

클러스터링기법을 이용한 3차원 모델의 법선 벡터 압축
Clustering based Normal Vector Compression of 3D Model

조영송¹⁾, 김덕수²⁾

Youngsong Cho and Deok-Soo Kim

Abstract

As the transmission of 3D shape models through Internet becomes more important, the compression issue of shape models gets more critical. The issues for normal vectors have not yet been explored as much as it deserves, even though the size of the data for normal vectors can be significantly larger than its counterparts of topology and geometry. Presented in this paper is an approach to compress the normal vectors of a shape model represented in a mesh using the concept of clustering. It turns out that the proposed approach has a significant compression ratio without a serious sacrifice of the visual quality of the model.

1. 서론

일상생활에서 인터넷이 차지하는 비중이 커짐에 따라, 데이터의 빠른 전송이 사업을 성공시키는 중요한 요인으로 자리잡게 되었다. 그러므로 데이터를 압축하는 기술은 네트워크를 통해 데이터를 좀더 빠르게 전송하기 위한 필수적인 기술들의 하나가 되었다. 형상 모델 압축에 대한 연구는 1995년 Deering의 연구[3] 이후, 형상 모델의 위상정보와 기하정보에 대한 압축 연구는 광범위하게 진행되어 왔다. 그에 비해 형상 모델을 좀더 현실적으로 가시화 하는데 필요한 법선 벡터 정보가 위상정보와 기하정보만큼이나 많은 데이터를 필요로 하지만, 활발한 연구가 진행되지 않고 있다.

일반적으로 법선 벡터는 IEEE 표준에 따라 32 bit를 사용하는 float으로 표현된 3개의 실수 값으로 구성된다. Float으로 표현된 두 법선 벡터가 구별할 수 있는 최소 각도가 2^{-46} radian이라는 것과 이렇게 자세한 정보가 일반적인 그래픽 모델을 가시화 하는데 사용되기에는 너무 상세한 정보라는 사실은 이미 잘 알려져 있다[3]. 그러므로 법선 벡터의 압축은 인터넷이나 그래픽 파이프라인을 통한 형상 모델의 교환에 있어 매우 중요한 이슈이다.

본 논문에서 제시하는 알고리즘은 잘 알려진 클러스터링 기술의 장점을 이용한다. 법선 벡터들은 자동으로 미리 정의된 다수의 클러스터로 그룹화 되고, 각 클러스터들은 클러스터에 속한 법선 벡터들을 대표하는 대표 법선 벡터를 가진다. 각 클러스터에 속한 법선 벡터들은 대표 법선 벡터로 표현된다.

제안된 알고리즘은 모델을 시각적인 측면에서 원 모델과 큰 차이를 주지 않으면서, 원 모델의 파일 크기를 약 8 %의 크기로 압축한다.

1) 한양대학교 산업공학과, yscho@cadcam.hanyang.ac.kr

2) 한양대학교 산업공학과, dskim@hanyang.ac.kr

2. 관련 연구

Deering의 형상 모델 압축에 대한 새로운 연구 이후, 형상 모델의 압축에 대한 연구 분야로서 위상정보 압축[2][3][5][7][9][10][12], 기하정보 압축[3][8][10][12], 칼라 압축[11], 법선 벡터 압축[3][11] 등의 연구들이 진행되어 왔다.

법선 벡터 압축에 대한 연구 중 Deering은 법선 벡터를 인덱스로 표현하는 방법을 제시하였다. 법선 벡터가 놓일 단위구 위에서 법선 벡터를 인덱스로 변환하기 위하여, 단위구를 48개의 동일한 크기를 가지는 영역으로 나누고, 각 영역 안에서 법선 벡터는 극좌표로 표현된 두 개의 각도를 두 개의 정수로 변환되어 표현되었다[3].

VRML compressed binary format 을 제안한 Taubin은 법선 벡터를 분할 기법을 이용하여 표현하였다. Base octahedron을 구성하는 하나의 octant가 분할 수준에 따라 재귀적으로 분할된다. 분할된 결과로 나타난 삼각형들을 순서에 따라 라벨을 부여하고, 해당 삼각형에 속하는 법선 벡터를 이 라벨로 표현하였다[11].

3. VRML 모델에서 법선 벡터의 표현

형상 모델을 표현하는 다수의 표현법들 중에서 메쉬 특히 삼각형 메쉬가 본 논문에서 사용하는 주된 표현법이다. 그리고 메쉬 모델은 orientable manifold임을 가정한다.

