

Superquadric과 z-buffer 알고리즘을 이용한 3차원 물체 표현 기법

김대현*, 현대환*, 이선호*, 최종수**
 **중앙대학교 전자공학과 영상정보연구실
 ** 한국과학재단 제어계측신기술연구소
 vante77@candy.ee.cau.ac.kr

3D object representation method using Superquadric and Z-buffer algorithm

Dae-Hyun Kim*, Dae-Hwan Hyeon*, Sun-Ho Lee*, Jong-Soo Choi**
 *, ** Dept. of Electronic Eng., Chung-Ang Univ.
 ** Engineering Research Center for
 Advanced Control and Instrumentation by KOSEF

요 약

효율적인 물체인식을 위해서는 물체의 형상특징을 직선적으로 기술할 수 있는 체적소 기반 물체 표현 방법이 필요하다. 본 논문에서는 몇 개의 계수를 가지고 3차원 정보를 효과적으로 표현할 수 있는 superquadric을 이용하여 기본적인 3차원 물체를 모델링 한다. 그리고 보다 복잡하고 정교한 물체의 표현을 위해서 변형된 superquadric을 함께 이용한다. 이렇게 만들어진 개개의 3차원 모델에 z-buffer 알고리즘을 적용하여 하나의 완전한 3차원 물체로 표현하는 방법을 제시하고 실험을 통해 그 유용성을 입증하였다.

1. 서 론

일반적으로 인간의 시각시스템은 3차원 체적의 물체를 단순한 육면체, 구, 원추, 원통 등의 기하학적인 기본 체적소(volumetric primitives)들로 쪼개어 보는 특성이 있으며, 이를 RBC(Recognition-By-Components)라 한다^[1]. 이러한 관점에서 3차원 물체 형상의 체적표현에 가장 알맞은 방법 중에 하나가 바로 기하학적 계수표현법인 superquadric 모델 표현법이다.

기존의 3차원 형상을 기술하는 많은 방법들이 면 방향이나 면곡률 등의 국소적인 특징들을 추출^{[2][3]}하는데 비해, superquadric은 직관적으로 물체를 인지하려는 인간의 인지체계와 유사한 특성 때문에 많은 주목을 끌고 있다. 또한 superquadric은 크기(size)인자, 위치(position)인자, 방향(orientation)인자, 변형(deformation)인자 등 몇 개의 계수들만으로 실세계에 존재하는 다양하고 복잡한 형태의 3차원 물체들을 효과적으로 기술하는데 유용하다^{[4][5]}.

본 논문에서는 superquadric과 변형된 superquadric을 이용하여 각각의 3차원 물체를 만든다. 그리고 만들어진 물체에 대해 z-buffer 알고리즘을 이용하여 하나의 물체로 합성하는 방법을 제안한다^[6].

기존의 superquadric과 CSG tree 기법을 이용한 3차원 물체 표현에서는 superquadric의 implicit함수로 정의된 집합 연산을 사용하였다^{[1][2][6]}. 그러나 표현하려는 물체가 복잡해질수록 implicit함수의 수식이 복잡해지는 단점이 있다. 이에 반해 z-buffer 알고리즘을 이용하여 3차원 물체를 합성하는 경우는 물체의 깊이값을 비교하여 두 개 이상의 물체를 합성하기 때문에 implicit함수를 이용하는 경우보다 물체의 표현 수식이 간단

하다^{[4][5][9][10]}. 또한 superquadric이 갖는 대칭성에 대한 문제점도 z-buffer 알고리즘을 이용하여 여러 개의 물체를 변형, 합성하여 표현함으로써 비대칭적인 물체의 표현도 가능하다.

II. Superquadric을 이용한 3차원 물체의 표현

2.1 Superquadric의 정의

Superquadric 표면은 다음의 3차원 벡터로 정의된다^{[1][4][10][16]}.

$$X(\eta, \omega) = \begin{bmatrix} a_1 \cos^{\epsilon_1}(\eta) \cos^{\epsilon_2}(\omega) \\ a_2 \cos^{\epsilon_1}(\eta) \sin^{\epsilon_2}(\omega) \\ a_3 \sin^{\epsilon_1}(\eta) \end{bmatrix} \quad (1)$$

$$-\frac{\pi}{2} \leq \eta \leq \frac{\pi}{2}, \quad -\pi \leq \omega \leq \pi$$

2.2 Superquadric Inside-Outside Function(Implicit Function)

식(1)은 superquadric 표면의 파라미터 방정식이다. $\cos^2(\alpha) + \sin^2(\alpha) = 1$ 을 이용하여 식(1)에서 η, ω 를 제거하면 다음의 implicit 함수를 구할 수 있다.

