극한 표면의 근접거리 메트릭를 이용한 적응적 Loop 메쉬분할법

정 원 기　　김 창 헌
{wkjeong, chkim}@cgvr.korea.ac.kr
고려대학교 컴퓨터학과

The Adaptive Loop Subdivision using the Closeness Distance of Limit Surface

Won-Ki Jeong　　Chang-Hun Kim
Dept. of Computer Science & Engineering, Korea University

요 약

본 논문은 메쉬 분할시 생성되는 정점들과 그 점들의 극한위치와의 차이로 정의되는 근접거리 메트릭을 이용한 적응적 Loop 메쉬분할법을 제안한다. 근접거리 메트릭은 모든 approximation 분할법에 적용가능하며, 이 메트릭을 이용하여 초기메쉬를 사용자 입력 허용치에 따라 적응적으로 분할하여 적은 데이터로 극한메쉬에 근접한 결과를 생성할 수 있다. 또한, 본 논문에서 제시한 적응적 Loop 분할법은 다단계 메쉬표현이나 메쉬 편집 등 Loop 메쉬분할법이 사용되는 알고리즘에 유용하게 적용시킬 수 있다.

1. 서론

메쉬분할법(subdivison)은 주어진 단순한 메쉬에 새로운 점들을 추가하여 부드러운 입체를 만드는 방법을 말한다. 메쉬분할법은 지역적으로 계산을 하고, 주어진 규칙에 따라서 새로운 점을 생성시키고, 기존의 점을 움직이기만 하면 되므로 기존의 곡선에 기반한 곡면을 이용한 방법보다 간단하며, 부드러운 결과를 쉽게 얻기 위하여 모델링에서 널리 쓰이는 방법이다. 그러나, 매 단계마다 거의 4 배씩 데이터의 크기가 증가하므로 몇 차례의 분할 후에는 많은 양의 데이터를 다루어야 하는 단점이 있다.

적응적 메쉬분할법은 필요한 지역에만 분할을 행함으로써 각 레벨마다 새로이 생성되는 불필요한 데이터의 크기를 줄이고 필요한 부분을 더욱 자세히 표현할 수 있다. 즉, 지역적 평편도(local flatness)나 원래 형상과의 가까운 정도를 측정하여 이를 기준으로 적응적인 분할을 행함으로써 균등하게 분할한 경우와 비슷한 결과를 적은 데이터로 표현할 수 있거나, 같은 크기의 데이터를 가지고도 더욱 자세한 표현이 가능하게 된다.

이 논문에서는 새로 생성될 점의 위치와 그 점의 극한위치와의 거리의 차이를 이용한 근접거리 메트릭(Closeness Distance Metric)을 정의하고, LOD 정보가 없는 초기메쉬를 위한 적응적 Loop 메쉬분할법을 소개한다. 이 방법은 극한위치와의 거리를 이용하여 적응적 분할을 하므로 극한표면에 근사한 결과를 얻을 수 있으며, 다단계 메쉬표현 및 편집 알고리즘[2,4]등에도 간단히 적용될 수 있다.

2. 관련 연구

적응적 메쉬분할법에 관한 관련연구는 크게 나누어 지형모델 및 surface fitting, 렌더링, 그리고 삼각형 메쉬의 다단계 표현 및 에디팅에서 사용된다. 지형모델은 원하는 지형과 근사한 모델을 얻기 위해 초기 평면을 부분적으로 분할하여 근사한 모델을 얻는다[1]. 렌더링에서는 관찰자의 시점에 따라 다른 레벨의 분할이 필요한 것을 적응적 메쉬분할법으로 극복하였다[3]. 즉, 복잡도가 높은 지역은 많이 분할하고, 복잡도나 작은 지역은 덜 분할함으로써 여러 시점에서 본 각기 다른 장면도 적절한 레벨의 상세함을 가진 렌더링이 가능하게 되었다. Zorin et al.[4]은 삼각형 메쉬모델의 다단계 표현을 위한 adaptive synthesis 방식을 제시하였다. 여기서는 지역적 평편도를 메트릭으로 하여 갖고 있는 LOD 정보를 이용하여 사용자의 입력값을 만족시킬 때 까지 적응적 분할을 한다.

