

삼차원 메시에 대한 기하학 라이브와이어와 기하학 라이브레인

유 관 희[†]

요 약

이차원 영상에서 정의된 에지와 유사하게 삼차원 메시에서도 주요 부위의 경계를 표현하는 기하학 특징을 정의할 수 있다. 삼차원 메시에서 기하학 특징은 메시 단순화, 메시 변형과 메시 편집 등과 같은 여러 응용에 기본적으로 사용되고 있다. 본 논문에서는 삼차원 메시의 기하학 특징을 효과적으로 찾기 위하여 이차원 영상의 라이브와이어와 라이브레인 기법을 삼차원 메시로 확장한 기하학 라이브와이어와 기하학 라이브레인 기법을 제안한다. 제안된 기법에서는 메시의 기하학 특징을 나타내기 위하여 근사곡률을 사용하였고 메시 그 자체를 정의된 비용함수를 에지의 가중치로 가지는 가중치 방향그래프로 나타내었다. 그리고 만들어진 가중치 방향그래프에 대해 잘 알려진 최단경로 탐색 알고리즘을 이용하여 사용자에게 의해 지정된 점들 사이에 존재하는 삼차원 메시에서의 기하학 특징을 추출하였다. 본 논문에서는 사람 얼굴, 소, 신발과 차아 메시 모델에 나타나는 기하학 특징을 추출하기 위하여 제안한 기법을 적용하여 얻어진 결과를 가시화한다.¹⁾

Geometric LiveWire and Geometric LiveLane for 3D Meshes

Kwan-Hee Yoo[†]

ABSTRACT

Similarly to the edges defined in a 2D image, we can define the geometric features representing the boundary of the distinctive parts appearing on 3D meshes. The geometric features have been used as basic primitives in several applications such as mesh simplification, mesh deformation, and mesh editing. In this paper, we propose geometric livewire and geometric livelane for extracting geometric features in a 3D mesh, which are the extensions of livewire and livelane methods in images. In these methods, approximate curvatures are adopted to represent the geometric features in a 3D mesh and the 3D mesh itself is represented as a weighted directed graph in which cost functions are defined for the weights of edges. Using a well-known shortest path finding algorithm in the weighted directed graph, we extracted geometric features in the 3D mesh among points selected by a user. In this paper, we also visualize the results obtained from applying the techniques to extracting geometric features in the general meshes modeled after human faces, cows, shoes, and single teeth.

키워드: 삼차원 메시(3D meshes), 기하학 특징(geometric features), 기하학 라이브와이어(geometric livewire), 기하학 라이브레인(geometric livelane)

1 서 론

컴퓨터에 좀 더 현실감있고 입체감있는 영상을 제공하기 위하여 실제계에 존재하는 객체를 삼차원으로 모델링하고 있다. 모델링은 순수 소프트웨어를 사용하여 이루어질 수도 있고 삼차원 스캐너에 의해서 이루어질 수 있다. 일반적으로 이들 삼차원 모델은 컴퓨터에서 효과적으로 처리될 수 있도록 삼차원 메시(mesh) 구조로 표현되어 다양한 분야에 응용되고 있다. 여러 응용에서 가장 기본적으로 요구되는 사항 중의 하나가 삼차원 메시에 나타나는 주요 경계를 표

현하는 기하학 특징을 추출하는 것이다. 예를 들어, 복잡하고 큰 물체를 좀 더 정밀하게 표현하기 위해서는 많은 메시 정보를 이용하여야 하는데 이 경우 가시화의 속도에 중대한 영향을 끼치게 된다. 이를 해결하기 위한 방법 중의 하나가 시점에서 가까이 있는 객체는 원래 메시를 가시화하고 시점으로부터 멀리 떨어져 있는 객체는 원래 메시로부터 단순화된 메시를 가시화하는 것이다. 이때 중요한 사항이 메시를 단순화할 때 메시의 형태, 즉 기하학 특징을 잘 보존하는 것이다. 또 다른 예로서 메시를 편집(editing)하거나 혹은 분할(segmentation)할 때 대부분의 경우 메시의 기하학 특징 부위를 자르거나 조작하거나 수정하는 경우가 자주 발생한다. 그리고 메시상에서 선택된 여러 개의 점 혹은 선을 변형시킴으로써 메시를 전체적으로 변형시키는 메시 변형

* 이 논문은 2004년도 충북대학교 학술연구지원사업의 연구비 지원에 의해서 연구되었음.

† 종신회원 : 충북대학교 컴퓨터교육과 및 정보산업공학과 교수
논문접수 : 2004년 12월 10일, 심사완료 : 2005년 1월 17일

(mesh deformation)에서도 선택된 점이나 혹은 선이 기하학 특징에 포함되는 경우가 대부분이다. 더욱이 메쉬 모핑(mesh morphing)에서는 두 메쉬간에 대응되는 기하학 특징들을 사용하여 메쉬의 자연스러운 단계별 변형을 처리하는 경향이 있다. 위의 예처럼 많은 응용에서 메쉬에서 보존하거나 혹은 처리해야 하는 부위가 기하학 특징으로 표현되기 때문에, 이들 응용을 효과적으로 처리하기 위해서는 기하학 특징을 효과적으로 추출하는 것이 무엇보다도 중요하다. 이러한 중요성 때문에 최근 들어 이들 기하학 특징을 추출하는 알고리즘들이 제시되었다.

2002년도에 이차원 영상의 이미지 스네이크 기법을 삼차원 메쉬로 확장한 기하학 스네이크 기법(geometric snakes)이 Lee 등[11]에 의해서 제시되었다. 기하학 스네이크 방법에서는 기하학 특징이 존재하는 메쉬상의 표면 부위를 매개 변수 사상(parameter mapping)[5]을 통하여 이차원 평면상에 사상시킨 후 이차원 영상을 만들어 이미지 스네이크 기법을 적용하여 이차원 영상에서 에지를 추출한다. 추출된 이차원 에지는 삼차원 메쉬상으로 복원되어 메쉬의 기하학 특징으로 표현된다. 그러나 이 방법은 삼차원 메쉬로부터 이차원 영상을 만들어야 하는 부담이 있을 뿐만 아니라 찾고자 하는 기하학 특징의 초기 곡선을 그려준 후 자동적인 경계 추적 기법에만 의존하고 있어 사용자가 의도적으로 지정한 경계부위를 대화적으로 제어하기 어렵다. 또 다른 연구로 Yoo 등[19]은 선택된 커서가 메쉬상의 주요 특징을 나타내는 정점으로 옮겨가는 기하적 스냅핑(geometric snapping)을 제시하였으며, 기하학 스냅핑을 이용하여 메쉬에 나타나는 기하학 특징의 일부를 찾을 수 있는 방법을 제시하였다. 그러나 이 방법 역시 궁극적으로 사용자가 의도한 부위의 기하적 특징을 찾기란 매우 어렵다.

