

체적 데이터의 특징점을 이용한 효율적인 볼륨 렌더링

김진열* 김형균*

*조선대학교 컴퓨터공학과
e-mail:zerokjy@hotmail.com

Efficient Volume Rendering using Characteristic point of volume data

Jin-Youl Kim* Hyeong-Gyun Kim*

*Dept of Computer Engineering, ChoSun University

요 약

본 논문에서는 3차원 형태로 체적 데이터를 효율적으로 렌더링 하기 위해서, 체적 데이터의 특징점을 추출하고 이를 이용하여 3차원 형태로 복원하기 위해 3D Points 추출을 위한 PEF 과정과 정점 변환 파이프라인 과정[6,7]을 통한 렌더링 방법을 제안한다. 기존의 광선 추적에 비해 저화질의 영상을 나타내지만, 추출된 데이터만 고려하기 때문에 계산량이 줄어 들어 그만큼의 렌더링 속도가 빨라짐을 볼 수가 있다. 또한, 기존의 광선 추적 기법에서 표현하는 회전, 절단, 축소/확대의 기능을 OpenGL을 이용하여 3차원 렌더링 프로그램으로 제작하였다.

1. 서론

볼륨 렌더링이란 현실공간에 있는 물체를 컴퓨터라는 가상 공간으로 이동시켜 가시화 하는 방법이다. 즉, 실제 물체를 CT나 MRI같은 장치로 샘플이 되어 얻어진 단층 이미지를 통해서 체적 데이터를 시각화 하는 방법이다. 볼륨 렌더링의 응용 분야를 살펴보면 Medical Imaging, Modeling, Paleontology, Computational Fluid Dynamics, Education, Nondestructive Testing, Microscopic Analysis, Oil Exploration 등 여러 분야에 응용이 되어 특수한 목적으로 활용되고 있는 실정이다.[1]

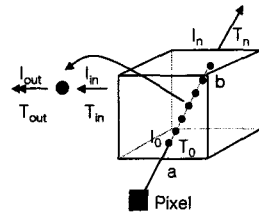
일반적으로, 볼륨 렌더링을 하기 위한 기법중 대표적인 광선 추적 기법이 있는데, 이방법은 고화질의 렌더링의 결과는 얻겠지만, 매번 특정 변환을 거칠때마다 다시 처음부터 체적 데이터에 대해 계산이

이루어져야 한다는 단점을 가지고 있어 많은 계산량과 처리 시간이 길어져 비효율적인 렌더링 방식이라 말할 수 있다.

따라서, 본 논문에서는 체적 데이터를 PEF 단계를 거치고, 거기서 얻은 특징한 체적 데이터를 컴퓨터 그래픽스에서의 정점 변환 파이프라인 단계를 통해 효율적인 체적 렌더링을 하는 방법을 제안한다.

여기서, 제안한 PEF(Point Extraction Function) 과정은 크게 3단계로 고주파 강화 필터링, 개량형 모폴로지 제거 연산, 특징점 추출로 분류되어 처리한다.[5] 그 다음으로, 정점 변환 파이프라인 단계에서 Modelview Transformation, Projection Transformation, Perspective division Transformation, ViewPort Transformation을 거쳐 화면에 출력이 이루어진다.[6,7]

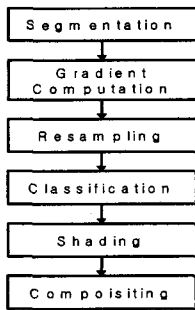
제안한 방식을 통한 렌더링은 기존의 광선 추적 기법에 비해 추출된 데이터만을 렌더링시에 고려하기 때문에 화질과 정교한 묘사가 좀 떨어지지만, 처리량 감소와 빠른 처리시간으로 효율적인 렌더링이 수행됨을 볼 수가 있다. 또한, 본 논문에서는 광선추적기법에서 표현하는 회전, 절단, 축소/확대 효과[2] 등을 제안된 기법을 통해 OpenGL를 이용하여 표현할 수 있도록 3차원 렌더링 프로그램을 제작 하였다.



