

개선된 공간 도약법을 이용한 고속 가상 내시경 기법 (High-Speed Virtual Endoscopy using Improved Space-Leaping)

신 병 석[†] 김 혁^{**}

(Byeong-Seok Shin) (JinGe)

요약 가상 내시경을 구현하기 위해서는 세밀하고 정확한 원근 투영 영상의 고속생성과 충돌검출이 필수적이다. 본 논문에서는 고속 원근 볼륨 렌더링 기법을 이용하여 정확하면서도 빠른 렌더링이 가능한 가상 내시경 기술을 제안한다. 이 방법은 기본적으로 화질이 우수한 광선 투영법을 기반으로 하며 거리 정보(distance information)를 이용하여 투명한 공간을 비약하도록 함으로써 렌더링 속도를 향상시키도록 하였다. 렌더링 시에는 광선의 진행 방향에 따라 샘플 간격을 가변적으로 조절하는 방법을 이용하여 화질의 손실을 최소화하면서 처리시간을 향상시킨다. 또한 전처리 단계에서 생성된 거리 정보를 이용하여 충돌 검출을 간단히 할 수 있는 방법을 제안한다.

키워드 : 공간도약법, 가상내시경, 볼륨렌더링

Abstract In order to implement virtual endoscopy, high-speed rendering algorithm that generates accurate perspective projection images and efficient collision detection method are essential. In this paper we propose an efficient virtual endoscopy system based on volume rendering technique. It is possible to skip over empty (transparent) space using the distance value produced in preprocessing time, and it does not deteriorate image quality since it is an extension of ray-casting. It also accelerates rendering speed with minimal loss of image quality by adjusting sampling interval along a ray according to direction of the ray. Using the distance information, we can simplify the collision detection of volumetric objects.

Key words : space-leaping, virtual endoscopy, volume rendering

1. 개요

광학 내시경(optical endoscopy)은 말단에 초소형 카메라가 장착된 가는 관을 환자의 체내에 삽입하여 식도, 위장, 기관지, 대장등 인체 내부의 병변을 진단하는 의료 장비를 가리킨다. 내시경은 침습적 수술법(invasive surgery)에 비해 환자의 고통이 적고 X선 영상보다 정확한 진단을 할 수 있지만 내시경을 체내에 삽입하는 것만으로도 환자에게 상당한 불편이 따르며, 보통 한번의

진단을 위해 상당히 긴 시간이 소요되는 문제가 있다.

광학 내시경의 문제점을 해결하기 위해 고안된 방법 중에 하나가 가상 내시경(virtual endoscopy)이다. 가상 내시경은 CT, MRI와 같은 전산화 단층 촬영장치로 인체 내부에 대한 연속된 단면 영상을 얻어낸 후 인체의 3차원적 구조를 재구성함으로써 내시경 카메라로 보는 것과 같은 가상의 3차원 영상을 만들어내는 기법을 말한다. 가상 내시경을 이용하면 환자의 고통이 전혀 없으며 인체의 어느 부위든 자유자재로 이동이 가능하다. 또한 수술기구와 장기 내벽이 충돌하더라도 실제 인체를 대상으로 한 것이 아니므로 외상이나 감염우려가 없다.

가상 내시경은 세밀하고 정확한 원근 투영 영상을 실시간으로 생성할 수 있어야 하며 가상의 내시경 장치와 인체 내벽간의 충돌여부를 신속하게 검사할 수 있어야 한다. 본 논문에서는 고속 원근 볼륨 렌더링 기법을 이용

· 이 논문은 2000학년도 인하대학교의 지원에 의하여 연구되었음(INHA-21086).

† 정 회 원 : 인하대학교 컴퓨터공학부 교수
bsshin@inha.ac.kr

** 비 회 원 : 조지 워싱턴대학교 대학원 전산과학과
jing@seas.gwu.edu

논문접수 : 2001년 8월 10일
심사완료 : 2002년 7월 15일

하여 정확하면서도 빠른 속도를 낼 수 있는 가상 내시경 기술을 제안한다. 기본적으로 화질이 우수한 광선 투영법을 기반으로 하며 거리 정보(distance information)를 이용하여 투명한 공간을 비약하도록 함으로써 렌더링 속도를 향상시키도록 하였다. 렌더링 시에는 광선의 진행 방향에 따라 샘플 간격을 가변적으로 조절하는 방법을 이용하여 화질의 손실을 최소화하면서 처리시간을 향상시키도록 한다. 또한 전처리 단계에서 생성된 거리 정보를 이용하여 충돌여부를 간단하게 검사할 수 있다.

논문의 구성은 다음과 같다. 2장에서는 가상 내시경을 구현하기 위한 기존 연구들을 소개한다. 3장에서는 거리 변환에 의한 거리 맵 구성 방법을 살펴보고, 광선 방향에 따른 적용형 샘플링 방법을 설명한다. 4장에서는 렌더링 속도 향상을 위해 사용된 몇 가지 방법들을 소개하고 5장에서는 거리 맵을 이용한 실시간 충돌 검출 방법을 설명한다. 6장에서는 실험을 통해 본 연구에서 제안하는 방법의 처리속도와 화질을 살펴보고 마지막으로 결론을 맺는다.

