

# 곡면의 연속 특성을 이용한 타원체 음함수 표면의 폴리곤화

박태정<sup>1\*</sup> 이해영<sup>2</sup> 박영준<sup>1</sup> 민홍식<sup>1</sup>

<sup>1</sup>서울대학교 전기컴퓨터 공학부

<sup>2</sup>홍익대학교 컴퓨터 공학과

(tjpark, ypark, hsmin)@amity.snu.ac.kr

lech@cs.hongik.ac.kr

## Polygonization of Ellipsoidal Implicit Surfaces Using Continuity of Surface

Taejung Park<sup>1\*</sup> Haeyoung Lee<sup>2</sup> Young-June Park<sup>1</sup> Hong-Shick Min<sup>1</sup>

<sup>1</sup>EECS/EE, Seoul National University

<sup>2</sup>Department of Computer Engineering, HongIk University

### 요 약

음함수 표면을 이용한 모델링 방법은 부드러운 곡면을 나타내기 위해 적합하며 날카로운 부분을 표현하기 위해서는 CSG 연산을 적용한다. 기존의 방식을 이용해서 얻은 메쉬에서는 흔히 음함수의 표면과 상당한 오차를 가지는 메쉬 꼭짓점(vertex)을 얻거나 겹쳐지는 삼각형 또는 날카로운 부분의 표현이 안 되는 점 등의 문제가 나타난다. 본 논문에서는 타원체의 특성을 이용해서 타원체 기반 음함수 표면을 정확하게 샘플링하고 동시에 날카로운 부분을 효과적으로 표현할 수 있는 메쉬를 얻기 위한 폴리곤화 방법을 제안한다. 이러한 목표를 위해 타원체의 두사 특성과 표면 법선 벡터의 연속 특성을 이용해서 음함수 표면 위에 정확하게 위치하는 꼭짓점(vertex)의 위치를 찾고 날카로운 부분을 효과적으로 표현하기 위해 점진적인 방법으로 정확한 꼭짓점(vertex) 위치를 찾는 방법을 제안한다. 지금까지 약점으로 지적되어 왔던 음함수 표면 모델링의 시각화 절차를 이 방법을 통해 개선함으로써 음함수 표면 모델링 기법이 제공하는 다른 장점들을 적극 활용할 수 있을 것으로 기대한다.

### 1. 서론

음함수 표면(implicit surface)을 이용한 모델링 방식은 메쉬(mesh) 모델에 비해 부피 데이터와 부드러운 곡면을 표현하기에 적합하다[1]. 또한 CSG(Constructive Solid Geometry) 방법을 사용하기도 용이하다.

그러나 음함수 표면을 현재 널리 사용되는 하드웨어 그래픽 파이프라인에서 표현하기 위해서는 반드시 폴리곤화 과정을 거쳐야 한다. 이에 따라 음함수 표면의 시각화를 위한 여러 가지 방법들이 제안된 바 있다 [2,3,4,5,6]. 이러한 방법들 중 공간 분할 격자를 이용한 방법들이 널리 사용되고 있다. 이러한 방식들 중 듀얼 공간 분할 격자 방식은 격자 모서리선에서 직접 얻은 샘플을 이용해서 격자 공간 내에서 꼭짓점(vertex)의 위치를 결정함으로써 날카로운 특성을 표현하는 부분들(모서리, 뾰족한 부분 등)을 포착할 확률을 증가시켰다. 그러나 듀얼 공간 분할 격자 방식들 역시 주로 CSG 연산을 통해 생성되는 날카로운 부분을 올바르게 표현하

지 못하는 경우가 빈번하게 발생한다. 본 논문에서는 이러한 문제의 원인을 고찰하고 점진적인 루틴을 통해 이 문제를 해결함으로써 기존의 연구[11]에서와 달리 최종 메쉬에서 별다른 후속 처리(예, Subdivision)없이도 날카로운 특성을 보다 확실하게 얻을 수 있는 방법을 제안한다.

듀얼 공간 분할 격자 방식에서의 또 다른 문제점은, 기존에 제안된 방식들에서 결정하는 분할 격자 내의 꼭짓점(vertex)들이 음함수 표면을 정확하게 표현하지 못하고 있다는 점이다. 기존 연구 결과에서 격자 공간 내 꼭짓점(vertex)의 좌표를 결정하는 방식을 고찰해 보면, 격자의 중심점(centroid)으로 지정[7]하는 방식이나 2차 에러 함수(QEF; Quadratic Error Function)와 평균점을 혼합하는 방식[8, 11]을 주로 사용하고 있음을 발견할 수 있다.

격자의 중심점을 적용하는 방식은 가장 간단한 방식으로, 빠른 속도와 함께 격자 공간 내에 꼭짓점(vertex)이

정확하게 위치한다는 장점을 가지고 있으나 매끄럽게 표현되어야 할 음함수 표면이 울퉁불퉁한 계단처럼 보이는 단점이 있다(그림 1의 중앙).

