

적응형 세분화를 이용한 3D 메쉬의 기하데이터 압축

(Adaptive Subdivision for Geometry Coding of 3D Meshes)

이 혜 영 [†]

(Haeyoung Lee)

요약 본 논문에서는 3차원 메쉬의 기하데이터 압축을 위한 새로운 알고리즘을 소개하고자 한다. 광역좌표계에 의거한 기하데이터 압축방법은 구현이 쉽고 단순하게 양자화가 결정 되지만 압축효율은 지역좌표계를 이용한 방법보다 떨어지는 단점이 있다. 반면에 지역좌표계에 기초한 방법은 광역좌표계 방법보다 압축효율은 우수하나 양자화가 사용자의 시행착오에 전적으로 의존하므로, 비체계적이고 시간이 많이 소요되는 단점이 있다. 본 논문에서는 지역좌표계영역에 적응형 세분화를 도입하여 체계적인 양자화가 가능하도록 하였다. 또한 문맥 모델링기법을 적용하여 연결데이터 압축효율도 더욱 향상시켰다. 결과적으로, 본 논문의 새로운 압축 알고리즘은 압축 효율성을 유지 하면서, 동시에 체계적이고 직관적인 방법으로 왜곡률과 압축률간의 균형을 제어할 수 있도록 하여 알고리즘의 신뢰성을 높였다.

키워드 : 3차원 메쉬, 기하데이터 압축, 문맥 모델링

Abstract We present a new geometry coding method for 3D meshes, an adaptive subdivision. Previous localized geometry coding methods have demonstrated better compression ratios than the global approach but they are considered hard to use in practice partly due to time-consuming quantization. Our new localized scheme replaces this quantization with an adaptive subdivision of the localized range. The deeper level a user chooses, the closer to the original the mesh will be restored. We also present an improved connectivity coder upon the current leading Angle-Analyzer's by applying a context-modeling. As a result, our new coder provides reliable and intuitive controls between bit-rate and distortion without losing efficiency.

Key words : 3D Mesh, Geometry Coding, Context modeling

1. 서론

최근 3차원 센싱 및 스캔 등 관련 기술의 발전에 따라, 섬세하고 복잡한 3차원 모델을 쉽게 만들 수 있게 되었다. 이에 따라 생성된 모델의 거대한 데이터 크기를 줄여 저장 및 전송이 용이하도록 다양한 압축[1-7] 알고리즘이 발표되고 있다. 현재 압축률이 가장 중시되고 있지만, 외부 메모리사용[8], 여러 복원력, 임의의 접근성[9] 등 다른 특징을 가진 알고리즘도 발표되고 있다.

최근, 기하데이터 압축 시 지역좌표계를 이용한 방법인 [4,10,6,11] Touma-Gotsman[1]의 광역좌표계를 이용

한 방법보다 압축률이 우수하다는 것을 보여주었다. 하지만 지역좌표계를 이용한 알고리즘은 적절한 압축률과 왜곡률을 위한 양자화수를 찾는 데 사용자의 많은 시행착오에 의존하기 때문에 실제로 사용하기에는 어려움이 있다. 본 논문에서는 지역좌표계를 사용해 압축 효율을 유지하면서도 사용이 쉽고 간단한 새로운 압축 알고리즘을 제안하고자 한다(그림 1, 2, 표 3 참조). 또한, 기존의 우수한 압축 방법인 Angle-Analyzer[4]의 연결데이터 압축알고리즘에 문맥 모델링(context-modeling)을 적용하여 압축률을 더욱 향상시켰다. 본 논문의 지역좌표영역 세분화 알고리즘은 지역좌표계를 사용하는 다른 압축알고리즘[3,11,6,10]에도 적용될 수 있다.

1.1 관련연구

기하데이터 압축은 1998년에 제안된 Touma-Gotsman [1]의 선형예측기법을 이용한 광역좌표계 알고리즘이 가장 많이 사용되었으나, 이보다 압축률이 우수한 지

· 이 논문은 2005학년도 과학재단의 국제협력(공동)연구기금 (KOSEF-042010702) 지원으로 수행되었음

† 정 회 원 : 홍익대학교 컴퓨터공학과 교수
leeh@cs.hongik.ac.kr

논문접수 : 2005년 8월 23일

심사완료 : 2006년 5월 16일

역좌표계를 이용한 알고리즘이[4,11,6,10] 최근 발표되었다.