2. 기존연구

볼륨 데이터로부터 3차원 영상을 만들어내는 방법은 크게 표면 렌더링(surface rendering)과 직접 볼륨 렌더링(direct volume rendering)으로 나뉜다. 표면 렌더링은 볼륨 데이터로부터 물체의 표면 정보를 표시하는 다각형 모델(polygon model)을 재구성한 후 기존 렌더링 알고리즘들을 활용하여 3차원 영상을 만들어낸다.[1] 표면 렌더링은 속도가 빠르지만 근육, 지방층 등 병변의 발생가능성이 높은 연조직을 가시화 하는데는 적절하지 못하다. 직접 볼륨 렌더링에서는 볼륨 데이터의 각 복셀(voxel)들이 가지는 스칼라 값(scalar value)과 그 값들의 공간적 변화량(gradient)을 기반으로 표면 정보를 추정하여 영상을 만들어 낸다.[2] 직접 볼륨 렌더링은 표면 렌더링에 비해 화질이 좋고 반투명한 인체 해부 구조물을 표현하는데 유용하지만 처리 시간이 오래 걸리는 단점이 있다.[3] 가상 내시경 기법에 대한 과거의 연구는 주로 표면 렌더링 기술에 집중하였다. 그러나 기관지의 말단이나 굴곡이 심한 대장의 내부 구조 등을 표현하는데는 한계가 있고, 정확한 영상을 기반으로 병변을 진단해야 하는 의학적 응용에는 부족한 점이 많다. 최근에는 기존의 직접 볼륨 렌더링 방법을 개선하여 가상 내시경 영상 생성에 적용하는데 관심이 모아지고 있다.

가상 내시경은 실시간에 가까운 고속 원근 투영 영상의 생성이 핵심이므로 여러 가지 가속화 방법들이 이용된다. 템플릿 기반 렌더링[4]은 광선 투영법에서 특정한 조건을 만족하는 경우 모든 광선들의 진행 궤적이 동일

하다는 공간일관성(spatial coherency)을 이용하여 한 광선의 궤적을 미리 계산한 후에 나머지 광선들의 궤적은 단순한 크기변환 연산을 이용하여 계산하도록 하는 것이다. 그러나 이 방법은 공간 일관성이 있는 평행 투영을 위해 고안된 방법으로 원근 투영을 해야하는 가상 내시경에는 적용하기 어렵다.

쉬어-와프 렌더링(shear-warp rendering)은 시각 변환을 단순화 하고 투명한 공간을 비약하도록 함으로써 렌더링 속도를 빠르게 한다.[5] 그러나 재 샘플링(resampling)을 위한 컨벌루션 커널(convolution kernel)로 쌍선형보간(bilinear interpolation)법을 이용하기 때문에 3쌍선형보간(trilinear interpolation)을 이용하는 광선 투영법에 비해 화질이 떨어지는 문제가 있다. 또한 원근 투영의 경우는 평행 투영시와 비교하여 가속효과가 떨어진다. 최근에는 VolumePro와 같이 쉬어-와프 렌더링을 실시간에 수행하는 하드웨어가 소개되었다.[6] 그러나 이것은 원근 투영법을 지원하지 않으므로 가상 내시경에 활용하기 어렵다. 데이터를 여러개의 슬랩(slap)으로 나누고 평행 투영법으로 원근 투영을 근사하는 방법도 고안되었으나 상당한 오차가 발생하는 문제가 있다.[7]

특수하게 고안된 병렬 처리 컴퓨터나 3차원 텍스처 매핑(3D texture mapping) 하드웨어를 이용하여 볼륨 데이터를 병렬처리하는 연구는 과거부터 계속되어왔다.[8][9][10] 이들은 화면 기반 분할, 데이터 기반 분할에 의해 입력된 볼륨 데이터를 여러개로 나누고 각각의 처리단위 (processing unit)들이 볼륨 데이터의 일부를 처리한 후 집약하도록 한다. 그러나 병렬 처리가 가능한 다중 프로세서 시스템은 비용이 많이 들어 일반적으로 사용하기 어려운 문제가 있다.

3. 거리 정보를 이용한 공간 도약법

본 논문에서 제안하는 방법은 광선 투영법의 확장인 공간 도약법(space leaping)[11][12]을 기반으로 한다. 기존의 광선 투영법은 광선이 볼륨 데이터에 진입하는 위치부터 샘플링을 시작한다. 그러나 투명한 샘플(transparent sample)의 값은 픽셀 색상을 결정하는데 기여하지 않으므로 이들을 식별하여 처리하지 않으면 계산 시간을 크게 단축할 수 있다. 많은 응용에서 실제 볼륨 데이터의 경계로부터 물체의 경계까지의 부분은 투명한 빈 공간이고, 가상 내시경의 경우에도 카메라가 볼륨 데이터의 안쪽에 위치하지만 인체 내부의 빈공간을 따라 이동하기 때문에 카메라로부터 물체 경계까지의 부분은 역시 샘플링할 필요가 없다. 그러나 기존 광선 투영법에서는 실제 샘플링을 해보기 전에는 물체 표

면에 도착했는지의 여부를 알 수 없기 때문에 불필요한 샘플링을 피할 수 없다. 공간 도약법에서는 전처리 과정에서 각각의 복셀들과 가시화하려는 경계 복셀(boundary voxel)까지의 최단거리를 계산하여 거리 맵(distance map)이라는 공간 자료구조에 저장하고 실제 렌더링 중에는 이 정보를 이용하여 투명한 복셀들을 비약하도록 함으로써 렌더링 속도를 향상시킨다.

공간 도약법은 기본적으로 광선 투영법을 이용하기 때문에 원근 투영을 쉽게 구현할 수 있고, 재샘플링 컨벌루션에 3쌍선형 보간 연산자를 사용하므로 쉬어-웁 렌더링에 비해 화질이 좋다. 또한 전처리 과정에서 생성한 거리 정보는 가상 내시경 처리에 기본이 되는 충돌 검출(collision detection)과 자동 경로 생성(automatic path generation)에 활용할 수 있어 다른 방법들에 비해 효율적이다.

3.1 거리 변환과 거리 맵

거리 맵(distance map)은 볼륨 데이터와 동일한 해상도를 가지는 3차원 공간 자료구조로서 각 점에는 그 점과 대응한 복셀에서 가장 가까운 물체 경계 복셀(object boundary voxel)까지의 거리 값이 저장된다.