QEF를 사용해서 첨점(vertex)의 위치를 결정하는 방식들은 결국, 격자 모서리선에서 샘플링한 좌표와 법선 벡터로 결정되는 각각의 평면들의 교점을 찾는 방법이라고 할 수 있다. 따라서 이 방법은 항상 유한한 해가 존재하지 않으며 해가 존재하더라도 해당 격자를 벗어나는 범위에서 해가 결정되는 경우가 흔히 발생한다. 이에 비해 이 논문에서 제안하는 방법은 타원체 음함수 표면의 특성을 이용해서 정확하게 해당 격자 내에서 음함수 표면 위에 위치하는 첨점(vertex)을 찾을 수 있다. 그림 1에서 볼 수 있듯이 본 논문에서 제안하는 방식(오른쪽)은 정확하게 음함수 표면 위에 첨점(vertex)이 위치하기 때문에 음함수 표면 모델링 방식의 가장 큰 특징인 부드러운 곡면 특성이 매쉬에서도 잘 나타난다. 또한 이러한 표면 위의 위치를 찾기 위해 기존에 주로 사용되었던 반복적인 특성의 이분할(bisection) 방식을 개선해서 직접 표면 위 위치를 찾는 방식도 제안한다.

**본 논문에서 새로 제안하는 내용**

- 듀얼 공간 분할 격자 폴리곤화 방식에서 각 격자 내에 포함되며 타원체 위에 정확하게 위치하는 매쉬 첨점(vertex)을 계산
- 음함수 표면의 날카로운 부분을 충실하게 표현
- 타원체 음함수 단위 상의 위치를 반복적인 방법이 아닌, 직접 바로 얻을 수 있는 타원체 투사 특성 제시

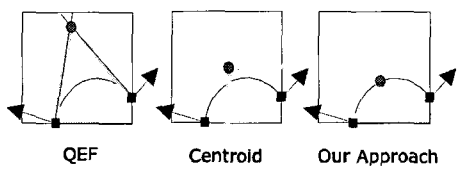


그림 1- 격자 공간 내 첨점(vertex) 결정 방식

빨간색 점은 격자 공간 내에서 결정된 첨점(vertex)의 위치. 검은색 사각형은 격자 모서리선과 음함수 표면(회색)이 만난 교점의 위치. 파란색 화살표는 이 교점에서의 법선 벡터. 녹색선은 생성되는 매쉬의 모서리선(edge).

**1.1 기존 연구**

**폴리곤화(polygonization)와 날카로운 부분(sharp-featuring)**

앞서 언급한 바와 같이, 음함수 표면의 폴리곤화를 위해 가장 널리 사용되는 방식은 공간 분할 격자 방식이다. 공간 분할 격자를 이용한 방법 중 대표적인 방식인 Marching Cube(MC)[6] 방법은 직관적이고 간단하다는 장점이 있으나 음함수 표면의 샘플링이 격자 모서리선(edge)을 따라서만 이루어 지기 때문에 음함수 표면의 뾰족하거나 각진 형태를 샘플링하지 못하는 단점이 지목된 바 있다. 이러한 문제의 해결을 위해 Marching Cube가 생성하는 매쉬의 첨점(vertex) 및 모서리선(edge)과는 각각 듀얼(dual)의 관계를 가지는 듀얼 공간 분할 격자 방식들[7, 8, 9]이 제안되었다. MC가 1차원 공간인 분할 격자의 각 모서리선에서 음함수 표면과의 교점을 직접 첨점(vertex)의 위치와 법선 벡터로 활용하는 것에 비해 듀얼 방식에서는 이렇게 격자의 모서리선에서 얻은 데이터를 바탕으로 격자 내 3차원 공간에서 첨점(vertex)의 위치를 결정하기 때문에 보다 날카로운 특징들을 잘 포착할 수 있다고 알려져 왔다. 그러나 지금까지, 이렇게 결정된 첨점(vertex)의 위치는 음함수 표면을 직접 샘플링한 데이터가 아니라 주변 격자 모서리선과 음함수 표면의 교점을 이용해서 추정된 위치이었기 때문에 최종적으로 얻은 매쉬가 만족스럽지 못한 경우가 빈번했다. 분할 격자 내에서 음함수 표면을 보다 충실하게 표현하는 문제는 최근 관련 연구 논문[7]에서 향후 보다 많은 연구가 필요한 분야라는 점이 언급되기도 했다.

