

표면의 방향정보를 고려한 메쉬의 특성정보의 보존

고명철[†] · 최윤철^{††}

요 약

대용량의 다각형 표면데이터를 효과적으로 감소시키는 많은 간략화 알고리즘들이 제안되었다. 이들 간략화 기법들은 정점, 에지, 삼각형 등과 같은 기본적인 간략화 단위에 대해 자신의 붕괴 비용함수를 적용하여 간략화 전후의 에러를 최소화 한다. 기존의 제안된 비용 함수들은 대부분 거리최적화에 기반 한 에러 측정방법을 사용한다. 그러나 기본적으로 스칼라 값인 거리요소 만으로는 현재 메쉬의 지역적인 특징을 정확히 정의하기 어렵다. 따라서 곡률이 심한 지역의 특징정보를 유지하지 못함으로써 간략화 단계를 높일수록 원래의 세부적인 모양을 잃어버리는 단점이 있다. 본 논문에서는 표면의 방향과 같은 벡터성분을 비용함수의 요소로서 고려한다. 표면의 방향성분은 거리와 같은 스칼라 양에 비의존적이다. 따라서 작은 스칼라 양을 갖는 요소라도 이의 벡터성분의 크기에 따라 보존 여부를 재고할 수 있다. 또한 제안된 비용함수를 바탕으로 하는 반-에지 붕괴에 기반 한 간략화 알고리즘을 개발한다. 이는 객체의 제거 후에 기존 에지의 두 정점 중 하나를 이용하여 새로운 정점을 표현하는 방법으로서 저장공간 상의 이점이 있으며 대용량 표면데이터의 실시간 전송을 요하는 렌더링 시스템에 매우 효과적으로 적용될 수 있다.

Mesh Simplification for Preservation of Characteristic Features using Surface Orientation

Myeong Cheol Ko[†] and Yoon Chul Choy^{††}

ABSTRACT

There has been proposed many simplification algorithms for effectively decreasing large-volumed polygonal surface data. These algorithms apply their own cost function for collapse to one of fundamental simplification unit, such as vertex, edge and triangle, and minimize the simplification error occurred in each simplification steps. Most of cost functions adopted in existing works use the error estimation method based on distance optimization. Unfortunately, it is hard to define the local characteristics of surface data using distance factor alone, which is basically scalar component. Therefore, the algorithms cannot preserve the characteristic features in surface areas with high curvature and, consequently, loss the detailed shape of original mesh in high simplification ratio. In this paper, we consider the vector component, such as surface orientation, as one of factors for cost function. The surface orientation is independent upon scalar component, distance value. This means that we can reconsider whether or not to preserve them as the amount of vector component, although they are elements with low scalar values. In addition, we develop a simplification algorithm based on half-edge collapse manner, which use the proposed cost function as the criterion for removing elements. In half-edge collapse, using one of endpoints in the edge represents a new vertex after collapse operation. The approach is memory efficient and effectively applicable to the rendering system requiring real-time transmission of large-volumed surface data.

Key words: 메쉬 간략화, 특징점, 방향성, Normal

[†] 준회원, 연세대학교 컴퓨터과학과 멀티미디어 연구실

^{††} 종신회원, 연세대학교 컴퓨터과학과 멀티미디어 연구실

1. 서론

다각형 표면데이터를 이용하여 객체를 표현하는 기법은 컴퓨터 그래픽스 분야에서 3D 모델을 디스플레이 하기 위해 매우 광범위하게 이용하고 있는 방법이다. 3D 객체를 화면상에 표현하는데 있어 형상이 복잡하거나 표면이 일정치 않은 자유곡면 형태를 갖을 경우 이를 수학적으로 정의하여 표현하는 것이 불가능하다. 따라서 객체의 외부 표면을 수많은 다각형 조각(polygonal patch)들로 나누고 각각의 면을 구성하는 정점과 연결선의 좌표로서 객체를 표현한 후 연속된 다각형 조각을 이용하여 실 객체의 표면에 근사(approximation) 시키는 방법을 이용하게 된다. 3D 레이저스캐너나 CAD 시스템을 통하여 생성된 객체의 표면데이터는 일반적으로 매우 복잡한 표면 구성과 대용량의 데이터로 구성된다. 객체의 표면을 근사 시키기 위한 다각형 조각들은 그 양이 많을수록 원본 모델에 대한 정확도를 높여 사실감을 증가시킬 수 있지만 실행시간의 효율을 더 중요시하는 시뮬레이션이나 실시간 응용시스템 등의 분야에서는 필요 이상의 데이터로 인해 오히려 문제가 될 수 있다.

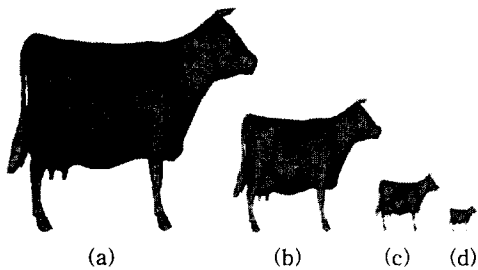


그림 1. 시점과의 거리에 따른 모델 크기의 변화
 ((a): A, (b): $\frac{1}{4}A$, (c): $\frac{1}{16}A$, (d): $\frac{1}{64}A$)

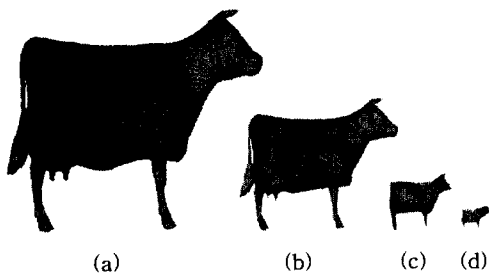


그림 2. 시점과의 거리에 따른 간략화 수행 결과
 ((a): 100%(5804t), (b): 9%(500t), (c): 2%(100t), (d): 0.5%(30t))

그림 1은 시점(viewpoint)과 객체간의 거리에 따른 모델크기의 변화를 보인 것이다. (a), (b), (c) 및 (d) 모두 5804개의 삼각형을 이용하여 표현한 것으로서 크기가 작아질수록 각 모델의 자세한 정도를 시각적으로 거의 인식할 수 없음에도 불구하고 이들은 모두 같은 수의 삼각형을 이용하여 표현된 것들이다. 반면에 그림 2의 경우는 시점과의 거리에 따라서 간략화를 수행한 것으로서 훨씬 적은 수의 삼각형을 이용하여 표현하였지만 원래 모델과의 시각적인 차이가 거의 없다. 만일 실시간 렌더링을 요하는 응용시스템에서 그림 1에서의 (d)와 같은 경우가 많을 경우 렌더링 부하로 인해 상당한 시스템 성능의 저하를 초래하게 된다.

