

3차원 메시모델 부호화기술 동향

Overview of 3D Mesh Model Compression Techniques

차세대 전파방송기술 특집

장은영 (E.Y. Chang)	3DTV시스템연구팀 연구원
허남호 (N.H. Hur)	3DTV시스템연구팀 팀장
김진웅 (J.W. Kim)	전파방송연구단 책임연구원
이수인 (S.I. Lee)	방송시스템연구그룹 그룹장

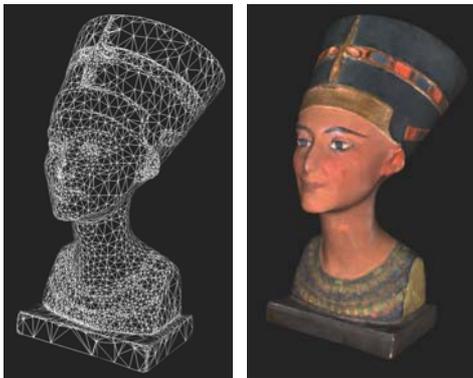
목 차

-
- I. 서론
 - II. 연구 개발 및 표준화 동향
 - III. 연구 활동 현황
 - IV. 결론

3차원 메시 모델은 컴퓨터 그래픽스, 애니메이션, 3D 모델링, CAD, 3D 게임 등의 다양한 응용 분야에서 광범위하게 사용되고 있다. 그러나, 현실감을 줄 수 있는 고해상도의 3차원 모델은 그 정보량의 방대함으로 인해 처리, 저장, 전송 등에 있어 여러 가지 어려움이 따른다. 따라서, 이를 효율적으로 압축하여 저장 및 전송하기 위한 3차원 메시 모델 부호화 기술이 연구되어 왔다. 본 고에서는 3차원 메시 부호화 기술에 대한 연구 개발 동향과 이에 대한 표준화 동향을 소개한다. 또한, 3차원 메시 부호화 기술과 관련하여 ETRI에서 연구중인 정점 및 면 순서 정보 부호화 기술과 텍스처 좌표 부호화 기술의 내용과 이에 대한 연구 활동 현황을 요약하고, 주요 이슈 논의 및 향후 전망에 대해 간략히 살펴보고자 한다.

I. 서론

물체의 3차원 형체 및 구조를 나타내는 3차원 메시 모델은 (그림 1)과 같이 각 정점의 3차원 위치를 나타내는 기하 정보, 정점들 간의 기하학적 연결 구조를 나타내는 연결성 정보, 그리고, 색상, 법선 및 텍스처 좌표와 같은 속성 정보로 이루어진다.



(그림 1) 3차원 메시 모델

이러한 3차원 메시 모델은 컴퓨터 그래픽스, 애니메이션, 3D 모델링, CAD, 3D 게임 등의 다양한 멀티미디어 응용 분야에서 광범위하게 사용되고 있다. 특히, 컴퓨터 그래픽스 응용 분야에서는 물체를 현실감 있게 표현하기 위해 고해상도를 가진 복잡한 3차원 메시 모델을 필요로 한다. 이러한 필요성을 만족하기 위해 고해상도를 가진 복잡한 모델이 획득되거나 생성된다. 그러나, 이러한 상세한 모델을 전송하기 위해서는 넓은 대역폭이 요구되고, 이를 저장 및 처리하기 위해서는 대용량의 저장 및 메모리 공간이 요구되는 등 많은 어려움이 있다. 따라서, 저장 공간의 저장 공간과 빠른 전송을 요구하는 응용분야는 이러한 문제점을 해결하기 위해 3차원 메시 모델을 효율적으로 압축하기 위한 3차원 메시 부호화 기술의 연구 개발을 유도하였고, 다수의 3차원 메시 부호화 기술이 고안되었다.

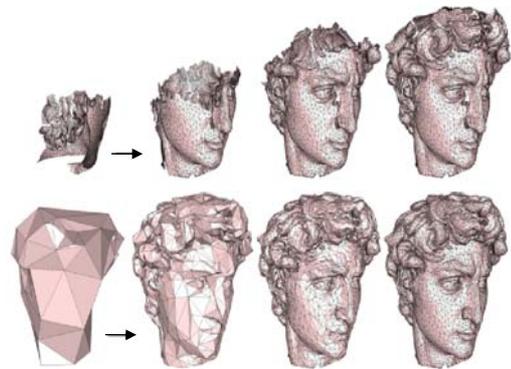
본 고에서는 개요에 이어 II장에서 3차원 메시 부호화 기술 연구 개발 동향과 이에 대한 표준화 동향을 소개하고, III장에서는 ETRI에서 수행중인 3차원

메시 부호화 기술 연구의 현황을 소개한다. 또한, IV장에서는 3차원 메시 부호화 기술과 관련된 주요 연구 이슈 논의와 함께 향후 전망을 소개한다.

II. 연구 개발 및 표준화 동향

1. 연구 개발 동향

3차원 메시 모델은 모델을 구성하는 기본 구성 단위가 삼각형인 경우 삼각형 메시, 다각형인 경우 다각형 메시로 나뉜다. 본 고에서는 일반적으로 사용되고 주로 연구되고 있는 삼각형 메시 부호화에 대한 연구 개발 동향을 다룬다. 3차원 메시 모델의 부호화 기술에 대한 접근 방법은 single-rate mesh coding과 progressive mesh coding의 두 가지 방법으로 분류할 수 있으며[1],[2], 두 방법에 대한 차이는 (그림 2)를 통해 알 수 있다.



