

빠른 기하 연산을 위한 변형 가능한 쿼드 메시*

강윤구⁰, 장준영, 김명수
서울대학교 컴퓨터공학부
(yunku2002, jyy14)@3map.snu.ac.kr, mskim@snu.ac.kr

Deformable Quad Mesh for Accelerated Geometric Operations

Yunku Kang⁰, Junyeong Jang, Myung-Soo Kim
Dept. of Computer Science and Eng., Seoul National University

요약

본 논문에서는 메시의 빠른 충돌 감지 및 거리 계산을 위해 매개 곡면의 장점을 공유하는 특수한 구조의 쿼드 메시의 설계와 BVH 구성 방법을 제안한다.

1. 서론

3차원 모델의 충돌 감지, 거리 계산 등의 기하 연산을 가속화하기 위한 효율적인 BVH의 설계에 있어서 모델을 분할하는 문제가 중요하다. BVH 상에서 조상-자손 관계가 아닌 BV들이 최대한 서로 겹치지 않는 것이 좋다. 매개화된 곡면 모델의 경우 이는 단순히 매개 구간을 분할하는 것과 대응시킬 수 있어 BVH를 빠르게 생성하는 것이 가능하다. Kim et al.[1]은 자유형상 NURBS 곡면에 대해 Coons 패치로 근사하는 BVH를 소개하였으며, 이를 활용해 Hausdorff 거리 계산[2]이나 컨벡스 헐의 계산[3]과 같은 복잡한 알고리즘이 가능하였다. 하지만 일반적인 폴리곤 메시 모델은 면들의 정형화된 구조가 없어 BVH를 효율적으로 구성하는 것이 어렵다. 모델이 차지하는 공간을 분할해 AABB 혹은 그와 유사한 BVH 트리를 생성하는 방식이 일반적이다. OBB 트리가 AABB 트리에 비해 충돌 감지 속도가 빠름이 알려졌지만[4], Bergen[5]은 변형하는 모델에 대해 AABB 트리를 새로 생성하는 대신 이전 단계에서의 트리를 재정비함으로써 충돌 감지를 가속화하는 방법을 소개하였다.

본 논문에서는 효율적인 기하 연산을 위해 격자 구조를 갖는 쿼드 메시 모델의 설계를 제시하고 그에 맞는 새로운 BVH의 생성 알고리즘을 소개한다. 이 BVH는 메시가 최초로 주어졌을 때 한 번만 생성하여 이후에 메시가 변형해도 메시가 같은 위상을 갖는 한 같은 골격을 유지하면 되기 때문에 각종 좌표 및 오차 등의 수치만 변경해 업데이트 속도가 빠르다. 이 BVH를 바탕으로 46208개의 면을 갖는 안면 쿼드 메시 모델(그림 1)에 대해 병렬 처리를 적용한 자기 충돌 감지를 수행한 결과 실시간의 속도를 보였다.

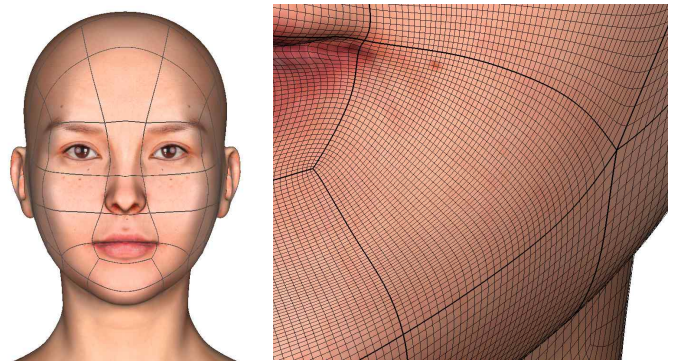


그림 1: 26개의 쿼드 패치와 46208개의 사각형으로 이루어진 안면 모델

2. 메시 설계

매개화된 곡면의 장점을 모방하기 위해 메시는 면이 모두 사각형으로 이루어진 쿼드 메시로 디자인하되 대부분의 정점의 차수가 4가 되도록 구성한다. 테두리를 제외한 모든 정점의 차수가 4이고 차수가 2인 정점(꼭짓점)이 네 개뿐인 서브메시를 하나의 쿼드 패치라고 정의했을 때, 디자인하고자 하는 메시는 최대한 적은 수의 쿼드 패치를 꼭짓점끼리 이어붙인 꼴이 되도록 하는 것이다.