<그림2> 광선 추적 과정

2. 전통적인 광선 추적 기법을 이용한 렌더링

블룸 렌더링은 CT나 MRI와 같은 장비를 통해 얻어진 3차원 데이터를 통해 가시화 한다. 복잡한 영역의 표현이 가능할 뿐 아니라 내부 영역의 구조도 표현할 수 있는 특징을 갖고 있다. 일반적인 블룸렌더링의 과정은 그림1과 같이 Volume Rendering Operations 과정을 통해 가시화 하게 된다.[1,3]



<그림1> Volume Rendering Operations

그리고, 광선 추적 기법은 체적 데이터의 모든 정보를 사용하여 2차원 평면에 3차원 물체를 표현하는 대표적인 알고리즘으로 컴퓨터 그래픽스에서 3차원 공간상의 물체를 정사 평행 투영하여 2차원 투사 평면에 투영이 되는데, 광선 추적 기법은 거꾸로 2차원 투사 평면에서 특정 화소로부터 3차원 공간상의 체적소들의 위치를 추적하는 방식이다. 여기서, 샘플링 값들은 3차원 격자인 블룸 데이터를 삼차선 정보간법에 의해 이루어지는 가상의 연속적인 공간으로 보고, 광선을 따라 일정 간격으로 체적 데이터가 광선의 밝기에 기여하는 정도를 계산하여 누적한다. 표현은 그림 2와 같이 Front-To-Back 방식으로 광선 추적 위치의 거리는 체적소들에 대해 표면법선 벡터 연산, 복셀간 보간 연산, 영역분할, 셰이딩, 누적작업을 통해 픽셀의 화소값을 결정하게 된다.

각 광선 추적 선상에 있는 체적소에 대해 계산되어 나온 결과를 Front-To-Back 방식을 통해서 누적 과정을 수식1에서 살펴볼 수 있다.[1, 4]

$$I(a,b) = \sum_{i=0}^n I_i \prod_{j=0}^{i-1} (1 - \alpha_j)$$

$$\sum_{i=0}^n I_i \prod_{j=0}^{i-1} (1 - \alpha_j) = I_0 + I_1(1 - \alpha_0)$$

$$+ I_2(1 - \alpha_0)(1 - \alpha_1) + \dots + I_n(1 - \alpha_0)(1 - \alpha_{n-1})$$

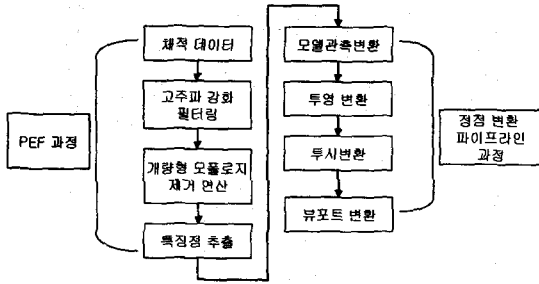
$$= I_0 OVER I_1 OVER I_2 OVER \dots OVER I_n$$

<수식1> Front-To-Back Compositing

지금까지, 광선추적 과정에 대해 간략히 살펴보았다. 광선 추적 기법은 고화질의 정교한 렌더링을 수행하고 블룸 데이터의 내부구조에 관한 정보를 유지함으로써 다양한 분석과 응용이 가능하다는 장점을 가지고 있는 반면에 많은 메모리와 렌더링 시간을 필요로 한다는 큰 문제점을 동시에 가지고 있는 점을 고려해서 3장에서 제안한 과정을 통해 계산량과 렌더링 시간을 단축 시키면서 효율적인 렌더링을 수행할 수 있는 방법을 제안한다.

3. 체적 데이터의 특징점을 이용한 효율적인 블룸 렌더링

본 논문에서 제안한 체적 데이터의 특징점 추출을 통한 체적 렌더링을 수행하기 위한 체적 묘사 과정은 그림3과 같다. 체적 데이터에서 물체를 표현하는 특징점들을 추출하여 이 데이터를 컴퓨터 그래픽스 기법을 이용하여 3D 체적 렌더링을 하게 된다.

