

가변 포톤 반지름을 이용한 Photon Mapping 렌더링 방법

정재광⁰, 김재정
 전남대학교 전산학과
 spacegate@naver.com⁰, jaykim@chonnam.ac.kr

Photon Mapping Rendering Method using Variable Photon Radius

Jae Kwang Chung⁰, Jay Jeong Kim
 Department of Computer Science, Chonnam National University

요 약

본 논문은 전역조명을 효율적으로 계산하기 위한 렌더링 방법인 photon mapping에서 물체의 조명값을 계산하는데 필요한 radiance를 구하는 새로운 방법을 제안한다. 기존의 방법은 포톤과 표면과의 반응 종류에 상관없이 포톤을 저장할 때 단지 위치와 기타 정보만을 저장하는 방법을 사용했었다. 본 논문에서는 각각의 포톤에 하나의 반지름을 부여하여 물체 표면과의 반응 종류에 따라 가변적으로 반지름을 조절하는 방법을 사용하였다. 구현 후 실험 결과 비슷한 렌더링 시간에 이전에 사용되던 방법에 비해서 caustic과 그림자 표현에서 더 나은 이미지 품질을 보여주었다. 또한 이전 방법에서 caustic을 명확하게 표현하기 위해서 사용했던 caustic map을 사용한 경우에 비해서 더 빠르게 렌더링 되는 결과가 나타났다.

1. 서 론

3차원 상에 존재하는 가상의 물체를 실사와 구별이 불가능할 정도로 사실적으로 렌더링하는 기술은 그래픽스 분야에서 매우 중요한 기술중의 하나이다. 이러한 사실적인 렌더링을 구현하기 위해서는 전역조명(global illumination)의 사실적인 표현이 필수적이라고 할 수 있다. 장면의 전역조명을 계산하기 위해서 radiosity[1], path tracing[2], bidirectional path tracing[3], metropolis light transport[4], photon mapping[5] 등 다양한 알고리즘들이 개발 되어왔다. 이 중에서 최근 실사 렌더링(photorealistic rendering) 분야에서 많이 쓰이고 있는 기술은 photon mapping이다.

기존에 사용되던 photon mapping 방법은 포톤(photon)과 표면과의 반응 종류에 상관없이 위치와 기타 정보만을 저장하였으며, radiance를 구하기 위해서 교차점에서 가장 가까운 특정 개수의 포톤을 검색하는 방법을 사용했었다. 그러나 이러한 방법은 노이즈를 줄이기 위해 많은 수의 포톤을 검색하는 경우 caustic이나 그림자마저 blurring되는 단점이 있었다.

본 논문은 photon mapping에서 물체의 조명값을 계산하는데 필요한 radiance를 구하는 새로운 방법을 제안한다. 포톤 각각에 하나의 반지름을 부여하고 물체 표면과의 반응 종류에 따라 가변적으로 반지름을 조절하는 방법을 사용하였다. 구현 후 실험 결과 비슷한 렌더링 시간에 이전에 사용되던 방법에 비해서 굴절에 의해 빛이 한곳으로 모이는 현상인 caustic과 그림자 표현에서 더 나은 이미지 품질을 보여주었다. 또한 이전 방법에서 caustic을 명확하게 표현하기 위해서 사용했던 caustic map을 사용한 경우에 비해서 더 빠르게 렌더링 되는 결과가 나타났다.

2. 기존의 photon mapping 방법

photon mapping은 기존의 광선추적법(ray tracing)과 radiosity에서 구현이 어려웠던 완전한 전역조명 효과를 Monte Carlo 기법에 기초하여 효과적으로 나타낼 수 있는 렌더링 방법이다. 또한 photon mapping은 Monte Carlo 기법에 기초한 path tracing과 같은 전역조명의 구현이 가능한 다른 렌더링 방법에 비해서 노이즈도 적으면서 획기적으로 빠르게 렌더링이 가능하다. 그리고 radiosity처럼 장면의 메쉬화 과정이 필요 없다는 장점을 지닌다.[5]

photon mapping은 photon tracing, rendering 두개의 단계로 이루어져 있는 two-pass method이다.