본 논문에서는 사용자가 지정한 부위를 충분히 반영하여 메쉬에 나타나는 기하학 특징을 찾기 위하여 Falcao 등[3]에 의해 제시된 이차원 영상에서 에지를 추출하기 위한 라이브 와이어(livewire)와 라이브 레인(liveline) 기법을 삼차원 메쉬로 확장하는 기법을 제시한다. 다시 말해 메쉬가 가시화된 컴퓨터 화면에서 사용자가 선택한 점들이 최대한 반영된 기하학 특징을 추출할 수 있는 기법을 제시한다. 본 논문에서는 이들 기법을 각각 기하학 라이브 와이어와 기하학 라이브 레인이라 부른다. 기하학 라이브 와이어와 기하학 라이브 레인 기법을 개발하기 위하여 고려해야 되는 가장 중요한 사항중의 하나가 메쉬의 기하학 특징을 정의하기 위하여 이차원 영상의 화소에 저장된 값과 유사하게 메쉬 표면에서 수치값으로 정의하는 것이다. 본 논문에서는 이를 위하여 메쉬 표면의 정점과 에지에서 계산가능한 근사곡률(approximate curvature)을 이용하였고 이웃한 영역에서 근사곡률을 부드럽게 나타내기 위하여 블러링(blurring) 기법을 적용하였다. 본 논문에서 제시된 기하학 라이브 레인과 기하학 라이브

와이어 기법에서도 사용자에 의해 선택된 점들 사이의 기하학 특징을 찾기 위하여 가중치 방향그래프(weighted directed graph)를 이용하고 있다. 그러나 이차원 영상에서와는 달리 본 논문에서 삼차원 메쉬의 정점을 그대로 사용하고 메쉬의 에지를 두 개의 방향 에지로 구성하여 가중치 방향그래프를 만들었다. 여기서 방향 에지의 가중치는 한 정점에서 다른 정점으로 이동하는데 소요되는 비용함수이다. 이차원 라이브 와이어와 유사하게 본 논문에서 제시한 기하학 라이브 와이어에서도 만들어진 가중치 방향그래프에 대한 최단경로 탐색 알고리즘을 이용하여 사용자가 선택한 점들 사이의 기하학 특징을 추출하였다. 기하학 라이브 와이어에서는 선택한 정점으로부터 모든 정점까지의 최단 경로를 매번 구해야 하기 때문에 처리시간이 많이 소요된다. 이를 개선하기 위하여 본 논문에서는 라이브 레인과 유사하게 선택된 정점으로 최단경로 탐색을 처리해야 하는 적용 범위를 지정할 수 있는 기하학 라이브 레인 기법을 제시한다. 본 논문에서 제시한 알고리즘을 등방성 메쉬(isotropic mesh)에 나타나는 기하학 특징을 추출하기 위하여 적용할 경우 등방성 메쉬가 격자 형태의 에지들로 구성될 수 있으므로 추출된 기하학 특징 역시 격자 모양으로 나타날 수 있다. 본 논문에서는 곡선 생성 기법과 메쉬 표면의 정규벡터를 이용하여 추출된 격자 모양의 기하학 특징을 부드럽게 하였다.

본 논문의 구성은 다음과 같다. 제 2장에서는 이차원 영상에서 에지를 추출하기 위한 기존의 연구 결과를 각각 살펴보고, 제 3장에서는 기하학 라이브 와이어와 기하학 라이브 레인을 적용하기 위하여 삼차원 메쉬 정점에서 근사곡률을 구하는 방법과 블러링하는 방법을 제시하고 정점사이를 이동하기 위하여 소요되는 비용함수를 계산하는 방법을 제시한다. 제 4장에서는 삼차원 메쉬의 기하학 특징을 추출하기 위한 기하학 라이브 와이어와 기하학 라이브 레인 기법을 설명하고, 제 5장에서는 본 논문에서 제시한 두 기법을 얼굴 모델, 소와 신발 모델, 인공 치아 모델에 적용하여 기하학 특징을 추출하였을 때 나타난 결과를 가시적으로 보여주며, 마지막으로 6장에서 본 논문의 결과를 요약하고 향후 연구 과제를 제시한다.

2. 관련 연구

삼차원 메쉬에 대한 기하학 특징 추출에 관한 연구 결과는 앞절에서 소개하였으므로 이번 장에서는 이차원 영상에서 나타나는 에지를 추출하기 위한 기존 연구의 주요 결과를 소개한다. 이차원 영상에서 에지를 추출하기 위한 대표적인 방법은 Kass 등[9]에 의해 제시된 스네이크(snake)이다. 이 방법에서는 사용자에 의해서 지정된 점들을 이용하여 초기 스네이크를 만든 후 스네이크에 존재하는 내부 힘

(internal forces)과 영상에 존재하는 외부 힘(external forces)의 합이 최소가 될 때까지 반복 처리하여 최종 스네이크를 구한다. 그러나 이 방법에서는 사용자에게 의해서 선택된 점들이 단지 가상의 점으로 밖에 사용되지 않아 사용자가 원하는 정확한 에지를 추출하기 어렵다.

스네이크와 달리 Mortensen 등[13]이 제시한 인공가위법(intelligent scissors)은 사용자가 원하는 점을 포함하는 에지를 추출할 수 있다. 이 방법의 기본 아이디어는 이차원 영상을 가중치 방향그래프로 표현한다. 즉 영상의 각 화소를 그래프의 정점으로 인접한 화소간의 강도(intensity)로부터 얻어지는 정보를 그래프 에지의 가중치로 부여한 후 사용자가 영상에서 특정한 화소를 종자점(seed point)으로 선택할 경우 그 화소에 해당하는 그래프의 정점으로부터 다른 모든 정점까지의 최단경로 트리를 구한다. 이렇게 구해진 최단경로 트리를 이용하여 사용자가 다른 정점을 선택할 경우 그 정점으로부터 종자점까지의 경로를 에지로 처리한다. 그러나 사용자가 선택한 정점이 종자점으로부터 매우 멀리 떨어져 있는 경우 의도하는 경로가 찾아질 수 없는 경우가 자주 발생한다. 이를 해결하기 위하여 사용자에게 더 많은 종자점을 부여하여 처리하는 라이브와이어, 라이브라인, 라이브와이어 온더플라이(livewire on the fly)가 Falcao 등[3,4]에 의하여 제시되었다.

라이브와이어에서는 인공가위법과 같이 전역적인 그래프를 탐색하지만 사용자가 대화적으로 커서를 특징의 경계 부근으로 옮기면서 종자점을 추가적으로 주어 새로 지정된 종자점으로부터 새로운 최단경로 트리를 계산해나간 후 이 트리에 나타나는 종자점간의 경로를 특징 경계로 추출한다. 그러나 라이브와이어는 인공가위법과 같이 종자점으로부터 다른 모든 정점까지의 최단경로 트리를 매번 구하기 위하여 그래프를 전역적으로 탐색해야 하므로 효율성이 떨어진다. 라이브라인과 라이브와이어 온더플라이는 효율성을 증가시키기 위하여 개선된 기법들이다. 라이브라인은 그래프 탐색 영역을 종자점을 중심으로 한 지역창(local window)으로 제한하고 대화적인 피드백을 제공하는 방식이다. 커서를 움직일 때마다 종자점에서 그 커서 정점까지의 최단경로 트리를 대화적으로 보여주고 커서가 그 윈도우를 벗어날 때 윈도우 경계와 특징 경계가 만나는 점에서 새로운 종자점이 자동적으로 설정되게 하였다. 이 때 특징 경계는 커서를 중심으로 하는 레인의 지역창 안에 있을 때만 추출된다. 바꾸어 말하면 정해진 레인 지역창 안에서 특징 경계로 커서를 움직여 나가야 하므로 잡음이 있거나 복잡한 물체가 있는 영상에서 사용자의 의도를 반영한 특징 추출이 가능하다. 라이브와이어 온더플라이는 전역적인 그래프 탐색이라는 기조를 유지하면서 잘 알려진 최단경로 탐색 알고리즘의 기본 성질을 이용한다. 이 방법은 최단경로 트리를 현 커서 위치의 누적 비용까지만 점증적(incremental)으로 확장하여 추출된 특징

경계를 보여주면서 커서가 목적 경계를 벗어날 때에 새로운 종자점을 설정해나간다. 따라서 불필요한 계산을 피함으로써 더 빠른 영상 특징 추출이 가능하다. 그러나 커서가 종자점으로부터 멀어질수록 전체적인 경로 트리가 커지므로 반응시간이 더 길어진다.

