

나선형 깊이 우선 탐색 방법을 사용한 3차원 데이터 압축 방법

이신준, 박우찬, 이현석, 양성봉
연세대학교 컴퓨터과학·산업시스템공학과

A 3D Data Compression Method Using Spiral Depth-First Search

Shin-Jun Lee, Woo-Chan Park, Hyun-Suk Lee, Sung-Bong Yang
Dept. of Computer Science & Industrial Systems Engineering,
Yonsei University

요 약

최근에 3차원 데이터는 그 크기가 상당히 커지고 있으며, 이에 따라 3D 그래픽 파이프라인에 데이터를 전송할 때 메모리 버스 대역폭의 병목 현상에 대한 문제가 크게 대두되고 있다. 이를 해결하기 위한 방법으로 모델링된 3차원 데이터의 위상 정보를 압축하여 보내는 방법이 연구되고 있다. 본 논문에서는 3차원 데이터의 위상 정보를 효율적으로 압축하는 방법과 자료구조를 제안하였다. 기존의 위상 압축 방법들의 선형적인 탐색 방법에 비해 보다 효율적인 나선형 깊이 우선 탐색 방법을 사용하였다. 제안하는 방법은 이미 탐색한 정점들의 재사용도를 높임으로써 압축 과정에서 생성되는 허프만 테이블의 크기를 줄였고 전체적으로 압축되는 데이터의 크기도 보다 줄였다. 실험한 결과, 비트수/정점이 평균 1.7 정도이었고, 비트수/폴리곤이 평균 1.3 정도로 이전 방법들보다 1.5배에서 2.5배정도 나은 결과를 보였다.

1. 서론

최근의 3D 컴퓨터 그래픽에서는 3D 데이터를 광범위하게 사용하고 있고, 다양한 기법들이 이 분야에 응용되고 있다. 그리고, 최근의 3D 데이터는 그 크기가 상당히 커지고 있으며, 그에 따라, 3D 데이터는 컴퓨터에서 저장하고 검색하는데 상당한 메모리의 용량을 필요로 한다. 3D 데이터가 상당히 커짐에 따라서, 현재의 3D 렌더링 하드웨어는 3D 그래픽 파이프라인(graphics pipeline)에서 시스템 메모리에 메모리 버스 대역폭의 병목 현상의 문제가 크게 대두되었다. 이를 해결하기 위한 한가지 방법으로 전처리(preprocessing) 단계에서 3D 지오메트리 정보를 압축하여 처리하는 방법이 제안되었다[1, 2, 3, 4, 5, 6, 8, 9]. 이 3D 그래픽 압축 방법은 전송할 3D 데이터를 줄이기 때문에 메모리 버스 대역폭의 병목 현상의 문제를 효과적으로 해결할 수 있다.

3D 데이터 압축 방법은 크게 두 가지로 정점 압축과 위상 압축으로 구성이 된다. 위상 압축은 3D 삼각

형 메쉬에서 삼각형간의 연결 정보를 압축하는 것으로, 정점 압축에 비해 최근에 연구가 활발히 진행되고 있다. 본 논문에서는 이 위상 압축 방법에 중점을 둔다. 제안하는 방법은 기존의 위상 압축 방법들의 선형적인 탐색 방법에 비해 보다 효율적인 나선형 깊이 우선 탐색을 방법으로 이미 탐색한 정점들의 재사용도를 높임으로써 압축 과정에서 생성되는 허프만 테이블의 크기를 줄였고 전체적으로 압축되는 데이터의 크기도 보다 줄였다. 실험 결과, bits/vertex의 평균값이 1.7정도이고 bits/triangle의 평균값이 1.3정도이며, 이는 현재까지 가장 나은 압축률을 보여주는 방식인 [3]에서 bits/vertex의 평균값이 3.8이고 bits/triangle이 2인 것과 비교하여 볼 때 매우 좋은 결과이다.

2장에서는 3D 데이터 위상 압축과 관련된 배경과 관련된 연구에 대해 살펴보고, 3장에서는 제안하는 위상 압축 방법에 대해서 설명을 하고 압축 과정에서 사용되는 탐색 방법을 소개한다. 4장에서는 실험 결과를 보여주고 5장에서는 결론을 언급한다.

