 얻는 것을 볼 수 있다. 이러한 사실은 포톤이 많이 저장된 경우보다 포톤이 적게 저장된 경우에서 조금 더 뚜렷하게 드러나는데 지역적으로 분포된 정보를 사용하기 때문에 발생하는 오차를 생각한다면 얼굴이나 손, 발과 같은 세밀하게 굴곡이 지는 부분에서 그 차이들을 확인할 수 있다.

그림 17의 장면 B는 비교적 넓은 공간에 다양한 형태와 세밀한 물체들이 다수 분포되어 있다. 또한 광원도 실외에 위치하여 있고 카메라가 보고 있는 장면도 전체 장면의 일부로써 장면 A와는 다른 특성을 가진다. 장면이 크고 복잡하여 포톤 매핑 기법에 기반한 방법이 전체적으로 장면 A에 비해 큰 오차를 생성하기 때문에 각 경우에 대한 정확도를 비교하기 쉽지 않으나 그림 11의 GPU에서의 65,536개의 최종 수집 광선을 샘플링 하여 생성한 조화 평균 거리 형태와 비교하였을 때, 광원보다는 카메라에서 샘플링 한 경우에 포톤 정보의 집중도가 높아져서 보다 정확한 결과를 생성하는 것을 볼 수 있다. 카메라에서 샘플링 되는 경우에는 최대 반사 횟수를 적게하고 생성되는 포톤의 개수를 크게 하는 경우에 보다 정확한 형태의 거리 값을 생성하는 것을 확인할 수 있다. 이는 장면이 복잡하고 크기 때문에 많은 횟수의

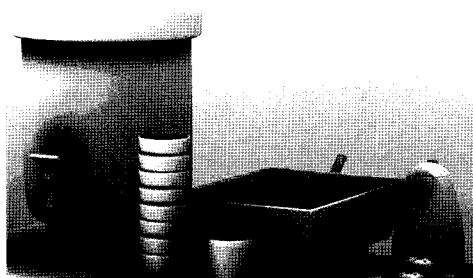


그림 11 장면 B에 대해 GPU에서의 최종 수집 기법을 통한 조화 평균 거리 계산한 결과. 65,536개의 수집 광선을 큐브 맵을 사용하여 샘플링한 결과이다.

반사를 추적한다고 하더라도 카메라가 바라보고 있는 일부의 장면에 정보를 다시 전달하는 것이 확률적으로 어려운 일이기 때문이다.

결국 포톤 매핑 기법을 활용하여 조화 평균 거리를 계산하고자 하는 경우에는 조화 평균 거리 계산이 필요한 지점들에 최대한 다양한 방향으로의 정보가 생성되는 것이 중요하며 이러한 정보가 최대한 국소 지역에서 사용 가능하도록 충분해야 함을 알 수 있다.

4.3 조화 평균 거리 계산 시간 분석

표 2는 최종 수집 기법에 기반한 조화 평균 거리 계산 시간을 나타내고 있다. 본 시간 측정에 사용된 프로그램은 고급 쇠적화 기법들의 적용 없이 최종 수집 알고리즘을 일반적인 C++ 코드로 구현한 것이며 GPU 계산에 사용된 프로그램은 OpenGL API를 기반으로 하여 어셈블러 형태의 쉐이더 코드로 작성되었다. GPU는 NVIDIA Geforce 8800 GTX를 사용하였으며 CPU는 Intel Xeon E5420 2.5GHz를 사용하였다.

CPU에서 1024×1024 해상도의 이미지에 대한 최종 수집계산을 위해 소요된 시간은 16개 수집 광선에 대한 계산 시간인 1분에서, 65,536개 수집 광선에 대한 계산 시간인 62시간까지 수집 광선의 개수에 따라 크게 증가하는 것을 확인할 수 있다. 그러나 GPU에서 최종 수집을 위해 소요된 시간은 수집 광선의 개수에 관계없이 평균적으로 약 40분가량의 시간이 소요된 것을 알 수 있다. GPU의 래스터화 연산을 이용한 큐브 맵 기반 최종 수집 기법에서는 장면이 포함하고 있는 물체를 투영하여 큐브 맵을 만드는 과정이 광선 수집 개수에 관계없이 동일하게 수행되며 이 생성시간이 큐브 맵을 샘플링하는 시간에 비해 매우 크기 때문에 그 소요시간 변동이 크지 않다고 할 수 있다. 즉 최종 수집 광선의 개수가 증가할 수록 GPU를 사용한 계산의 효율성이 커지게 된다.

표 3은 CPU에서 포톤 매핑 기반 조화 평균 거리 계산 소요 시간을 보여주고 있다. 저장되는 포톤의 개수가 증가 할수록, 수집 되는 최대 포톤 개수가 많을수록 시간이 증가하는 것을 볼 수 있다. 가장 많은 8,530,000개의 포톤이 저장되고 최대 수집 개수가 1024개인 경우에

표 2 CPU와 GPU에서의 최종 수집 기법에 기반 한 조화 평균 거리 계산 시간 비교. 시간단위는 초단위이다.