거리 맵을 생성하기 위해서는 먼저 기준이 되는 경계를 정의해야 한다. 볼륨 데이터에서는 기하학적 물체에 서처럼 경계가 명확하지 않으므로 구분(classification) 과정을 통해 간접적으로 경계를 정의한다. 식(1)에서와 같이 복셀의 밀도값이나 밀도값의 경사도와 같은 구분 인자(classification parameter)값에 따라 각 복셀의 불투명도(opacity)를 부여하는데 이것을 불투명도 전사함수(opacity transfer function)라 한다. 경계로 간주되는 복셀들의 불투명도를 높게 하고 그외의 복셀들에는 불투명도를 낮게 지정함으로써 경계를 정의할 수 있다.

$$a(x, y, z) = \mathcal{O}(d(x, y, z), \nabla d(x, y, z)) \quad (1)$$

여기서 $d(x, y, z)$ 는 밀도값, $\nabla d(x, y, z)$ 는 경사도

불투명도가 일정한 임계치 λ 를 초과하는 경우를 물체의 경계로 정하면서 임의의 복셀 $v(x, y, z)$ 을 다음과 같이 빈 복셀(empty voxel)과 경계 복셀로 분류할 수 있다.

$$\begin{aligned} \text{경계 복셀} &: \mathcal{O}(x, y, z) \geq \lambda \\ \text{빈 복셀} &: \mathcal{O}(x, y, z) < \lambda \end{aligned} \quad (2)$$

일단 경계조건이 정의되면 볼륨 $V(x, y, z)$ 의 거리 맵 $D(x, y, z)$ 는 다음과 같이 정의할 수 있다.

$$D(x, y, z) = \begin{cases} 0 & v(x, y, z) \text{가 경계 복셀인 경우} \\ \min(\bigcup_{i,j,k=0}^{L,M,N} \Psi(v(x, y, z), v(i, j, k))) & v(x, y, z) \text{가 빈 복셀인 경우} \end{cases} \quad (3)$$

여기서 $\Psi(v_1, v_2)$ 는 두 복셀간의 거리계산 함수 $v(i, j, k)$ 는 식별된 경계 복셀

여기서 함수 $\Psi(v_1, v_2)$ 는 두 복셀간의 거리를 계산하는 함수이다. 이상적인 거리 계산 함수는 유클리드 거리(Euclidean distance)를 계산하는 것이지만 이것을 계산하려면 복잡한 실수 연산을 필요로 하기 때문에 처리시간이 길어진다. 볼륨 데이터의 복셀 수가 상당히 많기 때문에 실제 응용에서는 유클리드 거리의 근사값인 도시구획 거리(city block distance)나 체스판 거리(chess-board distance)가 많이 사용된다. [13] 이러한 함수들은 단순한 정수 연산과 비교 연산만으로 이루어지기 때문에 계산시간이 상대적으로 적게 소요된다.

$$\begin{aligned} \text{유클리드거리: } \Psi(v(x_1, y_1, z_1), v(x_2, y_2, z_2)) &= \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2} \\ \text{도시구획거리: } \Psi(v(x_1, y_1, z_1), v(x_2, y_2, z_2)) &= |x_2 - x_1| + |y_2 - y_1| + |z_2 - z_1| \\ \text{체스판거리: } \Psi(v(x_1, y_1, z_1), v(x_2, y_2, z_2)) &= \max(|x_2 - x_1|, |y_2 - y_1|, |z_2 - z_1|) \end{aligned} \quad (4)$$

본 연구에서는 체스판 거리를 사용한다. 공간 도약의 목표는 임의의 광선 진행 방향으로 저장된 거리값 만큼 비약하는 것이다. 비약을 할 때 다음번 샘플링 위치는 빈 공간이거나 경계 복셀이어야 하며 어떤 경우에도 경계 복셀을 지나쳐서 물체의 내부 복셀 (inside voxel)로 비약해서는 안된다. 따라서 (x, y, z) 가 정수 좌표를 가질 때 광선의 진행방향과 무관하게 유클리드 거리보다 작은 값이 나오는 체스판 거리가 적당하다. 그림 1은 체스판을 이용하여 입력된 볼륨 데이터를 거리 변환한 결과를 2차원으로 도시한 예를 보여준다.

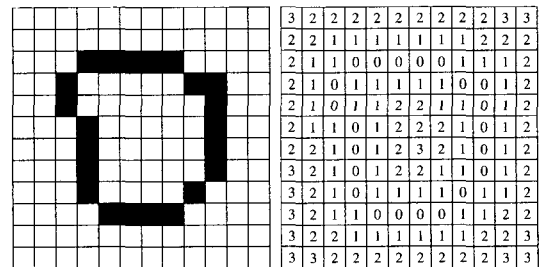


그림 1 체스판 거리함수를 이용하여 거리 변환한 결과 (a) 볼륨 데이터 (회색 : 비투명 복셀, 흰색 : 투명복셀) (b) (a)의 볼륨 데이터로부터 생성된 거리 맵

3.2 공간 도약법을 이용한 원근 볼륨 렌더링

광선 투영법에서 한 픽셀의 컬러는 그 픽셀에서 방사된 광선의 경로를 따라가면서 각 복셀들의 색상을 누적하여 계산한다. 이때 각 복셀이 최종 픽셀 색상에 기여

하는 정도는 불투명도 전사함수에 의해 지정된 각 복셀들의 불투명도 $a(x, y, z)$ 에 의해 결정된다. 그러면 특정 픽셀 $p(u, v)$ 의 컬러값 $I(u, v)$ 는 다음 식(5)와 같이 계산할 수 있다.