메쉬 간략화는 다각형으로 이루어진 3D 표면모델을 일정 수준의 정확도를 유지하면서 상대적으로 적은 수의 다각형을 이용하여 근사(approximation) 시키는 기술로서 시스템의 성능 혹은 응용분야의 특징에 따라 간략화 모델의 다각형 수에 따른 시각적인 정확도를 선택할 수 있게 한다. 즉 시뮬레이션이나 가상현실 시스템과 같이 실시간 렌더링 효율을 중요시 하는 분야에서는 모델의 정확도는 상대적으로 중요하지 않으므로 적은 수의 다각형을 이용하여 모델을 표현하고 모델의 사실감을 중요시 하는 응용분야에서는 시스템의 성능을 떨어뜨리지 않는 수준에서 상대적으로 많은 양의 다각형을 이용하여 모델을 표현하게 된다.

그 동안 메쉬 간략화와 관련하여 많은 알고리즘들이 제안되었다. 기존 알고리즘에서 사용하는 간략화 비용 함수들은 대부분 거리를 주 요소로 하여 이의 최적화에 초점을 맞춘다. 즉 간략화 전후 모델간의 거리를 최소화 할 수 있도록 간략화 대상객체를 선택하고 객체의 제거 후에 새로운 정점의 위치를 지정한다. 그러나 기본적으로 스칼라 값인 거리요소 만으로는 메쉬의 지역적인 특징을 정확히 정의하기 어렵고 간략화로 인해 발생하는 기하학적 정보의 손실 정도를 보장하지 못한다. 따라서 간략화 단계를 높일수록 원래의 세부적인 모양을 잃어버리는 단점이 있다. 이는 메쉬의 특징 요소들이 대부분 작은 스칼라 값을 가지고 좁은 지역에 밀집해 있기 때문에 제거비용이 낮아 간략화 후반까지 보존되지 않기 때문이다. 지역 표면의 방향요소는 매우 중요한 기하학적 정보로서 거리와 같은 스칼라 양에 비의존적이다. 따라서 이를

간략화 비용함수의 요소로 고려함으로써 원래 모델의 특징 요소들을 비교적 간략화 후반까지 보존할 수 있다.

본 논문에서는 표면의 방향요소와 같은 벡터 성분을 비용함수 요소로서 이용한다. 제안된 비용함수는 파격적인 수준의 간략화 수행 후에도 원래 모델의 세부적인 모양을 보존할 수 있게 한다. 또한 제안된 비용함수를 바탕으로 반-에지 붕괴(half-edge collapse)에 기반한 간략화 알고리즘을 개발한다. 본 논문은 실시간 렌더링 시스템에 적용 가능한 간략화 모델의 생성에 초점을 맞춘다. 따라서 메쉬를 점진적으로 렌더링 할 때 기존의 정점을 계속 이용할 수 있게 하는 반-에지 붕괴에 기반한 접근 방식은 렌더링에 드는 시간비용을 줄일 수 있고 저장공간 상의 이점이 있으며 특히 대용량 표면데이터의 실시간 전송에 매우 효과적이다.

2. 관련연구

간략화 알고리즘은 주로 간략화 대상 객체를 무엇으로 하는지에 따라 정점 클러스터링(Vertex Clustering), 정점 제거(Vertex Remove), 에지 붕괴(Edge Collapse) 및 면 붕괴(Face Collapse) 방식으로 구분할 수 있다. 본 절에서는 간략화 알고리즘에서 대상 객체를 선택하는 기준이 되는 제거 비용함수의 요소에 대해 기존의 알고리즘에서 사용하는 기법들에 대해서 설명한다.

① 정점 클러스터링(5, 16)

메쉬를 평면에 투영(projection) 시켰을 때 기하적으로 충분히 작은 지역에 밀집해 있는 정점들은 그룹화 하여 하나의 대표정점으로 표현하는 것이 기본 아이디어이다. [16]에서는 메쉬의 각 영역을 일정한 크기의 격자로 나누고 해당 격자 단위로 정점들을 병합시킨다. [5]에서는 격자구조가 아닌 Octree를 이용하여 메쉬를 삼차원적으로 분할하는 방법을 사용하였다. 클러스터링 방법은 일반적으로 간략화 수행 시간이 매우 빠른 반면 메쉬의 위상정보를 보존하지 않음으로 인해 결과 메쉬의 정확도가 떨어지는 단점이 있다.

② 정점제거(7, 12, 13)

비교적 간략화 연구 초기의 방법으로서 간략화 단위는 정점이며 정점이 제거된 후 생긴 구멍(hole)을 재삼각화(retriangulation) 하는 추가적인 과정이 필요하다. [7]에서는 각 정점에서의 지역곡률을 비용으로 할당하고 이를 이용하여 정점을 제거한다. [12]에서는 각 정점마다 인접 정점을 이용하여 구성된 평균 평면과의 거리를 계산하고 정해진 임계치보다 작을 때 이 정점을 제거한다. 정점제거 방법은 간략화 대상객체의 선택방법이 직관적이지만 간략화 결과의 정확도가 정점의 제거 후에 재삼각화 방법에 매우 의존적이다. 따라서 [13]에서와 같은 엄격한 삼각화 과정이 필요하다.