2.1 Photon tracing

광원에서 퍼져 나오는 수많은 빛의 입자들을 포톤(photon)이라고 하며, 포톤은 빛의 에너지 flux를 가지고 3차원 공간을 진행한다. 이러한 포톤이 물체에 반응하면서 무한히 진행하면 결국 에너지 평형상태에 이른다. 이러한 에너지 평형상태가 실제 전역조명이 부여된 전체 장면의 밝기를 나타낸다. photon mapping은 에너지 평형상태를 russian roulette이라는 확률적 시뮬레이션으로 근사한다.

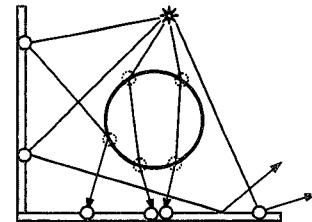


그림 1. 기존의 photon tracing 과정

[그림 1]에서와 같이 포톤은 광원의 랜덤한 지점에서 광원의 노멀벡터와 방출방향의 cosine에 비례하는 확률분포에 의해 랜덤한 방향으로 방출된다. 방출된 포톤은 광선추적법과 같은 방법으로 가장 먼저 만나는 물체와의 교차점을 찾는다. 교차하는 물체의 diffuse, specular 특성 값에 따라 russian roulette이라는 확률적 방법으로 diffuse 반사, specular 반사, 흡수라는 세가지 경로 중 하나를 결정한다. diffuse 반사하는 경우는 포톤의 정보(위치, flux, 입사방향, 기타 정보)를 photon map이라는 자료구조에 저장하고 diffuse 반사 방향으로 진행을 계속한다. 반면 specular 반사의 경우에는 포톤의 정보는 저장하지 않고 specular 반사 방향으로 진행한다. 마지막으로 흡수의 경우에 포톤의 수명이 끝나게 된다. 이러한 과정을 다음 [표 1]의 알고리즘에 따라 photon map이 가득 찰 때까지 반복한다.

이렇게 저장된 포톤은 rendering 단계에서 효율적으로 검색될 수 있도록 photon map에 balanced kd-tree 같은 이진 공간분할 자료구조로 전처리 되어 저장된다.

표 1. Photon tracing 과정을 나타내는 의사코드

```

PhotonTracing(dir, flux) {
  r = random() // r ∈ [0,1]
  If (r < diffuse) {
    // diffuse reflection
    photon map에 photon 저장
    dir = diffuse random direction
    PhotonTracing(dir, flux)
  } else if (r < diffuse + specular) {
    // specular reflection
    dir = specular direction
    PhotonTracing(dir, flux)
  } else {
    // absorption
  }
}
    
```

2.2 Rendering

rendering 단계에서는 광선추적법과 같은 방법으로 장면을 렌더링 한다. 광선과의 교차지점 x 에서의 조명값을 계산하기 위하여 photon map의 정보가 이용된다. 즉 장면에서의 반사 radiance $L_r(x, \vec{\omega}_r)$ 를 계산하기 위해서 교차점 x 주위에 존재하는 포톤들을 찾아 다음의 식으로 $L_r(x, \vec{\omega}_r)$ 를 구한다.[5]

$$\begin{aligned}
 L_r(x, \vec{\omega}_r) &= \int_n f_r(x, \vec{\omega}, \vec{\omega}_r) L_r(x, \vec{\omega}) (\vec{\omega}_r \cdot \vec{n}) d\vec{\omega} \\
 &= \int_n f_r(x, \vec{\omega}, \vec{\omega}_r) \frac{d^2\Phi_p(x, \vec{\omega}_r)}{(\vec{\omega}_r \cdot \vec{n}) d\vec{\omega} dA} (\vec{\omega}_r \cdot \vec{n}) d\vec{\omega} \\
 &= \int_n f_r(x, \vec{\omega}, \vec{\omega}_r) \frac{d^2\Phi_p(x, \vec{\omega}_r)}{dA} \\
 &\approx \sum_{p=1}^N f_r(x, \vec{\omega}_p, \vec{\omega}_r) \frac{\Delta\Phi_p(x, \vec{\omega}_p)}{\Delta A} \\
 &\approx \frac{1}{\pi r^2} \sum_{p=1}^N f_r(x, \vec{\omega}_p, \vec{\omega}_r) \Delta\Phi_p(x, \vec{\omega}_p)
 \end{aligned}$$