이러한 관련 연구들의 공통점은 히스토리가 있는 LOD 모델이나 지형모델과 같은 근사시킬 데이터를 갖고 있는 경우라는 것이다. 즉, 기존방법은 적응적 분할을 위한 메트릭의 계산을

위해 갖고 있는 데이터를 이용하지만, 일반적인 분할법을 이용한 시스템을 위해서는 초기메쉬만이 주어졌을 때에도 사용 가능한 메트릭이 필요하다. 그러므로 본 논문은 approximation 분할법에서 구할수 있는 극한위치를 이용한 메트릭을 사용하여 근사시킬 데이터가 없는 초기메쉬를 위한 적응적 분할법에 사용한다.

3. 적응적 Loop 메쉬분할법

3.1 Loop 메쉬분할법

이 논문에서는 근사분할기법 중 하나인 Loop 메쉬분할법[2,6]에 적응적 분할기법을 적용하였다. Loop 메쉬분할법은 다음과 같은 방법으로 새로운 점을 생성하고 기존의 점의 위치를 변경한다. Even vertex $v^i \in V^i$ 와 그 1-ring 이웃점 v_k가 주어지면 새로운 레벨의 점 v^{i+1}은

$$v^{i+1} = (a(k)+k)^{-1}(a(k)v^i + \textstyle\sum_{n=1}^{k} v_n), \quad a(k) = \frac{k(1-\alpha(k))}{\alpha(k)} \qquad (1)$$

$$\alpha(k) = \frac{5}{8} - \frac{(3+2\cos(2\pi/k))^2}{64}, \quad k = 1\text{-neighbor 개수}$$

Odd vertex v^{i+1}는 주변 점의 가중합으로 구해진다(그림 1).

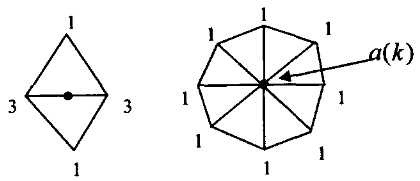


그림 1. Odd(좌) & Even(우)vertex 를 위한 분할 마스크

또한 어떤 한 점의 극한위치(limit position)을 구할 때에는 식 (1)에서 $a(k)$ 대신 $\omega(k) = 3k/8\alpha(k)$를 대입하여 구할 수 있다[2].

3.2 근접거리 메트릭

적응적 분할을 위해서는 분할될 지역을 선택할 기준이 있어야 한다. Zorin[4]은 지역적 편평도를 기준으로 하였다. 이 논문에서 제시하는 메트릭은 odd vertex와 그 점의 극한 위치와의 Euclidean 거리를 의미한다(그림 2). 여기서 근접거리 메트릭 d는 다음과 같이 정의 된다.

$$d = F\left\|v - v^{\infty}\right\| \quad , \ F: \text{normalize operator} \qquad (2)$$

Interpolation 분할법[5]의 경우 odd vertex가 극한위치가 되므로 본 논문과 다른 방식의 메트릭을 이용해야 한다. 그러나, Loop 분할법의 경우 이웃점들을 알면 극한위치를 구할 수 있으므로 이를 이용할 경우 극한메쉬에 근접한 결과를 얻을 수 있는 장점이 있다.

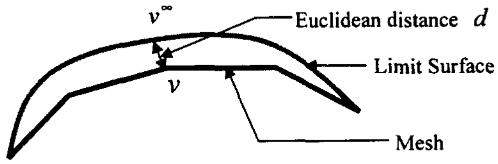


그림 2. 한 점 v와 그 점의 극한점 v^{∞}과의 Euclidean 거리

3.3 근접거리 메트릭을 이용한 적응적 메쉬분할 알고리즘

적응적 메쉬분할법의 전체적인 알고리즘은 다음과 같이 크게 세 단계로 구성된다. 기본메쉬가 주어지면 면 단위로 분할을 하게 되는데, 한 면의 세 변을 각각 분할하여 임시의 odd vertex를 3개 구하고, 이 점들에 대한 근접거리 메트릭을 구하여 사용자 허용치 이상의 점들만 택하고 나머지는 삭제한 후, 마지막으로는 택하여진 odd vertex들과 기존의 면을 구성하고 있던 세 꼭지점과 삼각화를 하여 새로운 면들을 생성한다.