③ 에지 붕괴(1, 2, 8, 9, 14)

가장 대표적으로 사용되는 간략화 방법으로서 간략화 후 두개의 정점을 하나의 정점을 이용하여 표현한다. 이 점은 원래 메쉬가 갖는 정점의 부분 집합일 수도 있고 전혀 새로운 점일 수도 있다. [1,2]에서는 정점과 이와 관련된 평면간의 평방거리의 합을 벡터적으로 해석한 QEM (Quadric Error Metric)을 각 에지의 제거비용으로 사용한다. [8,9]에서는 각 에지에 대해 자신의 붕괴 비용인 에너지 함수를 정의하여 이에 따라 에지를 순차적으로 제거한다. 에너지 함수는 원래 모델에서 샘플링 된 점과 간략화 후 메쉬간의 거리를 주된 요소로서 정의한다. [14]에서는 에지의 각 정점과 새로운 정점간의 거리 중 최대 거리를 비용함수로 사용한다. 새로운 정점의 위치는 먼저 에지의 인접 삼각형들로 구성된 평면을 2D 상에 투영시켜 후보 정점위치를 정하고 다시 3D로 역 투영 시켰을 때 에지에 인접한 삼각형과 후보 정점에 인접한 삼각형간의 거리를 최소화 할 수 있도록 정한다. 에지붕괴에 기반한 간략화 방법은 정점 제거방법에서의 재삼각화 과정은 필요하지 않지만 에지의 붕괴 후에 생성되는 정점에 대한 배치방법을 정해야 한다. 에지의 양 끝점 중 하나를 이용하거나 이의 중점 혹은 최적화 된 위치에 정점을 각각 배치시킬 수 있다. 많은 알고리즘들이 간략화 에러를 줄이기 위해 최적화 방법을 이용하고 있다. 최적의 위치를 찾는 기준은 간략화 전후 메쉬간의 거리량을 이용한다. 이 방법은 원래 모델에 없는 새로운 정점을 생성해 냄으로

써 저장공간 및 네트워크 환경 등에서 데이터의 전송 비용이 크다. 중점을 이용하는 방법의 경우도 역시 위 최적화 방법에서와 같은 단점이 있다. 또한 원래 객체가 불룩한 모양을 갖을 경우 간략화를 수행할수록 계속해서 부피가 작아지는 단점이 있다. 에지의 양 끝점 중 하나를 이용하는 기법은 위 두 가지 방법의 단점을 보완할 수 있는 방법이다. 또한 최적의 정점 재배치 방법을 사용하지 않지만 대부분의 경우에 원래의 모양을 잘 유지한다[4].

④ 면 붕괴(6,15)

간략화 단위는 면 객체로서 간략화 시 삼각형이 갖는 세 개의 정점이 하나로 병합된다. [6]에서는 각 삼각형에 대해 삼각형의 면적요소를 기반으로 지역 곡률 요소, 삼각형의 내각 요소, 인접 삼각형의 수적 균형을 조절하기 위한 위상 요소, 간략화 전후의 예측 에러 요소 등을 이용하여 비용함수를 정의한다. [15]에서는 삼각형의 세 정점 각각의 곡률을 간략화 비용함수의 요소로서 정의한다. 면 붕괴에 기반 한 간략화 방법은 한번의 간략화 연산으로 인해 제거되는 기하데이터의 양이 많은 반면 간략화 비용을 적절히 계산하지 못했을 경우 매 간략화 단계마다 에러가 누적되어 결과 메쉬의 정확도가 떨어질 수 있다.

본 논문의 목표는 파격적인 간략화 수행 후에 원래 모델의 특성정보의 보존 및 실시간 렌더링 시스템에 적용 가능한 간략화 모델의 생성이다. 따라서 위의 방법 중 에지붕괴에 기반 한 간략화 기법의 하나인 반-에지 붕괴 방법이 가장 적당하다. 이는 에지의 양 끝점 중 하나를 이용하여 새로운 정점을 표현하는 방법이다. 본 논문에서는 각각의 에지에 대해 방향성을 부여하여 이를 통하여 붕괴 연산 후의 지역 평면들을 일시적으로 재구성하고 간략화 에러를 예측할 수 있게 한다. 또한 방향성 에지를 이용하면 에지의 붕괴연산 후에 지역 메쉬의 재구성 과정이 매우 직관적으로 처리될 수 있다. 간략화 에러의 예측에 대해서는 3장에서, 보다 구체적인 반-에지 붕괴 연산에 대해서는 4장에서 각각 설명한다.

3. 기하학적 특성정보의 보존

간략화 기법들은 일반적으로 탐욕(Greedy) 알고리즘에 기반 하고 있는데 간략화 단계를 수행함에

따라 지역적으로 발생된 간략화 에러는 전체적으로 파급되어 간략화 결과 메쉬의 정확도에 영향을 미치게 된다. 따라서 간략화 결과의 기하학적인 정확도와 시각적인 정확도를 모두 높이기 위해서는 매 간략화 단계마다 메쉬의 지역적인 특징정보를 유지할 수 있어야 한다. 예를 들어, 피라미드의 꼭대기에 있는 정점이나 직육면체의 모서리에 있는 에지는 모두 모델의 정확도에 많은 영향을 미치는 요소들이므로 간략화 대상객체로서 적당치 못하다.

