전역 오류에 기반을 둔 삼각형 메쉬의 병합방법

안정환, 호요성
광주과학기술원 정보통신공학과
광주광역시 광산구 쌍암동 572 번지

A Mesh Merging Algorithm based on Global Errors

Jeong-Hwan Ahn and Yo-Sung Ho
Dept. of Information & Communications, K-JIST
572 Ssang-am Dong Kwang-san Gu, Kwang-ju, Korea
jhahn@gogh.kjist.ac.kr

Abstract

In recent days, applications using 3D objects are increasing rapidly. Since the 3D model contains a huge amount of information, a compact representation of the 3D object is necessary for efficient storage and transmission . In this paper, we propose an idea for generating a hierarchy of LOD (level-of-detail) approximations for a given 3D polygon. By applying multiple passes over triangle meshes and using local and global errors, we can control approximation errors accurately and treat multi-resolution easily.

1. 서론

최근 실제 사물을 가상 공간상에서 표현하기 위한 복잡한 3 차원 물체의 응용이 급증하고 있다. 기존의 MPEG-1 과 MPEG-2 표준은 자연적인 오디오와 비디오 데이터를 저장하거나 전송하는데 역점을 두었지만, MPEG-4 SNHC (Synthetic and Natural Hybrid Coding)에서는 컴퓨터 그래픽으로 만든 3 차원적인 물체를 이용하여 좀더 효과적이고 실감나는 영상을 표현하는데 중점을 두고 있다. 일반적으로 3 차원 물체는 삼각형 그물 구조(Mesh)로 나타내는데, 이러한 삼각형 데이터의 수는 보통 10^5 ~ 10^7 정도로 그 데이터의 양이 엄청나게 많아, 이러한 3 차원 데이터를 그대로 전송/저장,렌더링하는데 많은 문제점이 있다. 따라서 이런 문제점을 해결하기 위해 3 차원 물체를 이루는 삼각형 메쉬의 수를 줄여가면서 서로 다른 해상도를 지니는 다중 해상도 변이 (Level-of-Detail, LOD)가 필요하다. 본 논문에서는 지금까지 제안된 여러 가지 방법들의 장단점을 살펴본 뒤, 전역오류에 기반을 둔 삼각형 메쉬의 단순화 방법에 대한 삼각형 메쉬의 병합방법을 제안하고, 실제 VRML[1] 데이터로 LOD 의 실험 결과를 보였다.

2. 삼각형 메쉬의 단순화 방법

삼각형 메쉬를 단순화시키기 위해 Coplanar Facets Merging, Vertex/Edge/Face Decimation, Energy Function Optimization, Re-Tiling, Vertex Clustering, Multiresolution Analysis 등의 다양한 방법들이 제안되었다.

Coplanar Facets Merging 방법은 삼각형 메쉬들 중에서 같은 평면이나 비슷한 평면에 있는 면을 찾아 더 큰 다각형으로 만드는 방법이다 [2]. Re-Tiling 방법은 원래의 메쉬 표면에 불규칙하게 새로운 꼭지점들을 삽입하여 최대 곡률 위치에 옮겨지고 원래의 점들은 제거되거나 새로운 삼각형으로 만들어지는 방법이다 [3]. Energy Function Optimization 방법은 메쉬의 에너지 함수를 정의하여 이 함수가 최소가 되도록 꼭지점을 제거하거나, 삼각형 에지(Edge)의 붕괴(Collapse), 바꾸기(Swap)등을 하는 방법이다. 그러나 수행시간이 많이 걸리고 원래 메쉬의 점들이 유지되지 못하는 단점을 가지고 있다 [4]. Vertex Clustering 방법은 위상학적인 근접에 기반을 두고, 꼭지점들을 그룹으로 묶어서 하나의 새로운 꼭지점으로 대치하는 방법인데 Topology 와 Shape 가 보존되지 못하는 단점이 있다 [5]. Mesh Decimation 은 메쉬의 국부적인 해석을 통해 제거될 수 있는 점들을 판단하여 제거한 후 새로운 삼각형으로 만드는 방법이다 [6]. Mutiresolution Analysis 방법은 메쉬의 표면을 재메쉬화(Re-Meshing), 재표본화(Re-Sampling)하고 웨이블렛 계수를 이용한 부분할 방식으로 다양한 해상도로 표현하는 방법이다 [7].

위의 방법들 중 Mesh Decimation 방법은 다른 방법에 비해 수행 시간이 빠르며 예리한 에지(Sharp Edge)나 각도(Angle)는 보존을 하고 원래 메쉬 점의 위치가 변하지 않는다는 장점이 있다. 그러나 지역 오류만 고려하므로 이 방법으로 인해 생기는 근사화된 오류를 사용자가 제어할 수 없고, 우리가 원하는 LOD 를 쉽고 정확하게 표현할 수 없다. 따라서 본 논문에서는 지역 오류뿐만 아

니라 전역 오류를 고려하여 이런 문제점을 해결하였다.