3.3.1 Odd Vertex 생성

하나의 면(그림 3의 굵은선)이 입력되면 그 면의 모서리마다 생기게 될 임시의 odd vertex를 각각 구하여 temp[1] 부터 temp[3] 에 저장한다. 이를 위해서 주어진 면의 각 모서리에 접하는 1-neighborhood face 들을 구해야 한다(그림 3의 점선).

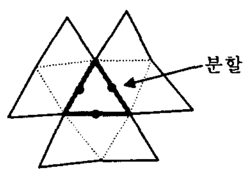


그림 3. 1-neighborhood faces

3.3.2 근접거리 메트릭을 이용하여 분할될 점 선택

식 (1)과 ω를 이용하여 temp[1]~temp[3]의 limit position을 각각 구하여 Limit[1]~Limit[3]에 저장한다. 이를 위해서는 그림 3의 점선으로 된 삼각형들의 1-neighborhood face들이 더 필요하게 된다. 즉, 결과적으로 주어진 하나의 면(그림 3 굵은선)을 분할하는데 필요한 인접면은 총 9개이다(그림 3). 이를 이용하여 odd vertex와 limit position과의 차이 d를 구한다. 이것을 사용자에게서 입력 받은 ε값과 비교하여 이보다 큰 점들은 극한위치와의 차이가 많으므로 남겨두고 반대로 작은 점은 극한위치에 근사했으므로 삭제한다.

For $i = 1,2,3$
if $d = \|temp[i] - Limit[i]\| < \varepsilon$ Take $temp[i]$
else discard

3.3.3 분할 후 새로운 면의 생성

입력으로 받은 하나의 삼각형을 적응적 메쉬분할법으로 세분화를 하였을 때 새로 생기는 odd vertex의 수는 각 면마다 다

르다. 즉, 한 삼각형의 세 변이 모두 분할 될 경우도 있고, 하나도 분할되지 않을 경우도 있다. 그러므로 그림 4와 같은 세 가지 경우에 있어서 각각의 경우에 적절하게 재삼각화를 하여 새로운 면을 생성한다.

a. 세 변 모두 분할될 경우

그림 4 (a)의 경우로 일반적인 방법과 같이 하나의 면이 4개의 면으로 나뉘게 된다.

b. 두 변만 분할될 경우

두 변만 분할될 경우는 그림 4 (b)와 같이 두가지의 경우가 생긴다. 두 경우 중 aspect ratio가 좋은 방향으로 삼각화를 한다. 삼각형의 aspect ratio는 다음 수식으로 구한다[3].

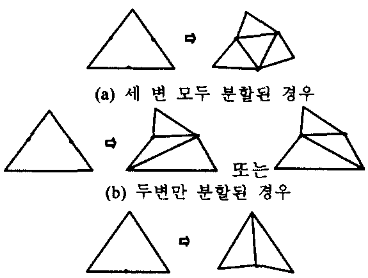
$$\text{Compactness} = \frac{4\sqrt{3}a}{l_0^2 + l_1^2 + l_2^2} \tag{3}$$

a: 삼각형의 넓이, $l_0 \sim l_2$: 삼각형의 세 변의 길이

Compactness가 1에 가까울수록 aspect ratio가 좋은 삼각형이다.

c. 한 변만 분할될 경우

그림 4 (c)와 같이 한 변만 분할될 경우는 분할되는 면의 마주보는 꼭지점과 연결해 준다.



(a) 세 변 모두 분할된 경우

(b) 두변만 분할된 경우

(c) 한변만 분할된 경우

그림 4. 세가지 재삼각화 경우

4 결과 및 결론

그림 5의 결과화면은 180Mhz R5000 CPU가 장착된 SGI O2 워크스테이션에서 OpenInventor 그래픽 라이브러리와 C++을 이용하여 구현하였다. 그림 5 (a)는 초기 cat 모델의 모습으로 366개의 점으로 이루어져 있다. (b), (c) 는 2단계 분할을 한 결과이며, (b)는 균등하게 분할하여 5649개의 점을 얻었고 (c)는 bound를 0.02를 주어 4100개의 점을 얻어서 약 28%의 데이터를 절약할 수 있었으며 외형상 비슷한 결과를 얻었다. (d)와 (e)는 (b)와 (c)의 머리부분을 확대하여 wireframe으로 표현한 모습으로 (d)는 균등하게 분할된 모습을 보여주고 (e)는 적응적으로 분할된 모습을 볼 수 있다.