3. 삼차원 메쉬에 대한 전처리 작업

Falcao 등[3]에 의해 제시된 라이브와이어와 라이브라인 기법에서는 이차원 영상을 가중치 방향그래프로 표현하였다. 그들은 영상의 화소를 그래프의 정점으로 나타내었고, 이웃한 화소간의 관계를 그래프의 에지로 표현하였다. 이차원 영상과 달리 삼차원 메쉬의 각 정점에는 영상의 각 화소 값과 같은 값이 존재하지 않는다. 따라서 영상의 화소 값으로부터 화소간의 경사도를 정의하고 중력(gravity)에 의한 에너지함수를 정의하여 그래프 에지의 가중치로 부여한 영상에서와는 달리 삼차원 물체는 그 방위(orientation)가 변할 수 있으므로 경사도에 의한 중력을 특징 추출에 사용할 수 없다. 따라서 제 3.1절에서는 삼차원 메쉬의 각 정점에 어떤 값을 부여할 것인지를 논의하고 제 3.2절에서는 정점에 지정된 값으로부터 방향 에지의 가중치를 어떻게 구할 것인지를 자세히 논의한다.

3.1 정점의 근사곡률 구하기

많은 연구에서 삼차원 모델에서 기하적인 특징은 특정한 점에서 기울기의 변화량으로 정의되는 곡률(curvature)을 기준으로 사용하고 있다[18]. 그러나 곡면 모델과는 달리 면의 집합으로 구성되는 메쉬 모델에서는 메쉬 표면의 임의의 점에서 정확한 곡률을 정의하고 구하기가 어렵다. 따라서 많은 경우에 근사곡률(approximate curvatures)을 정의하여 사용한다[1,7,10,14-17]. 근사곡률을 구하는데 가장 중요한 요소는 구해질 근사곡률이 메쉬 모델의 주요 기하학 특징을 얼마나 잘 설명하고 있는지일 것이다. 본 논문에서는 다양하게 제시된 근사곡률 계산 기법 중에 Alliez 등[1]이 제시한 곡률 텐서(curvature tensor)를 이용하였다. 그들이 제시한 임의의 정점 v 에서의 곡률 텐서 $AC(v)$ 는 사용자에게 의해서 지정된 이웃 영역의 범위 B 와 B 영역에 존재하는 이웃한 면들의 정규 벡터의 사이각을 이용하여 다음과 같이 정의하였다.

$$AC(v) = \frac{1}{|B|} \sum_{\text{에지 } e} \beta(e) |e \cap B| (\mathbf{e} \cdot \mathbf{e}) \quad (1)$$

식 (1)에서 $|B|$ 는 텐서가 계산될 v 주변에 있는 곡면 영역의 합을 의미하며 고려될 주변 영역의 범위는 사용자가 줄 수 있는 값으로 본 논문에서는 그 범위를 v 로부터 주어진 삼차원 메쉬를 포함하는 가장 작은 직육면체의 대각선

길이의 1/100 이내로 주었다. $\beta(e)$ 는 매쉬 에지 e 에 인접한 두 면의 정규 벡터사이의 부호가 있는 사이각을 의미한다. 따라서 볼록 형태로 인접해 있으면 $\beta(e)$ 는 양의 사이각을 오목한 형태로 인접해 있으면 음의 사이각으로 계산된다. $|e \cap B|$ 는 영역 B에 존재하는 에지 e 의 길이이고, \vec{e} 는 에지 e 와 같은 방향의 단위 벡터를 나타내고, \bullet 는 두 벡터의 내적을 의미한다.

이 방법 이외에도 다양한 다른 방법을 사용하여 매쉬의 각 정점에 대한 근사곡률을 구할 수 있다. 지금까지 제시되어 널리 사용되는 방법들을 살펴보면 인접한 에지 사이의 사이각과 인접한 면의 사이각을 이용하는 방법[14,15,16], QEM (quadric error metric)을 이용하는 방법[6], DDER(discrete differential error metric)을 이용하는 방법[10] 등이 있다. 일반적으로 매쉬의 정점에 지정된 근사곡률이 갖는 값의 범위는 매쉬의 형태에 따라 다르게 나타날 수 있다. 본 논문에서는 근사곡률을 일관되게 처리하기 위해, 정점 v 에 대한 기존 근사곡률 $AC(v)$ 을 $AC(v) / MAX_AC$ 와 같이 표준화하여 재정의하였다. MAX_AC 는 모든 정점에서의 최대 근사곡률을 의미한다. 이렇게 구한 매쉬 정점의 근사곡률을 가중치 방향그래프의 정점에 부여한다. Alliez 등[1]에서 제시된 방법이 이웃한 영역 정보를 이용하여 근사곡률을 구하고 있어 매쉬의 정점과 이 정점에 이웃한 정점에 지정된 근사곡률들 간에 유사성이 있긴 하지만 이웃한 영역에서 근사곡률이 자연스럽게 변화되어 나타나지는 않는다. 다시 말해 이웃한 영역내에서 급격하게 차이 나는 근사곡률이 존재할 수 있다. 이를 해결하기 위하여 삼각형 매쉬상의 정점 v 에 대한 근사곡률 $BAC(v)$ 을 이웃한 정점들의 근사곡률에 대해 식 (2)와 같이 가우시안 스무딩 필터(Gaussian smoothing filter)를 적용하여 다시 정의한다.

$$BAC(v) = \sum_{i=0}^{k-1} AC(nv_i) \times \frac{1}{\sqrt{2\pi\sigma}} \times e^{-\frac{(dx^2 + dy^2 + dz^2)}{2\sigma^2}} \quad (2)$$

위 식에서 정점에 대해 벡터 (dx, dy, dz) 는 $(v^x - nv_i^x, v^y - nv_i^y, v^z - nv_i^z)$ 이며 v 의 인접 정점 $nvi = (nv_i^x, nv_i^y, nv_i^z), i=0, \dots, k$ 이다. 위 식에 존재하는 σ 값은 정점 v 에 인접한 정점들의 수인 k 의 값에 따라 아래 표와 같이 주었다.