광선 수	CPU	GPU	광선 수	CPU	GPU
16	61.6	2412.4	2,048	7,756.2	2471.0
32	104.5	2426.9	4,096	14,825.3	2473.6
64	251.6	2427.8	8,192	29,689.7	2495.1
128	486.0	2433.7	16,384	56,647.8	2487.8
256	958.0	2430.5	32,768	111,470.1	2519.2
512	1,966.0	2442.0	65,536	225,910.1	2596.0
1,024	3,809.7	2448.4	-	-	-

표 3 장면 A에서 포톤 저장 개수 및 최대 포톤 수집 개수에 따른 포톤 매핑 기반 조화 평균 거리 계산 방법 I과 방법 II의 계산 시간 비교. 포톤 맵을 생성하는 시간은 포함되지 않고 각 지점에서 주변의 포톤을 수집하여 거리값을 계산하는 시간만을 측정한 것이다. 시간 단위는 초단위이다.

포톤 개수	방법	최대 포톤 수집 개수				
		64	128	256	512	1024
130,000	방법I	17.5	43.2	66.7	119.7	195.8
	방법II	19.2	40.8	71.9	131.4	221.2
260,000	방법I	20.5	51.8	71.5	126.7	214.0
	방법II	22.2	48.9	76.0	140.1	242.2
530,000	방법I	23.2	51.1	77.3	137.5	228.7
	방법II	25.7	52.5	83.5	153.6	256.7
1,060,000	방법I	29.6	59.6	88.4	154.3	150.1
	방법II	32.0	60.2	94.2	169.0	273.5
2,130,000	방법I	41.4	72.7	103.6	177.9	278.5
	방법II	43.2	78.2	109.3	199.4	314.5
4,260,000	방법I	75.4	98.0	131.2	221.7	335.8
	방법II	72.8	101.3	144.3	238.9	365.9
8,530,000	방법I	120.2	158.6	180.8	277.8	403.8
	방법II	122.8	148.1	190.7	289.2	436.1

1024×1024 해상도 이미지에 대해서 포톤 수집을 통해 조화 평균 거리를 계산하는데 436.1초의 시간이 걸렸는데 이는 CPU에서 픽셀 당 128개의 최종 수집 광선을 사용한 경우보다 적은 시간이다. 상당히 많은 포톤이 저장된 상태에서도 kd-tree 공간 자료구조를 사용한 포톤 수집이 비교적 효과적인 것을 알 수 있다. 128개의 최종 수집 광선을 사용한 경우에는 정보의 부족으로 인하여 거리 정보가 부드럽게 변하지 못하는 지역들이 발생하는데 포톤 매핑 기법을 사용한 경우에는 정보의 부족이 거리 값의 노이즈 형태나 세밀하게 변하는 지역의 거리 정보가 불러링 되는 형태로 나타나게 된다. 또한 포톤 매핑 기법은 메쉬 프리 기법으로써 저장된 포톤 맵은 장면의 복잡도와 관계없이 점정보로 저장되고 kd-tree로 구성되므로 수집 시간은 포톤의 개수에 크게 영향을 받게 된다.

표 4는 포톤 맵을 생성하는데 소요된 시간을 보여주고 있다. 최대 포톤을 저장한 경우에 포톤 수를 픽셀 수로 나눠보면 픽셀 당 8.1개의 포톤이 추적된 것으로 볼 수 있는데, CPU에서 16개 최종 수집 광선을 사용하였을 경우와 유사하거나 많은 시간이 소요되었으므로 확률 계산이나 난반사 방향 샘플링, 포톤 정보의 저장, kd-tree 구성 등에 소요되는 추가 시간이 포톤 맵을 생성하는 비용의 일정 부분을 차지하고 있음을 알 수 있다.

위의 시간 측정 결과들을 통해 GPU에서의 큐브 맵 기반 최종 수집 기법의 성능은 장면 복잡도에 크게 영

표 4 장면 A에서 포톤 저장 개수에 따른 포톤 맵 생성 시간. 각각 최대 80번의 반사 횟수를 가지고 포톤을 추적한 경우의 포톤 맵 생성 시간이며, 생성된 포톤 맵에 대한 kd-tree 생성시간을 포함한다. 시간 단위는 초단위이다.

저장된 포톤 개수	포톤 맵 생성 시간	
	광원에서 생성	카메라에서 생성
130,000	1.6	1.0
260,000	2.7	1.9
530,000	4.9	3.8
1,060,000	9.4	6.4
2,130,000	18.9	12.1
4,260,000	38.9	26.2
8,530,000	83.8	54.4

향을 받으며 수집 광선의 개수에는 큰 영향을 받지 않는 특징이 있음을 알 수 있다. CPU에서의 최종 수집 기법은 수집 광선의 개수가 증가함에 따라 계산 시간도 선형적으로 증가함을 볼 수 있다. 포톤 매핑 기반 방법에서 방법 II는 방법 I에 비해 적게는 1%에서 많게는 9%정도의 추가 시간을 소비하고 있음을 알 수 있다. 포톤 매핑을 통한 조화 평균 거리 계산 시간이 CPU나 GPU 기법에 비해 빠르게 수행될 수 있으나 생성되는 정보의 품질에 있어서 오차를 가지고 있으므로 보다 적은 포톤으로 빠르고 정확하게 거리 정보를 근사할 수 있는 방법이 요구된다.