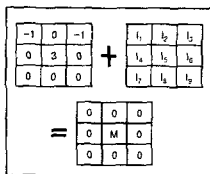


<그림3> 체적 묘사 과정

3.1 PEF 과정

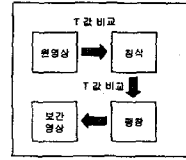
PEF(Point Extraction Function) 과정은 체적 데이터로부터 보간과정을 거친 체적소들에 대해 특징점을 추출하는 과정이다. 일반적으로, 컴퓨터 그래픽스에서 물체를 작고 단순한 모양을 다양하게 나열하고 조합함으로써 표현을 하게 된다. 여기서 표현하게 되는 작은 단위인 Point를 이용한다. 그리고, 해당 좌표 특징점을 추출하기 위해서는 3단계의 과정을 거친다.

(1) 고주파 강화 필터링 : 저주파 성분은 차단하고 고주파 성분을 부각시키는 필터링으로 영상에서의 윤곽선을 뚜렷하게 해주는 효과로 흐릿한 영상이나 블러링된 영상을 아주 선명하게 보기 위한 필터링으로 영상에서 주로 사용된다.[5] 여기서는 단층이미지의 각 영역의 경계에 대해 부각시키기 위한 효과로 간단한 고주파 강화 마스크를 제시했다.



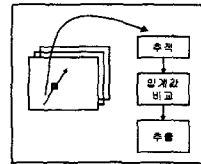
<그림4> 고주파 강화 마스크 연산 과정

(2) 개량형 모폴로지 제거 연산 : 그레이 영상에서 침식연산후 팽창연산을 수행하는 과정[5]인데, 개량형으로 임계값 조건이 맞는 체적소만이 고려되도록 하였다. 여기서, 마스크(3×3)로 9개의 체적소 중에서 가장 작은 세기 값을 가진 체적소를 결과 체적소로 세기 값으로 결정하는 침식연산과 가장 큰 세기 값을 가진 체적소의 세기 값을 결정하는 팽창연산 수행함으로써 단층이미지의 물체의 경계 부분의 노이즈를 다소 제거 할 수 있었고, 같은 선상의 체적소들의 세기가 약간 다를 경우 같은 세기로 보간하는 효과를 얻을 수 있었다.



<그림5> 개량형 모폴로지 제거 연산

(3) 특징점 추출 : 위의 두 단계를 거쳐 얻은 체적소들에 대해 그림6과 같이 물체라 인식되는 부분의 세기를 임계값으로 적용하여 특징점들을 추출하게 된다. 이 과정을 통해 체적소의 위치와 물체의 세기를 추출하게 된다.



<그림6> 특징점 추출

3.2 Vertex Transformation Pipeline 과정

3D 공간상에서 점, 선등 다양한 primitive들로 그린 모양들을 물체와 viewer와의 관계를 고려해서 하나의 화면으로 모으는 작업이 필요하게 된다. 그 과정이 바로 Vertex Transformation Pipeline (정점 변환 파이프라인)의 과정을 거치게 된다.[6,7]

그림3에서 보는것과 정점 변환 파이프라인 과정은 크게 네단계를 거쳐 실공간의 좌표 데이터가 원공간의 좌표 데이터로 이동해 화면상에 보여지게 된다. Modelview 단계에서 primitive들을 투영 범위내에 물체를 위치시키고 특정 변환을 주기 위해서 Translation, Rotation, Scaling의 변환을 주고 Projection 단계에서 Orthographic과 Perspective Projections이 있는데, 첫번째의 투영은 뷰잉 볼륨이 모든 면이 같은 크기의 사각형으로 화면에 정확한 모양과 치수를 표현할 때 자주 사용된다. 본 논문에서도 이 방식을 택했고, 두번째의 투영은 뷰잉 볼륨의 뒷부분과 앞부분의 넓이를 다르게 주어 같은 크기의 물체라도 관찰자로부터 멀리 떨어진 물체를 작게 나타내어 원근감을 제공한다. Perspective Division 단계에서 좌표계를 표준화의 작업을 수행하고, Viewport 단계에서 Clipping영역에 대한 지정으로 확대·축소 할 수 있게 한다.