<표 1>의 값 할당

인접정점 k의 갯수	σ 의 할당값
$k \leq 7$	0.85
$7 < k \leq 16$	1.7
$16 < k \leq 36$	2.5
$36 < k$	3.5

3.2. 방향 에지의 가중치 구하기

주어진 삼차원 매쉬의 두 정점 v 와 u 를 연결하는 에지는 가중치 방향그래프에서 두 개의 방향 에지 $e\langle v,u \rangle$ 와 $e\langle u,v \rangle$ 로 나타내어진다. $e\langle v,u \rangle$ 는 v 에서 u 로 가는 방향 에지를 표기하며, 이 방향 에지에 부여된 가중치의 의미는 기하학 특징을 정점 v 로부터 정점 u 로 이동시키는데 소요되는 비용을 말한다. 본 논문에서는 이 비용을 $cost\langle v,u \rangle$ 으로 표기하고 Mortensen 등[13]이 제시한 방법을 삼차원 매쉬에 다음과 같이 확장 적용하였다.

$$cost(u, v) = \omega_z f_z(v) + \omega_d f_d(u, v) + \omega_g f_g(v) \quad (3)$$

위 식에서, 세 함수 f_z, f_d 와 f_g 는 라플라시안 제로 교차, 곡률 방향과 곡률 크기를 나타내는 함수이다. 라플라시안 제로 교차 $f_z(v)$ 는 정점 v 가 주어진 매쉬의 기하학 특징 부위에 존재하는지 아닌지를 표현하기 위하여 사용된다. 다시 말해 v 가 기하학 특징 부위에 존재하면 $f_z(v) = 0$ 으로 그렇지 않으면 $f_z(v) = 1$ 을 부여하였다. 본 논문에서는 경험적인 결과를 기반으로 구해진 표준화된 근사곡률에 대해 기하학 특징을 나타내는 값으로 0.75을 부여하였다. 즉 정점 v 에 대해 $AC(v)$ 가 0.75보다 그 정점이 기하학 특징 부위에 있을 것으로 판단하고 $f_z(v) = 0$ 으로 설정하고 그렇지 않으면 그 정점이 기하학 특징 부위에 존재하지 않을 것으로 판단하고 $f_z(v) = 1$ 으로 설정한다. 일반적으로 보다 큰 근사곡률을 갖는 정점이 보다 작은 근사곡률을 갖는 정점보다 더 잘 기하학 특징을 표현하고 있으므로 곡률 방향 함수 $f_d(u, v)$ 가 두 정점 v 와 u 사이에 기하학 특징이 어떻게 이동하는지를 나타내기 위하여 사용된다. 정점 v 에 인접한 정점들 중 최대 근사곡률을 갖는 정점을 mv 라 할 때 이들 세 정점들에 의해서 만들어지는 세 벡터 $d(v, mv) = mv - v, d(v, u) = u - v$ 와 $d(u, v) = v - u$ 라 하자. 이들 세 벡터를 이용한 곡률 방향 함수 $f_d(u, v)$ 는 $AC(v) - AC(u)$ 의 부호에 따라 양수이면 $1.0 \cdot d(v, o) \cdot d(v, u) / (|d(v, o)| |d(v, u)|)$ 으로 음수 이면 $1.0 \cdot d(v, o) \cdot d(u, v) / (|d(v, o)| |d(u, v)|)$ 으로 정의하였다. $f_d(u, v)$ 가 거의 영에 가까우면 기하학 특징은 정점 v 에서 정점 u 로 이동할 확률이 높으며 그렇지 않으면 기하학 특징의 이동은 역방향으로 진행된다. 마지막으로 곡률 크기 함수 $f_g(v)$ 는 정점 v 에서의 근사곡률 $AC(v)$ 을 나타낸다. 위의 식에서 각 w 는 대응하는 함수의 가중치로 본 논문에서는 경험적인 결과를 기반으로 $w_z = 0.43, w_d = 0.43$ 와 $w_g = 0.14$ 로 설정하였다. 가중치의 의미는 라플라시안 제로 교차와 곡률 방향이 중요한 역할을 하는 반면 곡률 크기가 그 자체는 상대적으로 덜 영향을 끼친다.

4. 삼차원 메쉬에서 기하학 특징 추출 기법

이번 장에서는 삼차원 메쉬에 존재하는 기하학 특징을 추출하는 기하학 라이브라이어와 기하학 라이브레인 방법을 제시한다. 먼저 가장 단순하게 생각할 수 있는 주어진 메쉬를 이차원으로 사상시켜 처리하는 방법을 설명하고, 삼차원 메쉬로부터 직접적으로 구성된 가중치 방향그래프를 이용하여 처리하는 기법을 설명한다.

4.1 이차원 사상을 통한 기법

가장 직접적으로 생각할 수 있는 방법이 기하학 스네이크에서처럼 삼차원 메쉬의 표면을 이차원 평면에 사상시켜 기존의 이차원 영상에 대한 라이브라이어와 라이브레인 기법을 적용하여 기하학 특징을 추출하는 방법이다. 이 방법은 크게 세 과정으로 처리된다. 먼저 사용자에게 의해서 기하학 특징을 추출하고자 하는 메쉬 표면의 영역이 지정되면, 이 표면을 Floater [5]에 의해서 제시된 매개변수 사상 기법을 이용하여 기하학 이미지를 생성한다. 다음 단계가 이차원 영상에서 제시된 라이브라이어와 라이브레인 알고리즘을 기하학 이미지에 적용하여 에지를 구한다. 마지막으로 추출된 에지를 삼차원 메쉬 표면으로 복원하여 원하는 최종 기하학 특징을 얻는다.

위와 같은 방법이 삼차원 문제를 해결하기 위한 일반적인 방법이기도 하지만 다음과 같은 문제점이 존재한다. 첫 번째로 삼차원 메쉬 표면에 대해 기하학 이미지를 생성할 때, 삼차원 메쉬 형태로 나타나는 길이와 각도 등을 완벽하게 보존하기가 매우 어렵다. 더욱이 매핑을 처리하기 위하여 역행렬을 구해야 하기 때문에 삼차원 메쉬 표면의 정점의 개수가 많으면 많을수록 매핑 시간이 상대적으로 많이 소요됨을 알 수 있다. 또한 기하학 이미지를 화소 수준에서 생성해야 하고 또한 라이브라이어와 라이브레인 알고리즘을 적용하기 위해서는 화소 수준에서 가중치 방향그래프를 만들어야 하는 부담이 있다.

4.2 기하학 라이브라이어

제 4.1절에서 소개한 방법과는 달리 주어진 메쉬로부터 직접적으로 만들어진 가중치 방향 그래프를 이용하는 기하학 라이브라이어 기법을 이번 절에서 자세히 소개한다. 사용자는 화면상에 가시화된 삼차원 메쉬를 보면서 기하학 특징 부위로 판단되는 임의의 정점을 화면 공간에서 선택한다. 만약 선택된 정점이 기하학 특징 부위에 속하지 않으면 기하학 스내핑 [19] 기법을 이용하여 선택된 정점을 기하학 특징 부위로 자연스럽게 옮긴다. 종자점으로 정점 S_1 가 선택되었다고 가정하자. 그리고 제 3장에 기술된 방법에 따라 만들어진 가중치 방향그래프에서 널리 알려진 최단경로 탐색 알고리즘을 이용하여 정점 S_1 로부터 모든 정점까지의 최단