본 논문에서는 에지 붕괴를 기본 간략화 연산으로 하여 매 간략화 단계마다($M_i \rightarrow M_{i+1}$) 간략화 비용을 최소화 하는 에지를 반복적으로 제거 함으로써 전체적인 간략화 에러를 줄인다. 식 (1)은 메쉬 M_i 에서 각각의 에지 e_i 에 대한 비용함수 $f_{cost}(e_i)$ 를 표현한 것으로서 간략화 전후의 기하학적 변화량(f_i)과 현재 메쉬의 지역곡률(f_c)을 세부 비용요소로서 포함한다.

$$f_{cost}(e_i) = k(w_1 f_i(M_i, M_{i+1}) + w_c f_c(M_i)) . \quad (1)$$

간략화 전후의 기하학적 변화량을 측정하기 위한 요소로서 본 논문에서는 거리와 방향성을 이용하는 데 이에 대해서는 이후에 다시 설명한다. w_1 와 w_c 는 기하학적 변화량과 지역곡률 각각에 대한 실험적인 가중치로서 표면 데이터의 특징에 따라 적절히 지정한다. 예를 들어, 지형(terrain)이나 구(sphere)의 표면은 서로 다른 기하적 특성을 갖으므로 가중치를 적절히 지정해 줌으로써 각각의 요소들이 비용함수에 미치는 정도를 조절할 수 있게 한다. k 는 면 붕괴를 기본 간략화 연산으로 하는 [6]에서의 면(삼각형)의 면적과 비슷한 요소로서 에지 붕괴에 기반 한 본 논문에서는 에지의 길이(Euclidian length)를 의미한다. 큰 길이를 갖는 에지는 작은 길이를 갖는 에지에 비해 지역의 곡률이나 간략화 전후의 변화량이 상대적으로 크기 때문에 간략화 비용을 크게 함으로써 간략화 시점을 뒤로 미룰 수 있다.

본 절에서는 간략화 시 대상객체의 선택을 위한 비용함수를 설계하는데 있어 메쉬의 지역적 특징정보를 반영하기 위한 식(1)의 세부 비용함수 요소들에 대해서 설명한다.

3.1 거리 기반 비용함수

에지 붕괴에 기반 한 간략화 기법에서 우선적으로 생각해 볼 수 있는 간략화 대상 객체의 선택방법은

현재 메쉬 M_i 에서 가장 작은 길이의 에지 $\min(\lambda(u, v))$ 를 선택하는 것이다(식(2)).

이는 짧은 에지의 경우 지역의 곡률에 미치는 영향이 적어 상대적으로 덜 중요하다[3]는 의미로서, [5]와 같은 정점 클러스터링 기법에서 특정 크기의 Octree 셀 내에 있는 정점들을 하나의 정점으로 간략화 하는 개념과 비슷하다.

$$\lambda_{u,v \in M_i}(u, v) = \|u - v\| \quad (2)$$

매우 빠르고 간단한 간략화 방법을 원할 경우 실제로 위의 식(2)와 같은 비용함수만을 이용해도 어느 정도 원래의 모델과 비슷한 수준의 간략화 모델을 생성해 낼 수 있다. 특히 대부분의 지역이 굴곡이 적고 완만한 표면을 갖는 메쉬의 경우 복잡한 수식의 비용함수를 사용하는 것 보다 더 효과적일 수 있다. 그러나 일반적인 경우에 간략화 객체를 선택하는데 있어 에지의 길이만을 고려했을 경우 작은 길이의 특징 점 요소들이 간략화 초기에 모두 제거되고 그림 3에서와 같이 서로 다른 지역적인 특성을 갖는 메쉬에 대해 모두 같은 간략화 비용을 갖기 때문에 결과 메쉬의 정확도가 떨어지게 된다.

이러한 문제는 간략화 과정 전후($M_i \rightarrow M_{i+1}$)에 에지의 붕괴로 인한 기하데이터의 변화량 정도를 구분할 수 없기 때문이다. 이는 정점 u_i, u_2, u_3 에 인접한 면 요소를 고려함으로써 해결할 수 있다. 그림 4에서 메쉬 M_i 의 정점 u 에 인접한 면 f 는 에지 (u, v) 에 인접한 면을 제외하고 M_{i+1} 단계에서 새로운 정점 v' 을 갖게 된다. 이들 면과 새로운 정점 v' 간의 거리는 식(3)에 의해 구할 수 있다. 식에서 v_f 는 면 f 의 세 정점 중 임의의 한 점이고 n_f 는 f 의 법선 벡터이다. 이때 정점 v' 의 위치를 선택하는 방법은 간략화 과정에서 새로운 정점의 재배치 전략에 따른 문제로서 반-에지 붕괴 연산에 기반 하는 본 논문에서는 방향성 에지의 끝 정점인 v 와 일치한다.

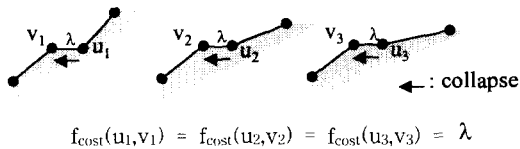


그림 3. 서로 다른 지역적 특성을 갖지만 에지의 길이가 같은 경우(2D)

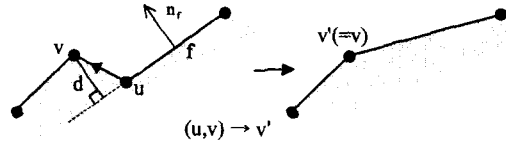


그림 4. 정점-면간의 거리

$$d(f, v') = \left| \frac{\overrightarrow{v'v_f} \cdot n_f}{|n_f|} \right| \quad (3)$$

여기서 n_f 를 단위 법선(unit normal) 벡터라고 했을 때 $|n_f|=1$ 이므로 위 식은 다시 식(4)와 같이 쓸 수 있다.

$$d(f, v') = \left| \overrightarrow{v'v_f} \cdot n_f \right| \quad (4)$$

식(4)를 이용하여 정점 u 에 인접한 면에 대한 간략화 전후의 거리 변화량을 식(5)와 같이 정의한다. 여기서 F_u 는 정점 u 에 인접한 면의 집합이고 F_{uv} 는 에지 (u, v) 에 인접한 면의 집합이다.

$$\sum_{f_j \in \{F_u - F_{uv}\}} (d(f_j, v')) \quad (5)$$

간략화 전후 정점-면간의 거리의 변화량을 고려한 식(5)도 여전히 문제를 가지고 있다. 그림 5에서 (a)와 (b)는 기본적으로 스칼라 성분인 거리요소만을 비용요소로서 고려했을 때의 문제를 보여주고 있다. 즉 메쉬의 기하학적 특징 요소들은 대부분 (b)와 같