2. 배경 및 관련 연구

3D 삼각형 메쉬(triangle mesh)는 3D 데이터 표현 방법 중 일반적으로 가장 많이 사용되는 형태로 메쉬를 이루는 폴리곤들을 삼각형으로 표현한다. 3D 삼각형 메쉬의 구성요소는 지오메트리 정보와 위상(topology) 정보로 크게 2가지로 나누어진다. 지오메트리 정보는 3D 삼각형 메쉬를 이루는 정점(vertex)들의 정보와 각 정점들의 3차원 (x, y, z)의 좌표이다. 위상 정보는 삼각형을 이루는 정점들간의 연결 정보로 1개의 삼각형은 3개의 정점으로 구성된다. 3D 데이터 압축(compression)은 3D 삼각형 메쉬를 입력으로 압축된 형태의 비트 스트림(bit stream)이 출력된다.

2.1 관련 연구

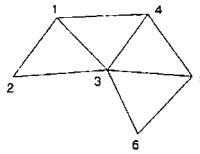
3D 데이터 압축 방법은 크게 두 가지로 정점 압축과 위상 압축으로 구성이 된다. 첫 번째, 정점 압축은 3D 데이터를 이루는 각 정점의 위치, 색상, 법선 등의 수치적인 정보들을 압축한다. 정점 압축의 일반적인 방법은 먼저 데이터 양자화(quantization)를 통해 일차 압축을 하고 그 다음에는 인접한 정점들간의 거리 델타(delta) 값을 구하고 그 델타 값들을 다시 허프만 인코딩(Huffman encoding) 함으로 최종적으로 압축이 이루어진다. 이 정점 압축은 Deering[1], Hoppe[2], Touma[6]에 의해 주로 연구되었고, 데이터 양자화와 엔트로피 인코딩을 통한 정점 좌표 압축에 중점을 두었다. 두 번째, 위상 압축은 3D 삼각형 메쉬에서 삼각형간의 연결 정보를 압축한다. 이는 삼각형을 이루는 정점들의 연결 정보를 압축하는 것으로 연결 정보는 특정한 순서를 갖는 정점들의 차수 리스트로 인코딩될 수 있다[6]. 이 위상 압축은 Deering[1]과 Taubin과 Rossignac[3], Floriani[5], Touma[6]에 의해 주로 연구되었고, 신장 트리 또는 정점의 차수 리스트로 표현되는 연결 정보를 압축하는데 중점을 두었다. 위상 압축 방법은 정점 압축에 비해 최근에 활발히 연구가 진행되고 있다.

이 위상 압축 방법에서 정점의 차수를 사용하여 삼각형 스트림을 정점의 차수 리스트로 재구성하는 방법 중에 한가지가 Park[9]에 의해 제안되었다. 이 방법에서는 현재의 정점 체인의 각 정점에서 이전 정점 체인과의 연결 선(edge)의 개수를 얻을 수 있다. 이 연결 선의 수를 그 정점의 차수(degree)로 정의하고 정점 체인과 함께 인코딩한다. 삼각형 스트림에서 임의의 정점에 대한 차수는 이전 정점 체인과 연결된

선의 개수로, 삼각형 스트림 내의 정점 체인 당 정점 차수는 평균 2정도의 값을 얻는다. 정점의 평균 차수의 빈도를 허프만 인코딩으로 엔트로피 인코딩을 압축할 수 있다.

2.2 위상 압축 방법의 과정

위상 압축하는 과정은 4단계로 첫 번째, 삼각형 메쉬의 위상 정보를 입력받아 삼각형 메쉬에서 탐색을 수행하고 삼각형 스트림을 형성한다. 두 번째, 탐색 수행 과정에서 정점을 방문하는 순서대로 각 정점의 차수를 구한다. 그리고 나서, 각 차수의 빈도를 계산하여 빈도수 테이블을 구한다. 세 번째, 정점 차수의 빈도수 테이블을 입력받아 허프만 테이블을 구한다. 그리고 나서, 각 정점의 차수에 따른 허프만 코드로 인코딩한다. 네 번째, 허프만 테이블을 참조하여 허프만 인코딩된 각 정점의 차수를 비트 단위로 파일로 저장하여, 최종 압축된 파일을 구한다.