경로를 나타내는 트리 SPT_{S_1} 를 구한다. 이 상태에서 정점 S_1 로부터 시작하는 메쉬의 특정한 정점 S_2 까지의 기하학 특징을 구하기 위하여 사용자가 정점 S_2 를 선택하면, 기하학 라이브라이어 알고리즘에서 미리 구한 SPT_{S_1} 에 존재하는 S_2 에서 S_1 까지의 가는 경로를 구하여 S_1 와 S_2 사이 존재하는 기하학 특징, $GF(S_1, S_2)$ 으로 사용자에게 제공한다. 사용자는 정점 S_2 가 아닌 또 다른 정점 SS_2 를 선택함으로써 S_1 로 출발하여 SS_2 까지 가는 또 다른 기하학 특징을 추출할 수 있다. $GF(S_1, S_2)$ 가 확정된 후에 사용자가 S_2 로 시작하는 새로운 기하학 특징을 추출하고자 할 때 기하학 라이브라이어 알고리즘에서는 정점 S_2 를 새로운 종자점으로 지정하고 S_2 로부터 시작하여 모든 정점으로 가는 최단경로를 나타내는 트리 SPT_{S_2} 를 Dijkstra 알고리즘을 이용하여 다시 구한다. 앞에서와 똑 같은 방법으로 이 상태에서 사용자가 다른 정점 S_3 를 선택하면 정점 S_3 를 새 종자점으로 지정한 후 위의 과정을 반복 적용하여 S_2 와 S_3 사이 존재하는 기하학 특징 $GF(S_2, S_3)$ 를 SPT_{S_2} 에서 추출한다. 이러한 작업을 사용자가 원하는 모든 종자점들에 대해 반복 적용한다. 만약 사용자가 k 개의 정점을 종자점으로 선택하였다고 가정하면 $GF(S_i, S_{i-1}), i=1, \dots, k$ 을 구한 후 이들 경로들을 처음부터 끝까지 연결한 경로, 즉 $GF(S_1, S_2)+GF(S_2, S_3)+\dots+GF(S_k, S_{k-1})$ 를 추출하고자 하는 메쉬상에서의 기하학적 특징으로 최종 지정한다. 그러나 기하학 라이브라이어 알고리즘은 사용자에게 의해 선택된 종자점으로부터 모든 정점까지의 최단경로를 나타내는 트리를 매 종자점마다 구해야 하므로 종자점이 많으면 많을수록 계산 시간이 많이 필요함을 할 수 있다.

4.3 기하학 라이브레인

제 4.2절에서 소개한 기하학 라이브라이어 기법에서의 계산 시간을 줄이기 위해서 제시된 기하학 라이브레인 알고리즘을 이번 장에서 자세히 기술한다. 라이브레인에서 에지 추출을 위한 시간을 단축하기 위하여 그래프 탐색 영역을 종자점을 중심으로 한 지역창(local window)으로 제한하고 대화적인 피드백을 제공했던 방식과 유사하게 기하학 라이브레인에서도 지역창을 주었다. 정사각형을 지역창으로 사용하였던 라이브레인과 유사하게 기하학 라이브레인에서는 정육면체를 지역창으로 사용하였다. 사용자가 선택한 첫 번째 정점을 종자점 S_1 으로 주었다고 하자. S_1 을 중심으로 하는 정육면체에 포함되는 삼차원 메쉬를 구하면 S_1 을 포함하는 객체와 그렇지 않은 객체로 분할될 수 있다. 종자점을 포함하지 않은 분할된 객체에 속하는 정점에서 종자점 S_1 까지 메쉬의 표면을 따라 가기 위해서는 주어진 정육면체를 벗어난 경로를 반드시 통과해야 함을 알 수 있다. 따라서 지역창의 개념에 따라 이들 객체를 더 이상 고려할 필요가 없음을 알 수 있다. 다시 말해 기하학 라이브레인 알고리즘이 종자점을 포함하고 정육면체 내부에 속하는 객체만을 고

려하여 기하학 특징을 추출하면 좀더 빠르게 처리할 수 있다. 종자점 $S = (S_x, S_y, S_z)$ 라 하고 정육면체의 변의 길이를 d 라 할 때 임의의 정점 $P = (P_x, P_y, P_z)$ 에 대해 $|P_x - S_x| \leq d, |P_y - S_y| \leq d$ 이고 $|P_z - S_z| \leq d$ 이면 정점 P 가 정육면체의 내부에 있으므로 S_1 으로부터 시작하여 가중치 방향 그래프의 너비 우선 탐색(breadth first search)을 하면서 이들 정점이 정육면체의 내부에 존재하는지 아닌지를 판단한다. 정점을 탐색할 때 그 정점이 정육면체의 내부에 존재하면 향후 탐색을 계속 진행하고 그렇지 않으면 탐색을 멈춘다. 잘 알다시피 두 정점사이의 유클리디안 거리를 이용하여 지역창을 고려하는 경우 두 정점을 연결하는 직선이 메쉬의 표면상에 존재하지 않을 수 있다. 본 논문에서는 이를 개선하기 위하여 지역창의 경계 범위를 종자점 S_1 으로부터 시작하여 메쉬의 표면을 따라 가장 짧게 이동할 수 있는 유클리디안 거리로 주었다. 특정한 유클리디안 거리로 주어진 지역창에 존재하는 정점들 역시 종자점 S_1 으로부터 너비 우선 탐색 방법을 통해 쉽게 구할 수 있다.

지정된 종자점 S_1 에서 시작하는 기하학 특징 추출을 위해서 마지막으로 가중치 방향그래프를 이용하여 종자점으로부터 지정한 지역창 내부에 속하는 정점까지의 최단경로 트리를 구한다. 이 상태에서 사용자는 다음 종자점 S_2 를 선택한다. S_2 가 미리 구한 최단경로 트리의 정점이면 S_2 에서 S_1 을 연결하는 경로를 두 점사이의 기하학 특징 $GF(S_1, S_2)$ 으로 제공하고 종자점 S_2 로부터 위의 작업을 반복 적용한다. 그렇지 않으면 가중치 방향그래프를 이용하여 S_2 로부터 가장 가까이 존재하는 최단경로 트리의 정점을 구하여 그 정점을 S_2 로 다시 지정한다. 그리고 S_2 에서 S_1 을 연결하는 경로를 두 점 사이의 기하학 특징 $GF(S_1, S_2)$ 으로 제공하고 종자점 S_2 로부터 위의 작업을 반복 적용한다. 기하학 라이브와이어와 유사하게 사용자가 k 개의 정점을 종자점으로 선택했다고 가정하면 $GF(S_1, S_2) + GF(S_2, S_3) + \dots + GF(S_k, S_{k+1})$ 가 추출하고자 하는 메쉬상에서의 기하학적 특징으로 최종 지정된다.

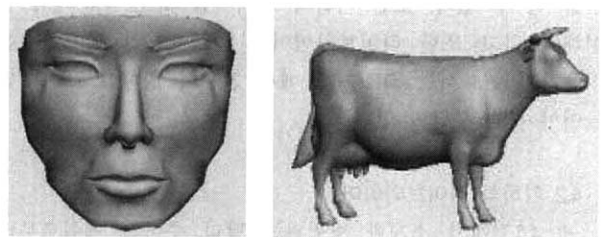
4.4 추출된 기하학 특징의 후처리

가상으로 존재하는 삼차원 객체를 표현하는 메쉬들이 설계자에 의해 만들어진 반면 실제 존재하는 삼차원 객체를 메쉬로 구성할 때 정확도와 효율성 등을 고려하여 삼차원 입력기를 사용하는 경향이 있다. 일반적으로 삼차원 입력기는 입력영역을 일정한 간격으로 나눈 후에 각 지점에서의 객체에 대한 깊이 정보 추출을 통해 처리된다. 따라서 많은 경우에 삼차원 입력기를 통해 만들어진 메쉬는 등방성 유형(isotropic type)이 될 수 있다. 이러한 등방성의 메쉬에 기하적 라이브와이어와 기하적 라이브레인 알고리즘을 적용하여 기하학 특징을 추출한다고 해 보자. 분명히 기하학 특징이 메쉬의 정점들을 연결하는 다각선 혹은 닫힌 다각선이기 때문에 등방성의 메쉬에서 추출된 기하학 특징은 계단 모양으로 나타난다.