(그림 2) Single-Rate Mesh Coding(위)과 Progressive Mesh Coding(아래) 방법의 일례

● 용어해설 ●

Single-Rate Mesh Coding: 원본 모델을 주어진 해상도 그대로 압축하여 점진적인 전송을 지원하지 않는 부호화 방법을 뜻한다.

Progressive Mesh Coding: 원본 모델을 최저 해상도의 LOD와 다수의 정제(refinements) 데이터로 변환 및 압축하여 점진적 전송을 지원하는 부호화 방법을 뜻한다.

우선 single-rate mesh coding 기술에 대해 살펴 보도록 한다. 앞서 기술한 바와 같이 3차원 메시 모델은 기하 정보, 연결성 정보, 속성 정보로 구성되어 있다. 대부분의 개발 방법은 이 세 종류의 데이터 중 어느 하나의 데이터만을 효율적으로 부호화하는 방법에 대해 다루었다. 특히, 다수의 기술이 연결성 정보의 효율적 부호화에 초점을 맞추어 진행되어 왔다. 더욱이, 속성 정보의 부호화에 대해서는 연구 개발이 거의 진행되고 있지 않아 대부분이 기하 정보와 동일한 방법으로 부호화하고 있는 실정이다. 따라서, 본 고에서는 연결성 정보와 기하 정보의 부호화에 대한 동향을 다룰 것이다.

주어진 3차원 정점들로부터 알고리즘적으로 이 연결성 정보를 계산해 낼 수 있으므로, 정점들 간의 기하학적 연결 구조를 나타내는 연결성 정보를 전송할 필요가 없다는 주장이 있다[3]. 그러나, 주어진 3차원 정점들로부터 연결성 정보를 복원하는 방식은 계산 복잡도가 높고 정확하게 복원하기도 어려울 뿐만 아니라 연결성 정보를 압축했을 때 하나의 삼각형 당 한 비트 이하로 압축할 수 있으므로, 이 정보를 효율적으로 부호화하여 전송하는 방식에 대한 다수의 연구가 진행되고 있다.

연결성 정보를 부호화하기 위한 다수의 기술 중 가장 잘 알려진 기술인 edgebreaker 방법[4]은 Depth-first TST 순서에 기반하여 삼각형을 탐색하고, CLERS 스트링을 생성한다. CLERS는 새로운 삼각형이 이전까지 복원된 삼각형 구조에 어떠한 형태로 접합되는지를 나타내는 5개의 라벨(C, L, E, R, S)로, 부호화 시 하나의 삼각형마다 하나의 라벨이 주어진다. 따라서, 복호화 시에는 이전까지 복원된 삼각형 구조에 CLERS 라벨에 따라 새로운 삼각형을 접합함으로써 3차원 메시 모델을 복원해낼 수 있게 된다. Edgebreaker 방법은 연결성 정보 압축의 worst-case bound인 정점 당 평균 4비트의 비트율로 주어진 삼각형 메시지를 부호화한다. 따라서, edgebreaker 방법을 개선하기 위한 다수의 시도가 이루어졌으며, 일례로 CLERS 스트링으로부터 워드를 구성하고 이를 Huffman 부호화기를 이용하여 압

축하는 방법은 삼각형 당 1비트까지 부호화 가능하였다. 이들 중 최고의 성능을 보인 Delphi 방법[5]은 edgebreaker 압축 방법에 의해 생성된 CLERS 스트링에 대해 예측 부호화를 적용함으로써 성능 향상을 꾀하였다. 실험 결과에 따르면 모델에 따라 다르긴 하지만 Delphi 방법의 예측이 최고 97%까지 적중하는 경우가 있었으며, 이 때는 삼각형 당 0.19비트의 부호화율을 달성하였다. 그러나, 예측 오류가 40%를 넘으면 예측 부호화를 통한 이득이 없다고 알려져 있다.

Edgebreaker 방법의 변형 방법인 cut-border machine 방법[6]은 edgebreaker 방법과 많이 유사하지만 S-삼각형의 오프셋을 부호화한다는 점에서 차이점을 갖는다. 삼각형 당 1.7~2.5비트의 부호화 성능을 가져 edgebreaker 방법에 비해 다소 비효율적이지만, single pass coding이 가능하다는 장점을 가져 대용량의 데이터를 부호화하는 데 많이 사용된다.

