

3D 메쉬 간략화 기술

김 창 현

고려대학교 컴퓨터학과

메쉬 간략화는 원래 모델의 형상과 특징을 유지하며 폴리곤의 수를 감소시키는 것이다. 즉 초기 모델에서 필요 이상의 위상학적, 기하학적 정보를 가지고 있는 폴리곤을 제거하여 단순한 모델을 만드는 과정을 뜻한다. 이러한 3D 모델을 간략화 하는 기술은 컴퓨터 애니메이션, 3D 게임의 LOD(Level of Detail) 제어 기술, 인터넷의 3D 그래픽 솔루션, 실시간 3D 그래픽스 시뮬레이션 등 많은 분야에 활용되고 있다.

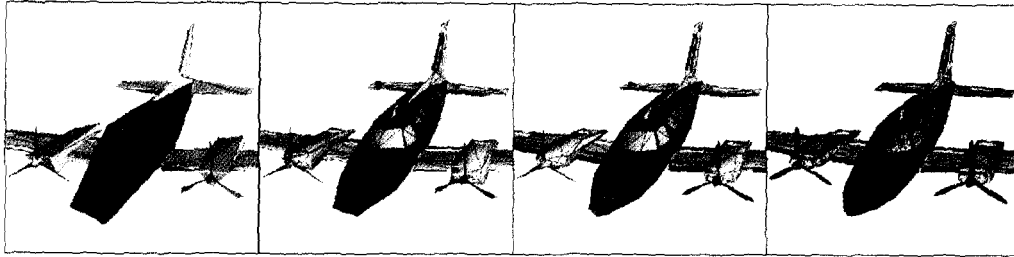
인터넷의 3D 그래픽 솔루션은 전자 카타로그나 쇼핑 물의 상품 전시 등 사용자가 원하는 방향으로 조작하여 볼 수 있는 장점이 있다. 인터넷 상에서 3D 모델을 사용하여 그래픽을 표현하기 어려운 이유는 모델 데이터 크기가 주 원인이었으나, 메쉬 간략화 기술을 이용하면 이러한 단점을 극복할 수 있다.

3D 게임의 LOD 제어 기술은 여러 가지 3D 게임이 리얼 타임으로 수정 가능하도록 하는 중요한 기술이다. LOD 그래픽 엔진은 3D 모델의 형상을 구체적 단계에서 단순한 단계까지 물체와 시점 사이의 거리에 따라 여러 단계로 LOD 모델 구조의 데이터로 최적화 시킨 후, 각 단계의 형상을 시점과 물체와의 거리의 이동에 따라 자동으로 축소 및 합성하는 기술이다. 이러한 기술은 자동차 레이싱 게임에서 도로 및 관중 배경 화면에 사용되며, 역동적인 3차원 자동차 모델의 변화를 처리해야 하는 기술 및 거리에 따른 배경의 조절 기술에 자주 이용되고 있다. 자동차의 역동성을 자연스럽게 표현하기 위하여 LOD 모델 생성 및 연속적인 LOD 전환 기술을 적용시켜야 하며, 거리에 따른 배경의 상세도 조절이 필요하다. 이러한 기술은 3D 압축 및 실시간 디스플레이, 전송의 기술로 해결할 수 있다. 또한 가상 시뮬레이션 및 3D 게임의 주요 요소 기술이 될 수 있다.

레이저 스캐닝 시스템과 같은 장치들의 출현으로 컴퓨터 그래픽스와 기하학적 모델링에서는 매우 복잡한 메쉬를 흔하게 볼 수 있었고, 이를 간략화 하여 LOD에 사용할 수 있는 3D 메쉬 간략화에 대한 연구가 최근 10년 동안 이루어져 왔다. Hoppe^{5,6,7}, Garland와 Heckbert^{3,4}, Cohen², Luebke와 Erickson¹², Lindstrom과 Truk⁹ 등의 간략화 기술이 주로 사용될 수 있고, 특히 Garland와 Heckbert^{3,4}이 가장 간편하게 사용될 수 있는 알고리즘으로 널리 알려져 있다. Hoppe⁵의 "Progressive Mesh"는 점진적으로 메쉬의 전송이 가능하도록 하였다. 이 방법은 에지 축약에 기반을 두었으며 복잡한 모델에서 단순한 모델로의 변환이 가능하도록 하였고, 특히 각 간략화 단계에서 잃어버리는 정보를 보존하는 것에 착안하였다. 이 알고리즘은 두 단계로 나눌 수 있는데, 첫째로 분석 단계로 세부 정보를 가지는 간략화 단계이다. 두번째는 합성 단계로 *edge split*를 하여 LOD 근사화를 한다. 그림 1은 Hoppe⁵의 점진적 메쉬를 나타낸 것으로 낮은 해상도에서 높은 해상도로 점진적으로 변화하는 것을 보여준다.

