

연기 파티클에 대한 포톤 매핑 기반의 렌더링 기법

송기동¹ 임인성²

midchow@hanmail.net ihm@sogang.ac.kr

¹매크로그래프²서강대학교 컴퓨터공학과

Photon Mapping-Based Rendering Technique for Smoke Particles

Kidong Song¹ Insung Ihm²¹Macrograph²Department of Computer Science & Engineering, Sogang University

연기와 같은 유체의 모습을 영화나 애니메이션에서의 특수 효과에 활용하기 위해서는 연기를 사실적으로 모델링하는 과정과 모델링된 연기 내부에서의 빛의 흐름이 잘 반영된 렌더링 과정이 필요하다. 컴퓨터 그래픽스 분야에서는 연기 모델링의 사실성을 살리기 위해 물리 기반의 유체 시뮬레이션 기법을 많이 차용하고 있는데, 그동안 시뮬레이션 기법으로 주로 연구되어 온, 격자 기반의 Euler 방법과는 근본적으로 다른, 파티클 기반의 Lagrange 방법이 시뮬레이션 단계에서 얻을 수 있는 장점 때문에 최근 관심이 높아지고 있다. 연기 렌더링은 연기 모델링 방법에 종속적일 수밖에 없으므로, 결과적으로 격자 기반의 시뮬레이션 결과에 대한 렌더링 방법은 많이 연구되고 있는 데 비해, 파티클 형태로 산출된 연기 데이터에 대하여 사실적인 영상을 생성해주는 렌더링 기술에 대한 연구는 아직 부족한 상황이다. 이에, 본 논문에서는 Lagrange 기법을 적용하여 생성한 파티클 집합 형태의 연기 시뮬레이션 데이터를 사실적으로 렌더링하기 위해, 전역 조영을 위한 최신 렌더링 기술인 포톤 매핑 기법을 파티클 데이터에 맞게 변형 및 확장한 파티클맵 기법을 소개하고, 개선된 파티클맵 기법을 제시하여, 기존 연구와의 차이점을 보여준다. 또한 렌더링 과정에서 효율성을 높이기 위해 볼륨 렌더링 방정식의 다중 산란 항을 미리 계산하는 광도맵이라는 방법을 제시한다.

Abstract

To realistically produce fluids such as smoke for the visual effects in the films or animations, we need two main processes: a physics-based modeling of smoke and a rendering of smoke simulation data, based on light transport theory. In the computer graphics community, the physics-based fluids simulation is generally adopted for smoke modeling. Recently, the interest of the particle-based Lagrangian simulation methods is increasing due to the advantages at simulation time, instead of the grid-based Eulerian simulation methods which was widely used. As a result, because the smoke rendering technique depends heavily on the modeling method, the research for rendering of the particle-based smoke data still remains challenging while the research for rendering of the grid-based smoke data is actively in progress. This paper focuses on realistic rendering technique for the smoke particles produced by Lagrangian simulation method. This paper introduces a technique which is called particle map, that is the expansion and modification of photon mapping technique for the particle data. And then, this paper suggests the novel particle map technique and shows the differences and improvements, compared to previous work. In addition, this paper presents irradiance map technique which is the pre-calculation of the multiple scattering term in the volume rendering equation to enhance efficiency at rendering time.

키워드 : 연기 파티클, 볼륨 렌더링 공식, 광선 추적법, 포톤 매핑, 파티클 맵, 광도 맵

Keywords : smoke particle, volume rendering equation, ray tracing, photon mapping, particle map, irradiance map

1. 서론

1.1 연구 배경 및 기존 연구

연기나 가스 같은 반투과 물질(participating media)

을 컴퓨터를 이용하여 사실적으로 합성해내기 위해서는 연기의 소용돌이(vortex)가 잘 반영된 물리 기반의 연기 모델링 과정이 필요하고, 생성된 연기 모델에 대해 빛이 통과하면서 발생하는, 빛과 연기의 상호 작용이 사실적으로 반영된 렌더링 과정이 필요하다[30]. 사실적인 연기 모

렌더링을 위하여, 컴퓨터 그래픽스 분야에서는 [14] 이래로 유체 역학 분야에서 오랜 기간 동안 연구되어온 물리 기반 유체 시뮬레이션 기법을 차용하여 연기, 가스, 폭발 등 다양한 유체의 모습을 영상 산업의 특수 효과 생성 기술에 활용하고 있다. 유체 시뮬레이션 방법은 크게 두 가지로 나뉘는데, 첫 번째는 격자 기반의 Euler 방법으로서, 유체가 존재하는 공간을 일정한 크기를 갖는 볼륨 데이터 형태의 격자로 나누고, 각 격자점에서 연기의 속도 및 밀도를 계산하는 방식을 취한다[6, 8, 14, 29]. 유체 시뮬레이션의 또 다른 방법은, 격자가 아니라 연속 공간에서 정의된 유체의 성질을 이산적으로 샘플링 한 파티클(particle)을 통하여 유체를 표현하는 Lagrange 방법이다. 파티클을 이용하여 연기를 표현하는 것은 [25]에서 도입된 이래로 [28] 등에서 시도되어 왔으며, 특히 천체 물리학 분야에서 수많은 파티클의 상호 작용을 모델링 하기 위해, Lucy[10]와 Gingold, Monaghan 등[16, 20]에 의해 제안된 SPH(Smoothed Particle Hydrodynamics) 방법은 [5]에 의해 컴퓨터 그래픽스 분야에 소개된 이후 가장 많이 사용되고 있는 Lagrange 방법이다. 이 외에도 또 다른 Lagrange 방법인 와법(vortex method)을 그래픽스 분야에 적용하여 연기 시뮬레이션을 시도하고 있다[1, 9].

현재 격자 기반의 방법은 파티클 기반의 방법에 비해 컴퓨터 애니메이션 분야에서 더 많이 활용되고 있으며, 상당히 자연스러운 연기의 모습을 생성할 수 있는 등 여러 장점이 있는 반면, 방법의 특성상 발생하는 수치적인 소실로 인해 시뮬레이션이 진행될수록 과도하게 스므딩이 발생하는 문제가 있다. 이러한 문제를 해결하기 위해, [7, 26]에서와 같이 격자-파티클 혼합 기법(hybrid method)도 제시되고 있는 등, 격자 기반 방법의 연구는 활발한데 비해, Lagrange 방법은 연기의 소용돌이를 강조하기 위해 격자 기반의 보조적인 수단으로만 주로 사용되고 있다. 하지만 최근, SPH의 확장인 MPS(Moving Particle Semi-implicit)를 물 시뮬레이션에 활용하거나[21, 23], 순수하게 Lagrange 방법에 기반을 두고 와법을 활용하여 연기 시뮬레이션 하는 등[22], 파티클 기반의 유체 애니메이션 기법에 대한 관심이 높아짐에 따라 파티클 형태로 산출된 유체 시뮬레이션 데이터에 대하여 사실적인 영상을 산출해주는 렌더링 기술의 중요성이 더욱 높아지고 있다.

연기에 대한 렌더링은 Blinn에 의해서 제시된 [4]의 방법을 시초로 활발히 연구가 진행되어 왔다. 과거 연기 렌더링은 계산 복잡도를 줄이기 위하여 단일 산란(single scattering)만을 고려한 단순화된 모델이었기 때문에, 알베도가 낮은 연기에 대해서는 적합했지만 알베도가 높은 연기에서는 사실성이 떨어지는 결과를 보여주었다[15, 18]. 하지만 물체 표면에 대한 전역 조명(global illumination)을 위해 [11]에서 새롭게 제시된, Monte Carlo 광선 추적법 기반의 포톤 매핑 기법을, Jensen 등이 [13]에서 반투과 물질에까지 확장하여 적용함으로써, 빛이 연기 안에서 산란되는 과정인 다중 산란(multiple scattering)까지 고려할 수 있게 되어, 사실성이 상당히 높아지게 되었다.

기존의 렌더링 방법들은 기본적으로 격자 기반의 볼륨에 대해 각 격자점에서의 연기의 밀도와 여러 가지 광학 성질을 구한 후, 렌더링 방정식에 적용하여 합산하는 방식을 취한다. 하지만 이런 격자 기반의 렌더링 방식은 격자의 해상도가 낮으면 앨리어싱이 발생하고, 해상도를

높이면 저장 공간이 낭비된다는 단점이 있다. 이러한 한계를 극복하기 위해 Foster 등은 [8]에서 파티클을 격자의 밀도 계산에 반영하는 방법을 제시했다. 이와는 달리 근본적으로 격자를 떠나서 파티클로 모델링 된 연기 데이터에 대해 직접 렌더링 하는 파티클 렌더링 기법 역시 널리 쓰여 왔으나, 상당히 제한적이었으며, 이미지의 사실성도 상당히 떨어지는 결과를 보여주었다[25]. 이에 파티클로 모델링 된 연기 데이터에 대해 볼륨 포톤맵 기법을 적용하는 시도도 있었는데, Adabala 등은 [1]에서 포톤의 산란을 계산하는 단계에서 포톤맵을 구성하지 않고 포톤의 정보를 포톤 주변의 파티클에 저장하는 파티클맵이라는 기법을 제시하였다. 이 방법은 연기 파티클에 대해 전역 조명이 지원되는 직접 렌더링 기법을 제시했다는 점에서 의미가 있으나, 시점에 독립적인 포톤맵의 장점을 살리지 못했기 때문에, 시점이 바뀔 때 마다 포톤을 새로 뿌리고 렌더링을 해야 하는 한계가 있었다.