VRML 97은 삼각형 메쉬를 표현하기 위하여 IndexedFaceSet이라는 노드를 제공하고, 이 노드는 법선 벡터를 표현하기 위하여 normalPerVertex(NPV)와 normalIndex(NI)라는 두 개의 필드를 가지고 있다[1]. normalPerVertex필드가 참(TRUE)일 경우, 법선 벡터는 모델의 꼭지점에 부여되고, 거짓일 경우 모델의 면에 부여된다. 그리고 모델을 정의하는데 normalIndex필드가 존재할 경우, 법선 벡터의 값은 한번 정의되고, 인덱스로서 법선 벡터를 표현한다. 그렇지 않을 경우 법선 벡터 값을 직접적으로 사용한다. VRML 97의 법선 벡터를 부여하는 방법은 그림 1에서와 같이 표현될 수 있다.

법선 벡터를 표현하는 4 가지 방법들 중에서, 본 논문에서는 법선 벡터가 모델의 꼭지점에 부여되고, 인덱스에 의해 표현되는 Case 4에 대한 압축 알고리즘을 제안한다. Case 4는 위의 4 가지 표현법들 중 가장 일반적인 표현법이다.

4. 법선 벡터의 클러스터링

Deering과 Taubin은 법선 벡터가 가리키는 방향을 동일한 크기를 갖는 다수의 영역으로 나누었다. 만약 법선 벡터들의 방향이 공간상에서의 분포가 비균일하다면, 일부 영역에 속한 법선 벡터의 수는 다른 영역에 비해 크게 작을 수 있다. 그러므로 Deering과 Taubin의 방법은 각 영역을 고정된 크기의 코드로 표현하므로 법선 벡터의 분포가 편중되어 있다면 bit 효율성이 크게 떨어질 수 있다.

이러한 요소를 고려하면서, 본 논문에서는 법선 벡터를 압축하기 위하여 클러스터링 기법이 사용되었다. 여러 가지 클러스터링 기법들 중 본 논문에서는 K-means 알고리즘을 사용하였다.

K-means 알고리즘은 반복 클러스터링 알고리즘이다. 클러스터의 수와 각 클러스터의 평균이 주어져 있다면, 각 데이터를 유사성 척도를 기준으로 하여 가장 가까운 클러스

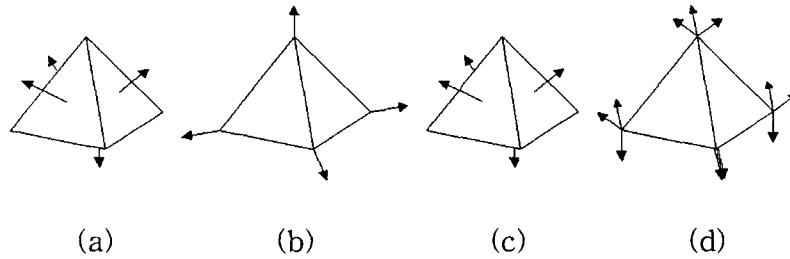


그림 1. VRML 모델의 법선 벡터 표현법 (a) Case 1 : NPV (FALSE) NI(non-existing) (b) Case 2 : NPV(TRUE) NI(non-existing) (c) Case 3 : NPV(FALSE) NI(existing) (d) Case 4 : NPV(TRUE) NI (existing)

터에 할당하고 새로운 데이터를 포함하여 클러스터의 평균을 갱신시킨다. 그리고 각 클러스터의 평균이 더 이상 변경되지 않을 때까지 반복적으로 이 과정을 수행한다[6]. 법선 벡터를 클러스터링하기 위한 유사성 척도는 법선 벡터간의 각도이며, 각 클러스터의 대표 법선 벡터는 클러스터에 속한 법선 벡터들의 평균 벡터로 정의하였다.