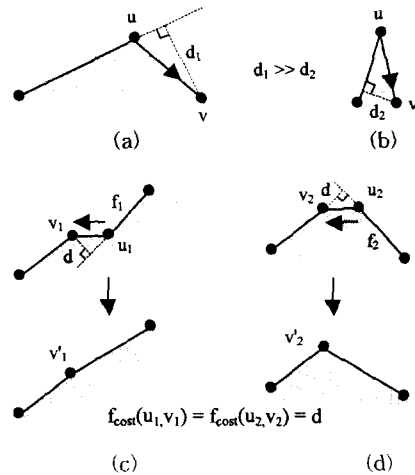


그림 5. 거리 기반 비용함수의 문제

이 작은 스칼라 값을 가지기 때문에 붕괴비용이 상대적으로 작다. 따라서 이러한 특징 요소들은 간략화 초기에 제거될 가능성이 높다. (c)와 (d)의 경우는 서로 다른 기하학적 특성을 갖는 지역이지만 거리에 기반 한 간략화 비용이 같은 예를 보여주고 있다. 그러나 이 둘 중 어느 쪽을 먼저 간략화 하는지에 따라서 결과 메쉬의 형태가 많이 달라지게 된다. 따라서 이러한 각각의 경우에 대해 서로 다른 간략화 비용을 생성하는 추가적인 방법이 필요하다.

3.2 면의 방향성을 고려한 비용함수

앞서 보았듯이 에지의 간략화 비용함수를 설계하는 데 있어 단순히 길이나 간략화 전후 면의 기하학적 거리와 같은 스칼라량만을 고려할 경우 좋은 품질의 간략화 메쉬를 생성할 수 없다. 이는 메쉬의 특징 요소들은 대부분 좁은 지역에 작은 크기의 기하학적인 요소로서 형성되기 때문에 초기 간략화 단계에서 먼저 제거되기 때문이다. 이러한 작은 스칼라량을 갖는 특징 요소들을 비교적 간략화 후반까지 보존하기 위한 방법은 지역의 방향성과 같은 벡터적인 성분을 고려하는 것이다.

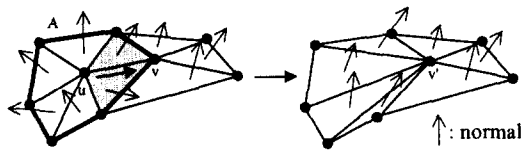


그림 6. 에지(u,v)의 붕괴로 인한 면의 방향 변화

그림 6에서와 같이 각각의 면들은 자신의 방향성을 가지고 있는데 간략화 후에 각 면의 방향이 변하는 것을 볼 수 있다. 따라서 A지역에 대한 현재 메쉬 M_i 에서의 곡률과 간략화 연산 후의(M_{i+1}) 방향 변화량을 계산하면 그 크기에 따라 이 지역을 보존 할 것인지의 여부를 선택할 수 있다. 간략화 지역 A에 대한 M_i 에서의 곡률함수 C 및 M_{i+1} 에서의 방향 변화량 함수 V 는 각각 식(6)과 식(7)과 같이 정의한다.

$$C(u,v) = \sum_{f_i \in \{F_u - F_v\}} \left(\min_{f_k \in \{F_{in}\}} \left((1 - n_{f_i} \cdot n_{f_k}) / 2 \right) \right) \quad (6)$$

$$V(u,v) = \sum_{\substack{f_i \in \{M_i \cap \{F_u - F_v\}\} \\ f_k \in M_{i+1}}} (1 - n_{f_i} \cdot n_{f_k}) / 2 \quad (7)$$

식(6)은 [3]에서의 곡률함수를 변형한 것으로서 정점 u 에 인접한 각각의 면에 대해 에지(u,v)에 인접한 면과의 내적을 계산함으로써 지역 A 의 곡률을 구한다. 또한 식(7)을 이용하여 에지(u,v)에 인접한 면을 제외한 나머지 면들이 간략화 후에 얼마만큼 방향이 변화 했는지를 계산한다.

지금까지 식(1)의 세부 비용 요소들에 대해 설명하였다. 즉, 식(2)는 식(1)에서의 k 에, 식(5)와 식(6)은 f_t 에, 식(7)은 f_c 에 각각 대응된다. 본 논문에서는 앞서 설명한 세부비용 요소들을 포함하는 식(1)을 에지(u,v)의 붕괴 비용함수로서 사용한다. 반-에지 붕괴에 기반 한 간략화 연산 시 기하데이터의 스칼라 및 벡터적 변화는 그림 6의 A지역에만 국한되어 일어나므로 식(1)을 이용하여 각 에지에 대한 붕괴 비용을 계산한 후 매 간략화 단계마다 최소비용의 에지를 반복적으로 제거한다.

4. 간략화 알고리즘

일반적으로 간략화 알고리즘에서는 제거 대상 요소의 선택 기준과 새로운 정점의 재배치 방법에 대해 제시하여야 한다. 또한 간략화 과정에서 변경될 수 있는 원본 메쉬의 위상적인 특징들에 대해서 적절한 대처 방안이 마련되어야 한다. 본 절에서는 이러한 각각의 사항들과 본 논문에서 구현한 알고리즘의 전체적인 개요에 대해서 설명한다

4.1 알고리즘 개요

알고리즘은 크게 초기화 과정과 간략화 수행과정으로 나눌 수 있다. 먼저 초기화 과정에서는 입력된 원본 메쉬의 각 기하학적 요소를 알고리즘의 내부 자료구조에 저장하고 모든 에지에 대한 초기 간략화 비용을 계산하여 우선순위 큐(heap)에 저장한다. 간략화 수행과정에서는 사용자가 정한 결과 정점 혹은 삼각형의 수를 만족할 때 까지 반복적으로 간략화를 수행하게 되는데 매 단계마다 에지의 붕괴로 인한 인접지역 에지의 비용을 새로 계산하고 큐를 동적으로 갱신 시킨다. 아래에 개략적인 알고리즘의 수행과정을 설명한다.