4.4 조화 평균 거리 계산 기법을 위한 주요 분석 결과

앞의 실험 결과를 통해 도출된 대표적인 사항들을 다음과 같이 정리할 수 있다.

- I. CPU 기반 최종 수집 기법을 사용한 조화 평균 거리 계산 시간은 수집 광선의 개수에 선형적으로 증가하며 주어진 1024개 이상의 광선을 사용하였을 때 일정 수준 이상의 정확도를 가지는 결과를 얻을 수 있었다.
- II. GPU의 큐브 맵 기반 최종 수집 기법을 사용한 조화 평균 거리 계산 시간은 수집 광선의 개수에는 크게 영향을 받지 않고 장면을 구성하는 기하 물체의 양에 종속적이며 세밀하게 변하는 지역에서 CPU에서 계산된 값과 오차를 보이지만 전체적으로 CPU와 적은 차이를 보인다.
- III. 포톤 매핑 기법에 기반한 방법은 최종 수집 기법에 비해 빠른 시간에 수행 가능하지만 일정한 영역 안의 정보를 사용하기 때문에 발생하는 정보의 세밀함과 부드러움에 대한 문제를 포함한다.
- IV. 세밀함과 부드러움은 저장된 포톤의 개수와 각 지점에서 수집하는 포톤의 최대 개수로 결정되는데 일정 개수 이상의 포톤이 저장된 경우에는 포톤 수집 개수가 적은 경우에 세밀함이 보존되고 부드러

움에 대한 오차는 노이즈 형태로 나타나게 된다.

V. 포톤 맵에 추가적인 정보를 저장하고 약간의 계산을 추가함으로써 거리 정보를 주어진 점을 중심으로 변환함으로써 보다 정확하고 부드러운 계산 결과를 얻을 수 있음을 보였다.

위의 내용들이 각 방법으로부터 어느 정도 예측 가능한 내용을 포함하고 있을 수 있으나, 실제로 각 조건들에 대한 실험 결과를 보임으로써 그 특성을 확인하고 개선된 기법을 모색하는 기반 자료로 활용할 수 있는 것에 의미가 있다고 하겠다. 특히 GPU를 사용한 기법에서는 장면을 구성하는 기하 정보를 세밀함의 손상 없이 간략화 하는 것이 성능향상을 위해 중요하며, CPU 기반 기법에서는 요구되는 조화 평균 거리의 정확도에 따라 최소의 수집 광선 개수를 사용하거나 각 지역의 복잡도를 예측하여 수집 광선의 수를 적응적으로 선택할 수 있는 기법의 개발이 가능하다. 더 나아가 광선의 수에 크게 영향을 받지 않는 GPU의 큐브 맵 기반 기법을 사용하여 CPU에 비해 빠른 시간에 정확도 높은 조화 평균 거리 값을 생성하고 오차를 발생할 수 있는 세밀한 지점에 대해서만 추가적으로 광선을 적접 샘플링하는 혼합 기법을 고려할 수 있으며 이때 추가적인 광선 샘플링 역시 CUDA등을 통해 GPU에서 가속화할 수 있다. 또한 포톤 매핑 기법 등의 확률적인 기법을 사용하는데 있어서는 각 지점에서 접근하게 되는 포톤들이 가지고 있는 방향의 다양성을 최대한 확보할 수 있는 포톤 생성 기법과 세밀한 지점과 부드럽게 변하는 지점에 대한 적응적인 포톤 수집 기법이 요구된다. 또한 본 논문에서 실현한 포톤 매핑 기법에 기반한 방법 II와 같이 각 포톤이 가지고 있는 정보를 조화 평균 거리를 생성하고자 하는 지점에 대해 오차를 최소화할 수 있도록 변환하는 기법을 응용할 수 있다.

5. 결 론

본 논문에서는 3차원 공간의 한 지점에 대한 가시성 및 주변 물체들과의 조명 관계를 효과적으로 표현할 수 있는 조화 평균 거리의 계산을 효과적인 고품질 전역 조명 알고리즘으로 알려진 최종 수집 기법과 포톤 매핑 기법을 통해 수행하였을 때의 결과를 분석하였다. 특히 최종 수집 기법에서는 광선 샘플링에 기반한 CPU 기법과 큐브 맵 샘플링에 기반한 GPU 기법에 대한 결과를 도출하였으며, 포톤 매핑 기법에서는 일반적인 포톤 매핑 기법을 그대로 응용한 방법과 약간의 추가 정보와 계산을 사용하여 오차를 줄일 수 있는 두 가지 방법에 대한 결과를 도출하여 비교 하였다. 또한 이를 각 방법을 다양한 조건에 대하여 실험하고 분석하였다.

본 논문의 분석 결과를 통해 주어진 환경에서의 조화

평균 거리 계산의 특성을 확인하고 향상된 조화 평균 거리 계산 기법 개발을 위한 주요 고려 사항들을 제시하였다. 이를 바탕으로 각 방법에서 요구되는 모듈의 성능 극대화 및 새로운 계산 알고리즘을 개발하고 전역 조명 계산에서 중요한 척도로 사용되는 조화 평균 거리를 효과적으로 계산함으로써 보다 향상된 전역 조명 계산 기법의 개발에도 기여할 수 있으리라 기대한다.

