

포워드 디퍼렌싱을 이용한 Catmull-Clark 서브디비전 서피스 렌더링

설주환⁰ 양성봉
연세대학교 컴퓨터과학과
(someone⁰, yang)⁰@cs.yonsei.ac.kr

Forward Differencing for Rendering Catmull-Clark Subdivision Surfaces

Joo-Hwan Sul⁰ Sung-Bong Yang
Dept. of Computer Science, Yonsei University

요 약

본 논문은 Catmull-Clark 서브디비전 서피스(subdivision surfaces)를 포워드 디퍼렌싱(forward differencing)을 이용하여 효율적으로 evaluation 해서 렌더링하는 알고리즘을 제안하고 있다. 포워드 디퍼렌싱은 순수한 다항식만을 evaluation 할 수 있다. 그러나 Catmull-Clark 서브디비전 서피스는 순수한 다항식이 아니다. 그러므로, Catmull-Clark 서피스를 정규 패치들(regular patches)로 분리하고, 그 패치들에 대한 다항식을 만들고, 포워드 디퍼렌싱을 사용해서 evaluation 하면 된다. 본 알고리즘의 장점은 전통적인 리커시브(recursive) 서브디비전 기법에 비해 메모리의 요구가 적다. 즉, [1]과 마찬가지로 서브디비전 깊이(subdivision depth)에 독립적으로 항상 상수(constant) 메모리 양 만큼만 요구된다.

1. 서론

서브디비전(subdivision) 기법은 임의의 베이스 메쉬(base mesh)들로부터 부드러운 서피스(surfaces)를 효율적으로 만들어내는 장점을 가지고 있기 때문에ジオ메트리 모델링(geometry modeling) 분야에서 많은 인기를 얻고 있다. 또한, NURBS의 trimming 처리 및 animating model의 smoothness 처리의 어려움에 대한 해결책으로 서브디비전 기법이 많이 사용되고 있으며, 최근에 나온 상업용 3D 모델링 및 애니메이션 관련 툴들(Maya, Renderman, ...)에도 서브디비전이 구현되어 있다[4].

전통적인 서브디비전 기법들은 서브디비전 깊이에 비해 해서 메모리 공간의 요구가 증가되고, 서브디비전 서피스를 계산하는데 시간도 많이 걸린다. 그 결과 3D 게임과 같은 실시간 렌더링(rendering)에는 사용되지 못하고, 3D 애니메이션과 같은 실시간으로 처리할 필요가 없는 응용에서 주로 많이 사용되어져 왔다. 서브디비전을 이용해서 만든 대표적인 3D 애니메이션 영화로는 Pixar Animation Studios에서 만든 "Toy Story 2"가 있다[3].

서브디비전 기법들에는 어프록시메이션(approximation) 기법과 인터폴레이팅(interpolating) 기법으로 나눌 수 있다. 어프록시메이션 기법에는 삼각형 메쉬를 위한 Loop 기법과 사각형 메쉬를 위한 Catmull-Clark 기법이 있고, 인터폴레이팅 기법에는 Butterfly 기법과 Kobbelt 기법 등이 있다. 일반적으로 정규(regular) 메쉬에서 어프록시메이션 기법은 C^2 -continuity를 만족하는데 반해

인터폴레이팅 기법은 C^1 -continuity를 만족한다. 그러므로 어프록시메이션 기법이 인터폴레이팅 기법보다 더 부드러운 서피스를 만들어낸다. 본 연구에서는 어프록시메이션 기법인 Catmull-Clark 기법을 포워드 디퍼렌싱을 사용하여 빠르게 evaluation 해서 렌더링하는 알고리즘을 제안하고 있다.

2. 관련 연구

서브디비전 서피스를 각각의 패치들로 나누고, 그 나누어진 각각의 패치들에 대해 evaluation 하는 알고리즘에 대한 다양한 연구가 있어 왔다. Catmull-Clark 기법과 Loop 기법을 Eigen-analysis를 이용해서 evaluation 하는 기법을 Jos Stam이 제안했다[5, 6]. 이 연구 결과로 인하여 서브디비전 서피스는 명시적으로 subdividing 하지않고는 직접적으로 evaluation 할 수 없다는 오랜 통념을 깼다. 그 이외의 다른 접근법은 Loop 기법을 포워드 디퍼렌싱을 이용해서 정규(regular) 삼각형 패치를 evaluation 하고, 렌더링하는 알고리즘을 Stephan Bischoff가 제안했다[1]. 이 연구로 인하여 Loop 서브디비전 기법을 하드웨어로 구현할 수 있는 간단한 알고리즘 제안 및 하드웨어로 구현했을 때 병렬처리를 통해 클럭 사이클(clock-cycle) 당 하나의 삼각형을 렌더링 할 수 있게 되었다. 본 논문은 [1]의 변형으로 Catmull-Clark 서브디비전 서피스를 포워드 디퍼렌싱으로 evaluation 하고 렌더링하는 방법을 나타내고 있다.