$$I(u, v) = \sum_{i=0}^N c_i \prod_{j=0}^i (1 - a_j) \quad (5)$$

$$c_0 + c_1(1 - a_0) + c_2(1 - a_0)(1 - a_1) + \dots + c_N(1 - a_0)\dots(1 - a_{N-1})$$

거리 맵에는 현재 위치로부터 최근접 경계 복셀까지의 거리가 저장되어 있으므로 이 정보를 이용하여 빈 공간에 대한 불필요한 샘플링을 피할 수 있다. 즉 한 복셀에서 샘플링을 하는 대신에 거리 맵에서 해당 위치의 값을 검사한다. 이 값이 0 이상인 경우는 빈 공간임을 의미하고, 그 점을 기준으로 저장된 값 $d(x, y, z)$ 에 해당하는 영역 내에서 경계 복셀이나 물체의 내부 복셀이 존재하지 않음을 의미한다. 따라서 저장된 값만큼 샘플링 위치를 도약할 수 있다. 이러한 도약은 거리값이 0이 될 때(경계 복셀에 도달할 때)까지 반복한다. 그림 2는 기존의 광선 투영법과 공간 도약법을 비교한 것이다. 그림에서 볼 수 있듯이 공간 도약법을 이용하면 샘플링 회수가 감소하기 때문에 상당한 속도 향상이 가능하다.

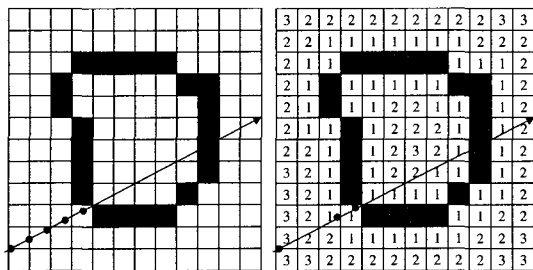


그림 2 광선 투영법(왼쪽)과 공간 도약법(오른쪽)의 샘플링 회수 비교

3.3 광선 진행 방향에 따른 적응형 샘플링 방법

기존의 광선 투영법에서는 단일 광선의 샘플 간격이 일정하다. 공간 도약법에서도 전처리 과정에서 미리 계산된 거리 값을 이용하여 그 값에 비례하도록 샘플 간격을 조정한다. 즉 고정된 기본 샘플 간격이 d 이고 도약할 거리가 ψ 라면 샘플 간격은 $d \times \psi$ 가 된다. 그러나 d 값이 고정되어 있으므로 단순히 선형비례 할 뿐이다. 그림 3는 이처럼 일정한 간격으로 샘플링 하는 공간 도약법이 비효율적임을 보여준다. 만약 O 점의 ψ 값이 7 이라면 $6 \times 6 \times 6$ 서브 볼륨 안에는 경계 복셀이 하나도

없다는 것을 의미한다. 그러나 광선의 방향에 따라 한번에 도약하는 거리는 달라지게 된다. 광선 $r(1,0,0)$ 은 한번에 서브 볼륨 밖으로 도약이 가능하지만 광선 $r(1,0,1)$ 과 $r(1,1,1)$ 의 경우 적어도 한번 더 거리 맵의 값을 참조해야 서브 볼륨 밖으로 벗어날 수 있다. 이 때문에 거리 맵 참조 회수와 도약 회수가 증가하게 되고 결과적으로 렌더링 속도가 느려진다.

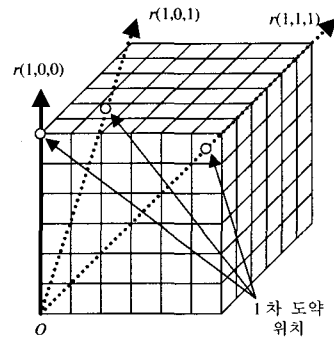


그림 3 고정 샘플 간격을 이용하는 경우 도약 회수의 증가 문제

이것은 유클리드 거리와 체스판 거리 사이의 오차 때문에 발생한 것이다. 본 연구에서는 이 오차를 보정하기 위해 광선의 진행방향에 따라 샘플 간격을 가변적으로 조정하는 “광선 방향에 따른 가변 샘플링”(ray direction dependent sampling) 방법을 제안한다. 광선 투영법에서의 샘플 간격을 d 라 할 때 제안된 방법에서의 샘플 간격은 다음 식(6)과 같다. 이 값은 주 시각축(major viewing axis)의 방향에 따라 달라진다.

$$\hat{d} = d \times \frac{\sqrt{(x_e - x_s)^2 + (y_e - y_s)^2 + (z_e - z_s)^2}}{\max(|x_e - x_s|, |y_e - y_s|, |z_e - z_s|)}$$

$$\hat{d} = d \times \frac{\sqrt{(x_e - x_s)^2 + (y_e - y_s)^2 + (z_e - z_s)^2}}{|x_e - x_s|} \quad (\text{주시각축} = X\text{축})$$

$$\hat{d} = d \times \frac{\sqrt{(x_e - x_s)^2 + (y_e - y_s)^2 + (z_e - z_s)^2}}{|y_e - y_s|} \quad (\text{주시각축} = Y\text{축})$$

$$\hat{d} = d \times \frac{\sqrt{(x_e - x_s)^2 + (y_e - y_s)^2 + (z_e - z_s)^2}}{|z_e - z_s|} \quad (\text{주시각축} = Z\text{축}) \quad (6)$$