삼각형 메시의 연결성 정보를 valence code로 변환하여 부호화를 수행하는 방법은 최상의 부호화 성능을 나타낸다. Valence code 기반의 방식인 deterministic conquest 방법[7]은 Euler 이론에서 정점의 valence가 평균 6이라는 점을 이용하여 부호화 효율 향상을 도모한다. 즉, 대부분의 모델의 valence 분포가 6에 집중되어 있으므로 엔트로피 부호화를 이용할 경우에 발생하는 적은 확률적 편차의 이점을 활용한 것이다. Deterministic conquest 방법은 edgebreaker와 동일한 순서로 삼각형을 탐색하지만 CLERS 스트링을 부호화하는 대신 각 정점의 valence와 dummy, merge, split 코드를 부호화한다. 실험 결과에 따르면 규칙적 메시(regular mesh)인 경우에는 정점 당 0.2비트 이하로, 일반적인 메시인 경우에는 정점 당 2~3.5비트 정도로 압축 가능하다. Valence code 기반의 부호화 방식이 최상의 성능을 달성하므로, 다각형 부호화 역시 상대성을 활용한 valence code 기반의 부호화 방식이 연구 개발되었다.

대표적 3차원 메시 모델 부호화 방법으로 상기에

기술된 방식 외에 MPEG 표준 기술인 topological surgery 방법[8]-[10]이 있다. Topological surgery 방법은 TST와 VST로부터 연결성 정보를 복원할 수 있다는 점, 이진 TST는 각 삼각형에 대해 왼쪽 또는 오른쪽 자식 노드가 존재하는지를 알려주는 2비트를 이용해 부호화할 수 있다는 점, VST는 현재의 노드가 잎 노드인지의 여부를 알려주는 1비트를 이용하여 부호화할 수 있다는 점을 착안하여 고안되었다. 따라서, topological surgery 방법은 우선 3차원 메시 모델로부터 메시를 절단하는 경로를 정점간의 연결 구조로 나타내는 정점 그래프(VG)와 삼각형 스트립으로 이루어진 이진 트리 구조의 삼각형 스페닝 그래프(TT)를 생성한다. VG의 경우는 last, (forward_run), (loop_index), run_length, leaf, (loop)의 필드로, TT의 경우 run_length, leaf, (marching pattern)의 필드로 표현된다. 이렇게 생성된 스트립은 추후 엔트로피 부호화를 거치게 된다. IBM에 의해 구현된 topological surgery 방법은 시각적 왜곡 없이 VRML 파일을 50:1로 압축할 수 있다.

이제, 기하 정보의 부호화 방법에 대해 살펴보면 정점의 3차원 위치를 나타내는 기하 정보는 VRML을 포함한 대부분의 표현법에서 부동 소수점으로 표현되며, 압축 효율을 높이기 위해 일반적으로 손실 부호화된다. 또한, 대부분의 방법에서 원본 모델 상에서의 순서와 관계 없이 연결성 정보 부호화 당시 탐색된 순서로 정점에 대한 기하 정보 부호화를 수행한다. 모든 부호화 방법이 그러하듯이 기하 정보의 부호화 역시 양자화, 예측 부호화, 엔트로피 부호화의 단계를 거친다.

기하 정보의 양자화는 부동 소수점으로 표현된 기하 정보가 제한된 범위의 정확도를 갖도록 정점의 기하 좌표 값의 끝을 자른다. 즉, x, y, z의 각 축에 정렬된 bounding box를 만들고 박스의 각 축을 크기로 동일하게 분할한 후 박스 내부의 각 셀 내부에 들어오는 정점을 셀의 중심으로 정규화하는 과정을 수행한다. 따라서, 양자화 에러는 셀의 대각선의 반절 크기만큼으로 제한된다. 통상 B를 12로 설정하

면 대부분의 응용 분야 및 모델에서 만족할 정도의 충실도를 제공할 수 있다. 초창기의 양자화 방법은 이와 같이 Cartesian 공간에 대해 각각의 좌표를 독립적으로 균등 양자화하는 방법을 많이 사용하였다. 이후의 방법은 벡터 양자화 및 주파수 영역에서의 양자화를 적용하였다. 특히, 인간의 시각 체계는 기하 왜곡에 더 민감하다는 점을 이용하여 시각 왜곡을 최소화하고자 “k-anchor Invertible Laplacian Transformation”을 수행한 후 생성된 변형된 좌표 공간에서 양자화를 수행하는 방법도 제안되었다[11].

대부분의 부호화 방식이 그러하듯 정점의 기하 정보 역시 예측 방법을 사용하여 부호화되므로 residual만을 전송한다. 송신단과 수신단에서 동일한 예측기를 사용하기 때문에 residual을 통해 전송된 기하 정보 값을 정확하게 복원할 수 있다. 전체 모델 크기에 비해 인접한 정점 간의 거리 차가 작기 때문에 새로운 정점을 예측할 때 이전에 전송된 이웃 정점들을 이용하게 된다. 하나의 이웃 정점을 이용하여 예측하는 방법(단순 델타 부호화)과 VST 순서 상에서 4개의 이웃 정점 값의 선형 조합을 이용한 예측 방법도 제안되었다. Single-rate mesh coding에서 가장 많이 사용되는 예측기는 평행 사변형 예측기[6]로 MPEG의 3DMC 기술에도 사용되었다. 이 방법은 이웃 3개의 정점을 이용하여 평행 사변형을 이루고자 할 때 새로운 정점의 위치를 이웃 3개의 정점이 차지하고 있지 않는 위치로 예측하는 것으로, 인접 삼각형이 이루는 각을 예측함으로써 성능 향상이 가능하다.