가장 널리 사용되는 듀얼 공간 분할 방식 중 하나인 Dual Contouring 방식[8, 10, 11]에서는 격자 공간 내에서 첨점(vertex)의 위치를 결정하기 위해서 한 격자 내에서 다음과 같은 2차 에러 함수(QEF)를 이용한다.

$$E(\mathbf{x}) = \sum_i (n_i \cdot (\mathbf{x} - p_i))^2 \quad (1)$$

이 때  $p_i$ 는 이 격자의 모서리선과 음함수 표면이 만난 교점의 위치이며  $n_i$ 는 그 교점에서의 법선 벡터이다. 따라서  $n_i \cdot (\mathbf{x} - p_i) = 0$  을 만족하는 점  $\mathbf{x}$ 의 자취는 점  $p_i$ 를 지나며 법선 벡터  $n_i$ 를 가지는 평면이며 따라서 식 (1)을 최소화하는 위치  $\mathbf{x}$ 는 각 평면의 교점임을 알

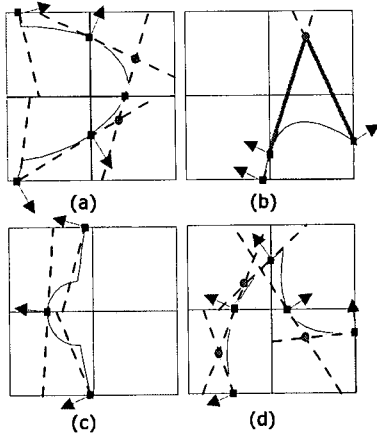


그림 2-QEF 최소화 방식의 문제점

점선은 격자 모서리선과 음함수 표면과의 교점에서 결정되는 평면. 녹색선은 올바르게 결정된 매쉬의 모서리선(edge). 빨간 색 선은 범위를 벗어나서 결정된 매쉬의 모서리선.

수 있다.

그러나 그림 2에서 제시한 대로 이 방법은 항상 해의 존재를 보장할 수 없으며(그림 2 (a), (c), (d)), 경우에 따라 올바른 격자의 범위를 벗어나서 첨점(vertex)의 위치가 결정되기도 한다(그림 2 (b)). 특히 그림 2 (b)의 경우 최종 매쉬에서 다른 삼각형 위로 겹쳐지는 삼각형(folding triangle)으로 나타나기도 한다. [10]에서는 이러한 문제의 해결을 위해 평면의 교점을 구할 수 없을 때 각 격자 모서리선에서의 교점의 위치를 평균해서 첨점(vertex)의 위치를 구하는 방법을 제시하고 있다. 그러나 이 방법 역시 오차가 있으며 더군다나 CSG 연산으로 인한 날카로운 부분이 포함된 경우(그림 2 (d)), 음함수 표면을 충실하게 표현할 수 없다. 이에 비해 본 논문에서 제안하는 방법은 (가) 한 격자 내에 단일 타원체 음함수 표면 단위(primitive)가 포함되어 있을 경우 첨점(vertex)의 위치가 격자 내에 포함되며 음함수 표면 위에 정확하게 위치하도록 보장하고 (나) 두 개 이상의 타원체 음함수 표면 단위가 CSG 연산으로 포함되어 있을 때에는 이 격자를 날카로운 부분이 포함된 것으로 판단하고 별도의 처리를 수행함으로써 정확한 음함수 표면의 표현을 보장한다.

[11]에서도 이러한 날카로운 부분의 충실한 표현을 위한 방법을 제안하고 있다. 그러나 이 연구는 본 논문에서 제안하는 방법과는 달리 별도의 추가 루틴(subdivision)이 후속 처리를 위해 필요하다.

## 타원체 표면 투사 특성

타원체는 적은 데이터로도 구현이 가능하며 CSG 연산 등으로 다양한 표현이 가능하다는 장점을 가진다. 특히 타원체는 허용 가능한 오차 범위 내에서 CSG 연산을 통해 직육면체 모델이나 원통 모델 등의 표현이 가능하다. 타원체를 음함수 표면의 기본 구성 단위(primitive)로 사용할 경우 폴리곤화나 기타 실시간 충돌 검사 등의 애플리케이션에서 타원체 위의 정확한 좌표를 구하는 작업이 필수적이다. 그러나 음함수 표면의 기본 구성 단위로서 타원체는 음함수로 표현되기 때문에 직접적인 방정식 해법을 적용하는데 무리가 있다. 기존 연구[2,3,12]에서는 타원체 표면의 정확한 위치를 얻기 위해서 이분할(bisection) 방법을 적용했다. 그러나 이 방식은 반복적(iterative) 방식이기 때문에 상대적으로 시간이 많이 걸리는 단점이 있다. 본 논문에서는 이러한 단점을 개선하기 위해 타원체의 투사 특성을 정의하고 간단한 연산으로 바로 타원체 표면 위의 위치를 얻는 방식을 제안한다.