참 고 문 헌

- [1] J. T. Kajiya, "The rendering equation," *Journal of ACM Transactions on Graphics*, vol.20, no.4, pp. 143-150, 1986.
- [2] T. Whitted, "An improved illumination model for shaded display," In *Proc. of ACM SIGGRAPH'79*, 1979.
- [3] R. L. Cook, T. Porter and L. Carpenter, "Distributed ray tracing," In *Proc. of ACM SIGGRAPH'84*, pp.137-145, 1984.
- [4] C. M. Goral, K. E. Torrance, D. P. Greenberg and B. Battaile, "Modeling the interaction of light between diffuse surfaces," In *Proc. of ACM SIGGRAPH'84*, pp.213-222, 1984.
- [5] M. F. Cohen and D. P. Greenberg, "The hemi-cube: a radiosity solution for complex environments," In *Proc. of ACM SIGGRAPH'85*, pp.31-40, 1985.
- [6] H. W. Jensen, "Importance driven path tracing using the photon map," In *proc. of Sixth Eurographics Workshop on Rendering*, pp.326-335, 1995.
- [7] H. W. Jensen, "Global illumination using photon maps," In *proc. of Seventh Eurographics Workshop on Rendering*, pp.21-30, 1996.
- [8] M. Reichert, "A two-pass radiosity method driven by lights and viewer position," Master's thesis, Cornell University, 1992.
- [9] G. Ritschel, T. Grosch, M. H. Kim, H. -P. Seidel, C. Bacher and J. Kautz, "Imperfect shadow maps for efficient computation of indirect illumination," In *Proc. of ACM SIGGRAPH Asia'08*, no.129, 2008.
- [10] R. Wang, R. Wang, K. Zhou, M. Pan and H. Bao, "An efficient GPU-based approach for interactive global illumination," *Proc. of ACM SIGGRAPH'09*, no.91, 2009.
- [11] T. Ritschel, T. Engelhardt, T. Grosch, H.-P. Seidel, J. Kautz, C. Dachsbaecher, "Micro-rendering for scalable, parallel final gathering," In *Proc. of ACM SIGGRAPH Asia'09*, no.132. 2009.
- [12] T. Ritschel, T. Grosch, H.-P. Seidel, "Approximating dynamic global illumination in image space," In *Proc. of the 2009 symposium on Interactive 3D graphics and games*, pp.75-82, 2009.
- [13] G. Ward and P. Heckbert, "Irradiance gradients," In *Proc. of 3rd Eurographics Workshop on Ren-*

- dering*, pp.85–98, 1992.
- [14] J. Krivánek and P. Gautron, "Radiance caching for efficient global illumination computation," *Journal of IEEE Transactions on Visualization and Computer Graphics*, vol.11, no.5, pp.550–561, 2005.
- [15] R. L. Cook and K. E. Torrance, "A reflectance model for computer graphics," In *Proc. of ACM SIGGRAPH'89*, pp.307–316, 1981.
- [16] S. Zhukov, A. Iones and G. Kronin, "An ambient light illumination model," In *Proc. of Eurographics Workshop on Rendering*, pp.45–56, 1998.
- [17] R. Herzog, V. Havran, S. Kinuwaki, K. Myszkowski and H. -P. Seidel, "Global illumination using photon ray splatting," *Journal of Computer Graphics Forum*, vol.26, no.3, pp.503–512, 2007.



차 득 현

2002년 8월 서강대학교 수학과 졸업(이학사). 2004년 8월 서강대학교 대학원 컴퓨터공학과 졸업(공학석사). 2004년 8월~현재 서강대학교 대학원 컴퓨터공학과 박사과정. 관심분야는 컴퓨터 그래픽스, 물리기반 시뮬레이션, 극사실적 렌더링



임 인 성

1985년 2월 서울대학교 자연과학대학 계산통계학과 졸업(이학사). 1987년 5월 Rutgers-The State University of New Jersey 컴퓨터학과 졸업(이학석사). 1991년 7월 Purdue University 컴퓨터학과 졸업(이학박사). 1999년 7월~2000년 7월 University of Texas at Austin의 TICAM의 연구 교수. 1993년 3월~현재 서강대학교 컴퓨터공학과 교수. 관심분야는 컴퓨터 그래픽스, 과학적 가시화, 고성능 계산