여기서 $v(x_s, y_s, z_s)$ 는 특정 광선이 볼륨에 진입할 때 교차하는 복셀이고, $v(x_e, y_e, z_e)$ 는 광선이 볼륨을 빠져 나갈 때 교차하는 복셀이다. 이 값은 한 광선에 대하여 1번만 계산하면 된다. 그림 4는 광선 방향에 따른 적응형 샘플링을 하는 경우의 의사 코드를 보여준다.

```

CastRaySpaceLeaping(pixel P)
{
    initialize U1, U2; /* U1 = 볼륨 시작 경계, U2 = 볼륨 끝 경계 */
    U ← U1; /* U = 현재 샘플 위치 */
    ΔU ← (Δx, Δy, Δz); /* ΔU = 단위 광선 진행량 (d) 계산 : 식 (6) 참조 */
    c ← 0, α ← 0; /* c는 픽셀 색상, α는 누적 불투명도 */
    while( U ≤ U2 and α ≤ 1.0)
    {
        if( distance(U) > 0) /* distance(U) = 현재 샘플 위치에서의 거리값 */
            U ← U + ΔU × distance(U); /* 빈공간 도약 */
        else U ← U + ΔU; /* 정상적인 광선 진행 */
        (c, α) ← shading(U);
        /* shading(U)는 현재 위치에서 색상과 불투명도를 계산하는 함수 : 식 (5) 참조 */
    }
}
    
```

그림 5 광선 방향에 따른 가변 샘플을 이용한 공간 도약 알고리즘

그림 5는 본 논문에서 제안하는 방법을 사용할 경우 광선에 따른 샘플링 패턴을 2차원으로 도시한 것이다. 그림에서 볼 수 있듯이 광선의 방향이 기준평면과 이루는 각도가 클수록 샘플간격이 커지고 이 방향으로의 도약회수가 감소하게 된다. 결과적으로 렌더링 속도가 빨라진다.

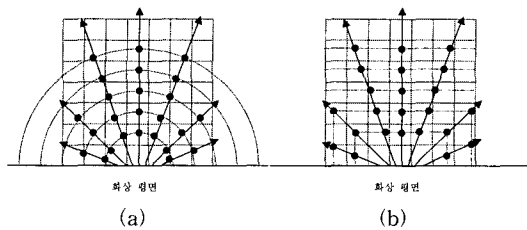


그림 5 광선 방향에 따른 샘플링 패턴 비교 (a) 기존 샘플링 방법 (b) 광선 방향에 따른 가변 샘플링 방법

4. 기타 가속화 방법

광선 방향에 따른 가변 샘플링을 이용한 공간 도약법 외에 고속 내시경 영상 생성을 위해서 몇 가지 추가적인 가속화 방법을 사용한다.

4.1 적응형 스크린 샘플링(adaptive screen sampling)

먼저 적응형 화면 샘플링 방법을 사용한다.[14] 화상 평면의 모든 픽셀들에 대해 광선투영을 할 경우 상당히 많은 시간이 소요되는 문제가 있다. 렌더링 속도를 향상시키기 위해 화면 최대 해상도의 1/4로 서브 샘플링(subsampling)하여 광선 투영을 하고 중간 픽셀의 컬러 값은 이들을 선형 보간하여 계산한다. 하나의 광선을 투영하는 속도에 비해 인접 픽셀값을 보간하는 시간은 무시할 정도이므로 이 방법을 사용할 경우 렌더링 속도를 최대 4배까지 향상시킬 수 있다.

4.2 경사도, 음영값 미리계산(gradient, shading value pre-calculation)

전처리 과정에서 각 복셀들의 경사도와 음영값을 미리 계산하여 저장해두면 렌더링 속도를 향상시킬 수 있다. 광선 투영을 위해서는 각 샘플 점에서의 밀도값과 경사도 값이 필요하다. 밀도값은 복셀 값을 3쌍선형 보간하여 계산하고, 경사도 값은 이웃한 두 복셀의 밀도값을 중심 편차 연산자(central difference operator)로 계산한 후 3쌍선형 보간으로 얻어낸다. 따라서 각 샘플에서 경사도 값을 계산하는데는 상당한 시간이 소요된다. 입력 볼륨 데이터가 변형되지 않는 한 경사도 값은 변경되지 않는 성질을 이용하여 전처리 단계에서 각 복셀들의 경사도값을 미리 계산한 후 입력 볼륨 데이터와 동일한 해상도를 가지는 “경사도 맵”에 저장한다.

각 픽셀의 음영 값을 계산할 때는 Phong 조명 모델(Phong illumination model)을 사용한다. 식 (7)과 같이 Phong 조명 모델은 주변광 성분(ambient light), 난반사 성분(diffuse reflection), 경면 반사 성분(specular reflection)으로 구성된다.

$$I = k_a I_a + \sum [k_d (\vec{L} \cdot \vec{N}) + k_s (\vec{V} \cdot \vec{N})^n] \quad (7)$$

광원의 위치가 고정되어 있고, 경면 반사 성분을 고려하지 않는다면 이것을 식 (8)과 같이 다시 쓸 수 있다. 광원 벡터가 상수이고 법선벡터는 경사도 맵을 계산하는 단계에서 만들어지므로 각 복셀의 음영값 (shading value)도 미리 계산하여 저장해 둘 수 있다.

$$I = k_a I_a + \sum [k_d (\vec{L}_{const} \cdot \vec{N})] \quad (8)$$

5. 볼륨 데이터의 충돌 검출

가상 내시경에서 고속 원근 렌더링과 더불어 중요한 것은 충돌 검출이다. 충돌 검출은 주로 computational geometry, 기하 모델링, 충돌 회피 경로 계획 등의 분

야에서 연구하고 있는데 이런 분야에서의 연구는 주로 표면 모델들 간의 상호작용에 초점이 맞춰져 있다. 불투화된 물체들에는 명시적 표면 정보가 없기 때문에 이를 처리하기 위한 적절한 방법이 제시되어야 한다.