3. 전역 에러에 기반을 둔 메쉬 병합 알고리즘

삼각형 메쉬는 점들을 연결하는 에지를 따라 삼각형으로 구성된 부분적인 선형 표면이다. 따라서 각 점들의 좌표 값과 그들의 연결성 정보가 아주 중요하다. 보통 삼각형 메쉬는 다음과 같이 정의할 수 있다.

$$M(V,C):\ V=\{v_1,v_2,\ldots,v_n\},\ v_i\in R^3$$
$$C=\{i,j,k\} \qquad (1)$$

즉 메쉬 $M(V,C)$에서 V는 점들의 위치로 메쉬의 모양을 정의하며, C는 이런 점들의 연결성 정보로 메쉬의 위상 형태를 정의한다. 이러한 메쉬 M에 다음과 같이 메쉬 병합 방법을 계속하여 적용하면 우리가 원하는 LOD의 메쉬를 얻을 수 있다.

$$M^0 \xrightarrow{\text{Mesh Merging}} M^1 \xrightarrow{\text{Mesh Merging}} \cdots \xrightarrow{\text{Mesh Merging}} M^l$$

전역오류에 기반을 둔 메쉬 병합에 대한 전반적인 알고리즘은 그림 1과 같다.

```
MeshMerging (V0,C0)
{
  ClassifyTriangleMesh(V0,C0)
  CalculateLocalandGlobalError(V0,C0)
  MakeOrderofCandidateVertex(V0,C0)
  : According to the target error
  SelectOffSetValue(V0,C0)
  : Threshold of Local error
  (V',C') = DeleteVertex(V0,C0)
  (V1',C1') = Retriangulation(V',C')
}
```

그림 1. 메쉬 병합의 PseudoCode

먼저 삼각형 메쉬 표면에서 임의의 하나의 점 V_i을 선택하여 이것을 정사영한다. 이렇게 정사영하여 얻어진 점과 V_i까지의 거리를 계산하여 이 값을 식(2)와 같이 지역 오류로 정의한다.

$$E_{local}(M)=dist(V_i,P(V_i,C)) \qquad (2)$$
Where $P(V,C)$: Projection Function.

그리고 V_i점을 기준으로 이 점과 연결되어 있는 임의의 점 V_j을 선택하여 이 두 점으로 만들어지는 에지를 식(3)을 이용하여 특성을 정의한다. 즉, 이 에지를 포함하고 있는 주위의 삼각형을 찾아 주위의 삼각형이 이루는 각으로서 에지의 성질을 알 수 있다.

$$Cos^{-1}\frac{(N1\bullet N2)}{\|N1\|\,\|N2\|}=\theta_j(V_i,V_j) \qquad (3)$$

여기서 N_1과 N_2는 V_i와 V_j 를 가지고 있는 삼각형면에서의 수직 벡터(Normal Vector)인데, 위에서 구한 E_{local}과 $\theta_j(V_i,V_j)$를 이용하여 Schroeder[6]이 구분한 방법처럼 삼각형 메쉬를 그림 2와 같이 5가지로 나눈다.

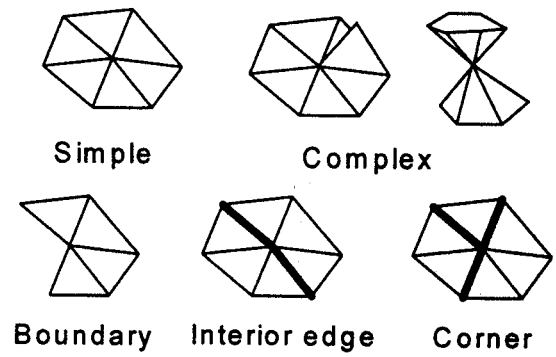

그림 2. 메쉬의 분류

어떤 임의의 점을 기준으로 주위의 삼각형이 완전한 360도를 이루거나, E_{local}가 어떤 임계값보다 작을 때는 간단한 경우로 설정을 하고, E_{local}가 어떤 임계값보다 작으면서 360도보다 작을 때는 경계점으로 설정한다. 그리고 앞에서 구한 식(3)이 특정 각도보다 작을 경우 그때의 에지를 특징 에지로 설정한다. 따라서 특징 에지의 개수가 2개이면 Interior Edge로, 1개이거나 3개 이상이면 Corner로 설정한다. 그밖에 세 개 이상의 삼각형이 만나서 이루어지는 에지를 포함하거나 기타 복잡한 점은 Complex로 설정한다. 보통 간단하거나 경계인 메쉬의 점은 쉽게 제거할 수 있지만, Interior나 Corner인 경우엔 메쉬의 기하학적인 특성을 가지고 있으므로 E_{local}나 $\theta_j(V_i,V_j)$를 고려하여 제거 여부를 판단한다. 그리고 Complex인 경우엔 메쉬의 점을 제거할 수 없다. 이렇게 n개의 점 중에서 E_{local}와 $\theta_j(V_i,V_j)$ 정보로 제거될 수 있는 m개의 메쉬의 점에 대해서만 식 (4)와 같이 전역 오류를 정의한다.