1.2 본 논문의 기여도

본 논문에서는 물리 기반의 유체 시뮬레이션 방법 중 하나인 Lagrange 기법을 적용하여 생성된 파티클 집합 형태의 연기 시뮬레이션 데이터로부터 사실적인 이미지를 생성해내기 위해, 전역 조명을 위한 최신 렌더링 기술인 볼륨 포톤 매핑(volume photon mapping) 기법을 파티클 데이터에 맞게 변형 및 확장한다. 특히 [1]에서 제시된 시점에 의존적인 방법과는 다르게, 시점에 독립적인 파티클맵을 구성하는 방법을 제안하고, 렌더링 과정에서 효율성을 높이기 위한 기법을 제시하여 성능을 비교한다.

2. 볼륨 포톤 매핑 기법 소개

2.1. 반투과 물질에서의 빛의 이동: 볼륨 렌더링 방정식

2.1.1. 볼륨 렌더링 방정식

빛이 반투과 물질을 통과할 때는 크게 세 가지의 물리적 작용, 즉 흡수(absorption), 발광(emission), 그리고 산란(scattering) 현상이 발생하는데, 이러한 현상을 수학적으로 표현해 주는 것이 볼륨 렌더링 방정식이다.

$$L(x, \vec{w}) = \int_0^s e^{-\tau(x, x')} \sigma_a(x') L_e(x', \vec{w}) dx' + \int_0^s e^{-\tau(x, x')} \sigma_s(x') \int_{\Omega_{x'}} p(x', \vec{w}', \vec{w}) L_i(x', \vec{w}') d\vec{w}' dx' + e^{-\tau(x, x+s\vec{w})} L(x+s\vec{w}, \vec{w}), \text{ where } \tau(x, x') = \int_x^{x'} \sigma_t(t) dt$$

이 방정식은 세 개의 항의 합으로 표현이 되는데, 여기서 $\sigma_a(x)$ 와 $\sigma_s(x)$ 는 각각 각 지점 x 에서 흡수와 산란 현상의 정도를 나타내는 계수를 나타내고, 그들의 합 $\sigma_t(x)$ 는 빛이 진행하는 과정에서 얼마나 빨리 감쇄되는지를 나타내는 척도가 된다. 또한 $p(x', \vec{w}', \vec{w})$ 는 빛이 산란되는 방향을 나타내는 위상함수로서 2.2.1절에 설명이 되어 있다. 이 방정식에서 첫번째항은 반투과 물질의 지점 x 에서 $L_e(x, \vec{w})$ 의 세기로 발광하고 있는 빛이 전체 경로 상에서 누적되는 현상을 표현하며, 두 번째 항은 바라보고 있는 방향에 해당하는 경로로 들어오는 산란 효과

(in-scattering)를 표현해주고 있다. 마지막으로 세 번째 항은 최초 $L(x + s\vec{w}, \vec{w})$ 의 세기로 반투과 물질에 들어온 빛이 경로를 따라 진행하면서 감쇄 현상으로 인하여 어두워지는 효과를 표현해주고 있다. 반투과 물질을 포함하고 있는 장면을 사실적으로 렌더링 하기 위해서는 이 방정식을 풀어, 반투과 물질을 통과하면서 복잡하게 반응하는 빛의 변화를 추적하여 결국 어떠한 빛이 눈으로 들어오게 되는지를 계산해야 하며, 물리적으로 정확한 계산을 위해서는 방대한 양의 계산을 요한다.

2.1.2. 광선 진행법(Ray Marching)

볼륨 렌더링 방정식은 복잡한 형태를 갖고 있기 때문에, 직접적인 계산이 용이하지 않다. 따라서 컴퓨터 그래픽스 분야에서는 수치 적분 방법을 사용하는 광선 추적법(ray tracing) 기반의 광선 진행법(ray marching)을 사용하는데, 이것은 빛이 투과하는 경로를 추적하면서 유한한 샘플링 지점에서의 빛의 변화를 계산하는 방법이다. 광선 진행법으로 볼륨 렌더링 방정식을 수치적으로 근사화하면 다음과 같은 식을 얻을 수 있다[12].

$$L_n(x, \vec{w}) = \sum_i^N L_i(x, \vec{w}_i) p(x, \vec{w}_i, \vec{w}) \sigma_s(x) \Delta x + \left\{ \frac{1}{S} \sum_{s=1}^S L_s(x, \vec{w}_s) p(x, \vec{w}_s, \vec{w}) \right\} \sigma_s(x) \Delta x + e^{-\sigma_t(x)\Delta x} L_{n-1}(x + \Delta x \vec{w}, \vec{w})$$

광선 진행법의 경우도 샘플링 간격을 촘촘히 해야 정확한 내부 산란(in-scattering)을 구현할 수 있다는 비용적인 부담을 안고 있다. 내부 산란은 단일 산란(single scattering)과 다중 산란(multiple scattering)으로 나눌 수 있는데, 다중 산란은 광원에서 출발한 빛이 연기 안에서 산란되면서 샘플링 지점에 미치는 영향을 계산하는 것이므로, 광원에서 직접 들어오는 빛만 고려하는 단일 산란에 비해 비용이 매우 크다. 하지만 비용을 줄이기 위하여 다중 산란을 생략하면, 대부분 경우에 회색조의 밋밋한 이미지로 렌더링 된다. 따라서 사실적인 연기 렌더링을 위해서는 다중 산란이 반드시 구현되어야 하며, 최소의 비용으로 매우 복잡한 다중 산란 효과를 구현하는 것이 중요하다.

2.2. 포톤 매핑

비교적 적은 비용으로 다중 산란을 구현할 수 있는 방법으로 볼륨 포톤 매핑 기법이 많이 쓰이고 있는데, 구현이 간단하면서도 매우 사실적인 이미지를 얻을 수 있다 [6, 13]. 렌더링 전처리 단계에 광원으로부터 일정 개수의 포톤을 빛의 진행 방향으로 뿌린 후, 각 포톤이 연기와 반응하여 산란되는 정보를 포톤맵(photon map)이라는 자료 구조로 구성한다. 렌더링 단계에서는 임의의 샘플링 지점에서 주변의 일정 개수의 포톤을 포톤맵으로부터 검색해서 광도를 계산하는 기법이다. 포톤맵을 구성하는 단계에서는 포톤의 이동 방향과 포톤의 이동 거리를 구하는 것과 포톤의 소멸 여부를 결정하는 방법이 중요한데, 여기서는 [12]에서 제시하는 방법을 사용하였다.

2.2.1. 포톤의 이동 방향

포톤이 저장된 후, 현재 포톤의 위치를 원점으로 하고, 포톤이 날아온 방향을 z축으로 하는 구면좌표계에서, 다음 이동을 위한 새로운 방향 $\vec{w}_d = (\theta, \phi)$ 은, 0과 1사이의 균일하게 샘플링 된 임의의 두 수, ξ_1, ξ_2 를 만들어서 다음과 같이 구한다.1)

$$\vec{w}_d = (\theta, \phi) = (\cos^{-1}(\sqrt{\xi_1}), 2\pi\xi_2), \xi_1, \xi_2 \in [0, 1]$$

이 때, 구면 위의 모든 지점에 대하여 동일한 가중치를 가지는 균일 샘플링(uniform sampling) 대신 특정 방향에 가까울수록 높은 가중치를 주는 중요 샘플링(importance sampling)을 적용할 수 있는데, 위상 함수(phase function)를 사용하여 포톤의 방향성에 물리적인 성질을 쉽게 부여할 수 있다. 여기서 물리적 성질이란, 포톤이 산란될 때, 날아온 방향성을 유지하는지의 여부인데, 포톤이 날아온 방향을 유지하면서 산란되는 것을 전방 산란(forward scattering)이라고 하고, 반대 방향으로 산란되는 것을 후방 산란(backward scattering)이라고 하며, 두 경우를 합쳐서 비등방성 산란(anisotropic scattering)이라고 한다. 반대의 경우는 등방성 산란(isotropic scattering)이라고 하는데, 포톤이 어디에서 날아왔는지에 상관없이 샘플링 지점에서 임의의 방향으로 산란되게 된다. 위상 함수는 반투과 물질에서 빛이 산란할 때, 분산되는 분포를 나타내주는 함수이다. 가장 많이 쓰이는 것은 Henyey-Greenstein 위상 함수로, 다음과 같은 형태를 갖는다.