Garland와 Heckbert³는 평면에서 한 점과의 거리 제곱의 합을 간단하면서도 빠르게 수행되는 이차 형식으로 표현함으로써 모델의 빠른 간략화와 품질을 향상시켰다. 그러나 기하학 정보만을 이용하기 때문에 컬러, 법선, 텍스처 좌표를 가지는 메쉬는 간략화하기에는 적합하지 못하다. 그래서 Garland⁴와 Hoppe⁷는 이차 형식을 확장하여 컬러, 법선, 텍스처 좌표를 고려한 새로운 간략화 방법을 제안하였다. 이차 형식을 이용한 것은 아니지만 Cohen²은 텍스처를 고려한 텍스처 오차 측정 알고리즘을 제안하였다. 그림 2는 Garland⁴의 확장 알고리즘을 이용한 것으로 기하학 정보만을 이용하여 간략화 하였을 때 보다 더 나은 모델의 품질을 보여준다.

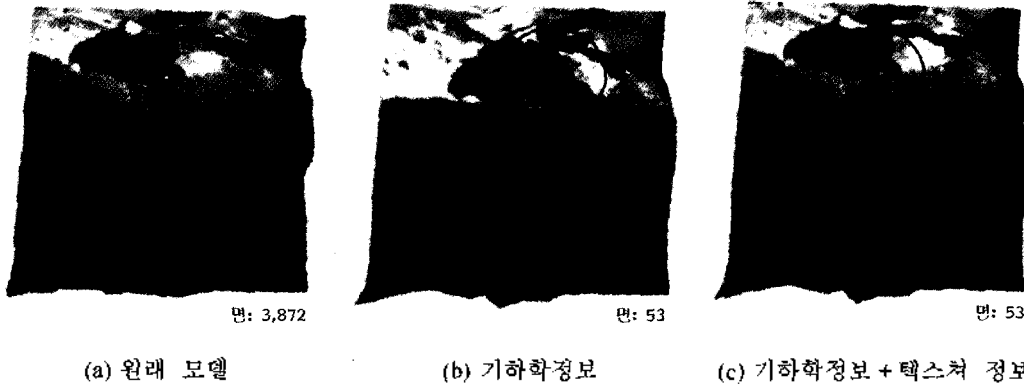
스텐포드 대학의 Levoy⁸는 "The Digital Michel-



단순한 메쉬 (낮은 해상도)

복잡한 메쉬 (높은 해상도)

그림 1. Progressive Mesh



(a) 원래 모델

(b) 기하학정보

(c) 기하학정보 + 텍스처 정보

그림 2. 확장된 QEM을 이용한 간략화

angelo Project”를 수행하면서 많은 대용량의 메쉬를 획득하게 되었고, 이렇게 획득한 대용량 메쉬를 단순한 모델로 만들 수 있는 간략화 연구가 활발하게 되었다. 그러나 기존의 방법은 대부분이 in-core 간략화 방법이기 때문에 대용량의 모델을 간략화 하는 것은 어렵다. 그림 3은 실제 David 상을 레이저 스캐너를 이용하여 스캔하여 3D 정보를 획득하고 이를 3D 폴리곤 메쉬 모델로 다시 재구성한 것이다.