$$F(x, y, z) = \left[\left[\left(\frac{x}{a_1} \right)^{2/\epsilon_1} + \left(\frac{y}{a_2} \right)^{2/\epsilon_2} \right]^{\frac{\epsilon_1}{\epsilon_2}} + \left(\frac{z}{a_3} \right)^{2/\epsilon_1} \right]^{\epsilon_1} \quad (2)$$

2.3 Superquadric의 변형

2.3.1 점감(Tapering)

점감은 일정한 기울기를 가지고 중심축(z축)과 직각인 두 방향(x, y축)의 좌표를 증가 또는 감소시킨다.

2.3.2 꼬임(Twisting)

꼬임은 꼬임 인자의 함수로부터 계산된 꼬임각도를 이용해, 중심축과 직각인 두 방향의 좌표를 회전시키는 것이다.

2.3.3 굽힘(Bending)

k를 굽힘 곡률이라 할 때, 굽힘각은 굽힘 곡률과 중심축의 함수 $\theta = z \times k$ 에 의해 계산되며, 계산된 결과를 이용해서 각 점을 회전시킨다.

2.3.4 복합 변형

변형은 점감, 꼬임, 굽힘의 순서에 의해 반복적으로 적용할 수 있다^{[11][16]}. 복합 변형된 영상을 그림 1에 보인다.



(a) 점감후 (b) 점감후 (c) 꼬임후 (d) 점감, 꼬임
꼬임 물체 굽힌 물체 굽힌 물체 후 굽힌 물체

그림 1. 복합 변형된 superquadric

III. Superquadric간의 집합 연산

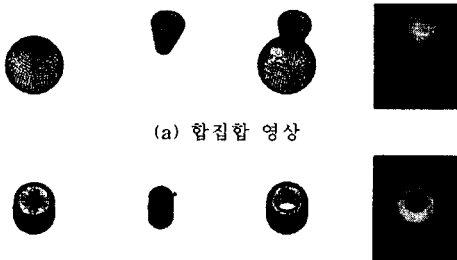
두 superquadric SQ1과 SQ2의 implicit함수를 각각 F1, F2라 하면, 두 입체간의 집합 연산은 다음과 같이 정의된다^{[12][16]}.

$$SQ1 \cup SQ2 = \{(x, y, z) | (F1(x, y, z) = 1 \text{ and } F2(x, y, z) > 1) \text{ or } (F1(x, y, z) > 1 \text{ and } F2(x, y, z) = 1)\} \quad (3)$$

$$SQ1 - SQ2 = \{(x, y, z) | (F1(x, y, z) = 1 \text{ and } F2(x, y, z) > 1) \text{ or } (F1(x, y, z) \leq 1 \text{ and } F2(x, y, z) = 1)\} \quad (4)$$

$$SQ1 \cap SQ2 = \{(x, y, z) | (F1(x, y, z) = 1 \text{ and } F2(x, y, z) \leq 1) \text{ or } (F1(x, y, z) \leq 1 \text{ and } F2(x, y, z) = 1)\} \quad (5)$$

그림 2 (a)는 구와 점감된 실린더의 합집합(union)을 보여주고 있다. (b)는 서로 다른 크기를 갖는 실린더간의 차집합(difference)을 보여준다.



(a) 합집합 영상

(b) 차집합 영상

그림 2. CSG를 이용한 집합 연산 결과와 합성 거리영상

IV. Z-buffer 알고리즘

3차원 물체를 현실감 있게 포착하기 위해서는 투영방향에서 관찰자에게 보이지 않는 모서리와 면들을 제거해야 한다. 어떠한 모서리나 면들을 제거할 것인가를 결정하는 작업은 은선/은면 제거(hidden line/surface removal) 문제라 한다^[17].

Z-buffer 알고리즘은 가장 간단한 은면 제거 알고리즘의 하나로 가장 작은 z값으로 z-buffer를 초기화하고 모든 화소의 위치에서 버퍼에 저장되어 있는 값과 화상의 z값을 비교한다. 만약 새로운 z값이 기존의 값보다 크다면, 이는 조사하고 있는 다각형이 관찰자로부터 더 가까운 것이므로, z-buffer는 이 새로운 값으로 대체된다. 이렇게 연산된 결과로 화상공간(screen space)에서 모든 다각형이 조사되고 나면, z-buffer는 은면 제거 문제에 대한 해를 갖게 된다^{[18][10]}.