$$P(\theta) = \frac{1-g^2}{4\pi(1+g^2-2g\cos\theta)^{1.5}}, g \in (-1, 1)$$

여기서 g 는 비대칭 인자(asymmetry parameter)로, 이 값을 통해 포톤의 방향성을 조절할 수 있다. 이 값이 0이면, 위상 함수는 $p(\theta) = \frac{1}{4\pi}$ 의 상수 함수가 되어, 등방성 산란을 하게 되고, 1에 가까울수록 전방 산란, -1에 가까울수록 후방 산란을 하게 된다. 위상 함수와 비대칭 파라미터를 이용하면, 포톤의 이동 방향 $\vec{w}_d = (\theta, \phi)$ 에 쉽게 물리적 성질을 반영할 수 있다(아래 식 참조). 새로운 방향을 결정하는 θ 와 ϕ 중, θ 가 구면좌표계의 z축과 관련이 있으므로 θ 를 중요 샘플링하고, ϕ 는 균일 샘플링 한다.

$$\cos\theta = \frac{1}{2g} \left[1 + g^2 - \left(\frac{1-g^2}{1-g+2g\xi} \right)^2 \right], \xi \in [0, 1]$$

2.2.2. 포톤의 이동 거리

포톤의 방향과 함께, 포톤의 이동 거리도 정해져야 하는데, 포톤이 다음 충돌 지점까지 이동하는 거리는 식 (4)에 의해 중요 샘플링 되어 구해진다. 여기서 흡광 계수(extinction coefficient, σ_t)는 밀도의 함수로써, [6]에서 사용된 바와 같이, 흡광 자름면(extinction cross-section,

1) 균일 샘플링 된 난수를 생성하기 위하여, 본 논문에서는 [17]에서 제시되었고, [24] 등에서 사용되어 효율성이 인정되고 있는 "Mersenne Twister"라는 난수 생성 알고리즘을 사용하였다.

C_{ext} 에 각 샘플링 지점에서 구한 밀도(ρ)를 곱한 값으로 정의되고 있다. 즉, $\sigma_t = C_{ext}\rho$ 이다. 밀도가 높으면 이동 거리가 짧아지고, 밀도가 낮으면 이동 거리가 길어지게 되어, 밀도가 높은 연기에서는 빛의 산란이 많이 일어나고, 밀도가 낮은 연기에서는 빛의 산란이 상대적으로 적게 일어나는 물리적인 의미가 반영되고 있다.

$$d = \frac{-\log \xi}{\sigma_t} = \frac{-\log \xi}{C_{ext}\rho}, \xi \in [0, 1]$$

2.2.3. 포톤의 생존

앞서 설명된 방법에 의해 포톤이 방향과 거리가 결정되어 새로운 위치로 이동하는 과정에서, 매번 포톤이 연기와 반응하여 포톤맵에 저장되고 다음 포톤의 위치를 결정하기 전에, 포톤이 계속 산란될 것인지, 아니면 연기에 흡수되어 사라질 것인지를 결정해야 한다. 빛이 날아오던 방향으로 계속해서 가지 못하고 산란될 확률을 의미하는 알베도(λ)를 이용하는데, [2]에서 Arvo 등에 의해 컴퓨터 그래픽스 분야에 소개된 러시안 룰렛(Russian roulette)이라는 확률적인 방법을 사용하여 포톤의 생존을 결정한다. 0과 1 사이의 균일 샘플링된 임의의 수를 발생시켜서, 미리 정해진 알베도보다 같거나 작으면 산란되어 계속 진행하는 것으로, 크면 흡수되어 사라지는 것으로 정한다.

$$\xi \in [0, 1] \rightarrow \begin{cases} \xi \leq \lambda, & \text{Photon is scattered.} \\ \xi > \lambda, & \text{Photon is absorbed.} \end{cases}$$

2.2.4. 변형된 볼륨 렌더링 방정식

포톤 매핑을 이용하면 볼륨 렌더링 방정식을 광선 진행법으로 구현한, 2.1.2절의 식에서 다중 산란을 표현하는 부분이 아래의 식의 두 번째 항과 같이 근사화 된다[13, 27].

$$L_n = \sum_i^N L_i(x, \vec{w}_i) p(x, \vec{w}_i, \vec{w}) \sigma_s(x) \Delta x + \left\{ \sum_{p=1}^n p(x, \vec{w}_p, \vec{w}) \frac{d\Phi_p(x, \vec{w}_p)}{4\pi r^3} \right\} \Delta x + e^{-\sigma_t(x)\Delta x} L_{n-1}(x + \Delta x \vec{w}, \vec{w})$$

포톤 매핑을 이용하여 다중 산란 효과까지 구현한 경우, 연기의 내부에서 반응하는 빛의 효과에 의해 명암이 살아나서, 단일 산란만을 사용하여 렌더링 한 미묘한 이미지와는 확연히 구분되는 결과를 얻을 수 있음을 알 수 있다.

3. 포톤 매핑 기반의 연기 파티클 렌더링 및 최적화 기법

3.1. 연기 파티클에 대한 렌더링 기법

3.1.1. SPH(Smoothed Particle Hydrodynamics) 보간법

파티클 렌더링을 위해서는 주어진 연기 파티클 데이

터로부터 임의의 샘플링 지점에서의 속성 데이터를 복원하는 것이 중요하다. 이를 위하여 [5, 21]에서 제시된 바와 같이 SPH(Smoothed Particle Hydrodynamics) 기법에서 사용하는 다음과 같은 보간 기법을 사용한다.

$$A_s(r) = \sum_i^N m_i \frac{A_i}{\rho_i} W(r - r_i, h)$$

여기서 r 은 (x, y, z) 로 표현되는 3차원 공간상의 한 점이고, m_i 는 i 번째 파티클의 질량, A_i 는 i 번째 파티클이 갖고 있는 속성 값, ρ_i 는 지점에서의 밀도이다. 한편, $W(r - r_i, h)$ 는 h 를 스무딩 반경(smoothing length)으로 갖는 커널 함수로써, $r - r_i$ 가 사용자가 설정하는 h 라는 반경 이내에 들어올 경우 적절한 가중치를 부여하기 위한 함수이다. 이 식에 의하면, 임의의 샘플링 지점 r 의 주변에 있는 N 개의 파티클로부터의 거리 $r - r_i$ 와 h 를 이용하여 가중 합을 구하여 샘플링 지점에서의 속성 값을 복원할 수 있다. 광선 진행법을 이용하여 연기 파티클을 렌더링 할 때, 빈번하게 필요하게 되는 임의의 지점에서의 밀도는 SPH의 복원식에서의 속성 데이터 A_i 부분에 밀도를 적용하여 다음과 같이 간단하게 얻을 수 있다.

$$A_s(r) = \sum_i^N m_i \frac{\rho_i}{\rho_i} W(r - r_i, h) = \sum_i^N m_i W(r - r_i, h)$$

3.1.2. 커널 함수와 스무딩 반경 설정

사용자가 설정한 커널 함수의 종류와 스무딩 반경에 따라서 특정 샘플링 지점에서의 속성값을 얻는데 사용되는 파티클들과 각 파티클에 적용되는 가중치가 다양해지므로 최종적인 렌더링 결과도 많은 영향을 받게 된다.

$$W_h(r) = \frac{15}{\pi(4h)^3} \begin{cases} (2 - \frac{r}{h})^3 & 0 \leq r \leq 2h \\ 0 & r > 2h \end{cases}$$

$$W_h(r) = \frac{315}{64\pi h^9} \begin{cases} (h^2 - r^2)^3 & 0 \leq r \leq 2 \\ \text{otherwise} & \end{cases}$$

위 두 식은 커널 함수의 예로, 전자는 [5]에서 제시하는 정규화 된 스플라인 가우시안(spline Gaussian) 함수이고, 후자는 [21]에서 제시하는 poly6라는 커널 함수이다. 샘플링 지점과의 거리(x 축)가 0에 가까울수록 큰 가중치(y 축)를 갖게 되고, 샘플링 지점과의 거리가 일정한 영역을 넘어가면 가중치가 0이 되는 모양을 하고 있다.

3.1.3. 포톤 매핑 기반의 파티클 렌더링

포톤 매핑 기반으로 파티클 데이터를 렌더링하는 방법으로 파티클 데이터를 볼륨 데이터로 변환한 다음에 기존의 볼륨 렌더링 방법을 적용하는 방법과 볼륨 데이터로 변환하는 중간 단계를 거치지 않고 파티클 데이터에 대해 직접 렌더링하는 방법을 우선적으로 고려해볼 수 있다.