초기에 제안된 방법[15]은 점유맵(occupancy map)을 이용하여 검사하고자 하는 위치에 다른 물체의 복셀이 매핑되어 있는지를 매번 검사해야 했다. 또한 물체의 복셀인지 아닌지가 명확하게 구별되지 않는 반투명 복셀에 대한 처리가 곤란했다. 불투화된 물체는 이산적으로 표현되므로 두 물체간의 상호작용을 명확하게 충돌/비충돌로 구분하는 것은 불가능하다. 다만 특정 위치에서 충돌여부를 확률적으로 표현할 수 있을 뿐이다. 이 때문에 불투 데이터의 충돌여부를 확인하기 위해 확률적 모델을 사용한다.[16] 불투에서 각 복셀들에는 충돌 확률 범위가 지정되는데 이것은 각 복셀의 밀도값과 밀도값의 경사도를 구분 인자로 사용하여 각각의 복셀에 [0,1] 사이의 불투명도를 지정하는 것과 동일하다.

렌더링 시간에 충돌 검출을 하려면 관심있는 각 복셀들의 충돌 가능성을 매번 계산해야 하는데 이것은 상당한 처리시간을 요구하게 된다. 공간 도약법에서는 전처리 과정에서 생성된 거리 맵을 충돌 검사에도 활용할 수 있기 때문에 다른 방법들에 비해 추가적인 비용이 들지 않는다. 거리 맵에는 가장 근접한 경계 복셀까지의 거리가 기록되어 있는데 이 값으로 충돌 가능성을 계산할 수 있다. 즉 특정한 복셀의 거리값이 작으면 충돌 가능성이 높은 것이고, 이 값이 크면 충돌 가능성이 낮다고 할 수 있다. 한번의 거리값 참조만으로 충돌 검사 가능하므로 복잡도는 $O(1)$ 이다.

$$\begin{aligned} D(x, y, z) &\geq \eta && : \text{충돌 안함} \\ \eta > D(x, y, z) &\geq 1 && : \text{충돌 임박} \\ D(x, y, z) &= 0 && : \text{충돌} \\ \text{여기서 } \eta &\text{는 임계치, } \eta > 1 \end{aligned} \quad (9)$$

6. 실험

여기서는 실험을 통해 기존 방법과 광선 방향에 따른 가변 샘플링 방법의 처리 시간을 비교하고 각각의 방법을 이용하여 생성된 영상의 화질을 비교한다. 실험은 Pentium III 933MHz CPU와 512MB의 메모리를 장착

한 PC에서 이루어졌고 프로그램은 Visual C++로 작성하였다. 실험 데이터는 $256 \times 256 \times 225$ 해상도로 인간의 두부를 CT촬영한 데이터(Bighead)와 대장 내시경을 위해 환자의 복부를 $256 \times 256 \times 256$ 해상도로 촬영한 데이터(Colon)를 사용하였다.

6.1 렌더링 시간비교

렌더링 시간을 비교하기 위해 기본적인 광선 투영법을 이용한 경우(RayCast), 광선 투영법과 광선 방향에 따른 가변 샘플링 방법을 결합한 경우(VarRC), 고정 샘플링에 기반한 공간 도약법(SpaceLeap)을 이용한 경우, 그리고 공간 도약법과 광선 방향에 따른 가변 샘플링 방법을 결합한 경우(VarSL)에 대해 각각 실험하였다. 데이터에 따른 렌더링 시간의 변이를 최소화 하기 위해 시선방향(viewing direction)을 종축방향(transverse direction)으로 0도부터 45도까지 각 5도씩 각도를 변경하면서 렌더링 시간을 측정 후 평균값을 사용하였다. 실험을 위해 1/4 해상도로 화면 서브 샘플링(screen subsampling)한 경우와 최대 해상도로 렌더링 한 경우를 각각 구현하여 측정 후 값을 비교하였다.

실험 결과를 보면 공간 도약법을 사용할 경우의 렌더링 시간은 기본적인 광선 투영법을 사용하는 것보다 20%(화면 서브 샘플링을 하는 경우) ~ 23%(최대 해상도로 렌더링 하는 경우) 정도 밖에 소요되지 않는다. 이는 공간 도약법을 사용하면 광선 투영법을 사용하는 경우보다 빈 복셀을 처리하는 시간이 상당히 많이 절약됨을 보여준다.

화면 서브 샘플링과 광선 방향에 따른 가변 샘플링을 병용하면 렌더링 시간은 고정 샘플링 하는 경우에 비하여 88%(광선 투영법) ~ 91%(공간 도약법) 정도 소요된다. 최대 해상도로 렌더링을 하는 경우는 85% ~ 90% 정도가 소요된다. 이것은 샘플간의 거리를 광선 방향에 따라 가변적으로 하면 그렇지 않은 경우에 비해 10 ~ 12% 정도의 속도 향상 효과를 얻을 수 있음을 의미한다. 뇌강 내시경이 대장 내시경에 비해 속도 향상 효과가 큰 것은 투명도를 조정하여 뇌내의 상당부분을 투명한 복셀로 지정하였기 때문이다. 결과적으로 기본적인 광선 투영법을 사용하는 경우에 비하여 광선 방향에

표 1 각각의 방법을 이용하는 경우 렌더링 시간비교(단위 msec)

		RayCast(A)	VarRC(B)	B/A(%)	SpaceLeap(C)	VarSL(D)	D/C(%)	D/A(%)
BigHead	subsampling	3196	2811	87.95	749	665	88.79	20.81
	full-resolution	12649	10806	85.43	2952	2494	84.46	19.72
Colon	subsampling	731	661	90.42	350	319	91.14	43.63
	full-resolution	2895	2604	89.95	1392	1252	89.90	43.25

따른 가변 샘플링에 기반한 공간 도약법을 사용할 경우 20%(bighead) ~ 43%(colon)의 시간만 소요된다는 것을 알 수 있다.

