

# 옥트리를 이용한 3차원 물체 추출에서의 모델링 향상 기법

임석현<sup>0</sup>, 한창호

인하대학교 전자계산공학과

e-mail : g1991275@inhavision.inha.ac.kr chhan@inha.ac.kr

## Modeling Improvement Algorithm for 3D Volume Refinement Using Octree

Suk Hyun Lim<sup>0</sup>, Chang Ho Han

Dept. of Computer Science and Engineering, Inha University

### 요약

옥트리(octree)는 3차원 물체를 복원함에 있어서 간편함으로 많이 이용되어지는 방법이다. 하지만 물체의 형태가 복잡해지고 물체 내부에 구멍이 뚫어져 있는 경우 옥트리로 표현된 결과만으로는 형태를 파악하기 어려운 경우가 발생한다. 본 논문은 옥트리를 이용하여 3차원 물체의 형태를 복원하는데 있어서 나타나는 문제점을 옥트리의 계층적 구조를 증가시키지 않고 현실감이 떨어지는 부분에 대해서 CSG모형을 이용하여 해결하는 방안을 제시한다. 옥트리로 만들어진 결과물과 3차원 물체와의 차이 가 나는 부분에 대하여 좀 더 현실감을 주기 위하여 CSG모형을 이용하여 기본도형을 만들고, 만들어진 도형을 다시 CSG모형을 이용하여 옥트리로 만들어진 물체에 결합시키는 형태로 문제를 해결한다. 본 논문에서는 실험을 위하여 3차원 물체를 만들고, 이를 옥트리를 이용하여 문제점을 확인하고, 이를 본 논문에서 주장한 방법을 이용하여 해결하는 방안을 제시한다.

### 1. 서론

영상(image)을 이용하여 3차원 물체의 형태를 복원하는 일은 컴퓨터 그래픽스 분야뿐만 아니라 컴퓨터 비전 분야에서 오래 전부터 관심의 대상이었다. 실세계의 물체와 거의 같은 형태의 물체를 만들어 내는 것은 의료분야, 건축 분야, CAD/CAM, 로봇분야 등 여러 분야에서 반드시 필요한 부분이다.

3차원 물체의 형태를 복원해 내는 방법은 크게 물체의 그림자(silhouette)를 이용하는 방법, 물체의 윤곽선(contour)을 추출하는 방법, 스테레오(stereo)를 이용하는 방법, 방향성을 가지는 입자를 이용하는 방법(oriented particle)등 기존의 많은 방법이 발표되었다[1].

이들 방법 중 영상의 그림자로부터 3차원 모습을 만들어 내고 이를 옥트리를 이용하여 3차원 물체를 모델링(modeling)하는 방법은 다른 방법에 비해 간편함으로 인하여 많이 사용된다. 하지만 옥트리를 이용하는 방법에는 몇 가지 문제점이 발생한다. 첫 번째로 옥트리 구조는 표면 정보를 이용하여 계산을 하므로 물체의 가운데 부분은 큰 정육면체로 채워지고 표면으로 갈수록 작은 정육면체로 채워지게 된다. 그러므로 컵과 같은 형태에서는 안쪽의 구멍이 없이 막힌 구조로 표현이 된다. 두 번째로 3차원 물체의 형태가 급격히 바뀌거나, 원한체(torus)나 튜브(tube)와 같이 안쪽에 구멍이 있는 경우 옥트리로 표현된

결과만으로는 현실감의 부족으로 인하여 물체의 형태를 제대로 파악할 수가 없다. 물론 옥트리의 계층적 구조 단계를 증가시킴으로서 해결은 가능하지만 메모리(memory)와 시간 비용이 증가한다.

본 논문에서는 CSG모형을 이용하여 물체의 현실감이 떨어진 부분과 비슷한 물체를 만들어 내고, 이를 다시 옥트리로 만들어진 결과물과의 불리안 연산(boolean set operation)을 통하여 최종 결과물을 만들어 낸다. 이런 방법으로 문제를 해결함으로써 기존에 옥트리를 이용하여 만들어진 형태를 실세계의 물체와 거의 같은 형태로 모델링하였다.