광역좌표계를 이용한 알고리즘이 모델 전체를 포함하는 영역의 각 축 범위를 8~12bit중 하나로 각 축을 동일하게 양자화(quantization) 하는데 반해, 지역좌표계를 이용한 알고리즘은 각 면에 지역좌표계를 정의하여 광역좌표 값을 지역좌표 값으로 변환하고 변환된 모든 점을 포함하는 한 개의 지역영역을 생성한 후 이 범위를 양자화 한다(그림 2). 지역좌표계를 이용한 알고리즘은 광역좌표계 보다 작은 영역을 각 축의 크기에 따라 다르게 양자화 함으로써, 압축률을 향상시켰다. 하지만, 양자화 수가 모델에 따라 다르고(표 3) 사용자의 시행착오에 의존하므로, 왜곡률을 감안한 적절한 양자화수를 찾는 것은 시간이 많이 걸리는 작업이다.

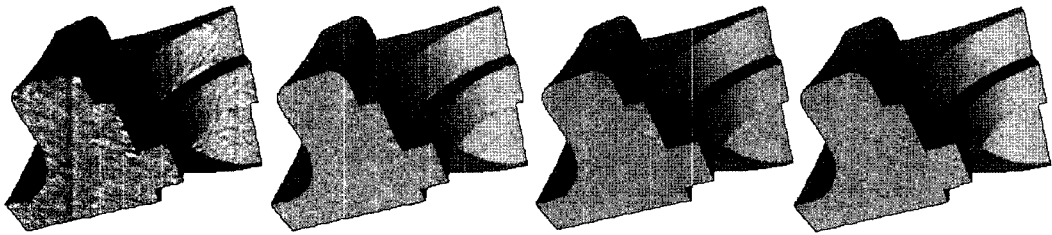
연결데이터 압축은 상대적으로 활발한 연구가 진행되었다[1-7]. 알고리즘 뿐만 아니라 이론적인 압축률의 한

계도 산출되었다. 주로 하드웨어의 지원을 받는 삼각 메쉬를 위한 방법이 제시되었으나, 최근 다각형 메쉬를 위한 방법도 발표되고 있다[12,13,7].

문맥 모델링의 개념은 2차원 이미지 프로세싱[14]에서 나왔다. 주변 화소(pixel)의 색과 현재 화소의 색 간에 관계가 있다는 개념으로 주변 화소의 색을 문맥(context)으로 사용하여 현재 화소의 색을 암호화한다. 즉, 암호화할 코드에 대한 조건부 확률을 산출하여 다음 코드의 예측 정확도를 높여준다. 최근 이러한 문맥 모델링을 3차원 등가곡면 모델에 적용하여 압축률을 향상시킨 방법이 발표되었다[15,10].

1.2 알고리즘의 개요

본 논문에서 제시된 3차원 메쉬 압축은 현재 최고의 압축률을 도출한 단순압축 알고리즘인 Angle-Analyzer [4]에 기초한다. 메쉬내의 모든 면을 순회하면서 압축을 수행하는데, 방향이 있는 선분(edge)인 게이트(gate)를



SubLevel	4	5	6	7
Geometry	6.51 b/v	8.26 b/v	10.64 b/v	13.74 b/v
Distortion	0.000304	0.000105	0.00004	0.000018

그림 1 지역좌표영역 적용형 세분화 적용 결과(Fandisk model) : 본 방법은 세분화 단계만으로 왜곡률과 압축률 간의 균형을 직관적으로 조절한다. 동일한 왜곡률의 경우, Angle-Analyzer[4]에 의해 압축한 경우 11.66b/v, Touma-Gotman[1]에 의한 13.79b/v보다 낮은 10.64b/v (sublevel 6)의 압축률을 도출한다. 왜곡률은 Metro[16]로 측정.

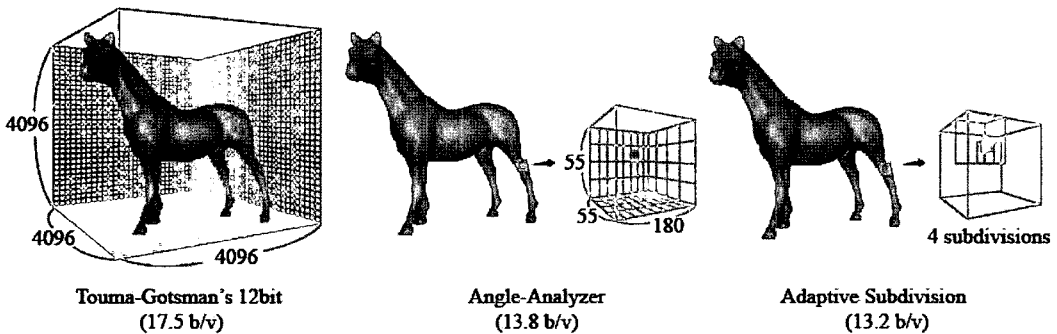


그림 2 기하데이터 압축 비교: Touma-Gotman은 광역영역을 균일하게 양자화 한다. Angle-Analyzer는 변환된 지역좌표영역의 비균일 양자화로 압축효율을 높였으나, 사용자의 시행착오에 의존하므로 실제 사용 시 어려움이 있다. 본 논문의 방법은 지역좌표영역 내의 점의 위치에 따른 적응적 세분화를 도입하여 압축률도 높이고 양자화도 용이하도록 하였다.