실질적으로, 체적소들을 Primitive 형태로 변환하여 그 데이터를 Vertex Transformation Pipeline 과정을 통해 렌더링을 수행하게 되며, 특정 변환위치에 제어 변수를 선언함으로써 자유자재로 회전, 이

동, 크기를 제어할 수가 있게 된다. 그리고 추출된 체적 데이터의 세기를 임계값을 부여함으로써 다양한 형태로의 렌더링을 가능케 한다.

4. 프로그램 구현 및 실험 결과

본 논문에서 제시한 과정을 통해 간단한 3D 체적 렌더링 프로그램을 제작하였다. 본 프로그램의 개발 환경은 Pentium(R) 4 CPU 2.00GHZ로 운영환경은 Window 2000 Sever, 메모리 512MB, 개발 툴은 Microsoft Visula Studio 6.0, OpenGL의 glut 라이브러리를 사용하여 프로그램을 제작 하였다.

프로그램은 전통적인 광선 추적 기법에서 표현가 능한 불투명도, 회전, 절단, 축소·확대의 효과를 본 논문에서 제안한 과정을 통해 시뮬레이션한 결과를 분석해 보았다.



<그림7> 기본 렌더링

그림7은 체적 데이터 파일인 엔진 이미지에 대한 렌더링을 한 결과이다. 여기서, 사용자의 입력에 의해 임계값에 따라 렌더링을 수행하도록 하였다. 임의의 특정 임계값을 주었기 때문에 완벽한 잡음 제거와 매끄러운 렌더링이 수행 되지 않았지만, 기존의 방식의 약간 차이가 있지만 효율적으로 렌더링이 수행됨을 볼 수가 있었다.

그림8은 광선추적기법의 몇가지 표현기법들을 그대로 적용하여 렌더링한 결과를 보여주고 있다.



(a)x축회전 0도 60도 120도



(b)x축회전 0도 60도 120도



(c)절단 y=50~100 x=50~100 x,y=50~100



(d) 원영상 축소 확대

<그림8> 영상 조작 예

5. 결론

본 논문에서는 체적 데이터로부터 특징점을 추출하여 3D 체적 렌더링하는 방법을 제안하였다. 이방법을 사용함으로써 광선 추적 기법에 비해 체적소가 상대적으로 적은 데이터가 사용되었다. 여기서, 제안한 PEF 과정과 정점 변환 파이프라인 과정을 통해 광선 추적 기법에 표현면에서는 어느정도 가깝게 렌더링을 수행했지만, 처리 속도면에서 확실히 빨라짐을 볼 수가 있었다. 향후의 연구과제로는 현재 기법에서 다른 Primitive 객체를 사용한 표현과 질리 효과를 추가시키면 보다 효율적인 렌더링을 수행할 수 있을 것이라고 기대가 된다.

참고문헌

[1] A. Kaufman, "Introduction to Voume visuali-zation", IEEE Computer Society press, 1991
 [2] K. Bo-Sun, C. Woo-Young, "Volume Rendering of Segmented Medical Images", Journal of Research institure of Industrial Technology, 2001
 [3] M. Levoy, "Display of surfces from volume data", IEEE computer Graphics and Applications, 1988
 [4] M. Levoy, "Efficient ray tracing of volume data", ACM Transctions on Graphics, 1990
 [5] C. Gonzale, E. Woods, "Digital Image Process-ing", Addison-Wesley, 1992
 [6] Richard S. Wright, Jr.Michael Sweet "OpenGL SuperBible" 2rd Ed. 인포북, 2000
 [7] M. Woo, J. Neider. T. davis, D. Sheiner "OpenGL Programming Guide" 3rd Ed. Addison-Wesley, 1999