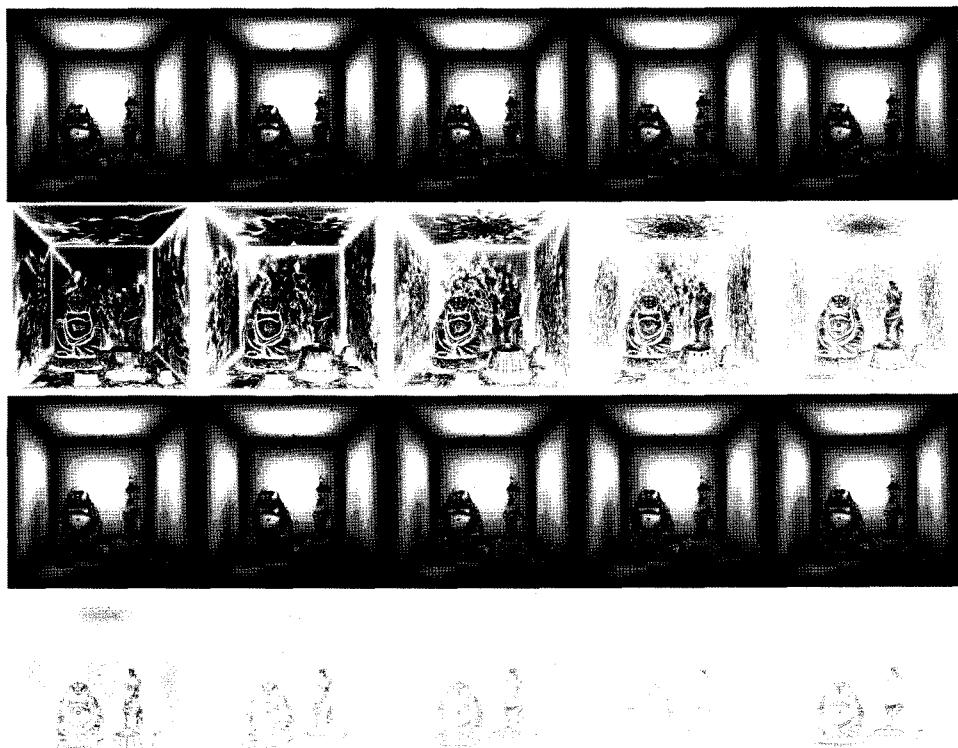
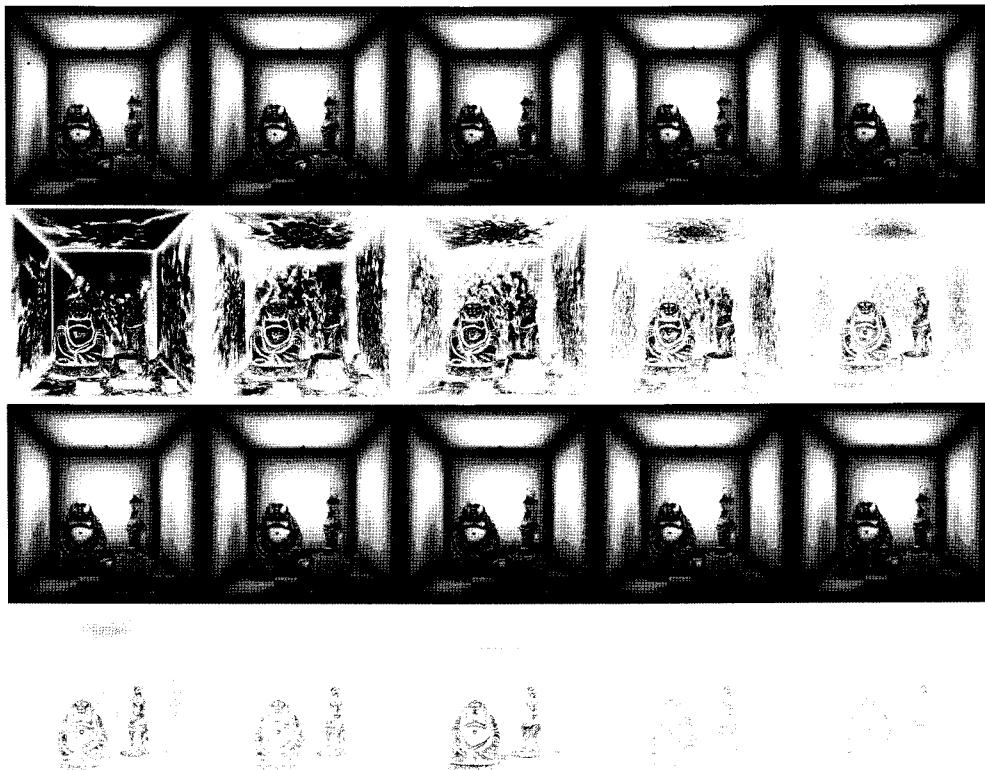
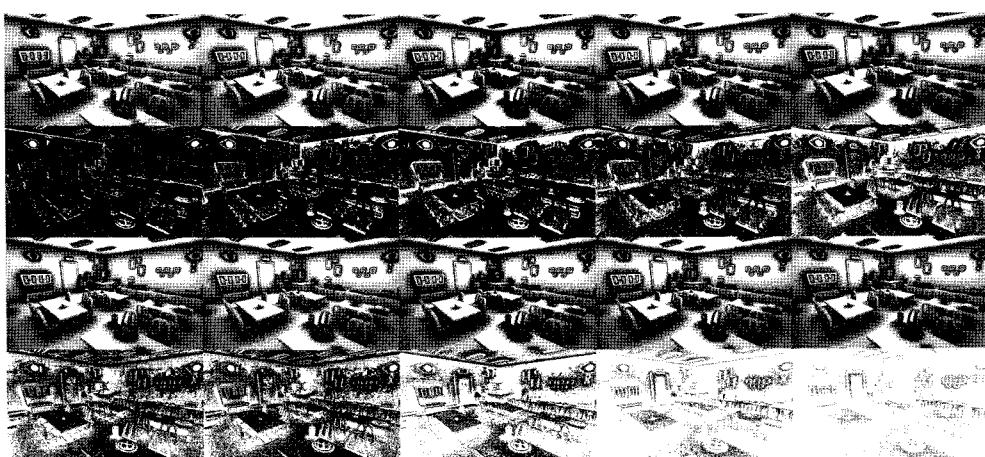