3D 대용량의 모델을 간략화 하는 방법은 out-of-core 구조를 갖는 것이 효과적인데, Bernardini^[11], Lindstrom^[10,11], Shaffer와 Garland^[13] 등이 연구하고 있다. 간단히 in-core과 out-of-core 두 가지 방법에 대해 설명하자면 다음과 같다. in-core 방법은 기존의 간략화 방법에서 많이 다루었던 것으로 모든 메쉬 정보를 메인 메모리에 넣어 간략화를 수행하는 것이다. 즉 메모리에 전적으로 의존해 간략화 하는 방법을 in-core 간략화라고 하는데 메쉬를

구성하는 기하학 정보, 위상 정보, 에러 척도를 위한 에러 정보 등을 모두 메모리에 담는 것이다. 그러나 대용량 메쉬의 경우 메인 메모리에 모든 대용량 메쉬 정보를 담아 둘 수가 없기 때문에 out-of-core 방법이 제안되었고, out-of-core 방법은 모든 메쉬 정보를 메인 메모리에 넣지 않고 일부분의 메쉬 정보에 대해서만 메모리를 이용하고, 나머지 메쉬 정보는 하드 디스크에 담아 두어 간략화 하는 것이다. 대용량 메쉬를 다루는 기술 중에서 주목 받는 방법으로는 Lindstrom^[10]의 “Out-of-Core Simplification of Large Polygonal Models”이 있다. 이 방법은 대용량 모델을 균일한 격자 셀로 나눈 후, 셀 안의 꼭지점에 대해 이차 형식을 구하고, 각 셀 내의 이차 형식을 합하여, 선형 시스템에 의해 새로운 점의 위치를 구한다. 그러나 이러한 방법도 해상 테이블을 구성하기 위해 많은 메모리가 사용된다. 그림 4는 OoCS(Out-of-Core Simplification)의 알고리즘을 이용하여 간략화 한 것으로 대용량 모델인

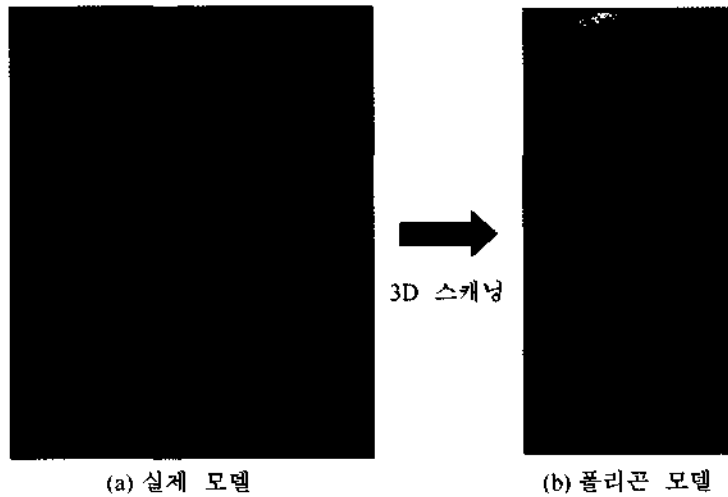


그림 3. The Digital Michelangelo Project

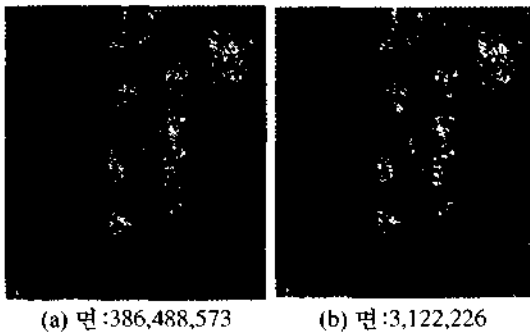


그림 4. St. Matthew 모델

St. Matthew 모델을 간략화 하였으나, 경계 부분의 보존을 고려하지 못했기 때문에 모델의 품질이 아직까지는 썩 좋지 못하다.

그래서 Lindstrom^[11]은 메모리의 비중을 좀 더 낮추면서 하드 디스크를 효율적으로 활용할 수 있고, 경계 부분을 고려한 방법인 "A Memory Insensitive Technique for Large Model Simplification"을 제안하여, 효율적으로 메모리를 이용하는 기법을 제안하였다. 또한 경계 부분의 보존을 위해 에지 대 한 이차식을 유도함으로써 모델의 품질을 향상시켰다. Shaffer와 Garland^[13]는 대용량 메쉬를 효과적으로 간략화 하는 다른 방법을 제안하였다. 입력 메쉬에 대해 2가지 패스를 수행한다. 첫번째 패스는 표면을 분석하고 그리고 공간을 적응적으로 분