- 초기화 과정
 - ① 입력된 메쉬 데이터를 읽어 들임

- ② 정점에 인접한 정점 혹은 삼각형을 계산하여 각 정점에 저장
- ③ 모든 정점에 대해 인접 정점과의 간략화 비용을 계산
- ④ 최소 비용의 대상 정점 v를 u의 대상 정점으로 저장
- ⑤ 간략화 비용을 키(key)로 각 정점을 힙에 저장

• 간략화 수행 과정

- ① 힙으로부터 최소비용의 에지를 가져옴
- ② 정점 u에 인접한 정점들을 T에 복사
- ③ 에지(u,v)에 인접한 삼각형 t(x, y, z)를 제거
- ④ t의 세 정점 x, y, z 각각의 인접 삼각형 리스트에서 t를 제거
- ⑤ u에 이웃 한 삼각형 각각에 대해 u를 v로 대체하고
- ⑥ T 내의 모든 정점에 대해 인접 정점과의 간략화 비용을 계산
- ⑦ 힙 갱신
- ⑧ 목표 정점 수 혹은 삼각형수를 만족할 때까지 ①~⑦을 반복

4.2 간략화 객체의 선택

원본 메쉬의 모든 에지들은 식(1)에 의해 붕괴비용이 계산된 후 크기에 따라 오름차순으로 힙에 저장된다. 또한 매 간략화 단계마다 에지의 붕괴로 인해 영향 받는 인접 에지의 붕괴비용이 재계산되어 동적으로 힙을 갱신 시킨다. 따라서 힙의 맨 상위노드에는 항상 최소비용의 에지가 저장되어 각 간략화 단계마다 제거 대상으로서 선택된다.

에지의 붕괴비용을 계산하는데 있어서 고려해야 할 사항은 기하데이터의 잡음(noise)이다. 예를 들어 간략화 전후 거리의 변화량을 계산하는 데 있어 거리의 합(식(5)) 대신에 다음의 식(8)과 같이 최대거리를 이용할 수도 있다. 이 중 어느것을 이용할지는 간략화 결과 메쉬의 정확도에 따라 실험적으로 선택하거나 표면데이터의 특성에 따라 정할 수 있다. 식(8)의 경우 일반적으로 기하데이터의 잡음에 민감하게 반응함으로써 정확도를 떨어뜨릴 수 있다.

$$\max_{r_i \in \{r_u, r_v\}} (d(r_i, v')) \tag{8}$$

4.3 새로운 정점의 배치

에지(u,v)의 붕괴 후에 새로운 정점 v'의 위치 선택방법으로서 가장 간단한 것은 u, v의 중점을 선택하는 것이다. 좀 더 고품질의 간략화 메쉬를 원할 경우 [8,9]에서와 같이 정점 u, v를 포함하는 가상의 윤곽선 상에 최적화 된 위치를 선택할 수도 있다. 그러나 중점 선택방법은 객체가 불룩한 모양인 경우 간략화 단계를 거칠수록 객체가 작아지는 단점이 있다. 또한 최적의 위치를 선택하는 방법의 경우 정점의 위치를 미리 계산해 둔다면 메모리가 많이 필요하고, 알고리즘적으로 결정한다면 추가적인 시간비용을 부담해야 한다.

본 논문에서는 에지의 양 끝점 중 하나를 선택하여 새로운 정점으로서 이용하는 반-에지 붕괴 방법을 사용한다. 이는 M_i 에서의 두 정점 중 하나를 M_{i+1} 에서도 그대로 이용하는 것이므로 추가적인 메모리 부담도 없고 시간도 빠르며 중점 선택방법에서와 같은 객체의 부피가 급격히 줄어드는 문제도 없다. 또한 대부분의 경우에 원래 모양을 제대로 유지한다[4].

4.4 메쉬의 위상적 특징의 보존

대부분의 간략화 알고리즘에서는 원본 메쉬로서 다면체(manifold)를 요구한다. 임의의 다각형 표면이 다면체이기 위해서는 경계에 있지 않은 모든 에지는 정확히 두개의 인접한 면을 가져야 하며(경계 에지의 경우 한 개), 경계에 있지 않은 모든 정점들은 인접한 면들에 의해 닫혀있어야(closed loop) 한다(경계 정점의 경우 인접 면으로 이루어진 하나의 부채꼴 모양). 이 중 한가지 조건이라도 위배될 경우는 비다면체(non-manifold)이다. 즉, 그림 7에서 (a)와 (b)는 다면체이고 (c)와 (d)는 다면체가 아니다. 탐욕 기법에 기반하고 있는 대부분의 간략화 알고리즘은 각 간략화 단계에서 비다면체 표면을 만들어 내기가 쉽다. 따라서 이를 세밀히 조사하기 위한 시간부담이 크고 알고리즘의 복잡도도 증가한다[1]. 또한 간략화 연산 시 다면체에 대한 제약조건은 결과 메쉬의 정확도를 떨어뜨릴 수 있다. 즉, 붕괴비용이 작은 요소라도 붕괴연산의 결과가 비다면체를 생성할 경우 이 요소의 제거를 허락하지 않기 때문이다. 간략화 수행 과정에서 비다면체의 부분적인 허용은 좀 더 융통성 있는 간략화 연산을 가능하게 한다. 따라서 본 논문

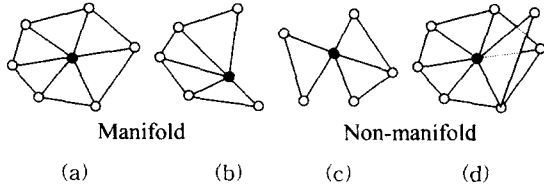


그림 7. Manifold와 non-manifold

에서는 이러한 점들을 고려하여 (c)나 (d) 같은 형태의 부분적인 비다면체를 허용한다.