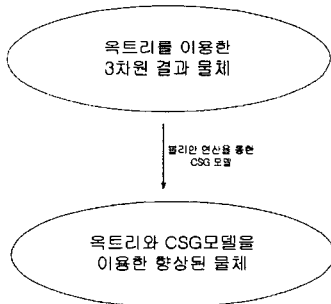
### 2. 관련연구

3차원 물체의 경계(bounding) 정보를 이용하여 물체를 근사 형태로 복원하는 알고리즘으로는 다면체(polyhedron)를 이용하는 방법[2]과 옥트리를 이용하는 방법[3]이 있다. 옥트리는 처음 기하학적인(geometric) 형태를 효과적으로 표현하기 위하여 개발되었다. 당시에는 컴퓨터 상에서 입체의 객체를 만들어 옥트리로 표현하였다. 이 후 실세계 물체에서 얻은 3차원의 영상을 이용하여 옥트리를 구성하였다. 초반부에는 실세계의 물체를 13개의 표준 뷰(standard view)로 나누어서 영상을 얻는 정사영 투시기법(orthographic projection)을 이용하여 옥트리를 구성하

게 되었다. 이 후에 임의의 뷰에서 영상을 얻어서 제작하는 옥트리 모델과 원근기법(perspective projection)을 이용한 옥트리 모델이 발표되었다. 최근의 연구에서는 극초단파 턴테이블(microwave turntable)에 3차원 객체를 올려 놓고 물체의 영상을 얻어 이를 옥트리로 표현하는 방법이 발표되었다.[6] 또한 옥트리의 속도를 향상시키기 위하여 2차원에 옥트리 객체를 투사시키고 투사된 결과를 이용하여 옥트리 표현될 영역을 제한시키는 방법이 발표되었다.[6]

### 3. 옥트리를 이용한 3차원 볼륨 추출에서의 CSG 모델을 이용한 개선된 모델링 알고리즘

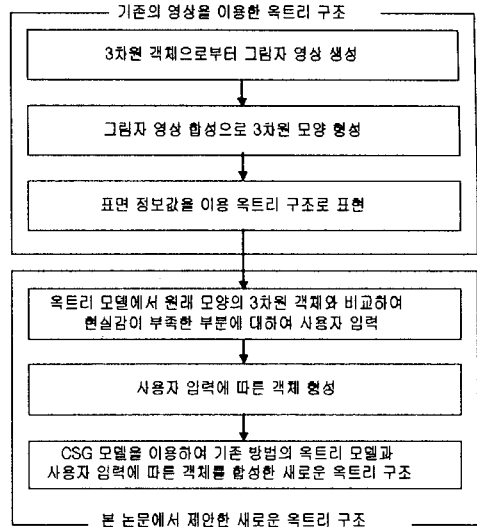
실제 세계의 물체와 거의 차이가 없는 물체를 모델링하는 과정은 매우 중요하다. 이를 위해 본 논문에서 주장하는 알고리즘은 실제 세계의 물체와 비슷한 형태로 모델링하기 위하여 옥트리만으로 표현된 3차원 결과물에 CSG 모델을 이용하여 좀 더 현실감있게 모델링하는데 그 목적이 있다. 다시 말해 옥트리의 계층적 구조를 증가시키지 않고 현실감이 떨어지는 부분에 대해서만 부분적으로 사용자 입력으로 받아드려 객체를 만들고 이를 기존의 옥트리 결과물과 합성하여 결과적으로 좀 더 현실감 있게 모델링하는데 그 목적이 있다(coarse-to-fine). [그림 1]은 본 논문에서 제시한 개선된 옥트리 모델을 도식화한 것이다.



[그림 1] 개선된 옥트리 모델

전체 알고리즘은 [그림 2]와 같다.

첫 번째 단계로 턴테이블에 3차원 물체를 올려놓는다. 턴테이블을 회전시켜 원하는 각도에서의 그림자 영상을 얻어낸다. 그림자 영상은 3차원 물체와 다른 색을 가지고 있는 배경과 픽셀(pixel) 비교를 통하여 얻어낸다. 이 때 물체의 각도를 알기 위해 턴테이블에 미리 붙여놓았던 이진 코드(binary code)와 3차원 객체의 영상을 동시에 얻는다. 두 번째 단계로 첫 번째 단계에서 얻어진 그림자 영상과 카메라의 투사 중심(center of projection)과의 일반화된 원뿔들(generalized cones)을 얻는다. 각 각도에서의 일반화된 원뿔을 서로 합성하여 표면 정보를 가지고 있는 3차원 형태를 만들어 낸다. 세 번째 단계로 두 번째 단계에서 만들어진 3차원 모양을 옥트리를 이용하여 3차원 객체의 형태로 만든다. 옥트