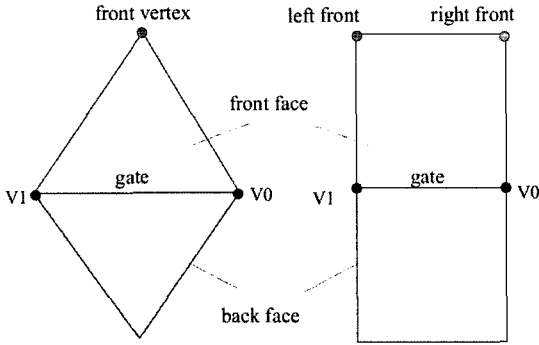


그림 3 용어 정의 (삼각/사각 메쉬): 게이트(gate)는 점 v0에서 점v1 방향으로 연결된 선분(edge). 뒷면(back face)은 이미 압축 처리된 면이고, 앞면(front face)은 현재 처리할 면이다. 게이트내의 두 점과 앞점(front vertex) 또는 왼쪽 앞 점(left front), 오른쪽 앞 점(right front) 간의 연결 상태는 연결데이터 압축으로, 앞 점의 위치는 기하데이터 압축으로 처리한다.

따라 면을 순회한다(그림 3). 생성된 게이트들은 리스트로 연결하여 관리한다.

Angle-Analyzer의 압축/해제 절차는 다음과 같다.

수도코드

다음 반복

초기화단계:

- 아직 압축되지 않은 컴포넌트를 가져온다.
- 1. 임의의 초기면(seed face)을 선택한다.
- 2. 초기면의 3개 또는 4개의 게이트를 연결하여 하나의 게이트 리스트를 만든다.
- 3. 스택에 생성된 게이트 리스트를 저장한다.

메쉬 순회(mesh traversal) 단계:

다음 반복

스택에서 한 개의 게이트 리스트를 꺼낸다.

다음 반복

- 1. 리스트내 게이트간의 연결각도를 계산하여 각도가 가장 작은 게이트를 선택.
- 2. 선택된 게이트의 앞면(front face)이 아직 처리되지 않았다면
 - 연결데이터 압축/해제를 한다.
 - 연결데이터 처리 결과에 따라 새 게이트 추가 또는 기존 게이트 삭제 실시.
 - 앞점(front vertex)이 처리되지 않았다면, 기하데이터 압축/해제를 수행. (*)
- 3. 처리된 게이트를 리스트에서 삭제한다.

더 이상 처리할 게이트가 없을 때 까지 반복.
 더 이상 처리할 게이트 리스트가 없을 때 까지 반복.
 더 이상 처리할 컴포넌트가 없을 때 까지 반복.

Angle-Analyzer는 위의 수도코드를 두 번 반복하는 two-pass 알고리즘이다. 첫 번째 패스에서는 기하데이터 압축을 위해 각 점의 좌표 값을 지역좌표계로 변환하고 변환된 모든 점을 포함하는 지역좌표영역을 구한다. 두 번째 패스에서 실제적인 기하압축이 수행된다. 압축해제는 똑같은 과정을 거쳐 게이트 리스트를 생성하고 게이트를 선택하여 연결데이터 및 기하데이터를 압축해제 한다.

본 논문의 새로운 기하압축 방법은 위의 수도코드에서 (*)부분에 적용된다. 본 방법의 핵심은 지역좌표영역내의 각 점의 위치에 따라서 세분화될 영역의 크기를 조정하고, 세분화 정도를 달리하는 것이다. 세분화시 점이 속한 내부 셀의 숫자를 부호화한다(그림 2, 5의 우측). 압축 시 Angle-Analyzer에서 지역좌표영역의 각 축을 양자화하기 위한 3개의 양자화수를 사용자입력으로 처리했던 것에 비해, 사용자로부터 오직 하나의 입력 즉, 세분화 단계만 있으면 된다(표 3). 그림 1에서 알 수 있듯이, 세분화 단계를 높일수록 같은 범위를 더 세밀하게 세분화하므로, 원형모델과의 왜곡이 감소된다. 본 기하압축 알고리즘은 왜곡률과 압축률간의 균형이 세분화단계에 의해 조정되므로 체계적이고 직관적으로 모든 모델에 일관적으로 적용할 수 있다. 또한, 연결데이터 압축은 기존의 Angle-Analyzer에 메쉬의 기하학적 특성을 이용한 문맥 모델링을 적용하여 압축률을 향상시켰다.