[파티클을 볼륨화한 후, 볼륨 렌더링] 파티클 데이터

를 포함하는 바운딩 박스를 적절한 크기의 격자로 나누고, 각 격자점에서 주변의 파티클을 검색하여 밀도를 구하는 커널 함수 기반의 보간법을 이용하면 격자 기반의 밀도장을 만들 수 있다. 이렇게 복원된 3차원 격자 상의 밀도장에 대해 포톤맵을 구성한 뒤, 위에서 설명한 광선 진행법을 적용하며, 임의의 샘플링 지점에서 밀도가 필요한 경우는 주변의 8 개의 격자점에 정의된 밀도로부터 삼선형 보간법을 적용하여 구한다. 파티클 데이터를 3차원 격자 상에 정의된 볼륨 데이터로 바꾸어 렌더링 하는 과정은 그림 1과 같이 간략하게 나타낼 수 있다. 기존에 많이 사용되어온 볼륨 렌더링 과정을 알아갈 수 있다는 장점이 있으나, 파티클을 볼륨화해야 하므로 추가적으로 적지 않은 메모리를 필요로 하며, 볼륨화 과정에서 격자 크기를 적절하게 결정하지 못하면 파티클이 갖고 있는 연기의 상세함이 사라지거나, 혹은 메모리가 너무 낭비되는 단점이 있다.

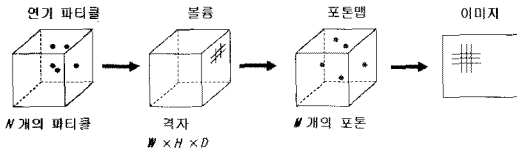


그림 1 파티클을 볼륨화한 후, 볼륨 렌더링 기법 적용.

[파티클 데이터의 직접 렌더링] 연기의 흐름을 이산적으로 나타내 주는 파티클의 집합으로부터 밀도장을 따로 만들지 않고, 파티클 집합에 대해 직접 포톤을 쬐서 포톤맵을 구성한다. 매 샘플링 지점에서 밀도가 필요한 경우, 커널 함수 기반의 보간법을 사용하여 밀도를 복원한다. 이렇게 구성된 포톤맵에 대해, 위에서 사용한 광선 진행법을 적용하는 방법이다. 광선을 추적하면서 발생하는 샘플링 지점에서의 밀도와 빛의 효과를 공간상에 분산되어 있는 파티클과 포톤을 통해서 직접 계산하여 이미지를 생성할 수 있다(그림 2). 3차원 격자 형태의 중간 단계를 거치지 않고도 연기의 모습이나 그에 반응하는 빛의 모습을 효과적으로 렌더링 할 수 있다.

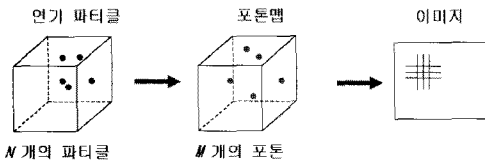


그림 2 파티클 데이터의 직접 렌더링. 대부분의 경우 포톤의 수(M)가 파티클 수(N) 보다 많다.

이 절에서 설명한 연기 파티클에 대한 두 가지 렌더링 방법은 직관적으로 쉽게 생각해볼 수 있는 것이지만, 렌더링 시간 측면에서나 메모리 사용 측면에서 개선될만한 여지를 갖고 있다. 3.2절에서 연기 파티클에 대해 좀더 효율적으로 렌더링 할 수 있는 방법을 제안한다.

3.2. 파티클맵: 파티클의 포톤화

3.2.1. 포톤맵 없는 포톤 매핑

광원으로부터 출발한 포톤이 산란되는 과정에서, 반응이 일어난 지점에서의 포톤을 따로 저장하지 않고 주변의 파티클에 포톤이 갖고 있는 빛의 강도와 포톤이 날아온 방향을 나누어 저장하면 파티클을 포톤화 하여 사용할 수 있다. 포톤을 뿌리는 과정을 마치고 광선 진행법을 적용하여 렌더링 할 때, 다중 산란의 광도는 포톤화 된 파티클 데이터로부터 계산한다(그림 3). 포톤맵을 구성하지 않는 대신 포톤의 정보를 파티클에 저장하므로, 이 포톤화 된 파티클 데이터를 기존의 파티클 데이터와 구분하기 위해, [1]에서 사용된 바와 같이 파티클맵(particle map)이라고 부르기로 한다.

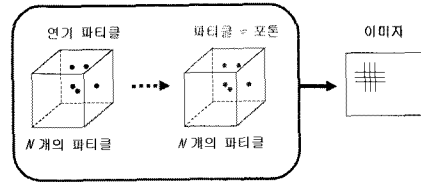


그림 3 파티클의 포톤화. 포톤맵 구성없이 광선 진행법 적용.

대부분의 경우에 포톤의 수가 파티클의 수보다 훨씬 많고, 렌더링 과정에서 일어나는 전체 샘플링 횟수가 포톤이 산란하면서 반응이 일어나는 횟수보다 크므로, 포톤맵을 검색하는데 드는 시간을 파티클맵을 검색하는 시간으로 대체함으로써 시간 비용을 줄일 수 있으며, 포톤맵을 따로 저장할 필요가 없으므로 메모리 사용에도 이점이 있다. 또한, 경우에 따라, 렌더링 과정에서 샘플링 지점에서의 밀도를 얻기 위한 파티클 검색 반경과 다중 산란의 광도를 얻기 위한 포톤 검색 반경이 같다면, [1]에서 주장한 바와 같이, 파티클에 대한 한번의 검색만으로도 밀도와 다중 산란의 광도를 구할 수 있으므로 효율성을 증대시킬 수 있다. 하지만, 두 반경이 항상 일치하는 것은 아니므로, 이것은 제한적인 상황에서만 얻을 수 있는 이점이라고 하겠다.

이 방법은 포톤의 정보를 주변의 파티클에 저장해둔다는 점에서 [1]에서 제시된, 알베도가 높은 경우의 렌더링 방식과 비슷하다. 하지만 서론에서 밝혔듯이 [1]의 방법은 시점에 독립적인 포톤맵의 장점을 살리지 못한, 시점 의존적인 파티클맵인 반면, 여기서 제시하는 방법은 시점 독립적인 파티클맵이라는 차이가 있다. 표 1은 한 번 구성된 파티클맵에 대해 카메라의 위치를 변경 시키면서 렌더링했을 때, 렌더링 시간을 비교한 표이다. 시점 의존적인 파티클맵이라면 카메라의 위치가 변경될 때마다 포톤을 추적하는 과정(photon tracing)이 추가되어 렌더링 시간이 늘어나겠지만, 표에서 볼 수 있듯이 카메라 1에서 구성된 파티클맵을 이용하여 렌더링을 하므로, 시점이 변해도(카메라 2, 3, 4) 포톤을 다시 뿌려줄 필요가 없다. 그림 4에 이에 대한 렌더링 결과가 나와 있다.

	카메라 1	카메라 2	카메라 3	카메라 4
포톤 추적 시간	289.12	-	-	-
광선 진행 시간	451.40	453.73	459.01	449.63
렌더링 시간 합	740.52	453.73	459.01	449.63

표 1 시점 독립적인 파티클맵 렌더링 시간 비교(단위:초).

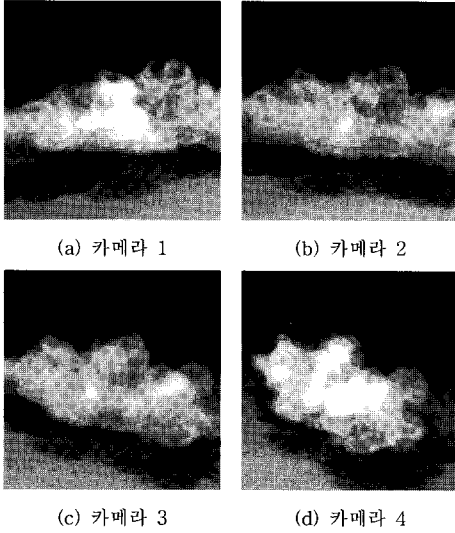


그림 4 시점 독립적인 파티클맵 렌더링.

그림 5는 포톤의 정보를 주변 파티클에 나눠주는 과정을 도식화한 것이다. 현재 위치에서 포톤맵에 포톤을 저장하지 않고, 주변의 일정 범위(h) 안에 들어오는 파티클을 검색해서 포톤의 정보를 더해지게 된다. 광원이 갖고 있는 포톤을 전부 방출한 뒤에는 파티클 사이를 돌아다니던 포톤들의 정보를 파티클이 갖게 되는데, 이를 통해 파티클맵이 구성된다. 이 때 반드시 고려되어야 하는 것은, 포톤의 정보를 주변 파티클에 '어떻게 더해줄 것인가'인데, 포톤의 지니고 있는 정보는 포톤의 위치와 빛의 강도, 그리고 포톤이 날아온 방향이 있다. 포톤의 위치는 주변 파티클을 검색할 때 중심점이 되므로, 따로 저장하지 않아도 반영이 되지만, 빛의 강도나 포톤이 날아온 방향은 주변 파티클에 나누어 주는 방법에 따라 포톤의 특성이나 형태가 반영되는 모습이 차이가 날 수 있으며, 렌더링 결과에 영향을 미칠 수 있다. 즉, 포톤을 버려도, 포톤의 정보를 잃어버리지 않고 파티클이 가지고 있도록 하는 것이 중요하며, 이 과정에서 포톤 정보를 분배하는 함수가 포톤의 특정 성질을 무시하거나 강조할 수 있는 또 다른 렌더링 조절 변수로 사용될 수 있다는 것 또한 중요한 점이다.