## 1.2 알고리즘 개요

전체 알고리즘에 대한 설명은 그림 3에서 제시되었다. 그림 3에서 볼 수 있듯이 이 알고리즘은 CSG 연산으로 결합된 음함수 타원체들의 데이터를 받아서 복셀(voxel)화한 다음 각 격자 내 포함된 타원체 음함수 표면 단위의 개수를 파악한다. 각 격자에서 이 개수가 1 개 일 경우 정확한 표면 위의 점 계산을 위한 처리로 넘어가고 2개 이상일 경우 CSG 연산으로 결합된 것으로 파악해서 날카로운 부분 처리를 위한 루틴을 실행한다. 특히 타원체 1개만이 포함된 경우에는 먼저 타원체 법선 벡터의 연속 특성을 이용해서 격자 내에 반드시 포함되는 법선 벡터를 구한 다음, 타원체 투사 특성을 이용해서 이 법선 벡터에 해당하는 유일한 위치를 타원체 표면 위에서 구한다.

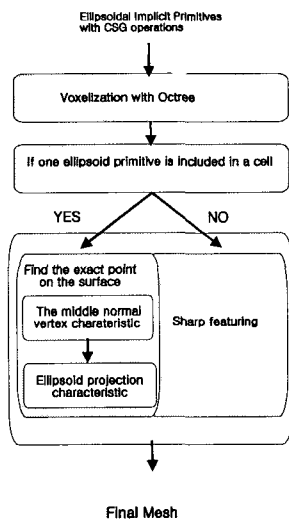


그림 3- 개요

입력은 CSG 연산으로 결합된 타원체 데이터로 주어지며 최종 결과로 날카로운 부분의 처리(sharp-featuring)가 끝난 메쉬를 얻는다. 또한 한 격자 공간 내 포함되는 타원체 단위(primitive) 개수에 따라 날카로운 부분 처리 여부를 결정한다.

## 2. 폴리곤화 방법

앞서 언급했듯이, 듀얼 폴리곤화 방식에서의 가장 중요한 문제 중 하나는 해당 격자 공간 내에 포함되며 음함수 표면을 가장 근접하게 대표할 수 있는 정점(vertex)을 결정하는 문제이다. 이 논문에서 제안하는 새로운 폴리곤화 방법은 타원체의 투사(projection) 특성과 타원체 표면 위의 법선 벡터(normal vector)의 연속성 특성을 이용한다. 또한 제안하는 방식은 정확하게 음함수 표면 위에 위치하며 해당 격자 공간 내에 포함되는 정점을 찾을 수 있다는 장점을 제공한다.

### 2.1 타원체의 투사 특성

타원체는 임의의 구(sphere)를 각 축의 방향에 따라 배율(scale)을 지정함으로써 얻을 수 있다.<sup>1</sup> 이 때, 같은 중심을 공유하며 각 축의 비가 동일한(isomorphic) 임의의 두 개의 타원체를 각각  $f_0$ 와  $f_{col}$ 이라고 하면,

<sup>1</sup> 일반적인 경우, 이동(translation) 및 회전(rotation) 변환이 함께 고려되어야 하지만, 이곳에서의 논의를 위해서는 배율만 지정한 경우도 충분히 일반성을 확보할 수 있기 때문에 원점에 중심이 위치하는 경우만 고려한다.

$$f_0(x, y, z) = \left(\frac{x}{a_1}\right)^2 + \left(\frac{y}{b_1}\right)^2 + \left(\frac{z}{c_1}\right)^2 - 1 \quad (2)$$

$$f_{col}(x, y, z) = \left(\frac{x}{a_2}\right)^2 + \left(\frac{y}{b_2}\right)^2 + \left(\frac{z}{c_2}\right)^2 - 1 \quad (3)$$

으로 나타낼 수 있고 두 타원체 위의 점 A, B에 대해서 간단한 연산을 통해 다음과 같은 관계가 성립함을 유도할 수 있다.

$$\vec{OB} = K \vec{OA} = \vec{OA} \sqrt{1 + f_0(\vec{OA})} \quad (4)$$

( $\vec{OA}$ 와  $\vec{OB}$ 는 각각 원점에서 점 A, B로 향하는 벡터)

이 투사 특성은 특히 입자 시스템(particle system)과 같은 실시간 적용 분야에서 충돌 처리(collision handling)에서 유용하게 사용할 수 있다. 충돌 처리는 일반적으로 충돌 검사(collision detection)와 충돌 복구(collision resolution) 두 루틴으로 나눌 수 있는데, 그림 4에서 타원체  $f_0$ 와 한 입자가 충돌한 상태(점 A)를 충돌 상태로 인식하는 단계가 충돌 검사 단계라고 하면 다시 이 입자를 표면 위의 점 B로 이동시키는 것을 충돌 복구 단계라고 할 수 있다. 또한 식 (4)는 그림 4에서와 같이 타원체  $f_0$ 의 내부에 있는 점 뿐만 아니라  $f_0$ 의 외부에 위치하는 점을 타원체 중심에 대한 투사를 이용해서 타원체 표면 위의 점을 구하는 경우에도 이용할 수 있다.