2. Angle-Analyzer(AA) 개요

AA는 하나 또는 여러 개의 컴포넌트형태로 된 삼각, 사각 및 삼각-사각 복합 메쉬를 압축할 수 있다. 본 방법은 삼각 메쉬에 중점을 두어 설명하나, 사각 및 삼각-사각 복합 메쉬에 동일하게 적용된다.

2.1 AA의 연결 데이터 압축

게이트와 앞 점간의 연결상태는 삼각메쉬의 경우 그림 4에서 나타난 것처럼 5개의 부호로 압축한다(사각메쉬는 8개, 그리고 복합메쉬는 12개의 부호로 나타낸다[4]).

앞 점이 처리되지 않은 새로운 점이라면 연결정보는 C로 부호화하고, 기하압축이 실행된다. 앞 점이 이미 처리된 경우는 현재 게이트를 기준으로 시계방향 또는 반시계방향으로 회전하여 찾을 수 있는 경우 각각 CW 또는 CCW로 부호화 한다. 게이트가 경계에 속해서 앞면이 없고 그 결과 앞 점이 없다면 S로 부호화된다. 앞 점이 이미 처리되었으나 회전으로 찾을 수 없는 경우 J로 부호화하고 점의 위치는 게이트에 속한 한 점과 나

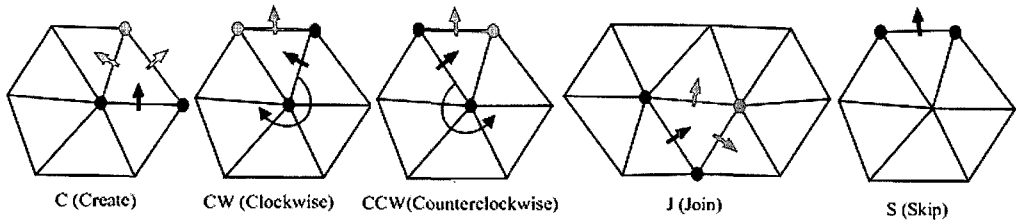


그림 4 AA의 삼각메쉬 연결 데이터 압축 부호: C는 새로운 앞점 일 때, CW 또는 CCW는 시계방향이나 반시계 방향으로 검색하여 앞점을 찾을 수 있는 경우, S는 앞점이 없을 때, J는 앞점이 이미 처리되었으나, 회전으로 찾을 수 없을 경우를 의미

머지 점간의 거리에 따라 정렬된 배열의 인덱스를 추가로 압축한다. J코드는 인덱스가 있어야 하고, 이를 위한 추가적인 처리를 하므로, J코드 발생을 줄여야 압축률 및 처리 속도를 향상시킬 수 있다. 이를 위해 AA는 현재 게이트와 다음게이트간의 연결각도가 가장 작은 게이트를 다음에 처리할 게이트로 선택하도록 하는 적응형(adaptive) 메쉬 순회 방법을 제시하였다. 메쉬의 기하학적 특성을 이용한 이 방법은 게이트 리스트의 오목한 (concave) 부분을 먼저 방문하게 되므로, J코드의 발생을 줄이고, 대신 CW, CCW코드를 발생시킴으로서 압축률을 향상시켰다. 임의의 초기면(seed face)의 3점과 지역좌표영역 값은 헤더파일에 저장하여 전달되고 헤제기는 위와 동일한 규칙으로 게이트 리스트를 만들고 동일한 적응적 순회를 통해 동일한 순서로 메쉬의 각 면의 연결데이터를 복원한다.

2.2 AA의 기하 데이터 압축

기하데이터 압축을 위해, AA는 두 가지 알고리즘을 제안하였다. 하나는 지역좌표계를 이용한 알고리즘이고, 다른 하나는 각(angle)을 이용한 알고리즘이다. 본 논문의 새로운 알고리즘은 전자를 개선한 것이므로, 본 논문에서는 지역좌표계를 이용한 알고리즘에 대해서만 설명한다. 각을 이용한 알고리즘은 논문[4]을 참고하기 바란다. 그림 5의 좌측에 보이는 것처럼 지역좌표계는 각 뒷면에 정의된다. 현재 게이트가 지역좌표계의 x축이 되고, 뒷면의 법선 벡터를 기준으로 -90도 회전시킨 것이 y축이 된다. x축과 y축을 외적인 벡터가 z축이 된다. 광역좌표계로 나타낸 앞 점의 좌표 값을 새로 정의된 지역좌표계의 좌표 값으로 변환한다. 첫 번째 패스에서는 변환된 모든 점을 포함하는 지역좌표계의 범위를 구하고, 두 번째 패스에서는 이 범위를 양자화하여 점의 위치를 압축하게 된다.