지형과 같이 명백한 경계를 갖는 메시는 경계지역의 기하요소가 제거되었을 때 간략화 결과 메시의 모양이 매우 일그러지게 된다. 따라서 경계에 위치한 요소에 대해서는 매우 높은 간략화 비용을 할당하여 간략화의 시점을 가능하면 뒤로 미루어야 한다. 본 논문에서는 에지(u,v)가 붕괴될 때 u, v가 모두 경계에 있을 때, u만 경계에 있을 때 그리고 v만 경계에 있을 때 등 모두 세가지 경우로 구분하여 각각 다른 간략화 비용을 할당한다. u만 경계에 있을 때는 매우 높은 간략화 비용을 할당한다.

에지(u,v)가 붕괴될 때 정점 u가 v로 이동하는 과정에서 인접한 면의 겹침(folding) 현상이 발생할 수 있는데 이의 결과로 인해 메시의 시각적인 질이 떨어질 수 있다(그림 8). 본 논문에서는 식(7)로부터 얻어진 식(9)의 값이 정해진 임계치를 넘을 때 높은 붕괴 비용을 할당함으로써 면의 겹침 현상을 제거한다. 이때 임계치 값은 실험적으로 정한다.

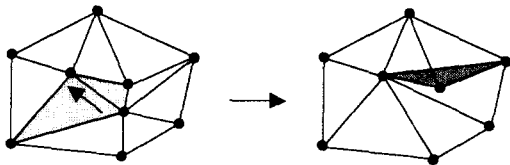


그림 8. 겹침(folding) 현상

$$\max_{\substack{f_i \in \{M_i \cap \{F_v - F_i\}\}, \\ f_i \in M_{i+1}}} ((1 - n_{f_i} \cdot n_{f_k}) / 2) \tag{9}$$

5. 실험 및 결과

본 논문의 알고리즘을 이용하여 간략화 한 결과 메시의 수치적인 에러율을 측정해 보고 시각적인 결과의 의미에 대해서 토의해 본다.

간략화 방법들은 동일한 비교기준을 적용하기가

어렵기 때문에 정성적이거나 정략적으로 객관적인 비교가 어렵다. 따라서 대부분의 간략화 알고리즘에서는 자체적으로 제작한 에러 메트릭(error metric)을 이용하여 에러율을 계산한다. 본 논문에서는 객관적인 에러의 측정을 위해 [10]에서 제안한 Metro 프로그램에 의하여 간략화 전후의 에러를 측정한다.

Metro를 이용하면 원본 메시에서 샘플링 된 임의의 점과 간략화 결과 메시의 점간의 거리를 측정하여 최대, 평균 및 평균평방 에러율을 계산해 낼 수 있다.

그림 9는 본 논문의 간략화 결과에 대한 최대, 평균 및 평균 평방 에러를 QSlim v2.0[2]과 비교한 것이다. 본 논문의 에러율은 식(1)의 w_t과 w_c 값을 변화시키면서 추출해 낸 값이다. 원본 메시에 대한 간략화율을 높일 수록 본 논문의 수치적인 에러율이 더 낮음을 볼 수 있다. 한가지 주목할 점은 QSlim의 경우 최적의 정점 재배치 방법을 사용하였는데도 반

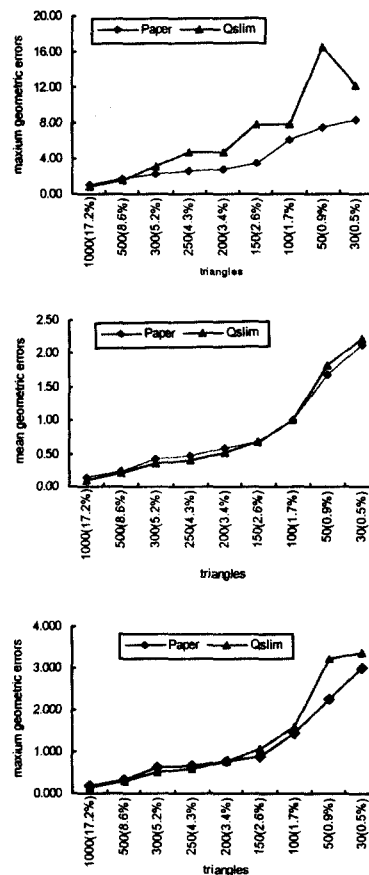


그림 9. 간략화 에러의 비교

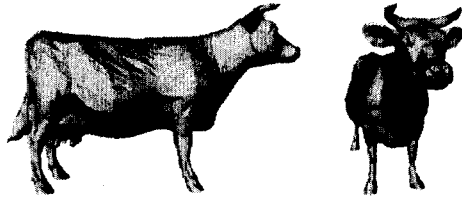


그림 10. Cow 모델(5804t)

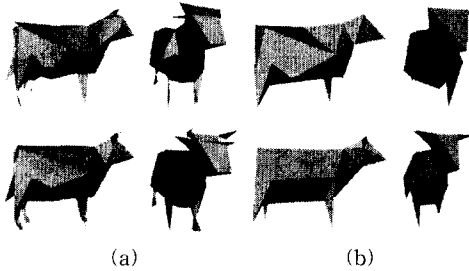


그림 11. 간략화 결과 모델의 시각적인 비교
(위: QSLim; 아래: 본 논문; (a): 150t(2.6%);
(b): 50t(0.9%))

-에지 붕괴에 기반 한 본 논문의 결과보다 에러 율이 높다는 것이다. 이는 거리와 같은 스칼라 양을 비용 함수의 주된 요소로 사용하는 QSLim에 비해 본 논문에서의 방향 변화량 및 곡률 요소와 같은 벡터 성분이 높은 수준의 간략화 율에서는 결과 메쉬의 정확도에 좀 더 많은 영향을 미치는 것으로 해석된다.

그림 11은 본 논문의 시각적인 간략화 결과를 비교한 것이다. 그림에서 QSLim은 앞서 설명하였듯이 최적화된 정점 배치 방법을 사용하기 때문에 간략화 결과 메쉬의 전체적인 볼륨이 비교적 잘 유지되고 있다. 그러나 곡률이 심한 지역(뿔, 귀, 머리, 다리 등)들은 본 논문의 결과에서 좀 더 잘 유지되고 있음을 볼 수 있다. 즉, 본 논문에서 제안한 간략화 비용 함수는 상대적으로 작은 곡률을 가지는 평평한 지역의 기하데이터를 잃어버리는 대신 높은 곡률 지역들은 간략화 후반까지 잘 보존할 수 있게 한다.