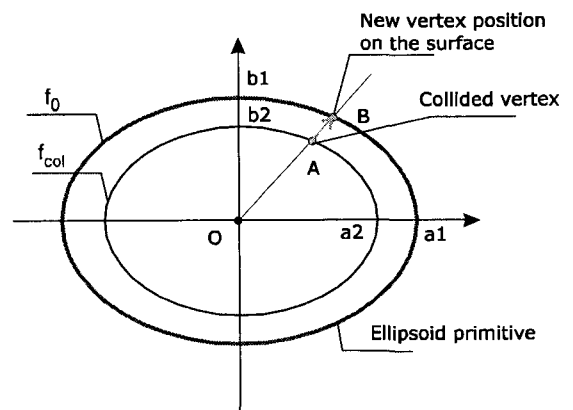


그림 4- 타원체의 투사 특성

빨간색 선으로 표시된 부분은 실제 타원체 단위, 파란색 선으로 표시된 부분은 입자가 충돌 위치 A에 있을 때 이 점을 지나며 실제 타원체와 닮은 꼴로 중심을 공유하는 가상 타원체. 위치 B는 위치 A를 표면 위로 투사한 결과

## 2.2 타원체 표면 법선 벡터 연속성 특성

일반적으로 임의의 타원체  $f$  위의 임의의 점에서 정의되는 단위 법선 벡터를 원소로 가지는 집합은  $\mathbf{R}^3$  공간에서 정의되는 모든 단위 벡터와 일대일 대응 관계를 가지며 따라서 두 집합은 동치이다.

타원체  $f$  위의 임의의 점  $\mathbf{x}$ 에서의 단위 법선 벡터

$\hat{h}(\mathbf{x})$ 를 다음과 같이 정의하면,

$$\hat{h}(\mathbf{x}) = \nabla f(\mathbf{x}) / |\nabla f(\mathbf{x})| \quad (5)$$

이 때  $\mathbf{R}^3$  공간에서 정의되는 모든 단위 벡터의 집합을  $\mathbf{V}$ 라고 하고 집합  $\mathbf{V}$ 의 서로 다른 두 임의의 원소를  $\mathbf{n}_1, \mathbf{n}_2$ 라고 할 때 임의의 타원체  $f$ 에 대해

$$\mathbf{n}_1 = \hat{h}(\mathbf{p}_1) \quad (6)$$

$$\mathbf{n}_2 = \hat{h}(\mathbf{p}_2) \quad (7)$$

라고 하면 다음 관계가 성립한다.

$$\mathbf{n}_1 \neq \mathbf{n}_2 \Leftrightarrow \mathbf{p}_1 \neq \mathbf{p}_2 \quad (8)$$

$$\mathbf{n}_1 = \mathbf{n}_2 \Leftrightarrow \mathbf{p}_1 = \mathbf{p}_2 \quad (9)$$

따라서 임의의 단위 벡터  $\mathbf{n}$ 은 타원체  $f$  표면 위의 점  $\mathbf{p}$ 와 고유하게 일대일 대응된다.

이러한 특성과 함수  $\hat{h}(\mathbf{x})$ 의 타원체  $f$  위에서의 연속성을 이용하면 공간을 분할하는 3차원 격자(직육면체)들 내에 위치한 음함수 타원체 표면에 위치하며 해당 격자 내에 정확하게 포함되는 점을 찾을 수 있다.

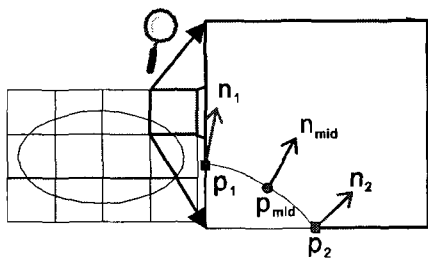


그림 5- 정확한 표면 위치 결정

검은색 사각점 - 샘플링된 음함수 표면의 점접 위치. 빨간색 원 - 최종 결정된 메쉬의 첨점(vertex). 빨간색 화살표 - 접점에서의 법선 벡터, 가운데 파란색 화살표 - 접점에서의 법선 벡터의 평균 벡터

그림 5는 설명을 위한 2차원 도식도이다. 이 그림은 2차원 예이지만 3차원 공간에도 동일한 방식으로 확장 가능하다. 그림에서 회색으로 표시된 영역은 음함수 타원이며 격자는 듀얼 폴리곤화를 위한 격자이다.

일반적으로 음함수 표면 애플리케이션에서 입력 데이터는 각 격자의 각 변과 음함수 표면이 만나는 (a) 지점의 좌표와 (b) 그 지점에서의 법선 벡터(normal vector)로 주어진다(Hermite data). 즉, 그림 5의 확대된 격자 공간에서 이 데이터는 각각 (a)  $\mathbf{p}_1, \mathbf{p}_2$  및 (b)  $\mathbf{n}_1, \mathbf{n}_2$ 이다.