[그림 2] 본 논문의 알고리즘

리는 단순히 정육면체 안에 물체의 표면 정보의 유무를 판단함으로써 이루어진다. 해당 정육면체에 물체의 표면 정보가 없으면 그 정육면체는 알고리즘 과정에서 제외된다. 또한 물체의 표면 정보가 정육면체 안에 있으면 그 부분은 해당 정육면체로 표현이 된다. 물체의 표면 정보가 있는 부분과 없는 부분이 해당 정육면체에 공존하고 있을 경우에는 재귀적 호출을 이용하여 사용자가 정해놓은 값 이하의 범위에서 분해된다.

네 번째 단계는 만들어진 기존의 옥트리 모델에서 실제 세계의 물체와 비교하여 현실감이 많이 떨어지는 부분에 대해서만 사용자 입력을 받는다. 이 부분은 일반적인 그래픽 툴(tool)에서와 마찬가지로 직선, 사각형, 원 등의 기본적인 모형을 자유곡선과 같은 모형을 이용하여 사용자와 옥트리 모델간의 대화식 방법을 통하여 유사한 모형을 만든다.

다섯 번째 단계는 이렇게 만들어진 모형과 기존의 옥트리 모델과의 합성이다. 합성 방법은 CSG 모델에서 제안한 방법과 유사하다. 합집합, 교집합, 차집합과 같은 불리언 연산자를 이용하여 기존의 옥트리 모델과 제작한 객체를 합성한다.

이와 같은 방식을 통하여 적은 수준의 옥트리 계층 구조로 표현된 결과물 중에서 현실감이 상대적으로 떨어지는 부분에 대해서만 사용자 입력으로 객체를 추가함으로써 좀 더 현실감있는 모델링 결과를 얻게된다.

### 4. 실험 결과

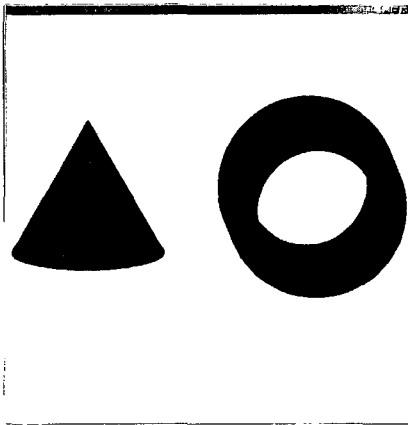
전체 과정의 실험은 실험장비의 부족으로 어려움이 있으므로, 본 논문에서는 PC에서 3차원 모델을 제작하고 이를 옥트리로 표현하여 문제점을 확인하고 이를 개선하는 방

안으로 진행하였다.

사용 PC는 P-III 550 (RAM 128MB) PC에서 진행하였고, OpenGL과 Visual C++ 6.0을 이용하여 구현하였다. 자세한 실험 과정은 다음과 같다.

1) 3차원 모델 제작과 로드(load)

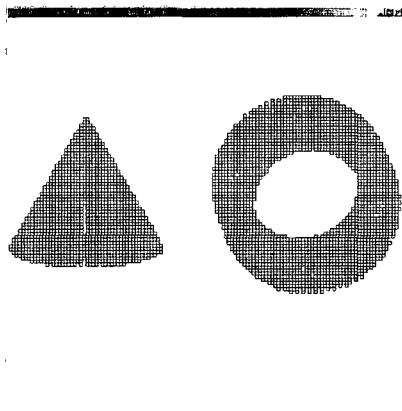
3차원 모델은 3D studio MAX R3.1에서 제작하였다. 본 논문에서는 간단한 모델링 실험을 위하여 원뿔과 튜브를 이용하였다. 튜브는 위쪽을 향하고 있고, 원뿔은 약간 아래쪽을 바라보고 있다. 제작한 모델을 PC상으로 가져오기 위하여 로더(loader)를 제작하였다.[10] 제작한 로더를 가지고 3D Studio MAX R3.1에서 작업한 객체를 표현한 그림이 [그림 3]이다.