검색반지름 r 인 구 안에 들어있는 N 개의 가장 가까운 포톤을 kd-tree 라는 자료구조를 통해서 검색한 후 각각의 포톤의 flux $\Delta\Phi_p(x, \vec{\omega}_p)$ 와 BRDF $f_r(x, \vec{\omega}_p, \vec{\omega}_r)$ 의 곱을 더하고 원의 면적 πr^2 으로 나누어 교차점 x 에서의 반사 radiance $L_r(x, \vec{\omega}_r)$ 를 계산한다. 또한 cone filter를 사용하여 각 포톤과 교차점과의 거리 d_p 에 따라 다음의 식으로 계산하면 이미지에 발생하는 노이즈를 줄일 수 있다.[5]

$$\begin{aligned}
 w_{pc} &= 1 - \frac{d_p}{kr} \quad k: \text{filter constant } (k \geq 1) \\
 L_r(x, \vec{\omega}_r) &\approx \frac{1}{(1 - \frac{2}{3k})\pi r^2} \sum_{p=1}^N f_r(x, \vec{\omega}_p, \vec{\omega}_r) \Delta\Phi_p(x, \vec{\omega}_p) w_{pc}
 \end{aligned}$$

3. 제안 방법

photon mapping에서 radiance를 구하기 위한 이전방법에서는 이미지의 노이즈를 줄이기 위해서는 검색반지름과 검색 포톤 수를 늘려야 하지만 그 때문에 caustic이나 그림자마저 blurring되는 단점이 있었다. 이러한 문제점을 해결하기 위해서 caustic map을 따로 사용하지만 또 하나의 photon map을 추가하는 것이므로 렌더링 속도가 느려지게 된다.

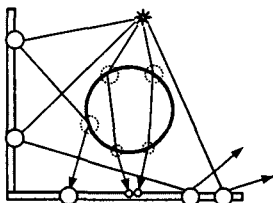


그림 2. 제안 방법에서의 photon tracing 과정

본 논문에서는 따로 caustic map을 만들지 않고도 [그림 2]와 같이 포톤마다 반지름을 부여하여 caustic이나 그림자를 명확하게 렌더링 하는 방법을 제안한다.

각각의 포톤에 초기 포톤반지름 r_i 을 부여한다. photon tracing 단계에서 포톤이 진행 도중에 굴절을 하게 되면 일정 비율 k 로 포톤의 반지름을 줄인다. 또한 굴절 하는 포톤의 flux에 k 로 나누어 flux를 조절한다.

rendering 단계에서는 kd-tree를 이용하여 모든 포톤의 최대반지름 r_{max} 안에 있는 포톤들을 검색한다. 검색된 각각의 포톤의 위치는 p_p , 포톤의 반지름은 r_p , 광선과의 교차점은 x 라고 할 때 다음의 식과 [그림 3]에서처럼 cone filter를 사용하여 포톤의 flux를 blending 한다.

$$\begin{aligned}
 w_{pc} &= 1 - \frac{\|x - p_p\|}{r_p} \\
 L_r(x, \vec{\omega}_r) &\approx \frac{1}{(1 - \frac{2}{3})\pi r_{max}^2} \sum_{p=1}^N f_r(x, \vec{\omega}_p, \vec{\omega}_r) \Delta\Phi_p(x, \vec{\omega}_p) w_{pc}
 \end{aligned}$$

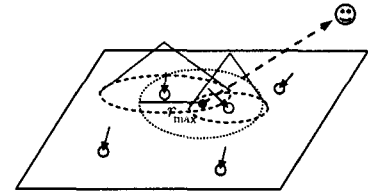


그림 3. cone filter를 사용한 blending

이러한 방법으로 포톤의 역할에 따라 포톤의 반지름을 조절하므로 노이즈가 없어야 할 부분은 부드럽게 blurring되며, 명확하게 나타나야 할 부분은 선명하게 나타나는 효과를 갖게 되는 것이다.