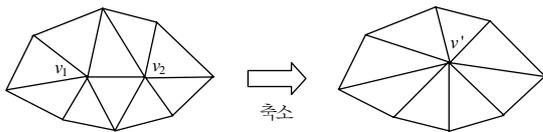
한 가지 짚고 넘어가야 할 점은 앞서 기술한 바와 같이 정점의 기하 정보가 VST 순서로 부호화되므로 이전에 전송된 이웃 정점들이 기하 정보를 효율적으로 예측하는 데에는 적합하지 않을 수 있다. 이 점에 착안하여 prediction tree[12] 방법이 제안되었고, 이 방법은 기존의 대다수 방법들처럼 VST 순서를 따른 예측 부호화를 수행하는 대신 기하 정보 자체만으로 탐색 순서를 결정한다. 실험 결과에 따르면 최대 50%로 비트율이 감소하였다.

다음으로, progressive mesh coding 방법은(그

림 2)에서 보는 바와 같이 refinement의 개념을 사용하는 방법으로, single-rate mesh coding과 다르게 주어진 원본 모델을 단순한 base mesh와 연속적인 refinement를 위한 데이터로 변환하여 전송한다. 수신단에서는 base에 연속적인 refinement를 적용함으로써 원본 모델을 복원하게 된다. 이 방법은 고해상도의 모델이 불필요하거나 즉각적인 피드백이 요구되는 경우에 사용할 목적으로 고안되었으며, 연속적인 LOD를 제공하는 장점을 갖지만 rate-distortion tradeoff의 최적화를 만족함에 어려움이 있다. 일반적으로 base mesh는 single-rate mesh coding 방법에 의해 압축 전송되고, refinement에 필요한 데이터가 progressive mesh coding 방법에 의해 압축 전송된다. 대부분의 점진적 부호화 방식은 (1) mesh decimation operator의 선택, (2) decimation될 요소를 결정하기 위한 여러 측정법, (3) decimation 과정을 역으로 수행하기 위해 필요한 정보의 효율적 부호화 등의 세 가지 요소에 따라 부호화 효율이 달라지고 각 방법의 독창성이 달라지게 된다.

가장 일반적으로 사용되는 방법은 Hoppe에 의해 제안된 progressive mesh 방법[13]이다. Progressive mesh 방법은 주어진 원본 모델에 (그림 3)의 에지 축소를 반복적으로 적용함으로써 base mesh를 만들고, 매 에지 축소 때마다 에지 축소 과정을 역으로 수행(정점 분할)하기 위해 필요한 최소한의 정보를 저장하는 방식으로 수행된다.

즉, progressive mesh 방법의 mesh decimation operator는 에지 축소이며, 에지 축소 과정을 역으로 수행하기 위해 저장하는 정보는 정점 분할되는 각 정점의 정확한 위치와 이 정점을 공유하는 에지의 식별자이다. 이 방법은 어느 시점에서든 전송 및 복호화를 중단할 수 있는 등의 장점을 갖지만, 부



(그림 3) 에지 축소의 일례

호화 효율이 낮다는 단점이 있다. 부호화 효율을 향상시키고자 많은 시도들이 이루어졌으며 정점 당 7.2비트까지로 부호화 효율을 낮추었다.

2. 표준화 동향

3차원 메시에 관련된 표준화 단체는 MPEG 외에도 X3D, COLLADA, Khronos, U3D 등이 있다.

X3D는 실시간 3차원 컴퓨터 그래픽스에 대한 ISO 표준으로 그 주요 특징은 VRML의 확장(Humanoid Animation, Nurbs, GeoVRML 등), XML Syntax 및 VRML97과 같은 open inventor를 활용한 장면 부호화, 강화된 API 등이 있다.

COLLADA는 인터랙티브 3D 응용 분야에 사용될 파일 포맷을 제정하기 위한 표준이다. 상기 포맷은 PlayStation 3와 PlayStation Portable 개발의 공식 포맷으로 이용하고자 Sony에 의해 최초 개발되었지만, Khronos 참여자의 노력으로 COLLADA로 계속해서 발전되게 되었다. 현재는 수십 개의 상업 게임 스튜디오와 게임 엔진 업체에서 표준으로 채택하였고, COLLADA Ver. 1.4 표준이 2006년 1월에 발표되었다. COLLADA는 파일 포맷 제정에 그 목적이 있는 점 때문에 Maya, 3D Studio Max, Softimage XSI 등을 개발하는 저작 툴 개발 업체가 표준화 활동을 활발하게 하고 있다.

Khronos는 2000년에 설립된 산업체 컨소시엄으로 다양한 플랫폼과 장치에서 미디어를 저장할 수 있고 재생할 수 있는 API를 제정하는 데에 그 목적이 있으며, 잘 알려져 있는 OpenGL, OpenGL ES, OpenML 등의 표준을 제정하고 있다.

상기에서 살펴본 X3D, COLLADA, Khronos 등의 표준화 단체는 3차원 (그래픽스) 미디어 데이터의 부호화보다는 그 표현법(파일 포맷) 자체 및 API 등의 표준 제정에 초점을 맞추고 있다. 그러나, 앞으로 살펴볼 U3D와 MPEG은 3차원 (그래픽스) 미디어에 대한 표현법 외에도 미디어에 대한 부호화까지도 다루고 있으며 특히 부호화에 대한 표준 제정에 초점을 맞추고 있다.