Z-buffer 알고리즘은 물체 표면의 각 점에 대한 깊이 정보를 계산할 수 있는 모든 형태에 적용이 가능하다. 뿐만 아니라 개별적으로 표현된 3차원 물체들의 z-buffer 정보와 CSG를 이용한 모델링이 가능하다.

V. 3차원 정보를 이용한 depth shading

5.1 z-level 양자화 (Quantization)

Superquadric을 이용하여 얻은 3차원 깊이값(z)은 물체를 중심으로 대칭인 관계에 있다. 따라서 깊이 정보를 이용해 물체의 표면을 나타내려면 이 값을 256 gray-level로 양자화시켜야 한다. 아래 그림 3에 변환 관계와 관계식 (6)을 보인다.

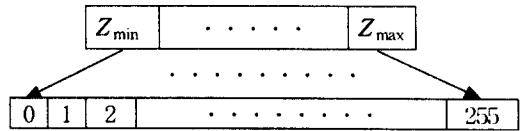


그림 3. 256 gray-level 양자화

$$g = \frac{(Z_{value} - Z_{min}) \times 256}{Z_{max} - Z_{min}} \quad (6)$$

5.2 영상 평면 보간 (Image plane interpolation)

각 꼭지점에서의 깊이값을 256 gray-level로 양자화시킨 후 이들 꼭지점으로 이루어진 다각형 내부 점들의 깊이값을 계산해야 한다. 여기에서는 이차 선형 보간법을 이용하여 깊이값을 보간한다^[10].

VI. 실험 및 고찰

그림 4는 두 개의 서로 다른 물체들을 하나의 물체로 모델링한 결과이다. (a)의 경우는 90°로 회전된 두 개의 실린더가 합쳐진 "해머(hammer)" 영상이다. (b)는 두 물체가 합성된 wireframe을 보여주고 있으며, (c)는 두 물체가 갖는 깊이값을 표면에 표현한 거리 영상(range image)이다. 마찬가지로 (d)는 회전된 실린더와 굽힘을 거친 실린더를 합성하여 만든 "머그컵(mug cup)"에 대한 모델이며, (g)는 실린더와 점감된 실린더가 합쳐진 "압정"의 모델링 결과를 보여준다.

그림 5는 세 개의 서로 다른 물체들의 모델링을 보여준다. (a)의 경우는 두 개의 이동된 구와 회전된 실린더가 합쳐진 "아령" 영상이다. (b)는 세 개의 물체가 합성된 wireframe을 보

여주고 있으며, 세 개의 물체들의 깊이값을 표면에 표현한 결과를 (c)에 보여준다. 마찬가지로 방법으로 (d)는 90° 회전된 두 개의 실린더와 육면체가 혼합된 모델을 나타내고, (g)는 이동된 육면체와 90° 회전된 실린더, 그리고 점감되고 이동된 실린더가 합쳐진 "갓등"의 모델을 나타낸다.