4. 실험 결과

제안 방법과 이전 방법을 모두 Pentium4 1.8GHz PC 환경에서 Visual C++로 구현하여 실험을 하였다.

그림 4-1, 4-2, 4-3은 모두 500개의 포톤을 이용하여 caustic map을 사용하지 않고 300x300 pixel의 해상도로 렌더링 하였다.

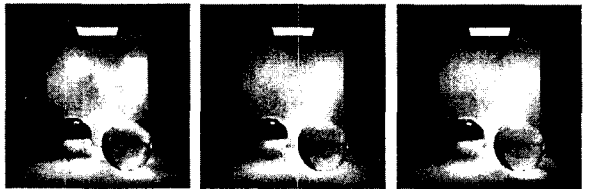


그림 4-1 그림 4-2 그림 4-3

	이전 방법		제안 방법
그림 번호	그림 4-1	그림 4-2	그림 4-3
검색 포톤수	20	80	
r	30	30	
r_i			30
k			70%
렌더링 시간	1.18 초	1.25 초	1.13 초
특징	caustic, 그림자 보임. 노이즈 발생	caustic, 그림자 상실. 약간의 노이즈	caustic, 그림자 보임. 약간의 노이즈

(r :검색반지름, r_i :초기검색반지름, k :굴절된 포톤의 반지름 조절비율)

그림 5-1, 5-2, 5-3 은 모두 10,000 개의 포톤을 이용하여 caustic map 을 사용하지 않고 300x300 pixel 의 해상도로 렌더링 하였다.

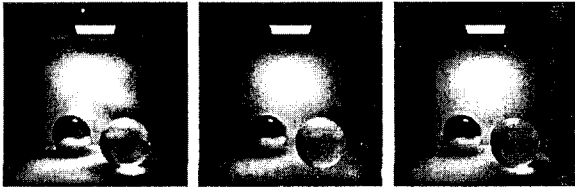


그림 5-1 그림 5-2 그림 5-3

그림 번호	이전 방법		제안 방법
	그림 5-1	그림 5-2	그림 5-3
검색 포톤수	300	900	
r	25	25	
r_f			25
k			70%
렌더링 시간	6.01 초	6.5 초	7.04 초
특징	caustic, 그림자 보임. 노이즈 발생	caustic, 그림자 상실. 노이즈 없음	caustic, 그림자 보임. 노이즈 없음.

그림 4-1, 5-1 은 유리구 아래에서 caustic 과 그림자가 나타났지만 검색포톤수가 작기 때문에 노이즈가 심하게 발생함을 볼 수 있다. 그림 4-2, 5-2 는 검색포톤수가 많아 노이즈는 발생하지 않으나 caustic 과 그림자가 상실되었다. 그림 4-3, 5-3 은 노이즈도 발생하지 않으며, caustic 과 그림자가 그대로 남아 있음을 알 수 있다. 또한 포톤 수를 줄일수록 더욱 많은 차이가 발생함을 알 수 있었다.

다음으로 이전의 방법에 caustic map 을 사용한 것과 제안된 방법만을 사용한 것과의 비교를 위해서 같은 품질의 이미지나 나옴도 설정값을 조절하여 렌더링 시간을 비교해보았다.

그림 6-1, 6-2 는 모두 300x300pixel 의 해상도로 100,000 개의 포톤을 방출해 바닥에 놓인 유리구를 렌더링한 이미지이다.