(그림 1) 간단한 삼각형 메쉬

<표 1> 위상 정보의 예

triangle	vertex index
T1	1, 2, 3
T2	1, 3, 4
T3	3, 4, 5
T4	3, 5, 6

3. 제안하는 위상 압축 방법

제안하는 위상 압축 방법의 주요 개념은 삼각형 메쉬를 나선형 형태의 삼각형 스트림으로 만들어 압축을 하려는 것이다. 나선형 형태의 삼각형 스트림을 만드는 것은 삼각형 스트림 형성 시 정점을 방문하는 회수를 최소한으로 하고 방문한 정점의 재사용도를 높이고자 하는 것이다. 실험 결과, 삼각형 스트림 형성 시의 정점의 방문 회수는 대부분 한번이었다. 제안하는 위상 압축 방법은 삼각형 메쉬의 연결 정보를 정점의 차수를 사용하여 정점 차수의 리스트로 인코딩하는 방식이다. 이 인코딩된 정점 차수의 리스트는 허프만 인코딩되어 최종 압축된다.

3.1 용어 정의

본 논문에서 사용하는 용어를 정의한다. 이들 용어는 본 논문에서 재정의 및 새로 정의하여 사용한다.

(1) 정점 체인(vertex chain) - 삼각형 스트림을 형성하는 과정에서, 나선형 깊이 우선 탐색 후의 결과로 형성되는 자료구조이다. 삼각형 스트림은 일련의 인접한 삼각형들의 집합이라면, 정점 체인은 일련의 인접

한 정점들의 집합이다. 정점 체인을 이루는 정보는 정점 리스트와 차수 리스트이다.

(2) 정점의 차수 - 한 정점의 차수는 나선형 깊이 우선 탐색 시, 삼각형 스트립을 형성하는 과정에서, 그 정점을 추가함으로써 형성될 수 있는 삼각형의 개수이다.

(3) 현재 정점 체인 - 삼각형 매쉬를 나선형 깊이 우선 탐색 방법으로 탐색하는 과정에서 생기는 정점 체인으로 시작 정점에서 현재까지 추가된 정점까지의 정점들로 구성된 정점 리스트이다.

(4) 이전 정점 체인 - 삼각형 매쉬를 나선형 깊이 우선 탐색 방법으로 탐색하는 과정에서 생기는 정점 체인으로 시작 정점에서 베이스 정점까지 추가된 정점들로 구성된 정점 리스트이다.

(5) 베이스 정점(base vertex) - 삼각형 매쉬를 나선형 깊이 우선 탐색 방법으로 탐색하는 과정에서 이전 정점 체인에 추가된 최근의 정점이다.

3.2 위상 압축 방법 개요

제안하는 위상 압축 방법은 매쉬 연결 정보를 일련의 순서로 나열되는 정점 차수의 리스트로 인코딩한다. 압축된 정보는 압축의 역과정 즉, 디코딩 과정을 거치면 원래의 매쉬 연결 정보로 구성된다. 제안하는 위상 압축 방법은 정점 체인을 1개만을 사용하여 임의의 삼각형의 정점으로부터 나선형 깊이 우선 탐색 방법을 통해 확장 탐색한다. 탐색 결과로는 삼각형 매쉬를 입력으로하여 정점 리스트와 차수 리스트가 나온다. 차수 리스트는 허프만 인코딩되어, 최종 위상 압축이 이루어진다.

3.3 삼각형 스트립 인코딩 방법

본 논문에서 제안하는 위상 압축 알고리즘은 나선형 깊이 우선 탐색 방법을 통해 삼각형 스트립을 형성한 후 이루어진다. 수행 방법은 다음과 같다.