3.2.2. 포톤이 지니고 있는 빛의 강도

포톤이 지니고 있는 빛의 강도는, 포톤과 연기가 반

응한 지점에서 일정 거리 안에 존재하는 파티클을 검색해서, 작용 지점의 중심과 각 파티클 사이의 거리를 이용하여 거리에 따른 가중 함수를 적용하여 각 파티클에 더해준다. 우선, 임의의 포톤 p_i 하나에 대해, p_i 가 갖고 있는 빛의 강도를 Φ_{p_i} 라고 하면, p_i 위치를 중심으로 스미딩 반경 h 안에 들어오는 L_{p_i} 개의 파티클 중 임의의 파티클 t_j 가 갖게 되는 빛의 강도 Φ_{t_j} 는 다음과 같다.

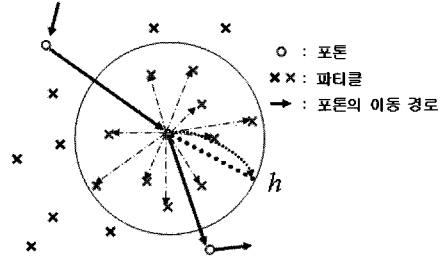


그림 5 포톤의 정보를 주변 파티클에 나눠주는 과정. 매 포톤이 이동할 때 마다 포톤맵을 구성하지 않고, 포톤이 지니고 있는 빛의 강도와 날아온 방향을 주변 반경 h 이내의 파티클을 검색해서 나누어 준다.

$$\Phi_{t_j} = \Phi_{p_i} + \frac{W(x_{p_i} - x_{t_j}, h)\Phi_{p_i}}{\sum_{k=1}^{L_{p_i}} W(x_{p_i} - x_{t_k}, h)}, t_j = \{t_k | k=1, 2, \dots, L_{p_i}\}$$

포톤 p_i 와 파티클 t_j 간의 거리 $x_{p_i} - x_{t_j}$ 에 따른 가중 함수 $W(x_{p_i} - x_{t_j}, h)$ 는 SPH 보간에 사용되었던 정규화된 스플라인 가우시안 함수(식 (9))를 사용하였다.

광원에서 포톤을 뿌리는 과정을 모두 마치게 되면, 포톤과 연기가 상호 작용한 N 개의 지점 p_i ($1 \leq i \leq N$)에 대해, 각 지점에서 L_{p_i} 개의 파티클을 찾았다고 했을 때, 파티클맵의 임의의 한 파티클 t 가 갖게 되는 빛의 강도 Φ_t 는 다음과 같이 된다.

$$\Phi_t = \sum_{i=1}^N \frac{W(x_{p_i} - x_t, h)\Phi_{p_i}}{\sum_{k=1}^{L_{p_i}} W(x_{p_i} - x_{t_k}, h)}$$

3.2.3. 포톤이 날아온 방향

포톤의 방향 정보 또한, 포톤과 연기가 상호 작용한 지점 p_i 에서 일정 거리 안에 존재하는 파티클을 검색하여 포톤이 날아온 방향 벡터 \vec{w}_{p_i} 를 각 파티클의 방향 벡터에 더해준다. 포톤을 뿌리는 과정이 끝나면, 임의의 파티클 t 는 다음과 같이 주변을 지나간 포톤들의 방향 정보를 모두 더한 \vec{w}_t 를 방향 벡터로 갖게 되며($\vec{w}_t = \sum_k \vec{w}_{p_k}$), 최

종적으로 다음과 같이 \vec{w}_i 를 정규화 하여 파티클 t 의 방향 벡터 \bar{w}_i 로 사용한다($\bar{w}_i = \frac{\vec{w}_i}{|\vec{w}_i|}$).

3.3. 광도맵을 통한 렌더링 속도 개선

렌더링 속도의 향상을 위한 방법으로써, 광선을 진행 시에 매 샘플링 지점에서 광도를 직접 계산하는 방법 대신, 3차원 격자를 통해 파티클 데이터가 있는 공간을 분할하고 각 격자점에서 다중 산란의 광도를 미리 계산한 후, 실제 샘플링 지점에서의 다중 산란의 광도는 주변 8개의 격자점을 찾아서, 각 격자점에 계산되어 있는 다중 산란의 광도를 삼선형 보간해서 얻는 방법을 제안한다(그림 6). 이 때 다중 산란의 광도를 구해서 저장해 놓는 3차원 격자들의 정보를 광도맵(irradiance map)이라고 부르기로 한다.

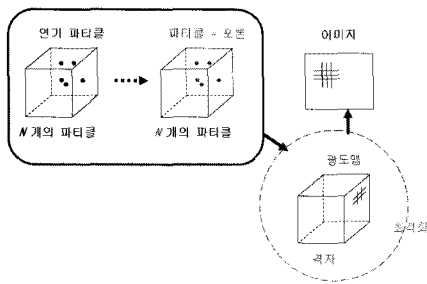


그림 6 광도맵을 통한 다중 산란 계산.

그림 7에는 하나의 격자 안에 발생한 샘플링 지점(붉은 사각형) 주변의 8개의 격자점에서 삼선형 보간 하는 과정이 도시화되어 있다. 다중 산란의 광도를 구하는 보통의 경우(광도맵을 사용하지 않았을 경우)라면, 모든 샘플링 지점에서 주변의 포톤을 검색해야 한다. 하지만 광도맵을 사용한다면, 임의의 격자 하나에 포함된 모든 샘플링에 대해 다중 산란의 광도를 구할 때, 주변 8개의 격자점에 미리 구해놓은 다중 산란의 광도로부터 보간해낼 수 있다.

대부분의 경우에 격자 하나를 통과하는 광선은 여러 개가 있으므로, 하나의 격자 안에서는 광선을 진행하기 위한 샘플링이 여러 번 발생하게 되며, 이미지의 품질을 높이기 위해서 픽셀 당 광선을 여러 번 샘플링 하는 슈퍼 샘플링을 사용하는 경우라면, 횡수는 더욱 늘어날 것이다. 하지만, 광도맵을 사용하게 되면, 하나의 격자에 대해서 8개의 꼭지점에서만 포톤을 검색하면 되므로, 각 샘플링 지점에서의 포톤 검색 시간을 현저히 줄일 수 있다. 또한 여러 개의 격자를 생각할 경우, 한 격자점에서 미리 계산된 다중 산란의 광도는 주변 8개의 격자에 대해서 재사용되기 때문에, 렌더링 속도를 개선할 수 있다.

광도맵을 만드는 경우에, 렌더링 장면의 상황에 따라 적용적으로 만들어서 효율성을 증대시킬 수도 있을 것이다. 우선, 사용되는 3차원 격자의 형태를 파티클이나 포톤

의 분포에 따라, 격자 간격이 균일한 격자를 사용하거나 격자 간격을 동적으로 정하는 적응적인 격자(adaptive grid)를 사용할 수 있다. 적응적 격자를 사용한 경우 광도의 분포에 따라 세밀하게 정보를 표현할 수 있으며, 격자가 낭비 되지 않는다는 장점이 있지만, 적응적 격자를 유지하기 위한 자료 구조의 복잡도가 증가함에 따라 샘플링 지점 주변 격자를 찾는 비용이 증가하는 단점이 있다. 또 다른 적응적 방법으로, 뷰잉 볼륨(viewing volume)과의 관계를 고려하는 것을 생각해볼 수 있다.

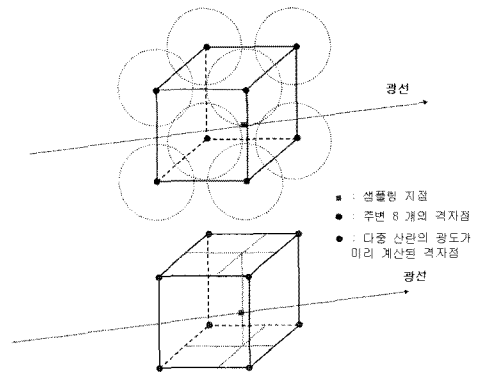


그림 7 삼선형 보간을 통한 다중 산란 광도의 보간. 임의의 샘플링 지점에서의 다중 산란 광도는 주변 8개의 격자점에 미리 계산된 다중 산란 광도로부터 삼선형 보간을 통해 계산한다.