또한, (a)에서 QSLim의 경우 간략화 과정에서 생성된 비다면체로 인해 메쉬의 시각적인 질이 떨어짐을 볼 수 있다. 본 논문의 경우도 부분적인 비다면체를 허용하지만 실험에 의해서 대부분의 경우 이와 같이 눈에 띄는 비다면체는 생성되지 않았다.

6. 결 론

메쉬 간략화는 대용량의 다각형 표면으로 구성된

원본 메쉬에 대해 가능하면 정확 도를 유지하면서 메쉬의 수를 효과적으로 줄이는 방법에 초점을 두고 있다. 현재까지 정점, 에지, 삼각형 등을 기본 간략화 단위로 하는 많은 알고리즘들이 제안되었는데 이들이 갖는 공통적인 문제는 제거할 대상요소를 어떠한 기준으로 선택하는가 하는 것과 간략화 후 새롭게 생성되는 정점을 어느 위치에 두는가 하는 것이다. 첫번째 문제의 경우 간략화 메쉬의 에러 율을 줄여 정확 도를 높이기 위한 것으로서 효과적인 간략화 비용함수를 정의하는 문제로 해석된다. 본 논문에서는 간략화 과정에서 발생하는 기하적 변화량을 스칼라 및 벡터 성분으로 구분, 해석하고 이들 각각의 경우를 반영한 간략화 비용함수를 제안하였다. 새로운 정점의 배치방법으로써 본 논문에서는 반-에지 붕괴에 기반 한 간략화 기법을 사용하여 중점이나 최적화된 방법에서의 단점들을 제거하였다.

제안 된 비용함수의 실험 결과 99% 이상의 과격적인 간략화 율에서 본 논문의 수치적, 시각적인 간략화 결과가 우수함을 보였다.

참 고 문 헌

- [1] Michael Garland, "Quadric-Based Polygonal Surface Simplification", PhD Thesis, School of Computer Science CMU, 1999.
- [2] Michael Garland and Paul Heckbert, "Surface Simplification Using Quadric Error Metrics", Proc. SIGGRAPH '97, pp. 209-216, 1997.
- [3] Stan Melax, A simple, Fast, and Effective Polygon Reduction Algorithm, *Game Developer*, pp. 44-49, November 1998.
- [4] Mark DeLoura, *Game Programming Gems*, Charles River Media, 2001.
- [5] Luebke, David and Carl Erikson, "View-Dependent Simplification of Arbitrary Polygonal Environments", Proc. SIGGRAPH '97, pp. 199-208, 1997.
- [6] Tran S. Gieng, Bernd Hamann, Kenneth I. Joy, Greg Schussman, Issac J. Trotts, "Constructing Hierarchies for Triangle Meshes", IEEE Transactions on Visualization and Computer Graphics, Vol. 4, No. 2, pp. 145-161, 1998.

[7] Greg Turk, "Re-tiling Polygonal Surfaces", Proc. SIGGRAPH '92, pp. 55-64, 1992.

[8] H. Hoppe, "Progressive meshes", Proc. SIGGRAPH '96, pp. 99-108, 1996.

[9] H. Hoppe, T. DeRose, T. Duchamp, J. McDonald, W. Stuetzle., "Mesh optimization", Proc. SIGGRAPH '93, pp. 19-26, 1993.

[10] P. Cignoni, C. Rocchini and R. Scopigno, "Metro: measuring error on simplified surfaces", Computer Graphics Forum, 17(2), pp. 167-174, 1998.

[11] Peter Lindstrom, Greg Turk, "Evaluation of Memoryless Simplification", IEEE Transactions on Visualization and Computer Graphics, Vol.5, No.2, pp. 98-115, 1999.

[12] William Schroeder, Jonathan Zarge, William Lorensen, "Decimation of triangle meshes", Proc. SIGGRAPH '92, Vol.6, No.2, pp. 65-70, 1992.

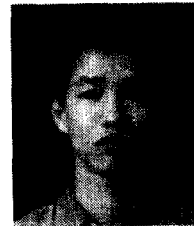
[13] K. J. Renze and J. H. Oliver, "Generalized Unstructured Decimation", IEEE Transactions on Visualization and Computer Graphics, Vol.16, No.2, pp. 24-32, 1996.

[14] Cohen, Jonathan, Dinesh Manocha, and Marc Olano. "Simplifying Polygonal Models Using Successive Mappings", Proc. IEEE Visualization '97, pp. 395-402, and 564.

[15] B. Hamann, "A data reduction scheme for triangulated surfaces", Computer Aided

Geometric Design, Vol.11, No.2, pp. 197-214, 1994.

[16] Jarek Rossignac and Paul Borrel, "Multi-resolution 3D approximations for rendering complex scenes", *Geometric Modeling in Computer Graphics*, B. Falcidieno and T. L. Kunii, Eds. Springer-Verlag, pp. 455-465, 1993.



고 명 철

1994년 제주대학교 정보공학과 (공학사)
 1997년 연세대학교 컴퓨터학과 (공학석사)
 1997년~현재 연세대학교 컴퓨터 과학과 박사과정

관심분야 : 3D 컴퓨터그래픽스, 실시간 렌더링, 가상현실, GIS



최 윤 철

1973년 서울대학교 전자공학과 (공학사)
 1975년 Univ. of Pittsburgh (공학석사)
 1979년 Univ. of California Berkeley Dept. of IE & OR (공학박사)

1979년~1982년 Lockheed사 및 Rockwell International 사 책임연구원
 1982년~1984년 Univ. of Washington 전산학과 박사과정
 1990년~1992년 Univ. of Massachusetts 연구교수
 1984년~현재 연세대학교 컴퓨터과학과 교수