5. 압축 알고리즘

5.1 법선 벡터의 표현

그림 2는 3 개의 면과 6 개의 법선 벡터를 가진 모델을 보여준다. 클러스터링의 결과로 3 개의 클러스터 A(n_1, n_4), B(n_2), C(n_3, n_5, n_6)가 생겨났다고 가정한다. 각 클러스터에 속한 법선 벡터들은 각 클러스터의 대표 법선 벡터로 표현된다. 면 f_1 는 면을 구성하는 3개의 꼭지점에서 n_1, n_2, n_3 이 부여되어 있고, 이는 Normal Index 배열의 처음 3 개의 인덱스 1, 2, 3으로 표현된다. 이 인덱스들은 f_1 의 각 꼭지점에 부여된 법선 벡터가 Cluster Pointer배열의 첫 번째, 두 번째, 세 번째 법선 벡터임을 가리킨다. 그리고 Cluster Pointer배열은 다시 각각의 법선 벡터들이 속한 클러스터를 가리키고, 이후 이들 법선 벡터들이 각각의 대표 법선 벡터로 표현될 것이다.

Cluster Pointer배열이 구성 요소들의 오름차순으로 정렬하고, Normal Index 배열의 인덱스들을 적절히 변경한다면, 모델은 정렬 전과 동일하게 표현된다. Cluster Pointer 배열은 순서대로 나타난 동일한 클러스터 값의 수를 의미하는 Cluster Size배열로 변환된

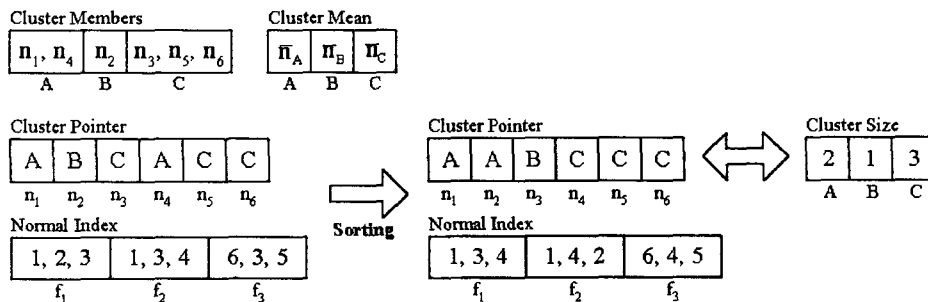


그림 2. 법선 벡터의 표현

다. 위의 예에서 Cluster Size배열은 3 개의 구성요소로 이루어지고, 각각 2, 1, 3의 값을 갖는다. 이는 클러스터 A에 속한 법선 벡터가 두 번 연속으로 나타나고, 다음에 클러스터 B에 속한 법선 벡터가 한 번 나타나고, 다음으로 클러스터 C에 속한 법선 벡터가 3 번 연속하여 나타나는 것을 의미한다. 정렬후의 Cluster Pointer배열은 Cluster Size 배열로 완벽하게 변환되며, 그 역변환 역시 완벽하게 이루어진다.

5.2 법선 벡터 인덱스의 표현

본 논문에서는 법선 벡터인덱스를 absolute index와 relative index로 나누어 표현한다. Normal Index배열에서 임의의 인덱스가 나타났다면, 그 인덱스가 정의된 최대 위상 거리, r , 내에서 두 번째로 나타날 경우, 이 인덱스는 먼저 나타난 인덱스와의 위상 거리를 값으로 갖는 relative index로 표현한다. 반면에 두 번째로 나타난 인덱스가 r 을 넘는 위상 거리를 갖거나, 새로운 인덱스는 인덱스 값을 갖는 absolute index로 표현한다. 그림 3(a)는 그림 2에서 정렬후의 Normal Index배열이다. 이때 $r=4$ 라 하면, f_2 의 첫 번째 인덱스 1은 f_1 에서 이미 나타났으므로, 위상 거리 3을 값으로 갖는 relative index로 표현된다. 이는 그림 3(b)에서 음영처리가 된 relative index로 표현된다.

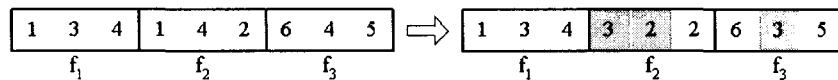


그림 3. absolute index와 relative index의 예 ($r = 4$)

absolute index와 relative index를 효과적으로 사용하기 위해서, 전체 인덱스에서 relative index가 차지하는 비중과 relative index의 bit크기를 고려해야 한다. 그림 4(a)는 $r=\infty$ 일 경우, 전체 인덱스에서 absolute index와 relative index가 각각 차지하는 비중을 보여준다. 본 논문에서 사용한 모델들은 약 80%가 relative index임을 알 수 있다. 그림 4(b)는 전체 relative index 중에서 위상 거리 1, 2, 3, 4, 5 이상을 갖는 relative index들을 구별하여 보여준다. 1, 2, 3, 4의 위상거리를 갖는 relative index가 약 80%이며 5이상의 위상 거리를 갖는 relative index가 약 20%임을 보여준다. 이 실험의 결과로 relative index의 bit크기를 2 bit로 결정할 경우, 반복하여 나타나는 인덱스의 약 80%의 relative index들이 표현될 수 있음을 알 수 있다.