본 논문에서는 이러한 계단 모양을 제거하기 위하여 먼저 추출된 기하학 특징을 구성하는 정점들을 제어 점으로 갖는 곡선을 생성한 후 이들 곡선을 메쉬의 표면으로 투영시켰다. 곡선 생성 단계에서는 기하학 특징에 나타나는 점들 중에 사용자에게 의해 제공된 종자점들을 보간시키고 이 이외의 점들을 근사시키는 곡선 생성 기법을 이용하였다 [2,18]. 따라서 종자점들이 항상 메쉬 표면에 존재하므로 두 종자점 사이에 존재하는 근사 곡선을 어떻게 메쉬 표면에 투영시킬 것인지를 생각해 보자. 근사 곡선상의 특정한 점 c 를 시작 종자점으로부터 t 만큼 떨어져 있다고 가정할 때 시작 종자점으로부터 t 만큼 떨어져 있는 기하학 특징 점을 포함하는 면을 모두 구한다. 이들 각각의 면에 대해 정규 벡터 방향으로 점 c 를 해당 면을 포함하는 평면에 투영시켜 투영 점을 구한 후, 해당 면에 속하는 투영 점들 중 점 c 에서 투영 점까지의 거리가 가장 짧은 점으로 최종 투영 점으로 선택하였다. 특히 본 논문에서는 두 종자점 사이의 곡선이 아주 많은 점들로 이루어질 수 있으므로 이 경우 시간이 많이 소요될 수 있으므로 단지 몇몇 점을 추출하여 처리하였다.

5. 구현 및 실험 결과

본 논문에서 제시된 기하학 라이브와이어와 기하학 라이브레인 기법은 MFC(Microsoft Foundation Class)와 OpenGL 라이브러리를 이용하여 PC환경에서 구현되었으며, 삼차원 메쉬를 컴퓨터에 표현하기 위하여 반에지 자료구조(half-edge data structure)[12]를 사용하였다. 특히 본 논문에서 사용하는 반 에지 자료 구조가 가중치 방향그래프의 모든 정보를 포함하고 있어 가중치 방향그래프를 위한 별도의 자료 구조를 사용하지 않았다.



(a) 사람 얼굴

(b) 소



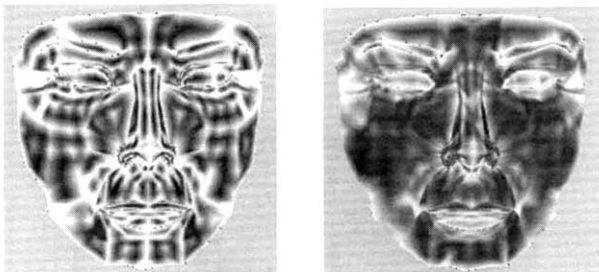
(c) 치아

(그림 1) 사람 얼굴, 소 및 치아 모델에 대한 그물 형태의 메쉬 가시화

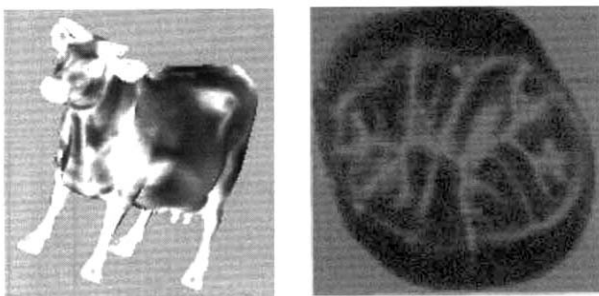
본 논문에서 제안한 기하학 라이브와이어와 기하학 라이브레인을 시험하기 위하여 사람 얼굴 모델, 소 모델, 치아 모델과 신발 모델을 사용하였으며 위 세 그림이 사람 얼굴 모델, 소 모델과 치아 모델을 보여주고 있다.

5.1 근사곡률과 근사곡률의 블러링 구현 예

식 (1)를 가지고 사람 얼굴 메쉬에 적용하여 계산한 근사곡률이 그림 2(a)와 같이 근사곡률의 크기를 가지고 가시화되었다. 이 그림에서 보다 밝은 부분은 상대적으로 큰 근사곡률을 나타내며 어두운 부분은 작은 근사곡률을 나타낸다. 그림 2(a)에서 보는 바와 같이 계산된 근사곡률이 여러 영역에서 심하게 자주 변함을 알 수 있다. 즉 사람 얼굴 영상 전반에 걸쳐 잡음이 낀 근사곡률이 구해졌음을 가시적으로 확인할 수 있다. 본 논문에서 이러한 문제점을 제거하기 위하여 식(2)를 이용하여 근사곡률을 블러링하였다. 그림 2(b)가 그림 2(a)에 블러링 기법을 적용한 후에 얻어진 결과이다. 그림 2(b)에서 보는 바와 같이 그림 2(a)에서 나타나는 급격하게 변하는 근사곡률들이 매우 부드럽게 변화됨을 알 수 있다. 소와 치아 메쉬에 대해 근사곡률을 구하여 블러링한 최종 결과가 각각 그림 2(c)와 그림 2(d)에 나타난다.



(a) 사람 얼굴(블러링전) (b) 사람얼굴(블러링후)



(c) 소(블러링후) (d) 치아(블러링후)

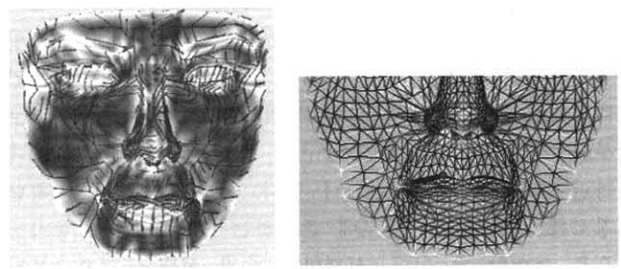
(그림 2) 사람 얼굴 모델, 소 모델, 치아 모델에 나타난 블러링한 근사곡률의 가시화

그렇게 얻어진 근사곡률을 이용하여 메쉬에 나타나는 기하학 특징을 추출하기 위하여 기하학 라이브와이어와 기하학 라이브레인 기법을 적용하고자 할 때 식 (3)을 이용하여 두 정점을 연결하는 방향 예지의 가중치를 계산하여 원하는

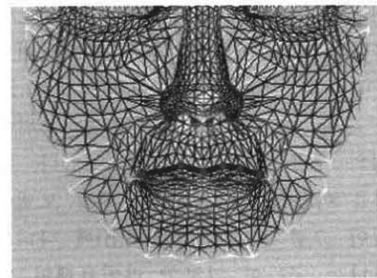
최종 가중치 방향그래프를 만들어야 한다. 따라서 앞으로 내부적으로 가중치 방향그래프가 구해졌다고 가정하고 구현 예를 설명한다.

5.2 기하학 라이브와이어

제 4.2절에서 제안한 기하학 라이브와이어 알고리즘을 적용하기 위해서는 사용자가 그림 2와 같이 블러링된 근사곡률을 갖는 메쉬를 가시화한 화면에서 상호동작하면서 기하학 특징에 존재하는 점들을 종자점으로 차례대로 선택한다. 그림 3에 나타나는 구들이 사람 얼굴 모델에서 입술 부위의 기하학 특징을 추출하기 위하여 사용자에게 의해 선택된 여섯 개의 종자점을 보여주고 있다. 선택된 종자점들이 기하학 라이브와이어 알고리즘의 입력 정보로 제공되면 제 3.2절에서 소개된 방법으로 먼저 첫 번째 종자점으로부터 가중치 방향그래프의 모든 정점까지의 최단경로를 구한다. 그림 3(a)에 나타나는 굵은 파란색 실선들이 구해진 최단경로를 가시적으로 보여준다. 이렇게 구해진 최단경로를 이용하여 두 번째 종자점으로 첫 번째 종자점으로 가는 최단경로를 두 정점을 연결하는 기하학 특징으로 선택한다. 그 결과가 그림 3(b)의 굵은 빨간색 실선으로 나타난다. 이 작업을 모든 종자점들에 대해 차례대로 반복 적용하여 결과를 얻는다. 그림 3(c)가 여섯 개의 종자점으로 추출한 기하학 특징의 최종 결과를 보여주고 있다. 그림 4(a)는 소 모델에서 아홉 개의 종자점 (구로 표시되어 있음)들을 사용자가 주었을 때 추출된 기하학 특징을 굵은 빨간색 실선으로 보여주고 있다. 그림 4(b)는 신발 모델에서 세 개의 종자점들이 주어질 때 추출된 기하학 특징을 보여준다.