- 1) 시작(또는 재시작) 삼각형 선택 및 탐색 초기화 단계: 시작(또는 재시작) 삼각형을 무작위로 선택한다. 시작 삼각형으로 새로운 삼각형 스트립을 구성한다. 시작 삼각형을 이루는 3개의 정점들에서 시작 정점을 무작위로 선택하고, 나머지 두 정점을 차례로 방문하는 정점 리스트를 만든다. 시작 정점을 베이스 정점으로 지정하고, 마지막 정점을 현재 체인 포인터가 가리키게 한다.
- 2) 스트립에 추가되지 않은 인접한 삼각형 검색 단계: 베이스 정점(베이스 정점 포인터가 가리키는 정점)과

현재 체인 포인터의 정점을 포함하는 삼각형을 검색한다. 검색은 스트립에 추가되지 않은 사각형 중에서 한다. 만약, 검색된 삼각형이 없으면 6)으로 간다.

3) 검색된 삼각형의 정점 검사 단계: 검색된 삼각형을 이루는 3개의 정점에서 베이스 정점과 현재 체인 포인터의 정점을 뺀 나머지 정점이 정점 리스트 순서에서 베이스 정점의 다음 정점과 일치하는지 검사한다. 일치하지 않으면 5)로 간다.

4) 정점 차수 증가 및 베이스 정점 포인터 이동 단계: 현재 체인 포인터가 가리키는 정점의 차수를 1 증가시키고, 정점 리스트 순서에서 베이스 정점의 다음 정점을 베이스 정점으로 지정하여 베이스 정점 포인터가 가리키게 한다. 2)로 간다.

5) 현재 체인 포인터 이동 및 삼각형을 스트립에 추가하는 단계: 검색된 삼각형을 이루는 3개의 정점에서 베이스 정점과 현재 체인 포인터의 정점을 뺀 나머지 정점을 정점 리스트에 추가시킨다. 현재 체인 포인터를 정점 리스트에 방금 추가된 정점을 가리키게 한다. 그 정점의 차수를 1로 설정하여, 차수 리스트에 추가한다. 2)로 간다.

6) 베이스 정점에 인접한 삼각형 검색 단계: 베이스 정점과 정점 리스트 순서에서 베이스 정점의 다음 정점을 포함하는 삼각형을 검색한다. 검색은 스트립에 추가되지 않은 삼각형 중에서 한다. 만약, 검색된 삼각형이 없으면 8)로 간다.

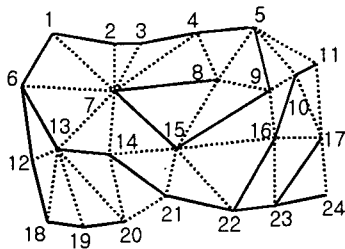
7) 검색된 삼각형을 삼각형 스트립에 추가하는 단계 : 베이스 정점을 정점 리스트에 또 한번 추가시키고, 베이스 정점의 차수를 0으로 하여 차수 리스트에 추가시킨다. 그리고 나서, 검색된 삼각형을 이루는 3개의 정점에서 베이스 정점과 베이스 정점의 다음 정점을 뺀 나머지 정점을 정점 리스트에 추가시킨다. 현재 체인 포인터를 방금 추가된 정점을 가리키게 한다. 그 정점의 차수를 1로 설정하여, 차수 리스트에 추가한다. 베이스 정점을 베이스 정점의 다음 정점으로 지정하여 베이스 정점 포인터가 가리키게 한다. 2)로 간다.

8) 베이스 정점 이동 단계: 베이스 정점 포인터를 베이스 정점의 다음 정점을 가리키게 한다. 정점 리스트 순서에서 베이스 정점의 다음 정점이 있으면 6)으로 가고, 그렇지 않으면 9)로 간다.

9) 남아 있는 삼각형 검색 단계: 아직 삼각형 스트립에 추가되지 않은 삼각형이 있으면 1)로 가고 그렇지 않으면, 끝낸다.

3.4 위상 압축 방법의 과정

위상 압축 방법의 과정에서 먼저 이루어지는 나선형 깊이 우선 탐색은 (그림 2)와 같다. (그림 2)에서 선택된 시작 삼각형이 (8, 7, 15)라고 할 때, 정점 8에서부터 시작되는 나선형 깊이 우선 탐색 결과를 보여 준다.