임의의 한 지점에서의 광도를 비교적 오랜 시간을 들여서 계산했는데, 뷰잉 볼륨 밖에 존재는 점이어서 최종 이미지에 전혀 반영되지 않는다면, 계산 시간을 낭비한 것이 된다. 따라서 광도맵을 구성할 때 격자점이 뷰잉 볼륨 밖에 있다면, 광도를 계산 하지 않고 넘어가는 방법으로 최적화 할 수 있다.

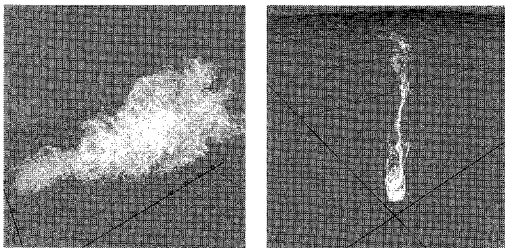
3.4. 3차원 kd-트리

파티클 및 포톤은 기본적으로 3차원 공간상에 정의된 점의 집합이므로, 임의의 샘플링 지점에서 파티클 및 포톤이 가지고 있는 특정 성질 값을 구하기 위해서 주변의 가까운 점들을 검색하는 일이 빈번하게 발생하게 된다. 따라서 파티클 기반의 데이터를 사용하고 포톤 매핑 기법을 적용한 본 논문에서와 같은 경우에, '가장 가까운 N 개의 점을 찾는 알고리즘이 얼마나 효율적인가'가 전체 렌더링의 효율성을 결정하는 중요한 부분을 차지한다고 할 수 있다. 여기서는 [3]에서 제시된 kd-트리를 사용하되, [1, 13]에서 사용한대로 균형 잡힌 3차원 kd-트리를 사용하였다. 3차원 kd-트리는 공간상의 점들을 이진 탐색 트리로 구성하여 공간을 나누는 자료 구조이다. 3차원 kd-트리를 구성하는 단계에서 주어진 장면에 대하여 파티클이나 포톤을 포함하는 전체 바운딩 박스를 설정하고, 이를 분할해 간다. 여러 가지 성질에 따라 분할 방식을 결정할 수도 있는데 kd-트리는 균일하게 분할해가는 것이 아니라, 물체가 분포된 형태에 따라 적용적으로 바운딩 박스를 나누어

간다. N 개의 점으로부터 균형 잡힌 kd-트리를 구성하는데 드는 시간 복잡도는 $O(\log N)$ 이며, 이 경우 N 개의 점으로부터 M 개의 점을 찾는 데 드는 시간 복잡도는 $O(M \log N)$ 이다. 3차원 kd-트리 구조를 사용함으로써, 렌더링 과정에서 밀도를 구하고, 포톤맵으로부터 포톤의 정보를 얻거나 포톤의 정보를 파티클에 나누어 주기 위한 검색을 효율적으로 할 수 있다.

4. 실험 결과

이 장에서는 다양한 시점, 광원, 데이터에 대해 본 논문에서 제시한 연기 파티클 렌더링 방법을 적용한 실험 결과에 대해 설명한다. 본 논문은 연기에 대한 렌더링이 주제이므로 실험에 사용한 연기 데이터의 모델링 정확도는 논의 대상에서 제외하며, 연기 모델링이 물리 기반으로 비교적 정확하게 이루어졌다는 가정 하에 데이터를 사용하였다. 실험에 사용한 연기 데이터는 유체의 일반적인 압축성 및 점성 유동을 기술하는 지배 방정식인 Navier-Stokes 방정식을 3차원 격자에서 풀어낸 속도장을 따라 밀도 파티클을 일정 시간 동안 흘려보낸 방식으로 시뮬레이션 한 데이터와, 여기에 SPH 모델에 의거하여 파티클의 상호 작용까지 고려한 연기 시뮬레이션 데이터, 두 가지가 있으며, 그림 8은 각 파티클 데이터를 시각화한 그림이다. 그림 9에는 SPH 모델에 의해 시뮬레이션된 파티클 데이터에 대한 렌더링 결과가 나와 있다. 이 장에서 설명되는 실험은 모두 Intel Xeon 3.0 GHz의 CPU와 2 Gbytes의 메모리를 장착한 컴퓨터에서 진행되었고, 렌더링된 이미지는 500×500 의 픽셀 해상도 가지며, 각 픽셀마다 광선 샘플링이 4번 이루어졌다.



(a) 속도장 만으로 만든 연기 데이터 (b) SPH 연기 파티클 데이터

그림 8 연기 파티클 데이터의 시각화.

4.1. 렌더링 방법에 따른 비교

이 절에서는 본 논문에서 소개한 연기 파티클 렌더링 방법에 따라 렌더링했을 때의 실험 결과를 보여준다. 파티클에 대해 포톤맵 구성 후 직접 렌더링하는 것과 포톤맵 없이 파티클맵으로 렌더링하는 것, 그리고 광도맵의 사용 여부에 따라 렌더링 방법을 네 가지 경우로 나눠서 비교를 해보았다. 실험 결과는 다음 두 가지 항목을 비교 대상으로 하여 살펴보았으며, 각 렌더링 방식에 따라 반영되는 요소는 약간씩 차이가 있다.

- 렌더링 시간 = 포톤 추적 시간 + 광도맵 구성 시간 + 광선 진행 시간

- 메모리 사용량 = 파티클맵 크기 + 포톤맵 크기 + 광도맵 크기

이 실험에서는 110,401개의 파티클에 대해 평균 640,000여 개의 포톤이 포톤맵에 저장되거나, 연기 파티클과 반응하여 파티클맵을 구성했으며, 다중 산란의 광도 계산을 위해 최대 500 개의 포톤을 수집했다.

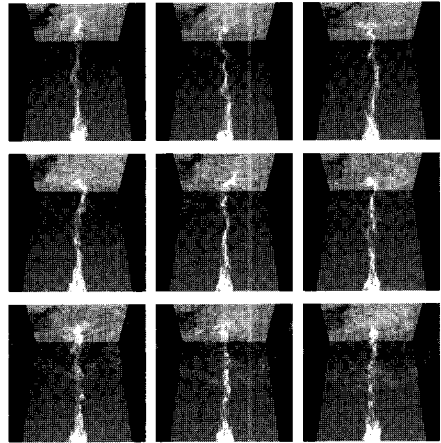


그림 9 SPH 연기 파티클 데이터 렌더링 결과.

광도맵 구성 시 격자의 크기를 단위격자($1 \times 1 \times 1$)의 1/4로 한 결과, 광도맵의 격자 해상도는 $260 \times 76 \times 104$ 가 되어, 격자점은 총 2,055,040개 이었다. 각 방법의 렌더링 시간과 메모리 사용량을 정리한 표가 표 2에 나와 있는데, 계산 시간 측면에서는 본 논문에서 제안한 파티클을 포톤화 하여 파티클맵을 사용하는 방식에, 추가적으로 광도맵을 사용하여 최적화하는 방법이 가장 효율적임을 알 수 있다. 메모리 사용에 심하게 제약받는 상황에서는, 렌더링은 파티클맵을 사용하는 방식을 쓰고, 광도맵을 통한 최적화는 사용하지 않는다면, 메모리 사용을 최소화할 수 있을 것이다. 이 경우, 렌더링 시간에서의 이점을 포기하는 대신, 포톤맵과 광도맵을 사용하지 않게 되어, 공간 제약을 최소화할 수 있으면서도 효율적인 렌더링이 가능하다. 렌더링 결과는 그림 10에 나와 있다.

4.2. 광도맵 사용 비교

이 절에서는 광도맵을 통한 최적화 시 성능을 비교해 본다. 표 3은 단위격자($1 \times 1 \times 1$)의 1/4 만큼의 크기를 갖는 격자를 파티클 데이터에 적용하여 구성한 광도맵을 사용했을 경우와 사용하지 않았을 경우, 렌더링에 걸리는 시간을 비교한 것이다. 여기서는 순수하게 광도맵의 성능을 비교하기 위해, 파티클맵을 구성하여 최적화 하는 방법을 쓰지 않고 파티클에 대해 포톤맵을 구성하여 직접 렌더링하는 방식을 사용했다. 광도맵을 사용했을 경우, 전처리 단계에서 광도맵을 구성하는 데에 비교적 많은 시간이 걸리

지만, 그 결과 광선을 진행하는 단계에서는 수행 시간이 현저히 줄고 있음을 알 수 있다. 전체적으로 봤을 때, 광도맵을 사용한 경우는 그렇지 않은 경우에 비해 대략 73%의 렌더링 시간 이득을 보이고 있다. 그림 11에는 표 3에서 비교한 네 개의 프레임에 대한 렌더링 결과를 비교하고 있는데, 광도맵을 사용한 경우와 그렇지 않은 경우가 거의 차이가 없는 것을 볼 수 있다. 광도맵을 사용할 때, 파티클 데이터의 특성과 격자 크기를 어떻게 결정하느냐에 따라 나타나는 결과가 달라지겠지만, 이 결과를 통하여 일반적인 경우 매우 효율적으로 사용할 수 있는 방법이라는 것을 알 수 있다.