U3D는 웹 상에서의 3D drawing(CAD)의 공유와 통상의 오피스 응용 분야에서 사용될 3D 그래픽스 포맷으로 Intel과 3DIF에 의해 개발되었다. 빠른 인터넷 다운로드와 스크린 상에서의 빠른 렌더링 등을 위해 U3D 파일 크기를 줄이고자 원본 CAD에 관련된 공학적 데이터의 대부분을 삭제하였다. U3D는 편집을 위한 새로운 3D 파일 타입을 제공하고자 함이 아니라, CAD 프로그램이 export 될 수 있는 단순한 포맷을 제공하는 데 그 목적이 있다. 또한, U3D 표준의 3차원 메시 부호화 기술은 progressive mesh coding 방법으로 앞서 기술한 progressive mesh 방법과 근본적 방법은 동일하며 CLOD를 제공하는 장점이 있으나 부호화 효율이 높지 않다는 단점도 있다. 2004년 말 adobe는 acrobat/PDF 제품 군의 버전 7에서 U3D를 지원하기로 하여 현재 사용중에 있다. 이는 3차원 메시 부호화 기술이 상용 제품에 처음으로 사용된 경우로 그 의미가 높다고 할 수 있다.

MPEG은 3차원 컴퓨터 그래픽스 데이터에 대한 표현법 및 부호화를 제정하고자 1995년 여름 도쿄 미팅에서 SNHC 그룹을 생성하였고, 현재는 3DGC 그룹으로 개명하여 활동하고 있다. 3차원 메시 모델 부호화와 관련하여서는 앞서 설명한 3DMC를 MPEG 표준으로 2003년에 제정하였다[14]. 3DMC 기술은 시각적 왜곡 없이 VRML 파일을 2~4% 정도로 압축할 수 있는 장점 외에도, incremental rendering, error resilience, progressive transmission, non-manifold 지원 등의 장점을 갖는다. 참고로, 3DMC 기술은 국내의 한국정보통신기술 협회 산하의 모바일 3D 표준화 포럼에서 그 활용을 계획하고 있다. 또한, 2005년 4월 부산 미팅에서 3DMC 기술을 보완 및 확장할 수 있는 유용한 기술들이 제안되어 3DMC 확장에 대한 표준화가 진행중에 있으며 현재는 TuC 단계에 있다. 현재 3차원 메시 부호화 기술의 상업 및 산업화가 활발치 않아 MPEG은 기술에 대한 표준 제정 외에도 산업화 방안에 대해서도 고려하고 있다.

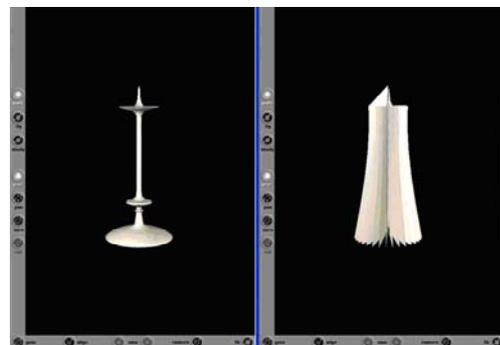
Ⅲ. 연구 활동 현황

3차원 메시 모델 부호화 기술과 관련하여 ETRI에서 개발중인 기술은 효율적인 애니메이션 지원을 위한 정점 및 면 순서 정보 압축 기술과 효율적인 텍스처 매핑을 위한 텍스처 좌표 압축 기술이 있다.

1. 효율적인 애니메이션 지원을 위한 정점 및 면 순서 정보 압축

MPEG-4 표준 기술인 3DMC를 포함하여 여타의 많은 3차원 메시 모델 부호화 기술은 원본 모델 상에서의 정점 및 면의 정보를 부호화 효율을 높이기 위해 순서를 변경하여 압축을 수행한다. 3DMC를 포함한 다수의 3차원 메시 부호화 방법이 연결정보를 VST와 TST 등의 형태로 변환하여 부호화를 수행하고 VST의 순서에 따라 기하 정보의 부호화를 수행함으로써 면과 정점의 순서에 변화가 발생하는 것을 그 예로 들 수 있다.

이렇게 정점과 면의 순서가 변경되어 복원된 모델이 정적인 형태로만 응용된다면 즉, 애니메이션, 편집, 갱신 등을 허용치 않는다면 심각한 문제는 발생치 않을 수 있다. 그러나, 복원된 모델을 애니메이션, 온라인 객체 갱신 및 편집 등의 응용 분야에 이용한다면 심각한 문제가 발생할 수 있으며, 그 예를 (그림 4)에 나타내었다. 이러한 문제점을 해결하고자 원본 모델 상에서의 정점 및 면의 순서 정보를 추