그림 12 장면 A에 대한 수집 광선 개수에 따른 GPU에서의 최종 수집 기반 조화 평균 결과 및 65,536(256×256)개 수집 광선을 사용한 결과에 대한 차이 이미지. 각 장면에 대하여 홀수 행은 조화 평균 거리를 0.14배 하여 그레이스케일로 나타낸 이미지이며, 짝수 행은 앞행의 동일 열의 경우에 대한 조화 평균 거리 값과 65,536개의 수집 광선을 사용한 계산 값의 대한 차이 값을 10배 한 이미지로 흰색이 차이 값 0인 경우를 나타내며 검은색에 가까울수록 차이 값이 커짐을 의미한다. 또한 좌상행에서부터 순서대로 32, 64, 128, 256, 512, 1,024, 2,048, 4,096, 8,192, 16,384 개의 수집 광선을 사용한 결과이다.



(a) 장면 A에서의 조화 평균 거리 계산 결과 및 차이



(b) 장면 B에서의 조화 평균 거리 계산 결과 및 차이

그림 13 수집 광선 개수에 따른 CPU에서의 최종 수집 기반 조화 평균 결과 및 65,536(256×256)개 수집 광선을 사용한 결과에 대한 차이 이미지. 각 장면에 대하여 훌수 행은 조화 평균 거리를 0.14배 하여 그레이스케일로 나타낸 이미지이며, 짹수 행은 앞행의 동일 열의 경우에 대한 조화 평균 거리 값과 65,536개의 수집 광선을 사용한 계산 값의 대한 차이 값을 10배 한 이미지로 흰색이 차이 값 0인 경우를 나타내며 검은색에 가까울수록 차이 값이 커짐을 의미한다. 또한 좌상행에서부터 순서대로 32, 64, 128, 256, 512, 1,024, 2,048, 4,096, 8,192, 16,384 개의 수집 광선을 사용한 결과이다.

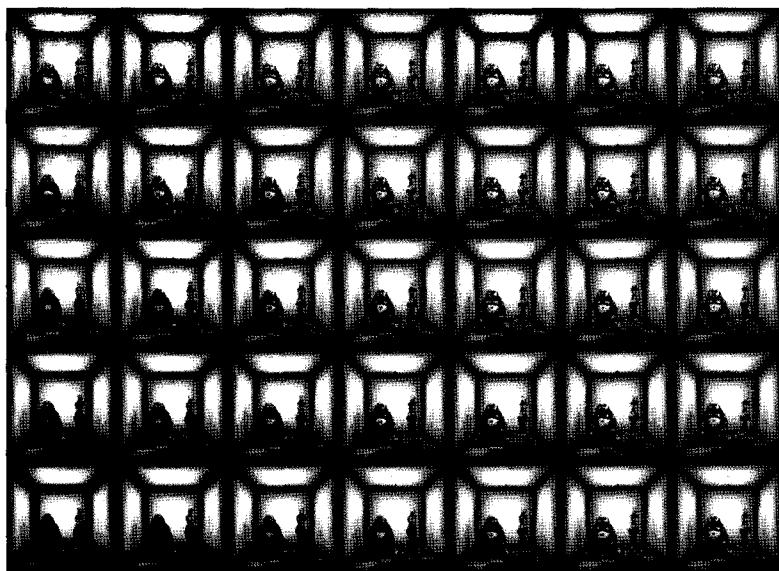


그림 14 포톤 매핑 기반 조화 평균 거리 계산 방법 I에서 포톤 저장 개수 및 수집 개수에 따른 결과. 각 열은 포톤이 저장되는 개수가 약 130,000, 260,000, 530,000, 1,060,000, 2,130,000, 4,260,000, 8,530,000개로 약 2 배씩 증가하고 있는 경우이며, 각 행은 최대 포톤 수집 개수가 64, 128, 256, 512, 1,024개로 증가하는 경우의 조화 평균 거리 값을 0.14배 하여 그레이스케일로 나타낸 결과이다(물체의 모든 표면에서 포톤은 난반사 성질에 의해 다음 방향이 결정되지만 20%의 흡수 성질이 정의되어 있어 저장되는 개수에 약간의 증감이 있다). 또한 포톤은 광원이 아닌 카메라 위치에서 샘플링되어 추적된 경우이며 각 포톤의 최대 반사 개수는 20인 경우이다.