3. Catmull-Clark 서브디비전 서피스

Catmull-Clark 서브디비전 서피스는 임의의 위상(arbitrary topology)을 가진 bicubic B-spline 서피스를 일반화한 기법이며, valence가 4인 regular 정점(vertex)들에서는 C^2 -continuity를 만족하고, valence가 4가 아닌 irregular 정점들에서는 C^1 -continuity를 만족한다. Catmull-Clark 서브디비전은 크게 splitting 단계와 averaging 단계로 나누어져 수행된다. Splitting 단계에서는 각각의 사각형에 edge들의 중점과 face의 중점에 새로운 정점들을 삽입해서 4개의 사각형으로 나누는 일을 하며, averaging 단계에서는 각각의 정점들을 그 정점과 이웃하고 있는 정점들의 weighted average와 함께 그 정점의 위치를 변경한다. Weighted average는 [2, 6, 7]에서 소개한 subdivision masks를 사용해서 계산한다.

4. 포워드 디퍼렌싱을 이용한 Catmull-Clark 서피스 분석

4.1 정규 Catmull-Clark 패치를 위한 포워드 디퍼렌싱

포워드 디퍼렌싱은 다항식(polynomial)을 evaluation 하는데 가장 효율적인 기법으로 알려져 있다. Catmull-Clark 서피스는 순수한 다항식이 아니다. 그러므로 서브디비전 서피스를 각각의 정규 패치들로 나누고, 나누어진 하나의 정규 Catmull-Clark 패치에 대한 다항식을 만들고, 포워드 디퍼렌싱을 적용하면 된다. 그러나 정규 Catmull-Clark 패치를 나타내는 bivariate 다항식은 너무 복잡하기 때문에 직접 포워드 디퍼렌싱을 분석하는 것이 어려우며, 분석한 결과의 정확성을 검증하는 것도 어렵다. 본 논문에서는 패치가 곡선들(curves)의 집합이라는 점에 착안하여, 먼저 cubic B-spline 곡선에 대한 다항식을 만들고, 이를 포워드 디퍼렌싱으로 분석한다. 포워드 디퍼렌싱과 관련된 더 자세한 내용은 [1]에 설명되어 있다. 위의 과정을 정확하게 수행하면 다음과 같은 행렬식(matrix)을 얻을 수 있다. 여기서 Δ 는 미분(derivatives)을 나타내며, $p[]$ 는 곡선의 컨트롤 포인트들

$$\begin{bmatrix} f \\ \Delta f \\ \Delta^2 f \\ \Delta^3 f \end{bmatrix} = \begin{bmatrix} \frac{1}{6} & \frac{4}{6} & \frac{1}{6} & 0 \\ -\frac{k^3}{6} + \frac{k^2}{2} - \frac{k}{2} & \frac{k^3}{2} - k^2 & -\frac{k^3}{2} + \frac{k^2}{2} + \frac{k}{2} & \frac{k^3}{6} \\ -k^3 + k^2 & 3k^3 - 2k^2 & -3k^3 + k^2 & k^3 \\ -k^3 & 3k^3 & -3k^3 & k^3 \end{bmatrix} \begin{bmatrix} p[0] \\ p[1] \\ p[2] \\ p[3] \end{bmatrix}$$

(control points)을 나타낸다. k 는 step width를 나타내며, $k = 2^{-r}$ 의 값을 갖는다. 여기서 r 은 서브디비전 깊이(subdivision depth)를 나타낸다.