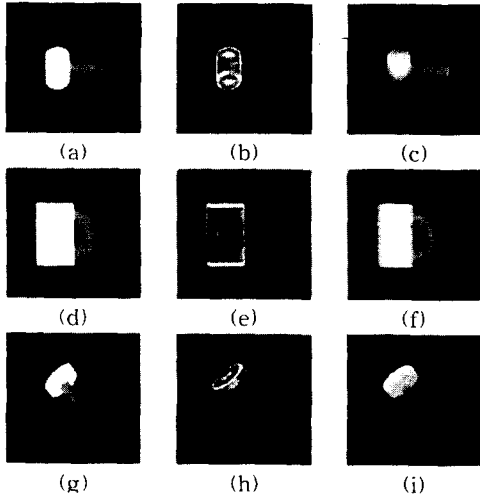


그림 4. 두개의 물체들간의 합성

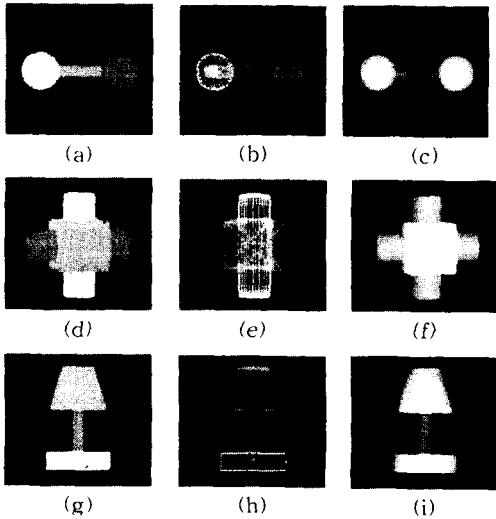


그림 5. 세 개 물체들간의 합성

Ⅶ. 결론

본 논문에서는 superquadric으로 얻어진 3차원 정보와 z-buffer 알고리즘을 이용한 3차원 물체 표현 기법에 대해서 기술하였다. 보의 실험 결과에서 보여 주듯이, 다양한 형태의 3

차원 물체를 superquadric을 이용하여 쉽게 구현할 수 있었으며, 보다 복잡한 물체의 표현에서는 변형된 superquadric을 함께 이용함으로써 쉽게 구현할 수 있었다.

기존의 implicit함수를 이용한 CSG tree 기법에서는 물체의 회전이나 이동, 변형에 대해서 implicit 함수의 수식이 복잡해진다. 따라서 물체의 자유로운 표현이 불가능하였다. 그러나 z-buffer 알고리즘을 이용하면 물체의 회전이나 이동, 변형에 따른 변화에 상관없이 물체의 깊이값만을 비교하여 합성하고 표현하기 때문에 기존의 방법보다 더욱 효율적이다.

본 논문에서 제안하고 있는 3차원 물체 표현 기법은 superquadric으로부터 얻은 기하학적 정보를 물체의 표면에 shading함으로써 합성 거리 영상 (synthetic range image)을 취득할 수 있다. 따라서 실험실 환경에서 고가의 계측장비 없이 원하는 형태의 다양한 3차원 거리데이터를 쉽게 취득하여 실험할 수 있다.

Z-buffer 알고리즘을 이용한 보다 정교한 3차원 물체 표현을 위해서는 CSG 기법에서처럼 집합 연산이 필요하다. 따라서 superquadric과 z-buffer 알고리즘을 이용한 CSG 집합 연산 알고리즘의 개발이 중요한 과제가 되고 있다.

참고 문헌

- [1] 황삼정, 양현승, "Superquadric을 이용한 3차원 물체의 표현과 인식", *한국정보과학회 논문지 '90*, Vol.17, No.5 September.
- [2] S. C. Hwang, H. S. Yang, "3D object Representation using the CSG tree and Superquadrics", *KITE Journal of Electronics Engineering*, Vol.2, No.1 June, 1991.
- [3] Nigel Stewart, Geoff Leach, "An Improved Z-Buffer CSG Rendering Algorithm", 1998 Eurographics/Siggraph Workshop on Graphics Hardware, pp.25-30.
- [4] N. Stewart, G. Leach, S. John, "A Single Z-Buffer CSG Rendering Algorithm for Convex Objects", *Siggraph '99* Submission.
- [5] Frank Solina, Ruzena Bajcsy, "Recovery of Parametric Models from Range Images: The Case for Superquadrics with Global Deformations", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol.12, No.2, Feb., 1990.
- [6] 김대현, 현대환, 이선호, 최종수, "Superquadric과 CSG에 기반한 3차원 물체 형상 모델링", 1999년도 하계 통신학회 학술대회 논문, Vol. 19, No.1 pp.521-524.
- [7] 송호근, 최종수, 미분 기하학 특징 정보를 이용한 3차원 거리 영상 분할, *대한 전자공학회 논문지*, 제31권, B편, 제 2호, pp. 123-134, 1995.
- [8] 박종훈, 최종수, 면 법선 영상 기반형 3차원 물체 인식에서의 새로운 매칭 기법, *대한 전자공학회 논문지*, 제30권, B편, 제 7호, pp. 695-705, 1993.
- [9] Vera B. Anand, "Computer Graphics and Geometric Modeling for Engineers", WILEY.
- [10] A. Watt, F. Policarpo, "The Computer Image", ADDISON-WESLEY.
- [11] I. Biederman, Human image understanding: recent research and theory," *Comput. Vision. Graphics. Image Processing.*, vol. 32, pp. 29-73, 1985.