기하압축해제는 기하압축과 대칭적으로 수행된다. 뒷면과 현재의 게이트를 이용하여 지역좌표계를 생성하고 단위벡터를 구한 후, 해제된 앞 점의 좌표 값을 광역좌표계의 좌표 값으로 복원한다.

3. 지역좌표영역 세분화 기법

지역좌표계를 이용한 알고리즘은 광역좌표계를 이용한 알고리즘보다 압축률이 더 높다는 것이 제시되었다. 특히 [11,4,10]에서 알 수 있듯이, 비균일적 양자화는 가장 높은 압축률을 보여준다. 지역좌표계를 이용한 기하 데이터 압축 알고리즘은 보통 다음의 단계들을 거친다.

- 1) 지역좌표계를 정의한다.
- 2) 광역좌표값을 지역좌표값으로 바꾼다.
- 3) 지역좌표값을 양자화한다. (*)
- 4) 양자화된 코드들을 암호화한다. (**)

(*)표시된 3번째 단계에서 적합한 압축률과 왜곡률을 위한 적절한 양자화 수를 찾는 것에 많은 시간을 소비해야 한다. 표 3에서 볼 수 있듯이, 양자화 수의 범위는 모델에 따라 달리 결정된다. 사용자는 다양한 양자화수를 넣어서 압축을 하고, 압축을 해제한 후 왜곡률을 측정하는 과정을 만족할만한 왜곡률과 압축률을 가진 결과가 나올 때까지 반복해야 한다.

본 논문의 새로운 기하압축 알고리즘은 (*)표시된 양자화를 영역세분화로 개선하였다. 세 개의 양자화 수 대신 세분화 단계를 나타내는 한 개의 입력이면 충분하다. 사용자가 더 깊은 단계를 선택할수록 원래모델에 가깝게 복원된다. 또한, 영역내의 점의 위치를 네 개의 타입으로 분류하여 적응형 세분화를 도입하였다(표 1, 그림 5의 우측). 영역내의 점의 위치를 통계적으로 분석한 결과, 평균적으로 아주 적은 수(전체의 0.14%)의 점으로 인해 전체영역의 크기가 커짐을 알 수 있다. smallest 범위의 크기는 largest 범위의 크기의 8분의 1로 설정하여 smallest에 속한 점은 largest에 속한 점보다 세분화를 3번 적게 하도록 영역의 범위를 구분 하였다.

한 점이 어느 타입에 속하는지는 가장 작은 영역에서 큰 영역으로 포함 여부를 점검하는 다음의 연산을 통해 정해진다(그림 5 우측).

```
if ( localrange* 7/16 < localvalue < localrange * 9/16)
then smallest에 속함 (세분화단계 = 주어진 단계 - 3)
else if ( localrange* 4/16 < localvalue < localrange *
```

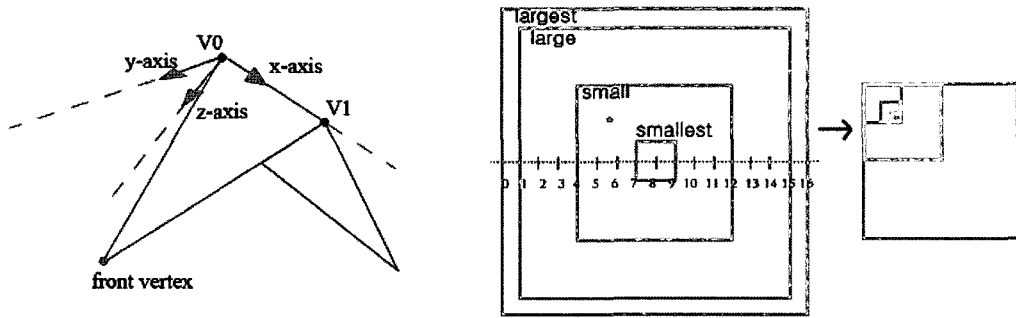


그림 5 좌측: AA의 기하데이터 압축을 위한 지역좌표계 정의 방법. 우측: 점의 위치에 따라 적응형세분화를 위한 네 타입으로 분류된 지역좌표영역

12/16)

```

then small에 속함 (세분화단계 = 주어진 단계 - 2)
else if (localrange * 1/16 < localvalue < localrange *
15/16)
    then large에 속함 (세분화단계 = 주어진 단계 - 1)
    else largest에 속함 (세분화단계 변화 없음)
    
```