다음은 cubic B-spline 곡선을 위한 포워드 디퍼렌싱 결과를 가지고 어떻게 정규 bicubic B-spline patch에 적용하는지에 대해 설명하겠다. 우선 그림1과 같은 정규 패치의 열 방향으로 있는 컨트롤 포인트들(Ex. P_0, P_4, P_8, P_{12})에 대해 포워드 디퍼렌싱을 적용해서 새로운 정점을 구한다. 나머지 3개의 열들에 대해서도 동일한 방법을 적용해서 새로운 정점들을 구하면 된다. 여기서 구

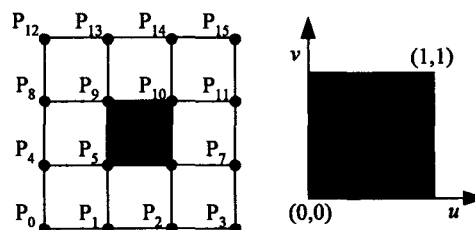


그림 1. Regular bicubic B-spline patch ($r=2$)

한 4개의 정점들에 대해 포워드 디퍼렌싱을 한번 더 적용하면 극한 서피스(limit surfaces) 위의 정점들을 구할 수 있다. 이와 같은 과정은 그림2에서 잘 보여주고 있다. 여기서 outer loop와 inner loop 중첩된 반복문이 사용되었는데, $r=2$ 일 때 outer loop에서는 인덱스가 하나 증가할 때마다 열 방향으로 step width 만큼 이동하고, inner loop에서는 인덱스가 하나 증가할 때마다 행 방향으로 step width 만큼 이동한다.

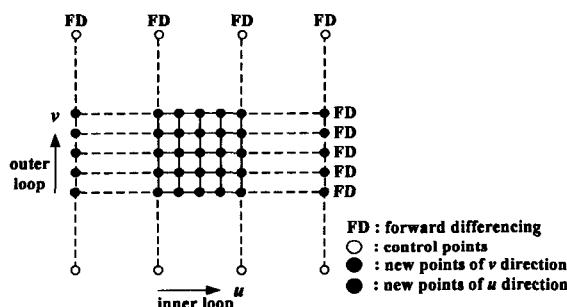


그림 2. FD로 Catmull-Clark 패치를 evaluation 하는 과정

다음은 메모리의 요구량에 대해 알아보고, 전통적인 리커시브 서브디비전(recursive subdivision) 기법과 비교해 보자. 본 논문에서 제안한 포워드 디퍼렌싱을 이용해서 사각형 패치(quad patch)를 서브디비전 하는데 요구되는 메모리는 항상 상수(constant) 만큼만이 요구된다. 왜냐하면, 구한 극한 정점(limit vertex)들을 quad strips를 이용해서 바로 렌더링 할 수 있으므로, 부파적인 저장 공간을 필요로 하지 않으며, 포워드 디퍼렌싱을 하는데 요구되는 메모리 공간만 있으면 된다. 반면에 전통적인 리커시브 서브디비전 기법을 이용해서 사각형 패치를 서브디비전 한다면 서브디비전 깊이에 의존해서 face의 수가 2^{2r} 만큼 증가하게 된다. 그러므로, 이들 정보들을 저장하기 위해서는 많은 메모리가 요구된다.

4.2 Irregular 정점들이 포함된 Catmull-Clark 패치들을 evaluation 하는 방법

일반적인 3차원 모델들은 많은 irregular 정점들을 포함하고 있다. 포워드 디퍼렌싱은 정규 사각형 패치들에만 적용할 수 있으며, irregular 정점을 포함하고 있는 패치들에는 적용할 수가 없다. 그림3은 irregular 정점을 포함하고 있는 패치를 evaluation하는 방법을 나타내고 있

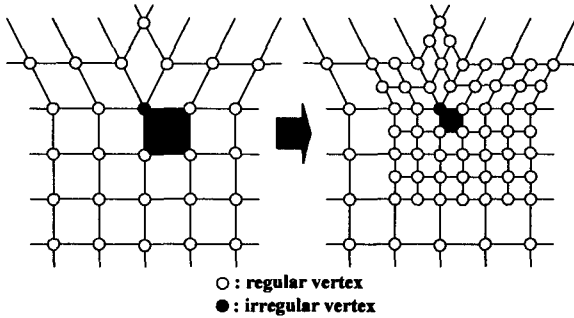


그림 3. Irregular 정점을 포함하고 있는 사각형 패치

다. 그림3에서처럼 만약 irregular 정점을 포함하고 있는 패치(왼쪽 그림)가 있다면, 우선 그 패치를 한번 서브디비전 한다(오른쪽 그림). 그러면 세 개의 정규 패치들과 한 개의 semi-regular 패치가 만들어진다. 세 개의 정규 패치들은 포워드 디퍼렌싱을 이용해서 렌더링하고, 나머지 하나의 semi-regular 패치는 위의 과정을 반복한다. 만약 위의 과정이 서브디비전 깊이 만큼 반복되었다면, 더 이상 서브디비전 하지않고 남은 하나의 사각형(quad)을 렌더링 한다. 이와 같은 방법으로 irregular 정점을 포함하고 있는 사각형 패치들을 evaluation 하면 된다.