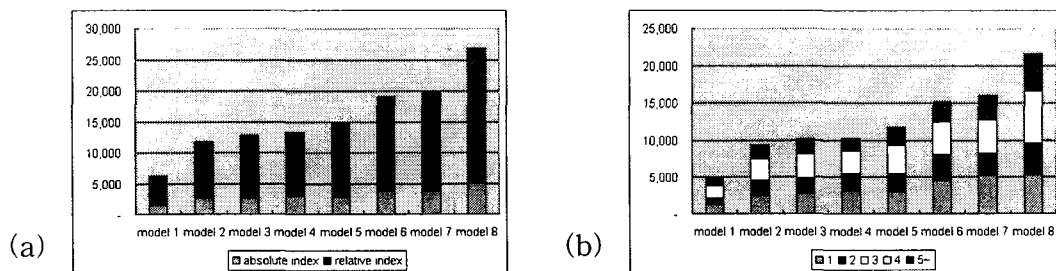


그림 4. relative index의 분포 (a) absolute index와 relative index의 비교($r=\infty$)
(b) 위상거리 1, 2, 3, 4, 5이상을 갖는 relative index들

6. 실험 결과

실험에서 사용된 모델들은 상용 CAD 시스템인 ProEngineer에서 만들어진 모델이다. 표 1은 제안된 알고리즘과 다른 방법으로 표현된 파일의 크기를 비교하여 보여준다. 첫 번째 열 A은 법선 벡터 정보만을 포함하고 있는 VRML ASCII파일의 크기이고, 두 번째 열 B는 WinZip으로 A를 압축한 파일의 크기이다. 그리고 세 번째 열은 법선 벡터 정보를 저장할 때 필요한 최소 메모리의 이론적인 값으로 계산한 값이다. 네 번째 열은 본 논문에서 제안한 알고리즘에 의해 압축된 법선 벡터의 파일 크기이다. 제안된 알고리즘은 A, B, C를 각각 약 8%, 약 30%, 약 14%로 압축함을 보여준다.

표 1. 파일 크기의 비교

	VRML File (A)	Zip of A (B)	MRM (C) (calculated)	Proposed Algorithm (D)	D/A	D/B	D/C
Model 1	68,392	18,730	41,892	5,934	8.68%	31.68%	14.16%
Model 2	132,491	36,888	77,952	10,706	8.08%	29.02%	13.73%
Model 3	139,411	38,637	82,524	11,450	8.21%	29.63%	13.87%
Model 4	150,678	40,990	89,736	11,781	7.82%	28.74%	13.13%
Model 5	158,900	43,664	92,796	13,001	8.18%	29.78%	14.01%
Model 6	205,110	54,436	122,172	16,218	7.91%	29.79%	13.27%
Model 7	210,446	55,578	124,500	17,190	8.17%	30.93%	13.81%
Model 8	291,132	77,532	168,348	24,789	8.51%	31.97%	14.72%

그림 5는 원 모델과 제안된 알고리즘에 의해 압축된 모델사이의 시각적인 차이를 보여준다. 두 모델의 기하정보 및 위상정보는 동일하다. 그림 5(a)는 2,861개의 법선 벡터로 표현된, 158,900byte의 파일크기를 가지는 model 5이다. 그림 5(b)는 model 5를 제안한 알고리즘으로 압축한 것으로, 64개의 대표 법선 벡터로 표현하였고, 13,001 byte의 파일 크기를 가진다. 그림 5에서와 같이 시각적으로 큰 차이가 없음을 알 수 있다.