광선 투영법은 별도의 전처리 과정이 필요 없으나 공간 도약법에서는 거리 맵을 생성하는 과정이 필요하다. SpaceLeap과 VarSpaceLeap의 경우 "Bighead"의 거리 맵 생성시간은 37.13초가 소요된다. 불투명도 전사 함수는 한번 정해지면 자주 변경되는 것이 아니므로 거리

맵은 빈번하게 생성할 필요가 없다. 따라서 특정한 응용 분야를 제외하고는 공간 도약법이 광선 투영법에 비해 처리 속도가 상당히 빠르다고 할 수 있다.

6.2 화질 비교

여기서는 4가지 방법을 사용하는 경우에 화질의 차이가 있는지를 비교한다. 공간 도약법은 광선 투영법과 동일한 샘플링 방법을 사용하기 때문에 화질의 변화가 없다. 그러나 광선 방향에 따른 가변 샘플링 방법은 단일

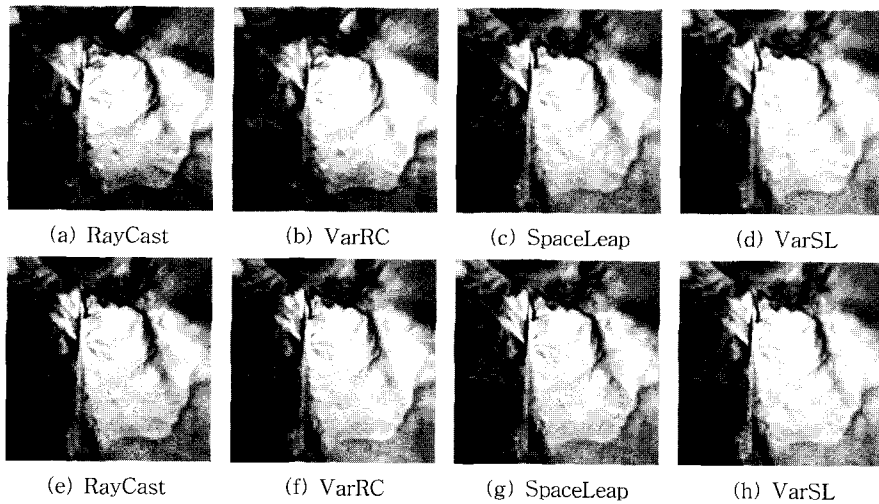


그림 6 BigHead의 내시경 영상 비교 (a)~(d) 1/4 해상도로 화면 서브 샘플링 후 보간하여 얻어진 영상 (e)~(h) 최대 해상도로 렌더링하여 얻어진 영상

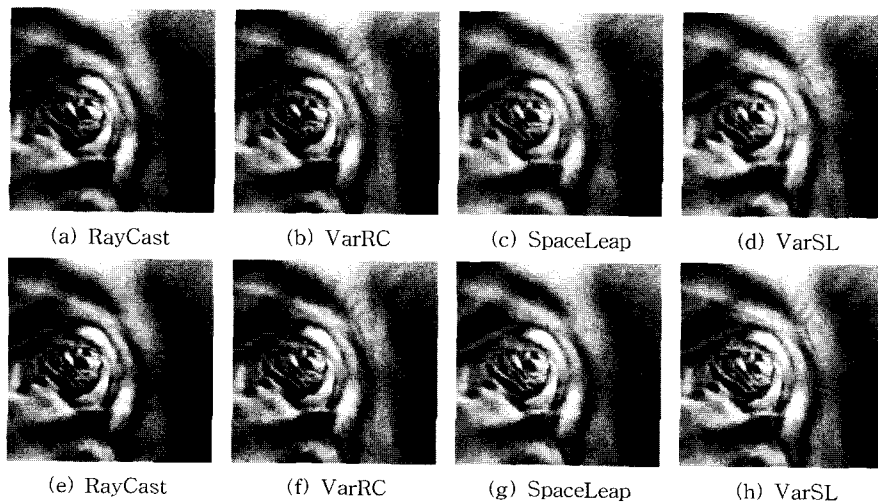


그림 8 대장 내시경 영상 비교 (a)~(d) 1/4 해상도로 화면 서브 샘플링 후 보간하여 얻어진 영상 (e)~(h) 최대 해상도로 렌더링하여 얻어진 영상

광선에서 샘플들 간의 간격이 넓어지기 때문에 화질이 다소 떨어질 수 있다. 여기서는 동일 각도에서 생성한 두 개의 화상을 비교하여 화질 차이가 얼마나 생기는지를 살펴본다. 그림 6은 두개강 내에 가상의 내시경을 삽입하여 촬영한 영상을 생성한 것으로서 관측방향은 경추에서 두정부쪽을 올려다 본 모양이 된다. 그림 7은 내시경 카메라로 대장 내부를 관찰한 영상이다. 그림에서 (a)-(d)는 1/4 해상도로 화면 서브 샘플링한 후 보간하여 얻은 영상이고, (e)-(h)는 최대 해상도로 렌더링하여 얻은 영상이다.

그림을 보면 최대 해상도로 촬영한 영상이 화면 서브 샘플링후 보간한 영상에 비해 다소 선명한 것을 볼 수 있다. 그러나 동일한 해상도에서 서로 다른 렌더링 방법을 사용한 경우는 보면 화질의 차이를 한눈에 알아보기가 어렵다. 이것은 광선 방향에 따른 가변 샘플링 방식으로 생성된 영상이 기존 방법들로 생성된 영상들과 비교하여 화질차가 그다지 크지 않음을 의미한다. 최대 해상도로 렌더링 할 것인지, 화면 서브 샘플링을 할 것인지는 카메라의 이동속도에 따라 결정한다. 즉 일정 속도 이상으로 카메라가 이동하는 경우는 다소 화질이 낮지만 신속하게 영상을 생성할 수 있는 화면 서브 샘플링 방법을 사용하고, 카메라가 천천히 이동하거나 정지하는 경우는 최대 해상도로 렌더링 하도록 한다.