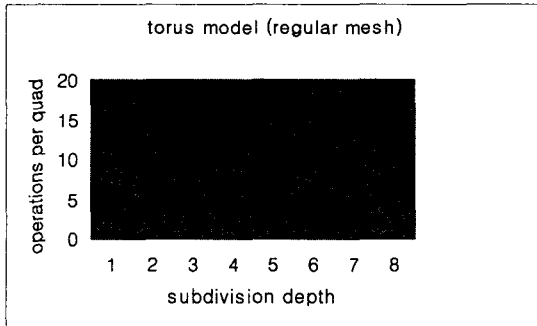


표 1. 16개의 정규 사각형 패치들로 이루어진 torus 모델에 대한 사각형 당 덧셈 연산수 (단, 반복문의 인덱스 증가와 포워드 디퍼렌싱을 setup 하는데 소요된 연산수는 제외)

5. 결론

본 논문은 Catmull-Clark 서브디비전 서피스를 포워드 디퍼렌싱을 이용해서 렌더링하는 알고리즘을 제안했다. 표1을 보면 서브디비전 깊이(subdivision depth)가 5이상이면 사각형(quad) 당 덧셈 연산수가 3임을 알 수 있다. 즉, 포워드 디퍼렌싱 계산을 병렬처리 하드웨어로 한다면 클럭 사이클(clock-cycle) 당 하나의 사각형을 렌더링 할 수 있다는 것을 의미한다. 이 결과는 Loop 기법[1]의 경우와 같은 결과이다.

향후 연구로는 [1]과 마찬가지로 본 알고리즘에 정확한 서피스의 normal을 계산하는 부분을 추가 시키는 것과 3차원 그래픽스 하드웨어 가속기에 적합한 서브디비전 모듈을 설계하는 것이다.

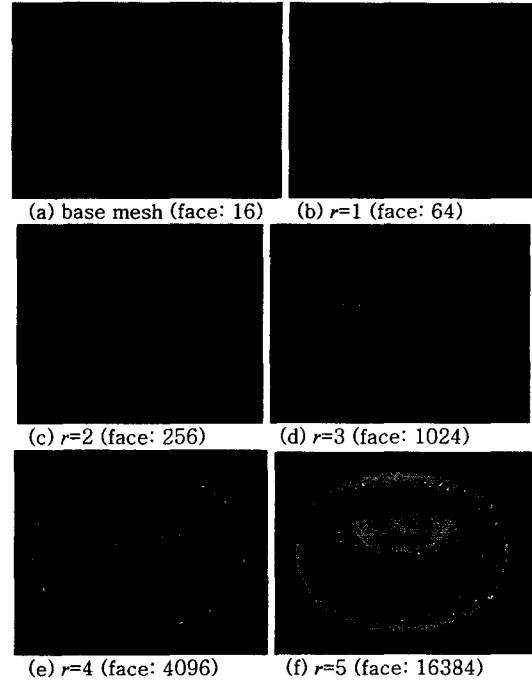


그림 4. torus 모델에 대한 서브디비전 결과

참고문헌

- [1] BISCHOFF, S., KOBELT, L. P., AND SEIDEL, H.-P. "Towards Hardware Implementation Of Loop Subdivision", *2000 SIGGRAPH / Eurographics Workshop on Graphics Hardware* (2000), 41- 50.
- [2] CATMULL, E., AND CLARK, J. "Recursively Generated B-Spline Surfaces on Arbitrary Topological Meshes", *Computer Aided Design 10*, 6 (1978), 350- 355.
- [3] DEROSE, T., KASS, M., AND TRUONG, T. "Subdivision Surfaces in Character Animation", *Proceedings of SIGGRAPH 98* (1998), 85- 94.
- [4] BIERMANN, H., LEVIN, A., AND ZORIN, D. "Piecewise Smooth Subdivision Surfaces with Normal Control", *Proceedings of SIGGRAPH 2000* (2000), 113- 120.
- [5] JOS, STAM. "Exact evaluation of Catmull-Clark subdivision surfaces at arbitrary parameter values", *SIGGRAPH 98 Conference Proceedings*, pages 395- 404, August 1998.
- [6] ZORIN, D., AND SCHRODER, P., Eds. "Subdivision for Modeling and Animation", *Course Notes. ACM SIGGRAPH*, 2000.
- [7] JEFFREY, BOLZ., SCHRODER, P., "Rapid Evaluation of Catmull-Clark Subdivision Surfaces", *Proceedings of SIGGRAPH 2001* (2001)