이 때 회색으로 표시된 타원 음함수를  $f$ 라고 하면  $f(\mathbf{x})=0$ 을 만족하는 자취를 따라서 식(4)로 주어지는 법선 벡터  $\hat{h}(\mathbf{x})$ 는 모든 구간에서 연속이기 때문에 중간 값 정리에 의해 그림 5의 확대된 영역에서 벡터  $\mathbf{n}_1$ 과  $\mathbf{n}_2$ 가 존재하는 두 점  $\mathbf{p}_1$ 과  $\mathbf{p}_2$  사이의 자취 위에 반드시  $\mathbf{n}_{mid} = (\mathbf{n}_1 + \mathbf{n}_2)/2$ 를 만족하는  $\mathbf{n}_{mid}$ 가 존재 하며 식 (8), (9)에 의해  $\mathbf{p}_{mid}$ 가 두 점  $\mathbf{p}_1$ 과  $\mathbf{p}_2$  사이에 존재하게 된다. 따라서 정점  $\mathbf{p}_{mid}$ 는 반드시 공간 분할 격자 내에 위치하게 되므로 최종적으로 얻게 되는 매쉬(mesh)에서는 QEF를 사용하는 Dual Contouring 방식에서 흔히 나타나는 겹쳐진 삼각형 현상이 나타나지 않는다. 동시에 이렇게 얻은 정점은 반드시 음함수 표면 위에 정확히 위치하기 때문에 다른 듀얼 폴리곤화 방식에서 사용하는 정점 위치 결정 방식에 비해 보다 정확하게 음함수를 폴리곤화 해서 표현할 수 있다.

특수한 경우로,  $\mathbf{n}_1 = \mathbf{n}_2$ 인 경우를 생각해 볼 수 있다. 이 경우는 타원체 음함수 표면 단위가 허용 가능한 오차 범위 내에서 평면으로 근사되는 경우에 해당한다. 이 경우에는 단순히 중심점(centroid)을 잡은 후 타원체 투사 특성을 이용하거나 분할 격자 모서리선과 음함수 표면 사이의 각 교점 좌표들의 평균으로 첨점(vertex) 좌표를 설정함으로써 정확한 위치를 설정할 수 있다.

## 2.3 날카로운 부분의 표현

일반적으로 음함수 표면은 부드러운 곡면을 나타내며 CSG 방법을 적용함으로써 날카로운 표면을 얻게 된다. 이러한 CSG 연산으로는 논리합(OR), 논리곱(AND), 논리빼셈(SUBTRACT) 등이 있다. 따라서 한 격자 내에 2개 이상의 음함수 표면 단위가 포함되면 이 부분에 모델의 날카로운 특징이 포함된다고 판단할 수 있다. 그림 6에서 위쪽 그림에서는 날카로운 부분의 처리를 위한 루틴을 도식적으로 표현하고 있다. 이 때 두 음함수 표면을 각각  $f_1, f_2$ 라고 하고 이 모델은 CSG 연산

에서  $\text{SUBTRACT}(f_2, f_1)$ , 즉  $f_2 - f_1$ 의 결과라고 가정한다. 일반적으로 날카로운 부분을 정확하게 표현하기 위해서는  $f_2 = f_1 = 0$ 을 만족하는 해를 찾아서 그 위치에 첨점(vertex)를 위치시켜야 한다.

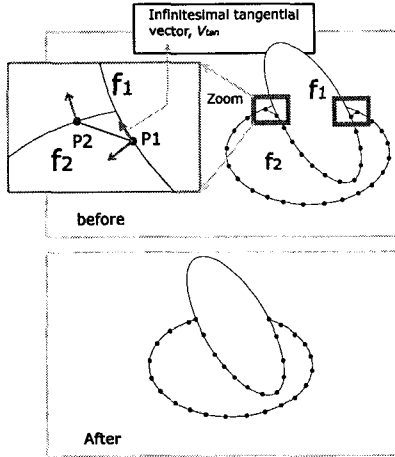


그림 6 - 날카로운 부분의 처리

Before : 파란색 화살표 - 각 타원체 단위(primitive) 위의 점 P1, P2에서의 법선 벡터. 빨간색 화살표 - 매우 짧은 길이를 가지는 P1에서의 접선 벡터. 오른쪽의 두 빨간색 삼각형은 날카로운 부분이 처리되지 않은 상황