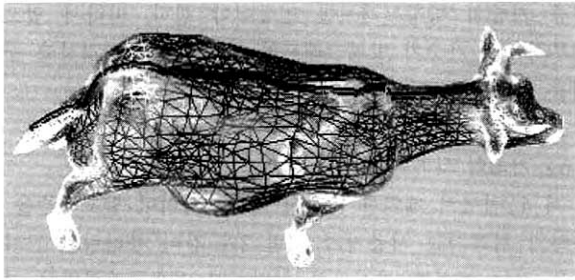


(a) (b)

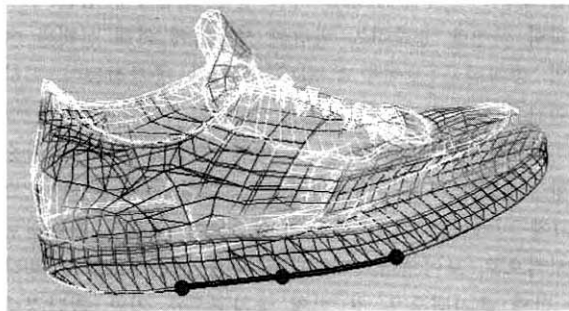


(c)

(그림 3) 사람 얼굴 모델에 대한 기하학 라이브와이어 적용 결과

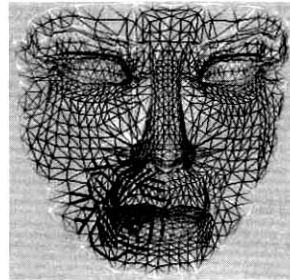


(a) 소 모델

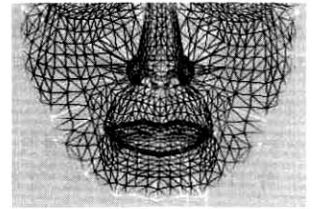


(b) 신발 모델

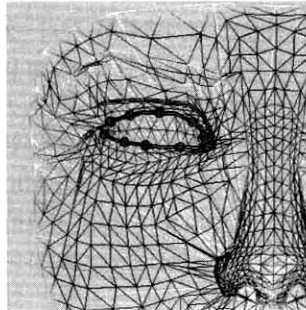
(그림 4) 소와 신발 모델에 기하학 라이브와이어 적용 결과



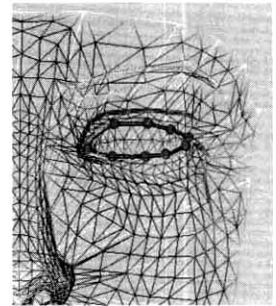
(a)



(b)



(c)

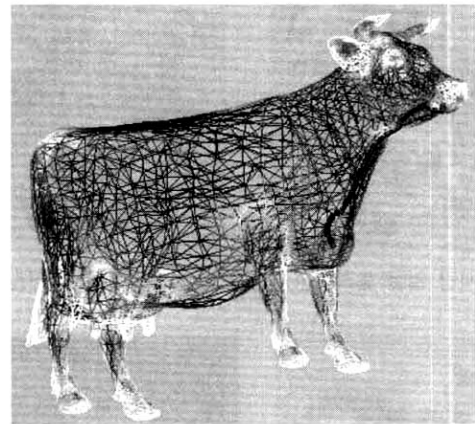


(d)

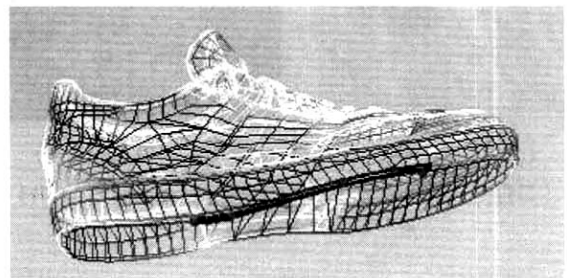
(그림 5) 사람 얼굴 모델에 대한 기하학 라이브레인 적용 결과

5.3 기하학 라이브레인

기하학 라이브와이어에서와 유사하게 사용자는 상호 동작하면서 기하학 특징을 추출하기 원하는 부위에서 첫 번째 종자점을 선택한다. 이 종자점을 이용하여 기하학 라이브레인 알고리즘은 사용자에게 주어진 가상 윈도우 값을 가지고 종자점으로부터 가상 윈도우 내에 존재하는 정점들까지의 최단경로들을 구한다. 주어진 메쉬의 경계 직육면체(bounding box)의 최장축의 길이를 ml 이라 할 때, 본 논문에서는 가상 윈도우 값으로 $ml/10.0$ 으로 주었다. 사람 얼굴 모델의 입술 부위에서 선택한 종자점으로부터 가상 윈도우 내에서 구한 최단경로가 그림 5(a)에 파란 굵은 실선으로 나타나 있다. 사용자는 구해진 최단 경로를 보면서 최단경로에 존재하는 정점을 다음 종자점으로 선택할 수 있다. 선택하는 순간 선택된 점으로부터 첫 번째 종자점까지 최단경로를 역으로 순회하면서 경로를 추출한다. 바로 이 경로가 두 점을 연결하는 기하학 특징으로 나타난다. 그림 5(b) 입술 전체를 기하학 특징으로 추출한 결과물 보여주고 있으며, 그림 5(c)와 그림 5(d) 각각이 왼쪽 눈 부위와 오른쪽 눈 부위에 나타나는 기하학 특징의 추출 결과를 보여주고 있다. 기하학 라이브레인 기법을 이용하여 소 모델과 신발 모델에서 추출한 기하학 특징이 그림 6(a)와 그림 6(b)에 나타나 있다. 그림에서 나타나는 구들이 기하학 라이브레인 기법에서 종자점들로 사용되었다.