차수 제약으로 인해 격자 구조가 되는 하나의 쿼드 패치는 “가로세로”를 따질 수 있다는 점에서 2변수 매개 곡면과 유사하다. 쿼드 패치의 한 꼭짓점에서 다른 꼭짓점에 도달할 때까지 “한 방향”으로 이동하는 횟수를 그 방향으로의 길이라고 보면 가로 혹은 세로 길이를 기준으로 삼아 쿼드 패치를 분할할 수 있다.

3. BVH 구성

3.1 메시 분할

메시가 n 개의 쿼드 패치로 이루어졌다고 할 때 기하 연산을 위한 BVH 또한 n 개를 이용한다. 각 BVH의 뿌리 노드가 하나의 쿼드 패치에 대응되며 여기에서부터 top-down 방식으로 BVH의 골격을 형성한다. 각 BV 노드의 두 자식 노드는 부모가 감싸는 쿼드 패치를 두 개로 분할해 각각 감싼다. 쿼드 패치를 분할할 때는 가로와 세로 중 길이가

* 구두발표논문
* 본 연구는 문화체육관광부 및 한국콘텐츠진흥원의 문화기술 연구개발 지원사업으로 수행되었음.

긴 방향으로 길이가 절반이 되도록 자른다. 잎 노드는 각각의 사각형을 감싸게 된다. 이 작업은 메시가 위상을 유지한 채 변형하는 한 한 번만 수행하면 된다.

3.2 사면체 근사

하나의 쿼드 패치를 감싸는 각 BV 노드의 모양으로는 그림 2와 같이 한 사면체를 특정 거리(오차)만큼 오프셋한 볼륨을 이용한다. 이 사면체는 쿼드 패치의 네 꼭짓점을 연결한 것으로, BV가 쿼드 패치를 감쌀 수 있도록 오차를 충분히 크게 정하는 것이 관건이다.

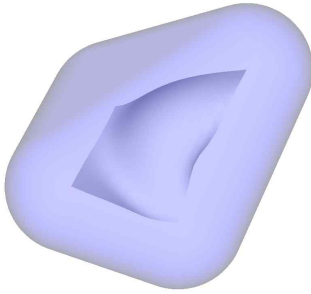


그림 2: 쿼드 패치를 감싸는 오프셋 사면체

잎 노드의 사면체는 대응되는 이중선형 곡면을 이미 포함하므로 오차를 0으로 고정하고 잎 노드에서부터 bottom-up 방식으로 필요한 오차를 계산한다. 메시가 변형될 때는 BVH의 빠른 업데이트를 위해 오로지 변형이 있었던 잎 노드에서부터 업데이트한다. 사면체 T 에 대응되는 쿼드 패치 P 가 꼭짓점 A 와 B 를 공유하는 두 쿼드 패치(사면체는 T_1 과 T_2 이다.)로 분할됐다고 했을 때 T 의 오차를 다음과 같이 설정하면 P 가 T 의 오프셋 안에 포함됨이 보장된다.

$$error(T) = \max(error(T_1), error(T_2)) + \max(dist(A, T), dist(B, T))$$

이는 삼각 부등식의 간단한 응용으로 증명 가능하며, 간단한 계산이기 때문에 BVH의 빠른 초기 구성과 업데이트가 가능하다.