$$E_{global}(M)=\sum_{i}^{m} dist(V_i,P(V_i,C)) \qquad (4)$$

전역 오류를 구한 다음 제거될 수 있는 메쉬의 점들을 지역 오류에 따라 내림차순으로 정리한다. 이렇게 하여 얻어진 점들의 순서는 1가지로만 나오므로 제거되는 점

들의 순서에 대해 다음 표 1 과 같이 유일 해 (Unique Solution)를 얻을 수 있다. 표 1 에서 보통 $m < n$ 이며 Error 는 지역 오류를 뜻한다. 그리고 E_{index} 는 식(5)에 정의 되어 있는데 이 식을 사용하여 얻어지는 E_{index} 값과 사용자가 입력한 E_{target} 을 비교하여 가장 근접한 값을 찾는다.

Index	1	2	3	. . .	n-1	n
Vertex	V1	V2	V3	· · ·	Vn-1	Vn
Error	E1	E2	E3		En-1	En

Original Order

Index	1	2	3	. . .	m-1	m
Vertex	V3	V15	Vm	· · ·	V1	Vm-1
Error	E3	E15	Em		E1	Em-1
E_{index}	0.97		0.82		0.10	

Candidate Deciamte Vertex Order

표 1. 제거될 수 있는 메쉬 점들의 표

그리고 역으로 m 부터 E_{index} 를 가리키는 Index 를 알고 있으므로 이 Index 까지의 Vertex 값만을 제거하면 우리가 원하는 에러를 가진 3 차원 물체를 얻을 수 있다. 위에서 살펴본 방법대로 지역 오류와 전역 오류를 이용하여 표 1 만 저장하고 있다면 최대 (n-m)개의 다양한 해상도를 표현하기 위해 단지 m 개의 메모리로 표현할 수 있으므로 메모리를 크게 줄일 수 있는 장점이 있다. 그리고 연속적인 다양한 해상도를 가진 LOD 를 구현할 수 있다.

$$E_{index}(k) = \frac{\sum_{j=0}^{k} dist(V_j, Proj(V_j))}{E_{global}} \qquad (5)$$

where $K \le M$

이렇게 메쉬의 점들을 제거하므로서 생기는 홀(Hole)은 다시 삼각형으로 만들어져야 한다. 본 논문에서는 다음과 같은 방법을 사용하였다.

$$retriangle: \min\left(\sum_{i}\sum_{j \neq i} dist(V_i, V_j)\right) \qquad (6)$$

즉, 홀에 관련된 메쉬의 점들을 찾아 그 중 임의의 V_i 으로부터 거리가 최소인 값을 찾아 삼각형이 되도록 만들어 준다. 나머지 점들에 대해서도 이런 방법을 반복하여 삼각형을 만든다. 이런 방법을 사용하면 결과적으로 만들어지는 삼각형은 상대적으로 세 변의 길이가 비슷한 정삼각형 모양이 나와 렌더링할 때 성능이 좋게 된다.

본 논문에서 제안한 일련의 메쉬 병합 방법을 초기의 메쉬에 대해서 i 번 계속하여 반복하면 우리가 원하는 에러를 가진 LOD 를 만들 수 있다.

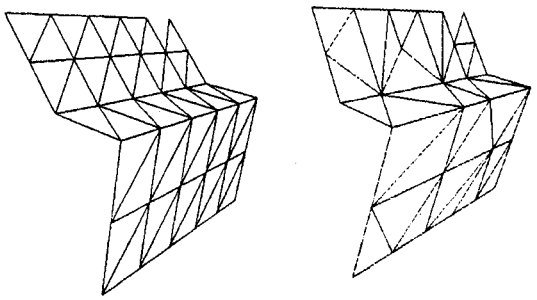

그림 3. 메쉬 병합방법에 의한 결과

4. 실험결과

본 논문에서는 현재 3 차원 데이터의 처리를 위해 널리 쓰이고 있는 VRML (Virtual Reality Modeling Language) 데이터에 대해 C 언어를 이용하여 제안한 방법을 구현하였다. 여러 VRML 데이터 중에서 특이한 기하학적인 특성을 많이 가지고 있는 Triceratops 데이터를 선택하였으며, 목표오류는 10%, 50%, 80%로 설정하였다. 목표오류에 따른 삼각형 메쉬 점들의 수와 VRML1.0 데이터의 크기를 표 2 에 정리 하였다. 표 2 에서 보듯이 목표오류가 클수록 삼각형 꼭지점들의 수와 화일 크기가 줄어드는 것을 알 수 있다.