7. 결론

일반적으로 화질과 처리속도는 상충관계에 있기 때문에 두 가지를 모두 만족하는 최적 렌더링 기법을 찾는 것은 쉽지 않다. 가상 내시경에서는 정확한 원근 투영 영상의 생성과 빠른 렌더링 속도가 모두 중요하다. 본 논문에서는 고속 원근 볼륨 렌더링 기법을 이용하여 정확하면서도 빠른 렌더링이 가능한 가상 내시경 기술을 제시하였다. 이 방법은 기본적으로 화질이 우수한 광선 투영법을 기반으로 하며 거리 정보를 이용하여 투명한 공간을 비약하도록 함으로써 렌더링 속도를 향상시키도록 하였다. 렌더링 과정에서 광선의 진행 방향에 따라 샘플 간격을 가변적으로 조절하는 방법을 이용하여 빈 공간을 신속하게 비약하도록 한다. 또한 전처리 단계에서 생성된 거리 정보를 이용하여 충돌검사를 간단히 할 수 있는 방법도 제안하였다. 실험 결과를 통해 광선 투영법이나 기존 공간 도약법으로 생성된 영상과 대등한 화질을 유지하면서 렌더링 시간을 단축시킬 수 있음을 보였다.

참고 문헌

- [1] Lorensen, W. E. and Cline, H. E., "Marching Cubes : A High Resolution 3D Surface Construction Algorithm," *Computer Graphics*, Vol. 21, No. 3, pp. 163-169, 1987
- [2] Levoy, M., "Display of Surfaces from Volume Data," *IEEE Computer Graphics and Applications*, Vol. 8, No. 3, pp. 29-37, 1988.
- [3] Yagel, R., "Volume Viewing: State of the Art Survey," *SIGGRAPH 97 Course Note 31*, 1997.
- [4] Yagel, R. and Kaufman, A., "Template-based volume viewing," *Computer Graphics Forum (Eurographics 92 Proceedings)*, pp. 153-167, Cambridge, UK, 1992.
- [5] Lacroute P. and Levoy, M., "Fast Volume Rendering Using a Shear-Warp Factorization of the Viewing Transformation," *Computer Graphics (SIGGRAPH 94 Proceedings)*, Orlando, Florida, pp. 451-458, 1994.
- [6] Pfister, H., Hardenbergh, J., Knittel, J., Lauer, H. and Seiler, L., "The VolumePro Real-Time Ray-Casting System," *Proceedings of SIGGRAPH 99*, pp. 251-260, Los Angeles, CA, 1999.
- [7] Wan, M., Li, W., Kreeger, K., Bitter, I., Kaufman, A., Liang, Z., Chen, D. and Wax, M., "3D Virtual Colonoscopy with Real-time Volume Rendering," *SPIE Medical Imaging 2000*, <http://www.cs.sunysb.edu/~vislab/papers/colonoscopy/final.pdf>
- [8] Westermann, R. and Eart, T., "Efficiently Using Graphics Hardware in Volume Rendering Applications," *Computer Graphics*, pp. 167-177, 1998.
- [9] Yagel, R. Kaufman, A., Cabral, B., Cam, N. and Foran, J., "Accelerated volume rendering and tomographic reconstruction using texture mapping hardware," *Symposium on Volume Visualization*, pp. 91-97, 1994.
- [10] Ma, K., Painter, J., Hansen, C. and Krogh, M., "A data distributed, parallel algorithm for ray-traced volume rendering," *Proceedings of the 1993 Parallel Rendering Symposium*, pp. 15-22, San Jose, 1993.
- [11] Yagel, R. and Shi, Z., "Accelerating volume animation by space-leaping," *Proceedings of IEEE Visualization '93*, pp. 62-69, 1993.
- [12] Sramek, M. and Kaufman, A., "Fast Ray-Tracing of Rectilinear Volume Data Using Distance Transforms," *IEEE Transactions on visualization and computer graphics*, Vol. 6, No. 3, pp. 236-252, 2000.
- [13] De Assis Zampiroli, F. and De Alencar Lotufo, R.

- "Classification of the distance transformation algorithms under the mathematical morphology approach," Proceedings XIII Brazilian Symposium on Computer Graphics and Image Processing, pp. 292-299, 2000.*
- [14] Danskin, J., and Hanrahan, P., "Fast algorithm for volume raycasting." *Workshop on Volume Visualization*, pp.91-98, 1992.
- [15] Gibson, S., "Beyond Volume Rendering : Visualization, Haptic Exploration, and Physical Modeling of Voxel-Based Objects," *6th Eurographics Workshop on Visualization in Scientific Computing*. May 1995.
- [16] He, T. and Kaufman, A., "Collision Detection for Volumetric Objects," *Proceedings of IEEE Visualization '97*, pp. 27-34, 1997.



신 병 석

1990년 서울대학교 공과대학 컴퓨터공학과. 1992년 서울대학교 대학원 컴퓨터공학과 석사. 1997년 서울대학교 대학원 컴퓨터공학과 박사. 2000년 ~ 현재 인하대학교 컴퓨터공학부 조교수. 관심분야는 실시간 렌더링, 볼륨그래픽스, 의료영상

김 혁

1997년 중국 북경대학교 전산학과. 2000년 서울대학교 대학원 전산과학과 석사. 2000년 ~ 현재 미국 조지 워싱턴대학교 대학원 전산과학과 박사과정. 관심분야는 볼륨그래픽스, 의료영상, 영상기반 렌더링, Global Illumination and BRDF modeling.