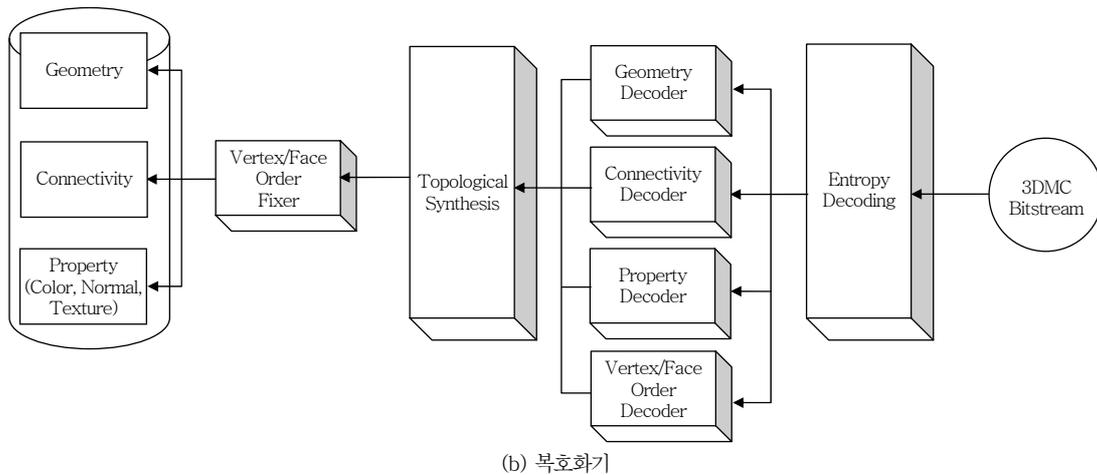
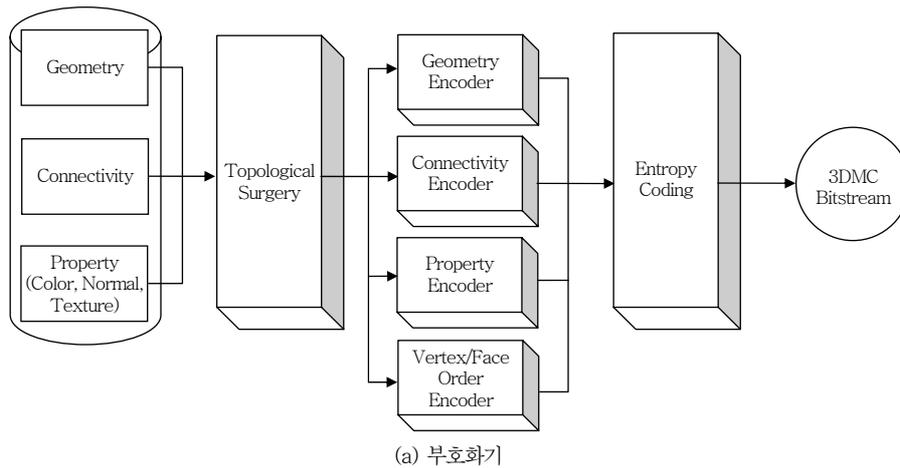
(a) 원본 애니메이션 모델 (b) 복원된 애니메이션 모델

(그림 4) 정점 및 면 순서 변화로 인한 문제점의 일례

가적으로 전송하는 방법과 이에 대한 부호화 방법을 개발하였다[15]. (그림 5)는 MPEG 3DMC를 기반으로 하여 정점 및 면의 순서 정보를 추가적으로 압축 전송하는 모듈을 포함한 확장된 3차원 메시 모델 부복호화기에 대한 블록도를 나타낸다.

정점 및 면 순서 변경의 문제점을 해결하려면, 원본상에서의 정점 및 면 순서 정보를 무손실 부호화하여 전송해야만 하고 이는 비트스트림 사이즈의 증가를 야기한다. 따라서, 이에 대한 효율적 부호화 방법이 필요하게 된다. 개발한 정점 및 면 순서 정보에 대한 부호화 방법은 (1) symbol의 전체 개수가 정해져 있다, (2) 각 symbol은 오직 한 번만 발생함의 두 가지 사실을 고려하여 적응적 확률 모델을 사용하여

압축률 향상을 도모한다. 즉, 이전에 부호화된 symbol을 고려하여 식별 가능한 단위로 symbol에 대한 코드워드를 한 비트씩 계속해서 줄여나감으로써, 약 8~13% 정도의 부호화 비트율 감소를 가능케 하였다. 또한, connected component 개념을 도입하여 기존의 IndexedFaceSet 기반의 정점 및 면 순서 정보를 connected component 기반의 정점 및 면 순서 정보로 변환하고 이를 앞서 설명한 방식으로 부호화함으로써 개발 기술의 성능을 더욱 향상시켰다. 실험 결과에 따르면 모델을 구성하는 정점 또는 면의 개수와 connected component의 개수에 따라 부호화 성능이 달라지긴 하지만 앞서 설명한 적응적 확률 모델기반 부호화 방식에 비해 최고 40%까지