렌더링 방식		직접 렌더링		파티클맵		
광도맵 사용 여부		미사용	사용	미사용	사용	
시간 (초)	전처리	포톤 추적	38.0	36.9	234.8	234.7
		광도맵	-	778.7	-	229.4
		전처리 합	38.0	815.6	234.8	464.1
	광선 진행 시간	5,512.4	686.6	4,456.7	659.5	
	렌더링 시간 합	5,550.4	1,502.2	4,691.5	1,123.6	
메모리 (KB)	포톤맵	41,526.4	41,775.4	-	-	
	파티클맵	364.8	364.8	729.7	729.7	
	광도맵	-	49,321.0	-	49,321.0	
	필요 메모리 합	41,891.2	91,461.2	729.7	50,050.7	

표 2 렌더링 방법에 따른 비교. 직접 렌더링은 파티클 데이터에 포톤맵을 구성하여 렌더링하는 것이고, 파티클맵 방법은 포톤맵 대신 파티클맵을 구성하여 렌더링하는 것이다.

4.3. 파티클 및 포톤 수에 따른 비교

이 절에서는 연기를 표현하는 파티클 수와 파티클에 뿌려지는 포톤 수에 따라 렌더링이 어떻게 달라지는지 비교한다. 파티클의 수가 너무 적으면 연기의 세밀함을 표현하기가 어려워질 뿐만 아니라, 연기의 밀도 같은 물리적 성질을 복원해내기 위한 샘플 수가 적어지게 되어, 사실적인 연기 렌더링이 어려워진다. 반면에, 파티클의 수가 너무 많으면, 파티클을 검색하는 데 드는 비용이 증가하게 되어, 불필요하게 샘플링 되었거나 노이즈로 섞여 들어온 파티클 때문에 계산 시간이 낭비된다.

파티클에 뿌려지는 포톤 수 역시 렌더링 결과에 영향을 미치는데, 포톤이 지나는 빛의 강도는 광원의 강도를 포톤의 개수로 나눠준 값이므로, 포톤 수가 너무 적으면

각각의 포톤이 지나는 빛의 강도가 커지게 되어, 연기 파티클을 포톤화 할 때, 빛이 연기에 끌고루 퍼지지 못하고 연기의 일부분에만 편중되는 현상을 초래할 것이다. 반대로 포톤 수가 너무 많으면, 각각의 포톤이 지나는 빛의 강도가 너무 약해져서 다중 산란의 효과를 충분히 얻을 수 없을 것이다.

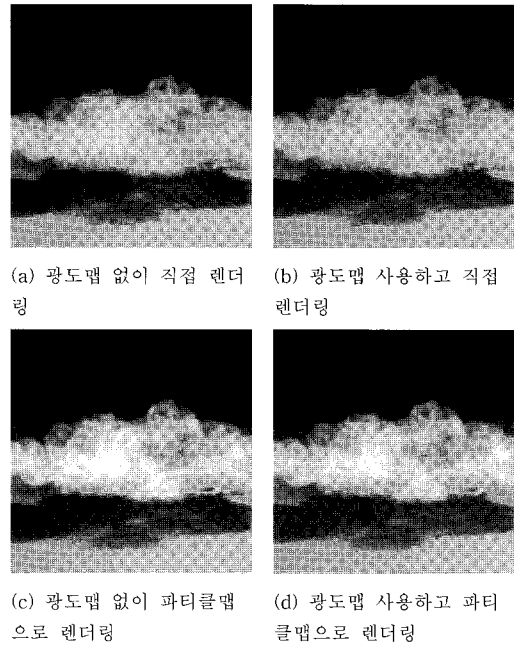
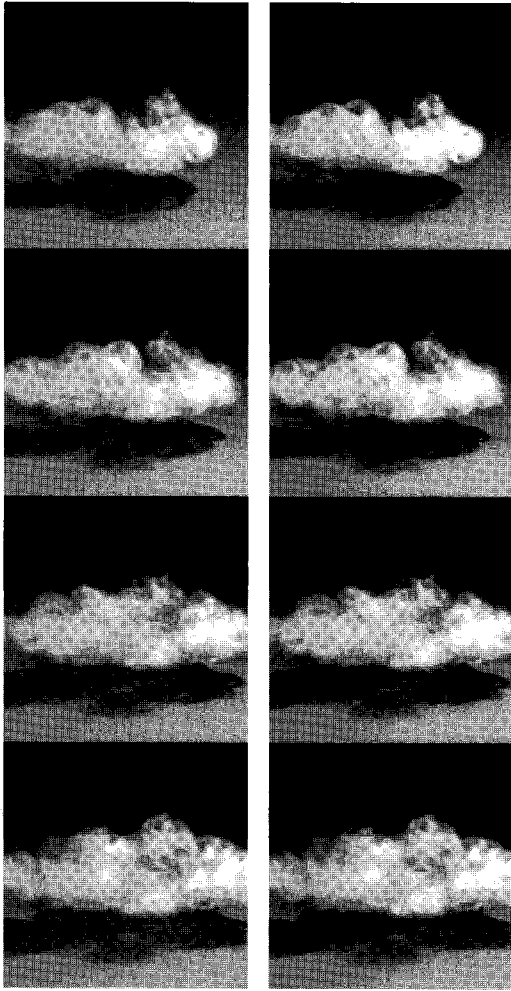


그림 10 연기 파티클 렌더링 결과 비교.

여기서는 시간 당 흘러가는 파티클의 수를 시뮬레이션 단계에서 임의로 조절해서, 일정 시간이 흘러 똑같은 프레임에 도달했을 때, 파티클 데이터에 존재하는 파티클 수를 다르게 하여 세 종류의 데이터를 만들었다. 또한 각각의 파티클 데이터에 대해 광원에서 방출되는 포톤의 개수가 많고 적음에 따른 두 가지 경우와 다중 산란 효과를 고려하지 않은 경우로 나누어서, 총 9 가지의 경우에 대한 렌더링 결과를 비교한다. 각각의 경우에 대해 렌더링의 세 가지 단계, 포톤 추적, 광도맵 구성, 광선 진행에 소요되는 시간을 비교한 결과가 표 4에 나타나있다. 포톤 추적 시간과 광도맵 구성 시간은 파티클 수나 포톤 수에 의존적이지 않지만, 광선 진행 시간은 포톤 수에 독립적임을 알 수 있으며, 이것은 본 논문에서 제시하는 광도맵을 사용하는 렌더링 방법의 장점이라 하겠다. 그림 12는 이에 대한 렌더링 결과인데, 파티클 수가 적은 첫 번째 열의 이미지의 경우, 스모딩 반경 안에 들어오는 파티클 수가 적기 때문에 연기의 밀도가 작아져서 사실성이 떨어지고 있다. 포톤 수가 적은 두 번째 행의 이미지의 경우는 각 포톤이 지니고 있는 빛의 강도가 커진데다가 광원에서 샘플링 되는 방향이 적을 수밖에 없으므로 연기의 일부분에 빛이 하얗게 물리는 현상이 생기고 있다.



(a) 광도맵 미사용

(b) 광도맵 사용

그림 11 광도맵 사용 여부에 따른 렌더링 결과 비교. 프레임 번호는 위에서부터 아래로 200, 225, 250, 275이다.

	프레임	200	225	250	275	
	파티클 수	80,401	90,401	100,401	110,401	
	평균 포톤 수	392,403	481,333	550,099	650,125	
광도맵 미사용	전처리	포톤맵 구성	24.1	27.1	31.9	37.2
	광선 진행 시간	3,543.3	4,244.2	4,949.4	5,570.6	
	렌더링 시간 합	3,567.4	4,271.3	4,981.3	5,607.8	
광도맵 사용	전처리	포톤맵 구성	23.6	27.8	31.6	38.8
		광도맵 구성	507.5	601.0	689.7	773.1
		전처리 시간 합	531.1	628.8	721.3	811.9
	광선 진행 시간	470.3	542.5	628.1	690.5	
	렌더링 시간 합	1,001.4	1,171.3	1,349.4	1,502.4	

표 3 광도맵 사용 여부에 따른 렌더링 시간 비교 (단위:초).

파티클 수	포톤 수	포톤 추적 시간	광도맵 구성 시간	광선 진행 시간	렌더링 시간 합
27,601	0	-	-	443.6	443.6
	443	0.1	99.6	447.2	546.9
	465,883	80.5	99.5	443.1	623.1
110,401	0	-	-	883.7	883.7
	486	0.2	228.5	888.9	1,117.6
	518,735	186.6	227.2	884.5	1,298.3
414,001	0	-	-	2,026.4	2,026.4
	607	0.3	345.7	2,023.7	2,369.7
	545,887	261.6	342.7	2,020.1	2,624.4

표 4 파티클 수 및 포톤 수에 따른 렌더링 시간 비교 (단위:초).