예로, 주어진 세분화 단계가 6인 경우, 앞 점이 속한 범위가 large이면 5번 세분화, small이면 4번 세분화, smallest이면 3번의 세분화를 하면 된다. 대부분의 점이 small 이나 large에 속하므로, 압축을 할 때 위치 타입의 순서열의 낮은 엔트로피도 압축률을 향상시키는데

기여한다. 또한, 세분화되는 셀은 $x*y*z$ 에 대하여 $2*2*4$ 로 세분화함으로써 AA 기하압축의 비균일적 양자화를 시뮬레이션 하였다. 다양한 형태의 세분화를 시도할 수 있지만, $2*2*4$ 만으로도 충분히 AA보다 더 높은 압축률을 얻을 수 있었다. 생성되는 부호는 점의 위치타입(0~3)과 점이 속한 세분화된 내부 셀의 위치(0~15)이다. 하나의 세분화단계로 세분화하는 것보다 적응형세분화가 압축효율이 향상되는 것을 표 2의 실험결과가 보여준다.

입력된 세분화단계와 지역좌표영역은 헤더파일에 저장되어 해제기에 전달되고 이를 이용 압축과정의 역으로 압축을 해제할 수 있다.

표 1 모델별 지역좌표영역 내 점의 분포도

모델	점의 수	largest	large	small	smallest	largest %
fandisk	6475	27	587	5717	141	0.4%
horse	19851	3	8929	10884	38	0.02%
maxpl.	25445	26	9230	16020	166	0.1%
tf2	14169	8	11326	2734	98	0.05%
dino	14070	4	3054	10974	35	0.03%
feline	19864	11	1812	46037	2001	0.05%
venus	8268	29	984	7180	72	0.35%

지역좌표영역내의 점의 위치에 따라 largest, large, small, smallest의 네 타입으로 분류. 영역의 크기를 결정하는 largest에 포함된 점의 개수가 아주 작음을 알 수 있다.

표 2 단순 세분화와 적응형 세분화간의 기하데이터 압축률 비교표

모델명	점의 갯수	세분화 단계	Subdivision	Adaptive Subdivision(AS)
			기하데이터압축률(b/v)	기하데이터압축률 (b/v)
bunny	35947	6	16.1	11.3
horse	19851	6	16.7	12.2
maxplanck	25445	6	19.6	15.1
tf2	14169	6	16.3	12.4
dinosaur	14070	7	23.1	18.2
feline	19864	6	18.95	13.9
venus	8268	7	23.4	18.5

동일한 단계로 단순 세분화 한 경우 생성되는 부호의 수가 적응형세분화보다 훨씬 많다. 적응형 세분화의 경우 위치타입이 추가되나 위치타입 열의 엔트로피가 낮고 생성되는 부호의 수가 훨씬 적어 압축률이 향상됨을 알 수 있다.

결과적으로 본 논문의 지역좌표영역 세분화에 기초한 알고리즘은 표 3에서 볼 수 있듯이 압축효율을 유지하면서, 그림 1 및 그림 2와 같이 왜곡률과 압축률간의 직관적인 조절이 가능하도록 하여 구현 및 사용 시 용이성을 향상시켜 알고리즘의 신뢰성을 높였다.

4. 문맥 모델링

엔트로피 기반 Arithmetic Coder[17]는 기하데이터 및 연결데이터 압축과정에서 생성된 순차적인 코드들을 암호화하기 위해서 사용되었다. 최근, 2D 이미지 프로세싱[14]에서 사용되었던 문맥 모델링을 등가ופן 압축[15,10] 등에 사용하여 더 높은 압축률을 얻을 수 있었다. 이에, 3차원 메시 압축에 확장 적용하기 위한 실험을 하였다. (**)표시된 3절의 4번째 단계인 암호화과정에 적용되는 기법으로 압축률 향상에 기여하였다.

연결데이터 압축을 위한 문맥 AA는 연속된 게이트 사이의 각도를 기반으로, 다음에 처리할 게이트를 결정한다. 우리는 이 각도가 연결데이터 부호와 관계가 있다는 것에 주목했다. 각도가 60도 보다 작으면 주로 CW, 각도가 200도 이상이면 주로 C가 발생한다. 따라서 우리는 이전각도, 이전코드, 현재각도 등의 세 값을 문맥으로 사용했다. 표 3에서 알 수 있듯이, 연결데이터 압축에 있어서 AA보다 평균 19%의 압축률을 낮추었다.