(그림 2) 입력 메쉬[9]에 대한 나선형 깊이 우선 탐색의 결과

탐색 결과로 나오는 정점 리스트와 차수 리스트는 다음과 같다.

정점 리스트 = <8, 7, 15, 9, 5, 4, 3, 2, 1, 6, 13, 14, 21, 22, 16, 10, 11, 6, 12, 18, 19, 20, 22, 23, 17, 23, 24>

차수 리스트 = <0, 0, 1, 1, 1, 2, 1, 1, 1, 1, 1, 2, 1, 1, 2, 2, 1, 0, 1, 1, 1, 3, 0, 1, 3, 0, 1>

4. 실험 결과

<표 2> 압축 결과

	Model				
	Bones	Bunny	Cow	Crater	Mech part
vertices	2154	35947	2904	100000	1572
polygons	4204	69451	5804	199114	3152
topology	201808	548688	278608	9557488	151312
bits/vertex	1.78869	1.70236	1.80316	1.74879	1.74216
bits/triangle	1.34705	1.10253	1.27688	1.40556	1.37405

본 논문에서 사용한 실험 데이터는 Microsoft사의 Hoppe의 웹사이트에서 공개된 데이터를 사용하였다 [7]. <표 1>은 각 데이터를 압축한 결과이다. 압축률은 일반적으로 평균 bits/vertex와 평균 bits/triangle로 평가한다. 본 논문에서 실험 후, 압축 결과는 bits/vertex가 평균 1.7정도의 결과를 보였고, bits/triangle이 평균 1.3정도의 결과를 보여 주었다. 이 결과는 Taubin과 Rossignac의 알고리즘[3]이 bits/vertex의 평균값이 3.8인 것과 bits/triangle이 2인 것에 비해 매우 좋은 결과를 보여준다.

5. 결론

본 논문에서는 3D 데이터 압축과 관련하여 새로운 위상 압축 방법을 제안하였다. 실험 결과, 제안된 위상 압축 방법의 결과는 bits/vertex의 경우 평균 1.7정도의 결과를 보였고, bits/triangle은 평균 1.3정도의 결과로 이전 위상 압축 방법들에 비해 더 나은 결과를 보였다. 이는 나선형 깊이 우선 탐색 방법으로 3D 데이터를 탐색하여, 삼각형 스트립을 형성하며 정점의 방문 회수를 줄이고, 정점의 재사용도를 높이므로 전체적인 삼각형 스트립을 형성하는 정점 체인의 자료 구조의 크기를 많이 줄이므로 다른 압축 방법에 비해 상대적으로 나은 결과를 보여주었다.

[참고문헌]

[1] M. Deering, "Geometry compression," In SIGGRAPH '95 Proc., pp. 13-20, 1995.
 [2] H. Hoppe, "Progressive meshes," In SIGGRAPH '96 Proc., pp. 99-108, Aug. 1996.
 [3] G. Taubin and J. Rossignac, "Geometric compression through topological surgery," Technical report, Research Report RC-20340, IBM Research Division, 1996.
 [4] M. Chow, "Optimized geometry compression for realtime rendering," In IEEE Visualization '97, pp. 347-354, Oct. 1997.
 [5] P. M. Leila De Floriani and E. Puppo, "Compressing tins," In ACM-GIS '98, pp. 145-130, Nov. 1998.
 [6] C. Touma and C. Gotsman, "Triangle mesh compression," In Proc. of Graphics Interface '98, pp. 26-34, 1998.
 [7] <http://www.research.microsoft.com/~hoppe>
 [8] S. Kim, Y. Kim, M. Cho, and H. Cho, "A geometric compression algorithm for masive terrain data using delaunay triangulation," In WSCG '99, Feb. 1999.
 [9] D. Park, Y. Kim, and H. Cho, "Triangle mesh compression for fast rendering," Information Visualization '99, pp. 280-285, July 1999.
 [10] G. Turan, "On the succinct representation of graphs," Direct Applied Mathematics, pp. 289-294, 1984.