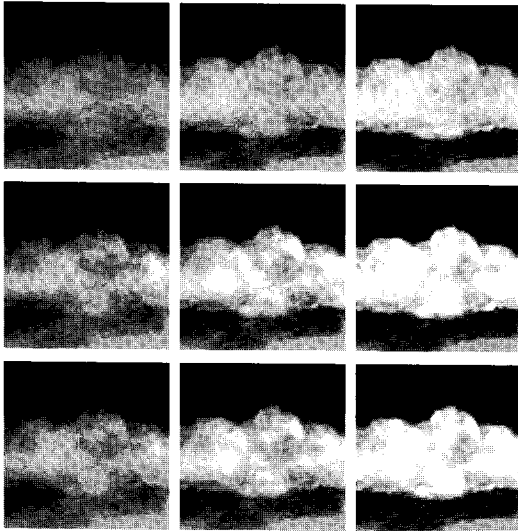


그림 12 파티클 및 포톤 수에 따른 렌더링 결과. 첫 번째 열에서 오른쪽으로 갈수록 파티클 수가 증가하고, 첫 번째 행에서 아래로 갈수록 포톤 수가 증가한다.

5. 결론 및 향후 연구

본 논문에서는 컴퓨터 그래픽스 분야에서 활발히 연구가 진행되고 있는 물리 기반 유체 시뮬레이션 기법 중, 파티클 기반으로 모델링 된 연기 데이터에 대해, 전역 조명이 지원되는 볼륨 포톤 매핑 기법을 적용하여, 이미지의 사실성을 살리면서도 렌더링의 효율성을 높일 수 있는 방법에 대해 살펴보았다. 특히, 시점에 독립적인 포톤맵 고유의 성질을 잃지 않으면서도 포톤맵을 파티클 데이터에 융화시키는 방법을 제시하여, 연기 파티클을 효과적으로 렌더링 할 수 있었다. 실험 결과에서 파티클 수에 따라 결과가 다르게 나오는 것을 살펴보았는데, 렌더링에 필요한 적절한 파티클 수를 결정할 수 있는 기준, 혹은 파티클 데이터의 노이즈를 판별할 수 있는 기준을 정할 수 있다면, 렌더링 시에 입력으로 들어온 파티클 데이터에 대해 동적으로 파티클을 추가하거나 제거하여, 렌더링의 사실성과 효율성을 높일 수 있을 것이다.

감사의 글

이 논문은 2008년도 정부(교육과학기술부)의 재원으로 한국과학재단의 지원을 받아 수행된 연구임 (No. R01-2007-000-21057-0(2008)).

참고 문헌

[1] N. Adabala and S. Manohar. Modeling and Rendering of Gaseous Phenomena Using Particle Maps. *The Journal of Visualization and Computer Animation*,

Vol. 11, No. 5, pp. 279-293, 2000.

[2] J. Arvo and D.B Kirk. Particle Transport and Image Synthesis. *ACM SIGGRAPH Computer Graphics*, Vol. 24, No. 4, pp. 63-66, 1990.

[3] J.L. Bentley. Multidimensional Binary Search Trees Used for Associative Searching. *Communications of the ACM*, Vol. 18, No. 9, pp. 509-517, 1975.

[4] J.F. Blinn. Light Reflection Functions for Simulation of Clouds and Dusty Surfaces. *ACM SIGGRAPH Computer Graphics*, Vol. 16, No. 3, pp. 21-29, 1982.

[5] M. Desbrun and M.P. Cani. Smoothed Particle: A New Paradigm for Animating Highly Deformable Bodies. In *Proceedings of the Eurographics Workshop on Computer Animation and Simulation*, pp. 61-76, 1996.

[6] R. Fedkiw, J. Stam, and H.W. Jensen. Visual Simulation of Smoke. In *Proceedings of ACM SIGGRAPH 2001*, pp. 15-22, 2001.

[7] B. Feldman, J. O'brian, and O. Arikan. Animation Suspended Particle Explosions. *ACM Transactions on Graphics(ACM SIGGRAPH 2003)*, Vol. 22, No. 3, pp. 708-715, 2003.

[8] F N. Foster and D. Metaxas. Modeling the Motion of a Hot, Turbulent Gas. In *Proceedings of ACM SIGGRAPH 1997*, pp. 181-188, 1997.

[9] M.N. Gamito, P.F. Lopes, and M.R. Gomes. Two-dimensional Simulation of Gaseous Phenomena Using Vortex Particles. In *Proceedings of the Eurographics Workshop on Computer Animation and Simulation*, pp. 2-15, 1995.

[10] R.A. Gingold and J.J. Monaghan. Smoothed Particle Hydrodynamics: Theory and Application to Non-spherical Stars. *Monthly Notices of the Royal Astronomical Society*, Vol. 181, pp. 375-398, 1977.

[11] H.W. Jensen. Global Illumination Using Photon Maps. In *Proceedings of the Eurographics Workshop on Rendering Techniques*, pp. 21-30, 1996.

[12] H.W. Jensen. *Realistic Image Synthesis Using Photon Mapping*. AK Peters, 2001.

[13] H.W. Jensen and P.H. Christensen. Efficient Simulation of Light Transport in Scenes with Participating Media Using Photon Maps. In *Proceedings of ACM SIGGRAPH 1998*, pp. 311-320, 1998.

[14] J.T. Kajiya and B.P. von Herzen. Ray Tracing Volume Densities. *ACM SIGGRAPH Computer Graphics*, Vol. 18, No. 3, pp. 165-174, 1984.

[15] R.V. Klassen. Modeling the Effect of the Atmosphere on Light. *ACM Transactions on Graphics*, Vol. 6, No. 3, pp. 215-237, 1987.

[16] L.B. Lucy. A Numerical Approach to the Testing of the Fission Hypothesis. *The Astronomical Journal*, Vol. 82, No. 12, pp. 1013-1024, 1977.

[17] M. Matsumoto and T. Nishimura. Mersenne Twister: A 623-Dimensionally Equidistributed Uniform Pseudorandom Number Generator. *ACM Transactions on Modeling and Computer Simulation*, Vol. 8, No. 1, pp. 3-30, 1998.

[18] N.L. Max. Light Diffusion through Clouds and Haze. *Computer Vision, Graphics, and Image Processing*,

Vol. 33, No. 3, pp. 280-292, 1986.

- [19] N.L. Max. Optical Models for Direct Volume Rendering. *IEEE Transactions on Visualization and Computer Graphics*, Vol. 1, No. 2, pp. 99-108, 1995.
- [20] J.J. Monaghan. Smoothed Particle Hydrodynamics. *Annual Review of Astronomy and Astrophysics*, Vol. 30, No. 2, pp. 543-574, 1992.
- [21] M. Müller, D. Charypar, and M. Gross. Particle-based Fluid Simulation for Interactive Applications. In *Proceedings of ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pp. 154-159, 2003.
- [22] S. Park, and M. Kim. Vortex Fluid for Gaseous Phenomena. In *Proceedings of ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pp. 261-270, 2005.
- [23] S. Peremože, T. Tasdizen, J. Bigler, A. Lefohn, and R.T. Whitaker. Particle-based Simulation of Fluids. In *Computer Graphics Forum 2003(Proceedings of EUROGRAPHICS)*, Vol. 22, No. 3, pp. 401-410, 2003.
- [24] M. Pharr, and G. Humphreys. *Physically Based Rendering*. Morgan Kaufmann, 2004.
- [25] W.T. Reeves. Particle Systems: A Technique for Modeling a Class of Fuzzy Objects. *ACM Transactions on Graphics*, Vol. 2, No. 2, pp. 91-108, 1983.
- [26] A. Selle, N. Rasmussen, and R. Fedkiw. A Vortex Particle Method for Smoke, Water and Explosions. *ACM Transactions on Graphics(ACM SIGGRAPH 2005)*, Vol. 24, No. 3, pp. 910-914, 2005.
- [27] R. Siegel, J.R. Howell. *Thermal Radiation Heat Transfer, 3rd Edition*. Hemisphere Publishing Corporation, 1992.
- [28] K. Sims. Particle animation and rendering using data parallel computation. *ACM SIGGRAPH Computer Graphics*, Vol. 24, No. 4, pp. 405-413, 1990.
- [29] J. Stam. Stable Fluids. In *Proceedings of ACM SIGGRAPH 1999*, pp. 121-128, 1999.
- [30] J. Stam and E. Fiume. Depicting fire and other gaseous phenomena using diffusion processes. In *Proceedings of ACM SIGGRAPH 1995*, pp. 129-136, 1995.