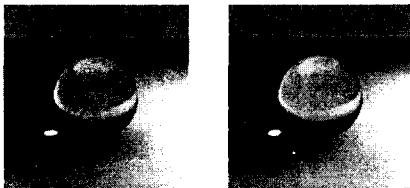


그림 6-1 그림 6-2

그림 번호	이전 방법		제안 방법
	그림 6-1	그림 6-2	그림 6-2
검색 포톤수	2000		
r	10		
r_f			10
k			40%
렌더링 시간	11.7 초		9.0 초
기타	caustic map 사용 (포톤 2,000 개)		

그림 7-1, 7-2 는 모두 300x300pixel 의 해상도로 500,000 개의 포톤으로 도넛 모양의 유리를 렌더링한 이미지이다.

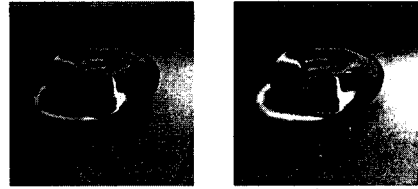


그림 7-1 그림 7-2

	이전 방법	제안 방법
그림 번호	그림 7-1	그림 7-2
검색 포톤수	9000	
r	10	
r_f		10
k		45%
렌더링 시간	97.51 초	66.41 초
기타	caustic map 사용 (포톤 40,000 개)	

여기서 그림 6-1, 7-1 은 각각 2,000 개, 40,000 개의 포톤을 담을 메모리 공간을 추가로 더 필요로 한다. 위의 결과에서 보듯이 같은 품질의 이미지를 생성하는데 제안된 방법이 caustic map 을 생성하지 않아도 되므로 더 빠르며, 메모리 사용량도 줄일 수 있다는 것을 알 수 있다.

5. 결론

기존의 photon mapping 방법은 포톤과 표면과의 반응 종류에 상관없이 단지 위치와 기타 정보만을 저장하는 방법을 사용했었다. 그러나 노이즈를 줄이기 위해 많은 포톤을 검색하게 되면 caustic 이나 그림자 마저 blurring 되는 단점이 있어 따로 caustic map 이라는 photon map 을 추가로 사용했었다. 본 논문에서는 caustic map 을 따로 사용할 필요가 없도록 포톤 각각에 하나의 반지름을 부여하여 물체 표면과의 반응 종류에 따라 가변적으로 반지름을 조절하는 방법을 사용했다. 실험 결과 제안된 방법이 같은 렌더링 시간에 이전에 사용되던 방법에 caustic map 을 사용하지 않은 경우에 비해서 caustic 과 그림자 표현에서 더 나은 이미지 품질을 보여줌을 알 수 있었다. 이러한 품질차이는 포톤 수를 줄일수록 더욱 많은 차이가 발생했다. 또한 이전 방법에 caustic map 을 사용한 경우에 비해서 더 빠른 결과가 나타났다.

앞으로 이전 방법에서 사용되던 다양한 전역조명 효과들이 제안된 방법으로 구현이 가능한지 여부를 살펴보는 연구가 필요하다. 또한 포톤의 반지름 조절을 굴절 외에 다른 반응에서도 응용할 수 있는지에 대한 연구도 필요하다.

참고문헌

- [1] Cindy M. Goral, Kenneth E. Torrance, Donald P. Greenberg, and Bennett Battaile. "Modelling the interaction of light between diffuse surfaces." Computer Graphics (Proc. SIGGRAPH '84) 18(3): 212-22 (July 1984).
- [2] James T. Kajiya. "The rendering equation." Computer Graphics (Proc. SIGGRAPH '86) 20(4): 143-150 (August 1986).
- [3] Eric P. Lafortune and Yves D. Willems. "Bidirectional path tracing." In Compugraphics '93, pp. 95-104, 1993.
- [4] Eric Veach and Leonidas J. Guibas. "Metropolis light transport." In Proceedings of SIGGRAPH 97, Computer Graphics Proceedings, Annual Conference Series, pp. 65-76, August 1997.
- [5] Hanrik Wann Jensen, "Realistic Image Synthesis Using Photon mapping", A K Peters LTD., 2001