기하데이터 압축을 위한 문맥 기하압축에도 문맥 모델링을 적용하였다. 여러 시도 중 연속된 3개의 상위 내부 셀 번호를 문맥으로 사용한 경우 최고의 압축률을 보여주었다. 이는 세분화시 점의 위치를 추적하는데 패턴이 있음을 알 수 있다.

5. 결과

지역좌표영역 세분화 기법은 기존의 우수한 기하데이터 압축알고리즘인 AA보다 평균적으로 약간 향상된 압

축률을 도출한다. 비록 압축효율성면에서는 개선점이 미미하지만, 용이성 및 알고리즘의 신뢰성면에서는 AA보다 많이 나아진 것을 알 수 있다. 사용자는 모든 모델에 대해서 같은 세분화 단계로 압축을 시도 하고, 압축을 해제한 후 왜곡률을 측정한다. 만족할 만한 결과가 나오지 않으면, 세분화 단계만 증가하여 압축하면 된다. 그림1에서 볼 수 있듯이, 왜곡률과 압축률간의 균형을 체계적으로 조절이 가능하다. 문맥모델링을 적용하여 연결데이터의 압축률도 기존 보다 평균 19% 향상시켰다. 실험에 사용된 모델은 미국 Caltech의 그래픽스 연구실과 Standford 그래픽스 연구실에서 제공된 데이터를 사용하였다.

본 압축방법은 게이트를 따라 모든 면을 중복되지 않게 한 번씩만 순회하면서 연결데이터 및 기하데이터의 압축이 수행된다. 한 개의 게이트에 2개의 점이 속하므로 압축 및 해제에 필요한 처리시간은 모델에 속한 점의 개수가 V라고 할 때 O(V)이다.

6. 결론 및 향후 연구과제

본 논문에서는 개선된 AA의 연결데이터 압축방법과 지역좌표영역 세분화에 기초한 새로운 기하압축 방법을 제시하였다.

광역좌표계를 이용한 기하압축 알고리즘은 비록 압축률은 떨어지지만, 사용이 간단하고 압축률과 왜곡률간의 직관적인 조절이 가능한 장점이 있다. 이에, 본 논문은 압축률이 우수한 지역좌표계에 기초한 알고리즘을 개선하여 압축효율을 유지하면서, 동시에 사용의 용이성 및 신뢰성을 높이는 방법을 제시하였다. 지역좌표영역내의 점의 분포에 대한 관찰을 통해 적응형 세분화를 제안하였고, 비균일적 세분화를 통해 비균일적 양자화를 시뮬레이션 하였다. 결과적으로 본 방법을 통해 지역좌표계 방법의 장점인 압축효율을 유지하면서 광역좌표계방법

표 3 압축결과 비교표

모델명	점의 갯수	Angle-Analyzer(AA)			Adaptive Subdivision(AS)		
		연결데이터 압축률(b/v)	기하데이터압축률(b/v)	양자화수 (x, y, z)	연결데이터 압축률(b/v)	기하데이터압축률(b/v)	세분화 단계
fantisk	6475	1.2	11.9	(50,50,280)	0.95	10.6	6
bunny	35947	0.9	12.3	(80,80,160)	0.8	11.3	6
horse	19851	1.4	12.5	(55,55,180)	0.94	12.2	6
maxplanck	25445	1.4	13.8	(50,50,90)	1.21	15.1	6
tf2	14169	1.0	11.5	(60,60,120)	0.6	12.4	6
dinosaur	14070	1.7	17.0	(100,100,200)	1.38	18.2	7
feline	19864	1.5	13.8	(60,60,120)	1.21	13.9	6
venus	8268	2.0	19.6	(110,110,300)	1.75	18.5	7

압축률 비교표. AA는 사용자의 시행착오에 의존한 세 개의 양자화수를 입력으로 요구하는 반면 본 논문에서 제안된 AS는 한 개의 입력(세분화단계)이면 충분하다. 동일한 왜곡률을 (Touma-Gotsman의 12bit양자화 경우) 갖도록 조정했으며, 왜곡률은 Metro[16]에 의해 측정하였다. 모델은 (그림 6) 참조

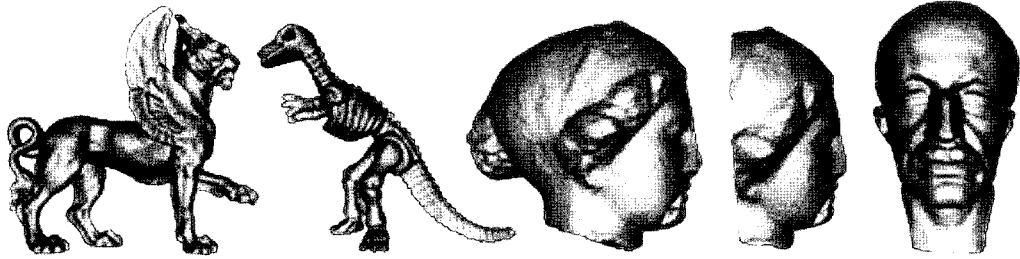


그림 6 실험에 사용된 모델: 좌로부터 feline, dinosaur, venus, tf2, maxplanck

처럼 체계적으로 압축률과 왜곡률간의 균형을 조절할 수 있게 되었다. 또한 기하학적 특성을 이용한 문맥모델링을 적용하여 압축률을 더욱 낮추었다.