[그림 3] 3D Studio에서 제작한 물체

2) 옥트리 모델 제작

PC에서 읽어드린 3차원 물체의 정보를 옥트리 표현 방법으로 바꾸어 표현한다. [그림 4]는 [그림 3]의 자료를 이용하여 8 단계 재귀적 호출 이용하여 옥트리로 표현한 그림이다. 왼쪽의 원뿔과 같이 닫힌 구조(close structure)인

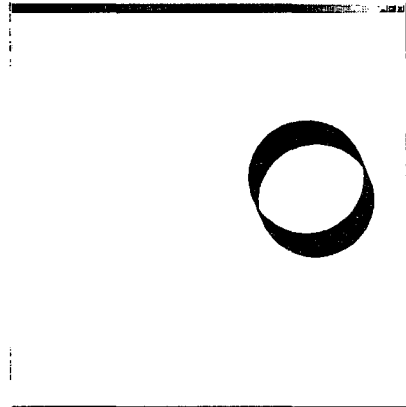


[그림 4] 옥트리 모델

경우에는 물체의 모델링 결과가 실제 3차원의 물체와 큰 차이가 없다. 하지만 오른쪽은 튜브와 같은 열린 구조(open structure)에서는 옥트리로 표현된 결과만으로는 튜브 모양을 가지고 있다는 것을 알 수 없다. 3차원 튜브의 형태가 아니라 타원형의 구멍이 뚫린 물체와 같은 결과를 보여주고 있다.

4) 옥트리 모델과 실제 3차원 물체와 비교하여 현실감이 떨어지는 부분에 대한 객체 형성

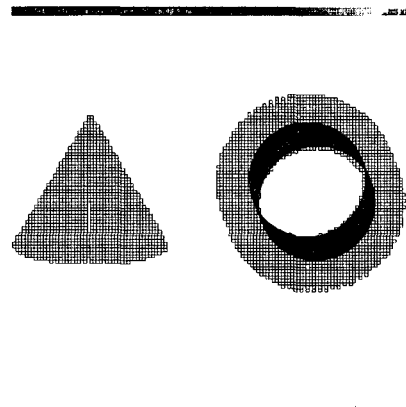
[그림 4]의 문제를 해결하기 위하여 현실감이 떨어지는 부분에 대하여 동일한 형태를 가진 객체를 사용자 입력으로 제작한다. 본 논문에서 사용한 예제에서는 튜브에서 윗부분과 아랫부분의 원을 연결하는 원통 구조이다. [그림 5]는 기존의 옥트리 모델과 원래의 3차원 모델의 차이를 줄이기 위한 사용자 입력 객체이다.



[그림 5] 사용자 입력 객체

5) CSG모델을 이용한 개선된 옥트리 모델 제작

이렇게 제작된 [그림 5]의 객체를 [그림 4] 옥트리 모델과 불리안 연상을 이용하여 합성을 한다. [그림 6]은 본 논문에서 제안한 방법으로 만들어진 최종 결과물이다.

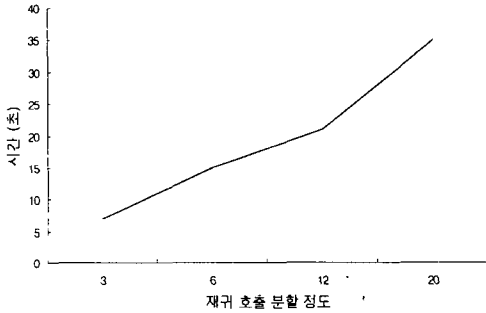


[그림 6] 개선된 옥트리 모델

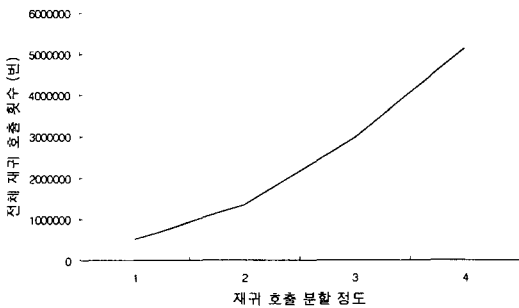
[그림 6]에서 보는 것과 같이 모델링되어진 결과에서 현실감이 떨어진 부분에 대해서만 사용자 입력을 통한 더욱 향상된 모델링 결과를 보여주고 있다.