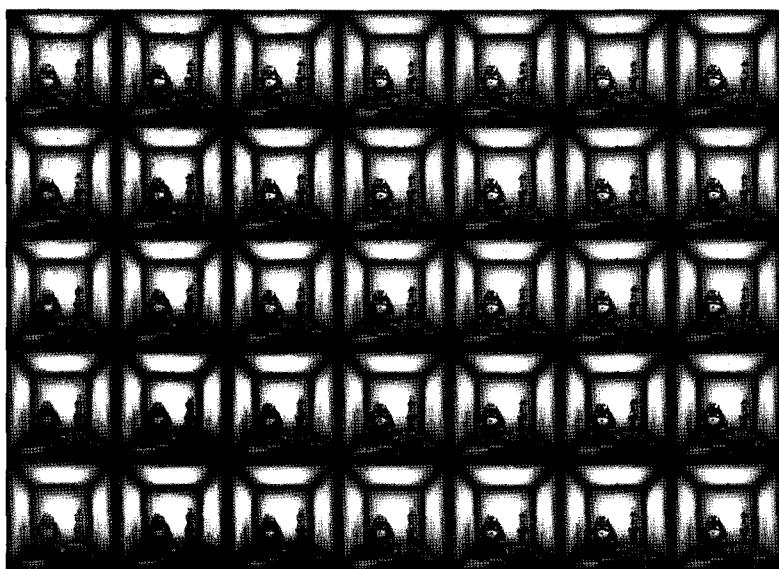


그림 15 포톤 매핑 기반 조화 평균 거리 계산 방법 II에서 포톤 저장 개수 및 수집 개수에 따른 결과. 각 열은 포톤이 저장되는 개수가 약 130,000, 260,000, 530,000, 1,060,000, 2,130,000, 4,260,000, 8,530,000개로 약 2 배씩 증가하고 있는 경우이며, 각 행은 최대 포톤 수집 개수가 64, 128, 256, 512, 1,024개로 증가하는 경우의 조화 평균 거리 값을 0.14배 하여 그레이스케일로 나타낸 결과로서 그림 14에서의 조건과 동일하다.