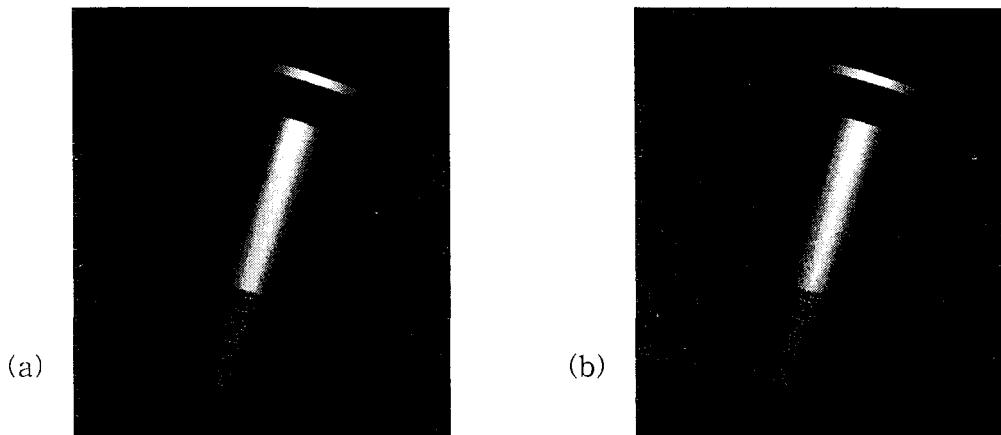


그림 5. 시각적 차이 비교 (a) 압축 전 model 5 (b) 압축 후 model 5

7. 결론

법선 벡터는 형상 모델을 좀더 현실적으로 표현하는데 필요하다. 법선 벡터 정보는 기하정보 및 위상정보 데이터에 비해 결코 작지 않은 양의 데이터를 가지게 된다. 이것

은 형상 모델을 네트워크를 통해 전송할 시에 상당한 장애요인이 될 수 있다. 제안된 법선 벡터 압축 기법은 클러스터링 기법을 이용하여 시각적으로 큰 차이 없이 원 모델의 법선 벡터 정보를 약 8 %정도로 압축하였다. 특히, 법선 벡터를 가리키는 인덱스를 absolute index와 relative index로 나누어 표현한다.

이후의 연구 과제로서, K-means 알고리즘보다 좀더 나은 클러스터링 기법을 연구하는 것이 중요한 연구 과제가 될 수 있을 것으로 사료된다. 그리고 각 클러스터의 대표 법선 벡터들이 좀더 압축될 수 있을 것으로 예상된다.

참고문헌

- [1] Carey, R., Bell, G, and Marrin, C.; ISO/IEC DIS 1477-1: 1997 Virtual Reality Modeling Language(VRML97), The VRML Consortium Incorporated, 1997.
- [2] Chow, M., M.; Optimized Geometry Compression for Real-Time Rendering, In Proceedings of IEEE Visualization '97, pp. 347-354, 1997.
- [3] Deering, M.; Geometry Compression, In Proc. SIGGRAPH' 95, pp. 13-20, 1995.
- [4] Foley, J.D., van Dam, A., Feiner, S. K. and Hughes, J. F.; Computer Graphics : Principles and practice (2nd ed.), Addison Wesley, 1987.
- [5] Gumhold, S., Strasser, W.; Real-Time Compression of Triangle Mesh Connectivity, In Proceedings of ACM SIGGRAPH '98, pp. 133-140, 1998.
- [6] Jain, A.K., and Dubes, R.C.; Algorithms for Clustering Data, Prentice-Hall, 1988.
- [7] Kim, Y.-S., Park, D.-G., Jung, H.-Y., and Cho, H.-G.; An Improved TIN Compression Using Delaunay Triangulation, In Proceedings of Pacific Graphics '99, pp. 118-125, 1999.
- [8] Lee, E.-S., and Ko, H.-S.; Vertex Data Compression For Triangle Meshes, Eurographics 2000, Vol. 19, No. 3, pp. 1-10, 2000.
- [9] Rossignac, J.; Edgebreaker: Connectivity Compression for triangle meshes, IEEE Transactions on Visualization and Computer Graphics, 5(1): 47-61, 1999.
- [10] Taubin, G., Rossignac, J.; Geometric Compression Through Topological Surgery, ACM Transactions on Graphics, 17(2): 84-115, 1998.
- [11] Taubin, G., Horn, W.P., Lazarus, F., and Rossignac, J.; Geometric Coding and VRML, Proceedings of the IEEE, 86(6): 1228 -1243, 1998.
- [12] Touma, C., and Gotsman, C.; Triangle Mesh Compression, In Proceedings of Graphics Interface '98, pp. 26-34, 1998.