6) 옥트리 분할 정도에 따른 시간과 메모리 측정

본 논문은 시간과 메모리 비용이 많이 들어가는 옥트리의 재귀 분할 정도를 줄이고 현실감이 떨어지는 부분에 대해서만 사용자 입력으로 객체를 추가하여 좀 더 현실감 있는 모델링을 하는데 그 목적이 있다. 아래의 [그림 7]과 [그림 8]은 옥트리 분할 정도에 따른 시간과 메모리 측정 결과이다. 재귀 호출 분할 정도에 따라 시간과 전체 재귀 호출 횟수가 급격히 증가함을 알 수 있다. 메모리 측정 결과인 경우는 측정의 어려움으로 인하여 전체 재귀 호출 횟수로 대신하였다.



[그림 7] 재귀 호출 분할 정도에 따른 시간 비교



[그림 8] 재귀 호출 분할 정도에 따른 전체 재귀 호출 횟수 비교

5. 결론 및 향후 연구 방향

본 논문에서는 옥트리를 이용한 3차원 물체 복원시 실제 3차원 물체와 차이가 나는 부분에 대해서만 사용자 입력으로 해당 부분의 객체를 만들고 이를 CSG모델을 이용하여 기존의 옥트리 모델과 합성하여 좀 더 현실감 있는 모델링 결과를 얻어냈다. 옥트리 모델은 알고리즘의 간편함으로 인하여 많이 사용된다. 하지만 단순히 3차원 영역에서 물체의 위치를 가지고 공간을 구분해서 표현하므로 물

체의 형태가 급격히 변하거나 물체 내에 구멍이 있을 때에는 기존의 옥트리 구조만으로는 현실감 있게 모델링하는데 한계가 있다. 이런 문제를 해결하기 위한 방안으로는 옥트리의 계층적 구조를 증가시키는 방법이 있다. 하지만 이는 메모리 비용과 시간적 비용의 증가를 가져온다. 그러므로 전체적으로 낮은 비용으로 옥트리 모델을 제작하고 부분적으로 현실감이 떨어지는 부분에 대해서만 사용자 입력으로 받아드려서 기존의 옥트리 모델과 합성하는 방법을 제안하였다.

향후에는 지금의 PC로부터 객체를 만들고 이를 실험하는 것이 아니라 본래의 알고리즘 흐름에 따라 영상으로부터 그림자를 받아드려 이를 옥트리 모델로 만들고, 본 논문에서 제안한 향상된 옥트리 모델링 기법을 적용할 필요가 있겠다. 또한 기존의 옥트리 모델과 실세계의 3차원 객체를 비교하여 사용자가 수작업으로 객체를 만들어 주는 것이 아니라 자동으로 해결해 주는 방안이 필요하겠다.

참고문헌

- [1] Richard Szeliski, "From Images to Models (and Beyond) : A Personal Retrospective", SIGGRAPH 99 Course 39, August 10, 1998
- [2] P. Srivasan, "Computational Geometric Methods in Volumetric Intersections for 3D Reconstruction", Computer Vision and Pattern Recognition, 23(8):843-857, 1990
- [3] I. Carlbom, "A Hierarchical Data Structure for Representing the Spatial Decomposition", Computer Vision, Graphics, and Image Processing, 5(4):24-31, 1985
- [4] Michael Potmesil, "Generating Octree Models of 3D Objects from Their Silhouettes in a Sequence of Images", Computer Vision, Graphics and Image Processing 40, 1-29(1987)
- [5] Irene Gargantini, "SGLib : An Octree-Based Library Report No. 452", Department of Computer Science, UWO, London, 1995
- [6] Richard Szeliski, "Real-Time Octree Generation from Rotating Objects", Digital Equipment Corporation Cambridge Research Lab, 1990
- [7] Alan Watt, 3D Computer Graphics 2nd Edition, Addison-Wesley, 1993
- [8] Donald Hearn, Computer Graphics C Version 2nd edition, Prentice-Hall, Inc., 1997
- [9] JaMes D. Foley, Introduction to Computer Graphics, Addison-Wesley, 1994
- [10] Keith Rule, 3D Graphics File Formats : A Programmer's Reference, Addison-Wesley, 1996