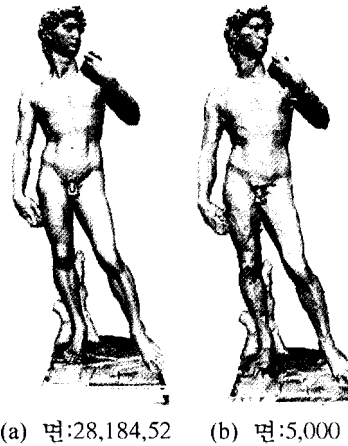


그림 5. David 모델

할한다. 즉 표면을 분석하여 BSP-트리를 구성한다. 마지막 패스는 BSP-트리를 사용하여 원래 꼭지점을 모으게 되고, 원래 메쉬에 대해 근사화를 하는 것이다. 이 방법은 OoCS 보다 메모리를 더 사용하는데, BSP-트리를 구성하고 이차 형식 정보를 위한 추가적인 메모리가 필요하다. 그러나 이 방법은 Lindstrom의 방법보다 더 나은 모델의 품질을 보여준다. 그림 5는 Shaffer^[13]의 알고리즘을 이용하여 David 모델을 간략화 한 것으로 대용량인 모델(면:28,184,526)을 5,000개로 간략화 하였을 때, 모델의 품질이 좋아진 것을 볼 수 있다.

이상과 같이 메쉬 간략화 기술은 대용량의 3차원 데이터를 온라인이나 인터넷 상에서 사용할 수 있는 작은 데이터로 자동으로 변환시켜 주는 기술에서 출발하여 최근에는 원래 데이터의 특징을 잘 보존하여 간략화 시키는 기술과 텍스처의 특징을 잘 유지하는 간략화 기술 등으로 발전하고 있다. 또한, 간략화 기술과 서브디비전 기술을 기반으로 메쉬를 다단계로 표현하여 LOD에 따라서 다단계 메쉬를 제어하는 기술로 발전되고 있다.

다단계 메쉬는 인터넷 상의 그래픽 솔루션과 3D 게임의 실시간 제어, 다단계 편집 등으로 활용 범위를 넓혀가고 있는 흥미로운 분야라 할 수 있다.

참고문헌

1. F. Bernardini, J. Mittleman, H. Rushmeier, and G. Taubin. Case study: Scanning michelangelo's Florentine pieta, SIGGRAPH 99, Course Notes, Course 4, ACM SIGGRAPH, August 1999.
2. J. Cohen, M. Olano and D. Manocha. Appearance-preserving simplification, In SIGGRAPH'98, pp. 115-122, July 1998.
3. M. Garland and Paul S. Heckbert. Mesh simplification with quadric error metrics. In SIGGRAPH 97, pp. 209-216, August 1997.
4. M. Garland and P. Heckbert. Simplifying surfaces with color and texture using quadric error metrics, IEEE Visualization'98, pp. 263-269, October 1998.
5. H. Hoppe. Progressive Meshes. In SIGGRAPH 96, pp. 99-108, August 1996.
6. H. Hoppe. View-dependent refinement of progressive meshes. In SIGGRAPH 97, pp. 189-198, August 1997.
7. H. Hoppe. New quadric metric for simplifying meshes with appearance attributes, IEEE Visualization'99, pp. 59-66, October 1999.
8. M. Levoy, K. Pulli, B. Curless, S. Rusinkiewicz, D. Koller, L. Pereira, M. Ginzton, S. Anderson, J. Davis, J. Ginsberg, J. Shade, and D. Fulk. The digital Michelangelo project: 3D scanning of large statues. In SIGGRAPH'00, pp. 131-144, July 2000.
9. P. Lindstrom and G. Turk. Fast and memory efficient polygonal simplification, IEEE Visualization98, pp. 279-286, October 1998.
10. P. Lindstrom. Out-of-core simplification of large polygonal models. In SIGGRAPH 00, pp. 256-262, July 2000.
11. P. Lindstrom and C. Silva. A Memory Insensitive Technique for Large Model Simplification, IEEE Visualization'01, October 2001.
12. D. Luebke and C Erikson. View-dependent simplification of arbitrary polygonal environment. In SIGGRAPH'97., pp. 199-208, August 1997.
13. E. Shaffer and M. Garland. Efficient adaptive simplification of massive meshes. IEEE Visualization'01, October 2001.