After : 이 부분의 처리가 끝난 후의 최종 매쉬

그림 6에서 위쪽 부분은 날카로운 부분의 처리를 고려하지 않았을 때 폴리곤화의 결과이다. 이러한 결과는 공간 분할 격자의 위치에 따라  $f_2 = f_1 = 0$ 을 만족하는 교점이 반드시 격자 내에 포함되지 않을 수도 있기 때문에 발생한다. 이 결과를 살펴 보면, 위쪽에 있는 두 개의 빨간색 사각형 내부에 날카롭게 표현되어야 할 부분이 뭉뚱하게 표현된 상황을 살펴 볼 수 있다. 따라서 이렇게 서로 다른 두 개의 음함수 단위(여기에서는  $f_1, f_2$ )에 각각의 첨점(vertex)이 닿아 있는 매쉬의 모서리선(edge)을 적절하게 처리해 주어야 할 필요가 있다. 제안하는 방법에서는 이 모서리선의 임의의 한쪽 첨점(vertex)(그림에서는 P1)을 점진적으로 이동시켜서 그림 6의 아래쪽 결과와 같은 매쉬를 얻는다. 이 때 주목할 만한 점은 일반적으로 직육면체의 한 모서리와 같은 부분은 두 개의 타원체 단위들이 AND 연산으로 생성되기 때문에 이 때  $f_1 = f_2 = 0$ 을 만족하는 방정식의 해는 공간상에서 부정(직선에 가까운 곡선 상의 점들)으로 나타난다. 따라서 직접 방정식 해법(direct solver)으로는 이러한 점을 구하는 것이 무리가 있으며 따라

서 이 논문에서는 점진적인 방정식 해법을 제안한다. 이 루틴의 의사 코드(pseudo code)는 다음과 같다.

```

loop()
{
  while(f2(p1)<0)
  {
    Move incrementally P1 at the
    tangential direction toward
    the surface f2 = 0
  }
}

```

이 과정 중에 첨점(vertex)들 사이의 균형을 위해서 매쉬의 첨점(vertex)을 질점(mass point)으로, 모서리선(edge)을 스프링 및 댐퍼(damper)로 모델링한 물리 시뮬레이션 루틴을 적용할 수도 있다. 그림 6에서 점 P1에서 표면에 접하는 매우 짧은 길이를 가지는 벡터,  $\vec{V}_{tan}$  은 다음과 같이 지정한다.

$$\vec{V}_{tan} = \varepsilon(\vec{n}_2 - \vec{n}_1(\vec{n}_1 \cdot \vec{n}_2)), (0 < \varepsilon \ll 1) \quad (10)$$

이 때

$$\vec{n}_1 = \nabla f_1(\mathbf{x}) / |\nabla f_1(\mathbf{x})| \Big|_{\mathbf{x}=P1} \quad (11)$$

$$\vec{n}_2 = \nabla f_2(\mathbf{x}) / |\nabla f_2(\mathbf{x})| \Big|_{\mathbf{x}=P2} \quad (12)$$

### 3. 결과

그림 7은 본 논문에서 제안하는 방법의 결과를 보여주고 있다. 가장 왼쪽 그림은 입력으로 사용된 타원체 음함수 표면 단위(primitive)들의 포텐셜을 표현하고 있다. 중앙의 결과는 타원체 투사 특성과 법선 벡터의 연속 특성을 이용한 결과를 이용해서 매쉬의 첨점(vertex)을 각 타원체 위에 정확하게 위치시킨 그림이다. 마지막으로 가장 오른쪽 그림은 최종적으로 날카로운 부분을 처리(sharp-featuring)가 끝난 매쉬를 보여주고 있다.

이 결과에서 생성된 매쉬의 크기는 표 1과 같다.

표 1 - 생성된 매쉬 데이터

매쉬 이름 (그림 7)	첨점(vertex) 개수	삼각형(face) 개수
Tire	19840	39676
Dice	17176	34368
Part	5920	11848
Cup	16968	33936

각 모델의 폴리곤화에 사용된 시스템의 사양은 Intel Pentium M 1.5GHz, 512MB이며 Microsoft Windows XP 운영 체제에서 DirectX 9.0b를 사용해서 구현되었다. 이때 주요 루틴별 소요 시간은 표 2에서 정리했다. 전체적으로 소요되는 시간은 매쉬의 크기에 비해 그렇게 크지 않다고도 볼 수 있으나, 이 표를 통해 가장 많은 연산 시간을 차지하는 부분은 날카로운 특성을 처리하는 부분임을 알 수 있다. 그 이유는 앞서 설명한 바대로 이 루틴이 점진적으로 방정식의 해를 찾아 가고 있기 때문이다. 특히 식 (10)에서  $\epsilon$ 의 크기를 증가시키면 이 루틴의 처리 시간이 줄어 드는 것을 확인할 수 있었다. 그러나 지나치게 큰  $\epsilon$ 을 적용할 경우 날카로운 모서리가 약간 시각적으로 불균일하게 나타나는 현상이 발생했다.