(a)

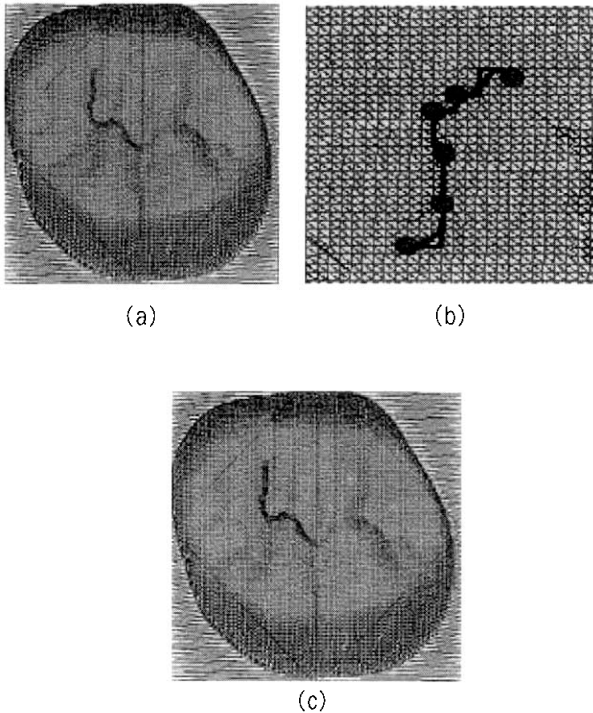


(b)

(그림 6) 소와 신발 모델에 대한 기하학 라이브레인 적용 결과

5.4 구해진 기하학 특징의 후처리

앞 장에서도 설명한 바와 같이 접촉식 삼차원 입력기에 통해 얻어진 점들로부터 만들어진 메쉬가 등방성의 메쉬가 될 수 있는데 그림 7과 같이 그물 형태로 가시화된 어금니 치아 모델이 0.05 mm 간격으로 얻어진 점들로부터 만들어진 메쉬이다. 그림 1(c)가 이 치아 모델을 가시화한 그림이고 그림 2(d)가 근사곡률을 불러링한 결과이다. 그림 1(c)와 그림 2(d)에서 보는 바와 같이 치아 모델에서 찾고자 하는 기하학 특징은 치아 모델의 위 부분에 나타나는 계곡처럼 흐르는 부위들이다. 기하학 라이브레인 알고리즘을 적용하여 추출한 기하학 특징이 그림 7(a)의 파란색 굵은 실선과 같다. 이 부분을 자세히 보면 그림 7(b)의 파란색 굵은 실선과 같이 추출된 기하학 특징이 격자 형태로 되어 있음을 알 수 있다. 격자 형태의 기하학 특징을 부드럽게 만들기 위하여 제 4.4절에서 소개된 방법을 적용하였다. 그 결과가 그림 7(b)의 빨간색 굵은 실선으로 나타나 있으며, 치아 표면에 이를 표시하면 그림 7(c)와 같다.



(그림 7) 치아 모델에서 추출된 기하학 특징의 후 처리

6. 결 론

본 논문에서는 삼차원 메쉬에 존재하는 기하학 특징을 추출하기 위하여 이차원 영상에서 에지를 추출하기 위하여 사용한 라이브와이어와 라이브레인을 확장한 기하학 라이브와이어와 기하학 라이브레인을 제시하였고, 제 5장의 예제에서 보는 바와 같이 이들 알고리즘을 적용하면 삼차원 메쉬

모델에서 나타나는 기하학 특징을 매우 효과적으로 추출할 수 있음을 알았다.

본 논문에서 제시된 연구 결과는 향후 두 가지 측면에서 좀더 발전적으로 진행되어야 할 것으로 판단된다. 첫 번째 방향은 Falcao 등[4]와 Kang 등[8]이 각각 제시한 와이어온더플라이와 향상된 라인 기법을 메쉬로 확장하는 연구이며, 두 번째 방향은 추출된 기하학 특징에서 발생하는 계단 형태가 후처리를 통해 많이 개선되었긴 하지만 본 논문에서 제시된 방법이 완벽한 해결 방법이 아니므로 좀더 좋은 해결 방법을 찾는 것이다.

참 고 문 헌

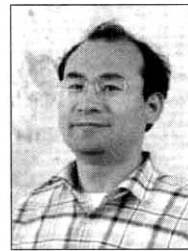
- [1] P. Alliez, D. Cohen-Steiner, B. Levy and M. Desbrun, "Anisotropic polygonal remeshes," *ACM Computer Graphics (Proc. of SIGGRAPH '03)*, 2003, pp.485-493.
- [2] D. Hearn and M.P. Baker, *Computer Graphics*, Prentice-Hall, 1994.
- [3] A. X Falcao, et. al, "User-steered image segmentation paradigms: livewire and livelane," *Graphical Models and Image Processing*, Vol. 60, 1998, pp.223-260.
- [4] A. X. Falcao, et al, "An Ultra-fast user-steered image segmentation paradigm: live wire on the fly," *IEEE Tr. on Medical Imaging*, Vol.19, No.1, 2000, pp.55-62.
- [5] M.S. Floater, "Parameterization and smooth approximation of surface triangulation," *Computer-Aided Geometric Design*, Vol.14, No.3, 1997, pp.231-250.
- [6] M. Garland and P.S. Heckbert, "Surface simplification using quadric error metric," *ACM Computer Graphics (Proc. of SIGGRAPH '97)*, 1997, pp.209-216.
- [7] X. Gu, S. Gortler and H. Hoppe, "Geometry images," *ACM Computer Graphics (Proc. of SIGGRAPH '02)*, 2002, pp.355-361.
- [8] H. W. Kang and S. Y. Shin, "Enhanced lane: interactive image segmentation by incremental path map construction," *Graphical Models*, Vol.64, No.5, 2002, pp.282-303.
- [9] M. Kass, A. Witkin and D. Terzopoulos "Snakes, active contour models," *International Journal of Computer Vision*, Vol.1, 1987, pp.321-331.
- [10] S.-J. Kim, S.-K. Kim and C.-H. Kim, "Discrete differential error metric for surface simpliciation," In *Proceedings of Pacific Graphics 2002*, pp. 276-283, Beijing, China, October 2002.
- [11] Y. Lee and S. Lee, Geometric snakes for triangular meshes, *Computer Graphics Forum*, Vol.21, No.3, 2002, pp.229-238.
- [12] M. Mantyla,, *Introduction to Solid Modeling*. W.H. Freeman & Co, 1988.
- [13] E. Mortensen and W. A Barrett, "Intelligent scissors for

image composition," *ACM Computer Graphics (Proc. of SIGGRAPH '95)*, 1995, pp.191-198.

- [14] A. Rosenfeld and E. Johnston, "Angle detection in digital curves," *IEEE Transactions on Computers*, Vol.22, 1973, pp.875-878.
- [15] A.D.C. Smith, *The Folding of the Human Brain: from Shape to Function*, University of London, PhD Dissertations, 1999.
- [16] G. Turk, "Re-tiling polygonal surfaces," *ACM Computer Graphics (Proc. of SIGGRAPH '92)*, Vol.26, No.2, 1992, pp.5564.
- [17] J. Vorsatz, C. Rossl, L. Kobbelt, and H. Seidel, "Feature sensitive remeshing," In *Proc. of EUROGRAPHICS '01*, 2001, pp.392-401.
- [18] F. Yamaguchi, *Curves and surfaces in Computer Aided Geometric Design*, Springer-Berlag, 1988.
- [19] Y. H.-Yoo and J. S. Ha, "Geometric snapping for 3d meshes," *Workshop on Computer Graphics and Geometric Modelling (Lecture Notes on Computer Science 3039)*, 2004, pp.90-97.

유 관 희

e-mail : khyoo@cbucc.chungbuk.ac.kr



1985년 전북대학교 전산통계학과 졸업
 1988년 한국과학기술원 전산학과 석사학위 취득
 1997년 한국과학기술원 전산학과 박사학위 취득
 1988년~1997년 (주)데이콤 종합연구소 선임연구원

1997년~현재 충북대학교 컴퓨터교육과, 정보산업공학과, 컴퓨터정보통신연구소 교수

2003년~2005년 카네기멜론대학교 로보틱스 연구소 방문교수
 관심분야 : 컴퓨터 그래픽스, 계산 기하학, 덴탈 및 메디칼 로보틱스