본 논문에서 제시한 알고리즘은 LOD 정보가 없는 초기메쉬를 적응적으로 분할하며, Loop 메쉬분할법을 사용하는 기존 시스템에 그대로 적용하여 새로이 생성되는 데이터의 크기를 사용자 입력값에 따라 줄일 수 있다. 향후 연구로는 메쉬 분할법을 이용하는 편집 시스템이나 다단계 메쉬 표현에 적용시킬 수 있다.

참고문헌

[1] Gross, M.H., Gatti, R., And Staadt, O. *Fast Multiresolution Surface Meshing*. IEEE Visualization '95 Proceedings, pp. 135-142, 1995.

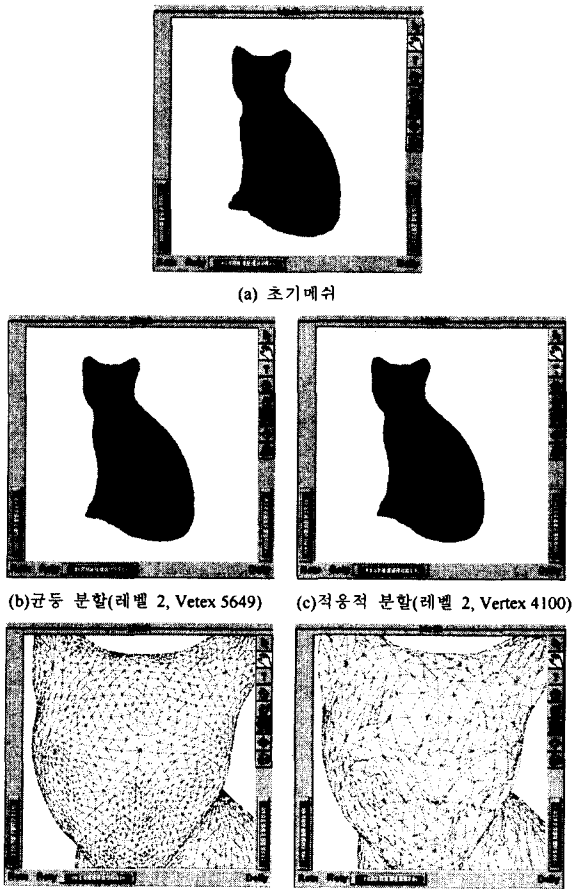
[2] Hoppe, H., DeRose, T., Duchamp, T., Halstead, M., Jin, H., McDonald, J., Schweitzer, J., And Stuetzle, W. *Piecewise Smooth Surface Reconstruction*. SIGGRAPH 94 Conference Proceedings, pp. 295-302, 1994.

[3] Xia, J., El-Sana, J., And Saxe, E. *Adaptive Real-time Level-Of Detail-Based Rendering for polygonal Model*. IEEE Transactions on Visualization and Computer Graphics, pp.171-183, 1997.

[4] Zorin, D., Schroder, P., And Swelden, W. *Interactive Multiresolution Mesh Editing*. SIGGRAPH 97 Conference Proceedings, pp. 259-268, 1997.

[5] Zorin, D., Schroder, P., And Swelden, W. *Interpolating Subdivision for Meshes with Arbitrary Topology*. SIGGRAPH 96 Conference Proceedings, pp. 189-192, 1996.

[6] Zorin, D., Schroder, P. *Subdivision for Modeling and Animation*. SIGGRAPH 99 Course Note, 1997.



(a) 초기메쉬

(b)균등 분할(레벨 2, Vetex 5649) (c)적응적 분할(레벨 2, Vertex 4100)

(d) 균등분할 (e) 적응적 분할

그림 5. Cat 모델을 이용한 균등분할 및 적응적 분할 결과