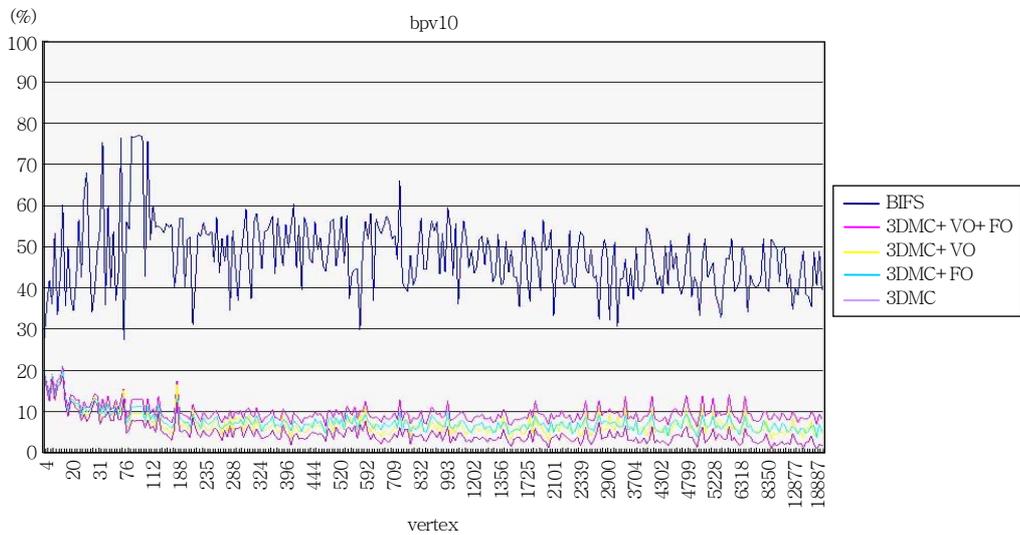


(그림 5) 정점 및 면 순서 정보 부복호화기를 포함한 3차원 메시 모델

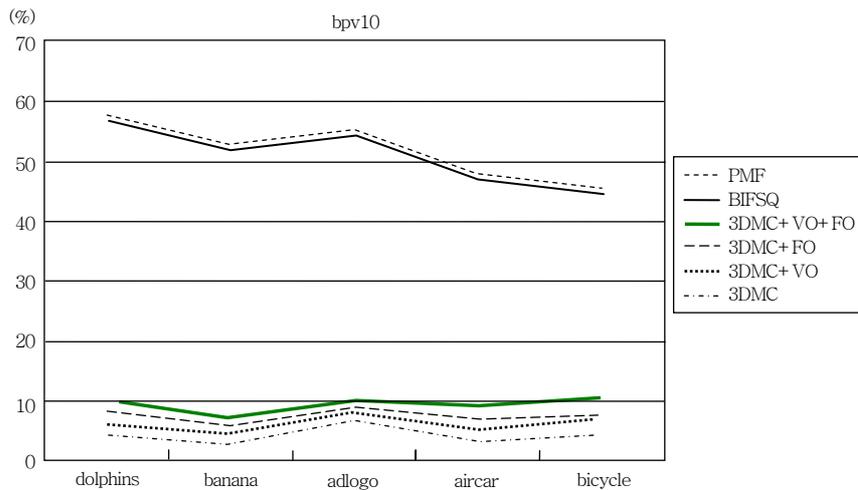
부호화 비트율을 감소시켰다.

여러 가지 방법을 도입하여 정점 및 면 순서 정보의 압축을 시도하였음에도 불구하고 여전히 그 정보의 전송으로 인해 증가하는 비트스트림 크기가 너무 크다는 지적이 있어 왔다. 이에 대한 확인을 위해 여러 모델에 대해 모델을 MPEG 3DMC로 압축한 부호화 비트스트림에 정점 및 면 순서 정보 부호화 비트스트림을 추가하여 이 둘 사이의 비트스트림 크기

를 구하였다. 실험 결과에 따르면 약 3%의 비트율 추가만으로 구현 가능하며, 이는 정점 및 면 순서 정보를 보존할 수 있는 ZIP 부호화 방식에 비해서도 매우 효율적이라는 것을 확인할 수 있었다[15]. 또한, MPEG에서 사용되는 실험 데이터에 대해 BIFSQ와 PMF 방법과도 부호화 효율을 비교하였는데[16], 통상적으로 사용되는 정점 당 10비트(bpv=10)로 성능 비교를 실시하였을 때의 실험 결과를 (그림 6)



(a) 개발기술과 BIFSQ와의 성능 비교



(b) 개발기술 BIFSQ 및 PMF와의 성능 비교

(그림 6) 부호화율에 따른 개발기술의 성능 평가 결과

에 나타내었다. 상기의 두 가지 실험 결과를 통해 개발 기술의 유용성을 입증할 수 있었다.

2. 효율적인 텍스처 매핑을 위한 텍스처 좌표 압축

속성 정보의 부호화에 대해서는 연구 개발이 거의 진행되고 있지 않으며, 대부분이 기하 정보와 동일한 방법으로 부호화하고 있는 실정이라는 것을 앞서 설명하였다. MPEG 3DMC 역시 속성 정보를 부호화함에 있어 기하 정보의 부호화에 사용되는 방법을 동일하게 적용하고 있다. 그러나, 이러한 접근법은 각 속성 정보의 특성을 고려하지 않아 속성 정보에 대한 효율적 부호화를 불가능하게 만든다. 특히, 텍스처 좌표의 경우에는 부정확한 텍스처 매핑 등의 결과를 초래한다. 이러한 문제점을 해결하고자 적응적 양자화 스텝 크기 개념을 제안하였다[17]. 텍스처 좌표는 부동 소수점 값으로 표현된 어느 특정한 값이 의미가 있는 것이 아니고 텍스처 영상의 픽셀 위치를 나타내는 정보이므로, 이러한 의미에 대한 고려 없이 균등 양자화하는 방법은 왜곡된 텍스처 매핑 동작을 초래한다. 따라서, 부동 소수점 값에서 변화가 발생하더라도 참조 픽셀 위치를 부호화한 후 정확하게 복원할 수만 있다면 복원된 렌더링 영상에서 왜곡이 발생하지 않는다. 하지만, 텍스처 영상의 크기를 텍스처 좌표 값을 부호화하기 이전에 반드시 알 수 있는 것은 아니므로, 텍스처 영상의 크기를 알 수 없을 때에는 주어진 텍스처 좌표 값으로부터 추산된 일정한 간격으로 부호화를 수행하여야 한다.