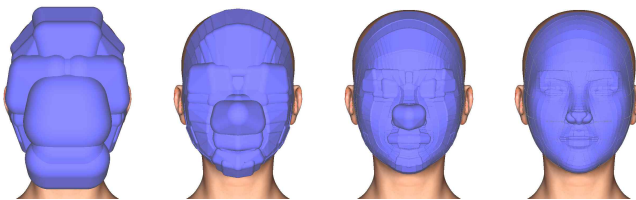


그림 3: 안면 모델에 대한 여러 단계에서의 BVH

4. 자기 충돌 감지 결과

위에서 다룬 BVH에 전통적인 자기 충돌 감지 알고리즘을 적용할 수 있다. 이때 두 BV 사이의 거리를 구하기 위해서는 사면체 사이의 거리에서 두 오차의 합을 빼면 된다. 사면체 사이의 빠른 거리 계산은 GJK 알고리즘[6]을 이용한다. 메시가 변형할 때엔 이전 단계의 충돌 감지 결과를 바탕으로 변형이 일어난 부분에 대해서만 새로 수행해 충돌 감지를 가속화할 수 있다.

본 논문에서 제시하는 조건을 만족하는 안면 메시 모델을 설계해본 결과 46208개의 면들이 26개의 쿼드 패치를 구성하게 되었다(그림 1). 이 모델에 대해 그림 3의 맨 왼쪽과 같이 26개의 BVH를 생성하고 메시 변형을 시켜가며 자기 충돌 감지를 수행했을 때 변형이 심하지 않은 경우 실시간으로 감지가 이루어졌다. 충돌 감지는 독립된 여러 BVH에 걸친 병렬화가 가능하기 때문에 OpenMP를 이용한 병렬 처리를 추가로 적용해 완전한 실시간 자기 충돌 감지를 달성할 수 있었다(i7-3770K CPU에서 5~25ms).

5. 결론

본 논문에서는 빠른 기하 연산을 위해 격자 구조의 쿼드 패치로 이루어진 특수한 메시 모델을 제안하고 이에 대한 효율적인 BVH를 위해 오프셋 사면체로 쿼드 패치를 감싸는 방법을 소개하였다. 이를 바탕으로 디자인하고 BVH를 생성한 안면 메시 모델을 변형할 때의 자기 충돌 감지는 실시간으로 이루어졌다.

추후 연구로는 본 논문에서 제시한 메시 모델을 생성하는 데에 도움을 주는 툴의 개발이 도움이 될 것이다. 더 나아가 이미 존재하는 메시 모델이 제약 조건을 만족하는 쿼드 메시가 되도록 변환하는 방법에 대한 연구도 필요할 전망이다.

6. 감사의 글

실험에 쓰인 안면 메시 모델을 디자인해준 비주얼아트의 이소성 씨에게 감사의 인사를 전한다.

참고문헌

[1] Y.-J. Kim, Y.-T. Oh, S.-H. Yoon, M.-S. Kim and G. Elber, Coons BVH for Freeform Geometric Models, *ACM Transactions on Graphics*, 30(6): Article 169, 2011.
 [2] Y.-J. Kim, Y.-T. Oh, S.-H. Yoon, M.-S. Kim and G. Elber, Efficient Hausdorff distance computation for freeform geometric models in close proximity, *Computer-Aided Design*, 45(2): 270-276, 2013.
 [3] Y.-J. Kim, M.-S. Kim and G. Elber, Precise convex hull computation for freeform models using a hierarchical Gauss map and a Coons BVH, *Computer-Aided Design*, 46(1): 252-257, 2014.
 [4] S. Gottschalk, M. C. Lin and D. Manocha, OBBTree: A Hierarchical Structure for Rapid Interference Detection, *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, 171-180, 1996.
 [5] G. V. D. Bergen, Efficient Collision Detection of Complex Deformable Models using AABB Trees, *Journal of Graphics Tools*, 2(4): 1-13, 1997.
 [6] E. G. Gilbert, D. W. Johnson and S. S. Keerthi, A Fast Procedure for Computing the Distance Between Complex Objects in Three-Dimensional Space, *Robotics and Automation*, 4(2): 193-203, 1988.