E_{target}	Vertex 수	전체 Bytes
0%	2832	203,680
10%	2519	180,438
50%	1701	140,583
80%	1351	124,043

표 2. 목표 오류에 따른 결과

그림 4 부터 그림 7 까지는 표 2 에 대한 결과 그림으로 각각 Shading 한 결과이다. 그림 7 에서 목표 오류가 80% 인 경우에 원 데이터가 거의 절반으로 줄어들었지만 뿔이나 발가락같은 예리한 에지는 그대로 보존되어 적은 데이터의 양으로도 원 물체를 표현할 수 있는 것을 알 수 있다. 그리고 위의 데이터 경우 약 1000 여개의 꼭지점을 저장할 메모리만 더 있다면 1000 여 개의 다양한 연속 해상도 변이 LOD 를 사용자가 임의로 선택하여 표현할 수 있다.

그림 4. 원 3 차원 물체(# of Vertex: 2832)

그림 5. 10%의 에러(# of Vertex:2519)

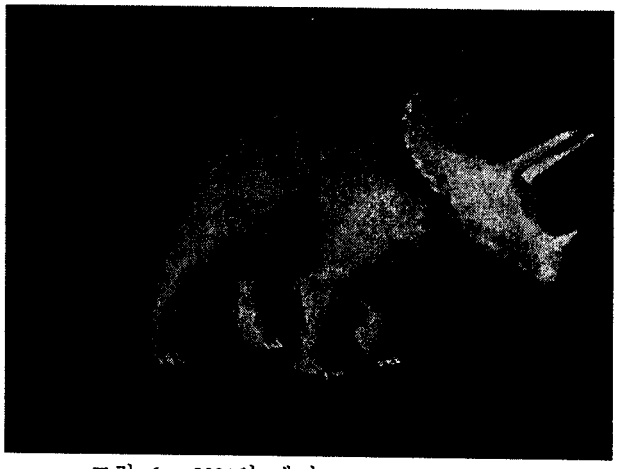

그림 6. 50%의 에러 (# of Vertex:1701)

그림 7. 80%의 에러 (# of Vertex :1351)

5. 결론

본 논문에서는 전역 오류에 기반을 둔 메쉬 병합 방법을 제안하였다. 메쉬 병합 방법은 삼각형 수를 줄이기 위해 빠른 수행을 하는 기존의 Mesh Decimation[6]방법에 기반을 두고 있다. 기존의 방법은 단지 메쉬의 지역적인 해석만을 통해 제거될 수 있는 점들을 판단하여 제거한 후 새로운 삼각형으로 만드는 방법이었다. 하지만 본 방법은 정사영을 통한 지역 오류와 전역 오류를 통해 크게 다음과 같이 3 가지의 장점을 가지고 있다. 첫째 3 차원 데이터의 수를 줄일 때 쉽게 정확한 오류의 제어가 가능하고, 둘째 유한 값의 오류가 존재하며, 셋째 제거되는 점들의 유일한(Unique) 순서를 가지고 있으므로 LOD 를 구현할 때 적은 메모리를 가지고도 사용자가 선택적으로 다양한 연속 해상도변이 LOD 를 효과적으로 구현 할 수 있다.

참고문헌

[1] Ames, Nadeau, *VRML 2.0 Sourcebook*, Wiley, 1997

[2] P.Hinker and C.Hansen, "Geometric optimization," *IEEE Visualization '93*, pp.189-195, Oct. 1993.

[3] Greg Turk, "Re-tiling polygonal surfaces," *SIGGRAPH '92*, pp.55-64, July. 1996.

[4] Hugues, Tony DeRose, Tom Duchamp, John McDonald and Werner Stuetzle, "Mesh optimization.", *SIGGRAPH '93*, pp.19-26, July 1993.

[5] J.Rossignac and P.Borrel, "Multi-resolution 3D approximation for rendering complex scenes," *Geometric Modeling in Computer Graphics*, pp. 455-465, 1993.

[6] W.J.Schroeder and J.A.Zarge, "Decimation of triangle meshes," *SIGGRAPH '92*, pp.65-70, July 1992

[7] Michael Lounsbery, Tony D. DeRose, Joe Warren, "Multiresolution Analysis for Surfaces of Arbitrary Topological Type,", Tech. Rep, Univ. of Washington.