dinosaur의 경우 압축률이 저하되었는데, 향후 연구 과제로 우선 모델의 특성이 압축률과 관계가 있는지를 분석하고자 한다. 또한, 본 알고리즘을 사면체 메쉬 압축에 적용할 수 있도록 확장하고, 문맥 모델링에 대한 연구를 추가하여 압축률 감소를 위한 지속적인 연구를 할 예정이다.

참고 문헌

[1] C. Touma and C. Gotsman: Triangle Mesh Compression. Graphics Interface 98 Conference Proceedings. pp. 26-34, 1998.

[2] J. Rossignac: EdgeBreaker : Connectivity Compression for Triangle Meshes. IEEE Transactions on Visualization and Computer Graphics. Vol. 5(1). pp. 47-61, 1999.

[3] P. Alliez and M. Desbrun: Valence-Driven Connectivity Encoding of 3D Meshes. Computer Graphics Forum (Proc. Eurographics'01), Vol. 17(3). pp. 480-489, 2001.

[4] H. Lee and P. Alliez and M. Desbrun: Angle-Analyzer: A Triangle-Quad Mesh Codec. Computer Graphics Forum (Proc. Eurographics'02), Vol. 22(3), pp. 383-391, 2002.

[5] H. Hoppe, Progressive Meshes. Siggraph 96 Conference Proceedings. pp. 99-108, 1996.

[6] Pierre Alliez and Mathieu Desbrun: Progressive Encoding for Lossless Transmission of 3D Meshes. ACM Siggraph Conference Proceedings. pp. 198-205, 2001.

[7] P-M. Gandoin and O. Devillers: Progressive Lossless Compression of Arbitrary Simplicial Complexes. ACM Trans. on Graphics. Vol. 21(3). (2002) 372-379.

[8] M. Isenburg and S. Gumhold: Out-of-core compression for gigantic polygon meshes. ACM Transactions on Graphics. Vol. 22(3) (2003) 935-942.

[9] Sungyul Choe, Junho Kim, Haeyoung Lee,

Seungyong Lee, Hans-Peter Seidel: Mesh Compression with Random Accessibility. The 5th Korea-Israel Bi-National Conference on Geometric Modeling and Computer Graphics, October 2004.

[10] H. Lee and M. Desbrun and P. Schrder: Progressive Encoding of Complex Isosurfaces. ACM Transactions on Graphics (Proc. Siggraphics'03), Vol. 21(3). pp. 471-476, 2003.

[11] Andrei Khodakovsky and Peter Schrder and Wim Sweldens: Progressive Geometry Compression. ACM Siggraph Conference Proceedings. pp. 271-278, 2000.

[12] Martin Isenburg and Jack Snoeyink: Face Fixer: Compressing Polygon Meshes With Properties. ACM SIGGRAPH 2000 Conference Proceedings. (2000) 263-270.

[13] Andrei Khodakovsky and Pierre Alliez and Mathieu Desbrun and Peter Schrder: Near-Optimal Connectivity Encoding of 2-Manifold Polygon Meshes. Special Issue of Graphical Model. (2002).

[14] W. B. Pennebaker and J. L. Mitchell: JPEG: Still Image Data Compression Standard. Van Nostrand Reinhold. (1993) 135-252.

[15] G. Taubin: BLIC: Bi-Level Isosurface Compression. Proc. of IEEE Visualization. (2002) 451-458.

[16] P. Cignoni and C. Rocchini and R. Scopigno: Metro: Measuring Error on Simplified Surfaces. Computer Graphics Forum. Vol. 17(2) (1998) 167-174.

[17] F. Wheeler: Adaptive Arithmetic Coding Source Code. <http://www.cipr.rpi.edu/~wheeler/ac>. (1996).



이혜영
1986년 연세대학교 수학과 학사. 1992년 미국 San Jose State University 전산과 석사. 2003년 미국 University of Southern California 전산과 박사. 1992년 (주)비손텍 3D director. 1993년~1999년 한국통신 연구센터 전임연구원. 현재 홍익대학교 컴퓨터 공학과 조교수