표 2 - 실행 시간

매쉬 이름 (그림 7)	격자 내 첨점 계산(초)	투사 시간 (초)	날카로운 특 성 처리(초)
Tire	0.922212	0.001292	2.861121
Dice	0.763765	0.001137	3.145645
Part	0.240006	0.000417	0.974836
Cup	0.785904	0.001086	2.021166

#### 4. 결론 및 향후 연구 과제

본 논문에서는 음함수 표면 모델의 폴리곤화 (polygonization) 방법으로 널리 쓰이는 기존의 듀얼 공간 분할 격자 방식에서의 문제점을 해결하기 위해 타원체 음함수 단위의 투사 특성과 법선 벡터의 연속성을 이용함으로써 격자 내에서 정확하게 표면 위에 위치하며 동시에 날카로운 부분의 처리(sharp-featuring)할 수 있는 방법을 제안한다. 이 방법은 지금까지 널리 사용되어 왔던 QEF를 이용한 격자 공간 내에서의 첨

점(vertex) 결정 방식에서의 겹침 문제, 음함수 표면에 대한 위치 오차 발생 등의 문제를 해결했다. 또한 기존 연구와는 달리 음함수 타원체에서 투사 특성을 이용함으로써 반복적으로 수행되는 이분할(bisection) 방식보다 신속한 성능을 보장한다.

또한 음함수 표면 단위(primitive)로 타원체 표면과 함께 널리 사용되는 골격 기반(skeletal based) 음함수 표면 단위에까지 이 방식을 확장 가능할 것으로 기대하고 있으며 향후 연구로서 현재 날카로운 부분 처리 (sharp-featuring) 과정을 보다 최적화해서 빠른 속도로 처리하기 위한 연구를 수행하고 있다.

마지막으로 음함수 표면의 가장 큰 특징 중 하나인 블렌딩(blending)을 매쉬와 음함수 표면이 혼합된 모델링 방식에서 구현하기 위한 연구도 함께 진행 중이다.

#### 감사의 글

본 연구는 과학기술부 과학재단 목적기초연구(R01-2005-000-10120-0) 지원으로 수행되었다.

#### 참고문헌

- [1] James F. Blinn. A generalization of algebraic surface drawing. *ACM Transactions on Graphics*, 1(3):235-356, 1982. ACM Transactions on Graphics.
- [2] Mathieu Desbrun, Nicolas Tsingos, and MariePaule Cani. Adaptive sampling of implicit surfaces for interactive modeling and animation. In *Implicit Surfaces '95*, 1995. Published under the name MariePaule Gascuel.
- [3] Jules Bloomenthal and Keith Ferguson. Polygonization of non-manifold implicit surfaces. *Computer Aided Geometric Design*, 5(4):341-355, 1988. Computer Aided Geometric Design.
- [4] Andrew P. Witkin and Paul S. Heckbert. Using particles to sample and control implicit surfaces. In *Proceedings of SIGGRAPH 94*, pages 269-277, 1994.
- [5] Barton T. Stander and John C. Hart. Guaranteeing the topology of an implicit surface polygonization for interactive modeling. In *SIGGRAPH '97: Proceedings of the 24th annual conference on*

- Computer graphics and interactive techniques*, pages 279–286, New York, NY, USA, 1997. ACM Press/AddisonWesley Publishing Co.
- [6] William E. Lorensen and Harvey E. Cline. Marching cubes: A high resolution 3d surface construction algorithm. In *Proceedings of the 14th annual conference on Computer graphics and interactive techniques*, pages 163–169, 1987.
- [7] Gregory M. Nielson. Dual marching cubes. In *Proc. of IEEE Visualization 2004*, pages 489 – 496, 2004.
- [8] Tao Ju, Frank Losasso, Scott Schaefer, and Joe Warren. Dual contouring of hermite data. In *Siggraph 2002, Computer Graphics Proceedings*, pages 339–346. ACM Press / ACM SIGGRAPH / Addison Wesley Longman, 2002.
- [9] Schaefer S. and Warren J. Dual marching cubes: Primal contouring of dual grids. In *Pacific Graphics 2004*, pages 70 – 76, 2004.
- [10] Schaefer S. and Warren J. Dual contouring: "the secret sauce". Technical Report TR 02408.
- [11] Yutaka Ohtake and Alexander G. Belyaev. Dual/primal mesh optimization for polygonized implicit surfaces. In *Seventh ACM Symposium on Solid Modeling and Applications*, pages 171–178, 2002.
- [12] S. Akkouche Antoine Leclercq and E. Galin. Mixing triangle meshes and implicit surfaces in character animation. In *Proceedings of the Eurographic Workshop on Computer Animation and Simulation*, pages 37–47, 2001.



*Tire*

*Dice*

*Part*

*Cup*

### 그림 7- 결과

왼쪽 - 모델 구성에 사용된 다원체 단위. 중앙 - 다원체 투사 특성 및 법선 벡터 연속 특성을 이용해서 매쉬의  
침점(vertex)들을 정확한 표면 위에 위치시킨 결과. 오른쪽 - 날카로운 부분 처리(sharp-featuring) 후의 최종 매쉬