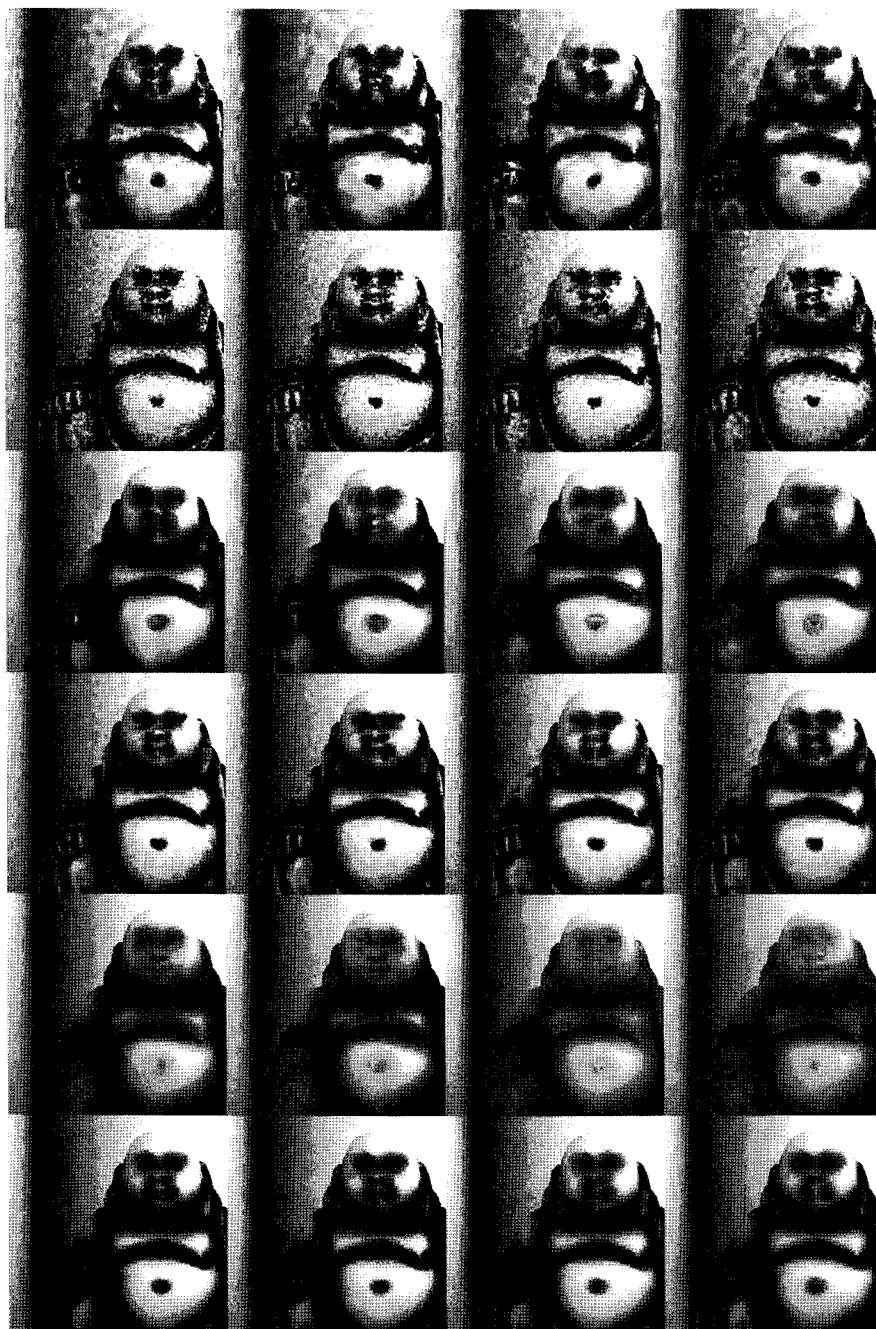


그림 16 장면 A에서 포톤 매핑 기반 조화 평균 거리 계산 방법 II를 사용하여 동일하게 저장된 포톤 개수에 대한 최대 반사 횟수 및 포톤 샘플링 지점에 대한 비교. 각 열은 원쪽에서부터 광원에서 샘플링한 포톤의 최대 반사 횟수가 20번, 80번, 카메라에서 샘플링한 포톤의 최대 반사 횟수가 20번, 80번인 경우이며 각 행은 위쪽에서부터 두 행씩 포톤의 최대 수집 개수가 64, 256, 1024인 경우이다. 또한 홀수 행은 약 1,060,000 개 포톤이 저장된 경우이고 짝수 행은 약 8,530,000개 포톤이 저장된 경우이다. 저장된 포톤 개수를 동일하게 하기 위해 반사 횟수가 20번인 경우에는 80번인 경우에 비해 4배 많은 포톤을 샘플링하는 방식으로 실험하였다. 결과 이미지는 각 픽셀당 거리값에 0.14배 하여 그레이스케일로 표현한 이미지이다.

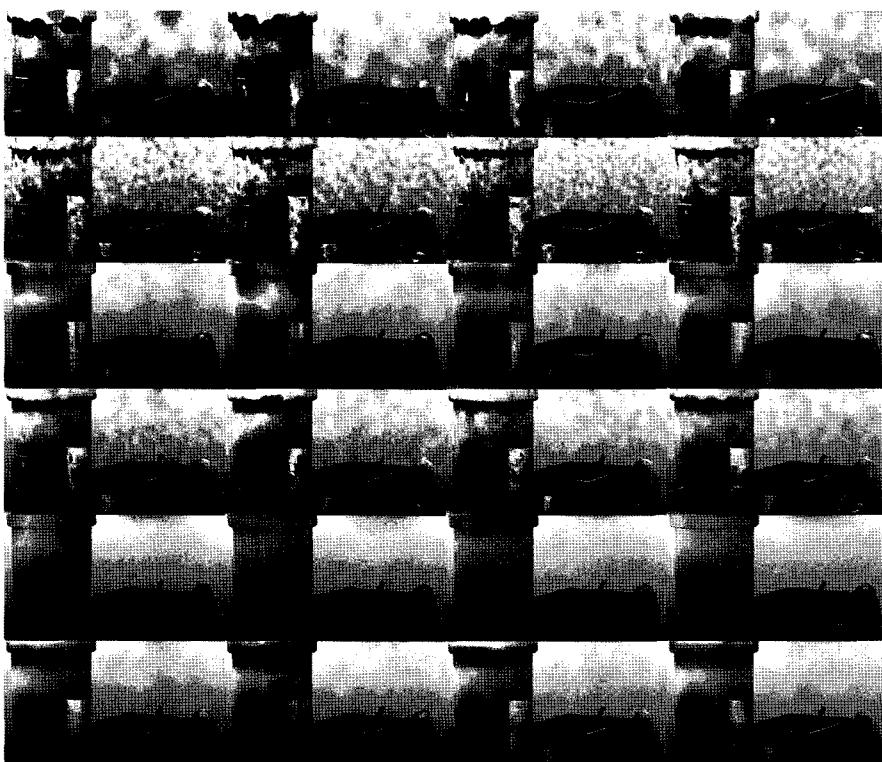


그림 17 장면 B에서 포톤 매핑 기반 조화 평균 거리 계산 방법 II를 사용하여 동일하게 저장된 포톤 개수에 대한 최대 반사 횟수 및 포톤 샘플링 지점에 대한 비교. 각 열은 원쪽에서부터 광원에서 샘플링한 포톤의 최대 반사 횟수가 20번, 80번, 카메라에서 샘플링한 포톤의 최대 반사 횟수가 20번, 80번인 경우이며 각 행은 위쪽에서부터 두 행씩 포톤의 최대 수집 개수가 64, 256, 1024인 경우이다. 또한 홀수 행은 약 1,060,000개 포톤이 저장된 경우이고 짝수 행은 약 8,530,000개 포톤이 저장된 경우이다. 저장된 포톤 개수를 동일하게 하기 위해 반사 횟수가 20번인 경우에는 80번인 경우에 비해 4배 많은 포톤을 샘플링하는 방식으로 실험하였다. 결과 이미지는 각 픽셀당 거리값에 0.025배 하여 그레이스케일로 표현한 이미지이다. 장면 A에 비해 공간의 크기가 큰 장면이므로 평균 거리값을 더 작게 스케일 해주어야 명암에 의한 거리 값 계산 형태를 이미지로 확인할 수 있다.