적응적 양자화 스텝 크기를 적용한 텍스처 좌표 부호화 방법은 3DMC의 성능을 개선시켰다. 우선 참조 픽셀 위치의 정확한 복원 여부를 기준으로 무손실 여부를 판단해 볼 때, 3DMC는 모든 실험 모델에 대해 무손실로 부호화할 수 없었던 반면, 개발 기술은 비트율을 10%에서 최대 40%까지 절감하면서 모든 실험 모델에 대해 무손실로 부호화할 수 있었다. (그림 7)에 3DMC와 개발 기술을 통해 복원된 모델에 대한 렌더링 결과를 나타내었다. 참고로, 개



(그림 7) 3DMC(좌)와 개발 기술(우)에 의한 복원 모델

발 기술을 통한 부호화스트림은 3DMC 비트율 대비 10%를 절감하면서 무손실로 부호화하였다.

상기의 두 가지 개발 기술은 현재의 MPEG 기술의 문제점을 보완하거나 확장할 수 있는 유용한 기술임을 확인할 수 있었다. 이 기술은 2005년 부산에서 개최된 MPEG 표준화 회의에서 표준안으로 제안하였고, 그 타당성 및 유용성을 검증 받아 MPEG 표준으로 채택되었다. 현재는 TuC 단계에 있으며 2007년에 표준 완료될 예정이다.

IV. 결론

본 고에서는 3차원 메시 부호화 기술의 연구 개발 동향 및 표준화 동향과 ETRI의 관련 연구 활동 현황에 대해 살펴보았다. 3차원 메시 부호화 기술은 국내외적으로 활발히 연구 및 개발되고 있고 표준화 작업이 활발하게 진행중에 있으나, 산업에서는 광범위하게 사용되고 있지는 않은 실정이다. 그러나, 3차원 게임, 3차원 애니메이션 영화, 3차원 TV 산업 등이 활성화되면서 점차적으로 주목을 받고 있다. 3차원 메시 모델 부호화 기술은 이러한 응용에서 사용되는 멀티미디어 콘텐츠의 효율적 저장 및 전송을 위해 반드시 필요한 기술임은 자명하다고 판단된다. 따라서, 산업에서 요구하는 필수 기술 개발과 이에 대한 적극적 표준화 활동이 기대된다.

약어 정리

3DGC	3D Graphics Compression
3DIF	3D Industry Forum

3DMC	3D Mesh Coding
API	Application Programming Interface
BIFS	Binary Format for Scenes
BIFSQ	BIFS Quantization
CAD	Computer Aided Design
CLOD	Continuous Level of Detail
COLLADA	COLLABorative Design Activity
LOD	Level of Detail
MPEG	Moving Picture Experts Group
OpenGL ES	OpenGL for Embedded Systems
OpenGL	Open Graphics Library
OpenML	Open Media Library
SNHC	Synthetic-Natural Hybrid Coding
TST	Triangle Spanning Tree
TT	Triangle Tree
TuC	Technologies under Consideration
U3D	Universal 3D
VG	Vertex Graph
VRML	Virtual Reality Modeling Language
VST	Vertex Spanning Tree

참 고 문 헌

- [1] J. Rossignac, *Handbook of Discrete and Computational Geometry(Second Edition)*, CRC Press, 2004.
- [2] P. Alliez and C. Gotsman, "Recent Advances in Compression of 3D Meshes," *In Proc. of the Symp. on Multiresolution in Geometric Modeling*, 2003.
- [3] N. Amenta, M. Bern, and M. Kamvysselis, "A New Voronoi-based Surface Reconstruction," *In Proc. of SIGGRAPH*, 1998, pp.415-421.
- [4] J. Rossignac, "Edgebreaker: Connectivity Compression for Triangle Meshes," *IEEE Transactions on Visualization and Computer Graphics*, 1999.
- [5] V. Coors and J. Rossignac, "Guess Connectivity: Delphi Encoding in Edgebreaker," *GVU Tech. Report*, 2002.
- [6] S. Gumhold and W. Strasser, "Real Time Compression of Triangle Mesh," *In Proc. of SIGGRAPH 98*, 1998.
- [7] C. Touma and C. Gotsman, "Triangle Mesh Compression," *In Proc. of Graphics Interface 98*, 1998.
- [8] Fernando Pereira and Touradj Ebrahimi, *The MPEG-4 Book*, Prentice Hall, 2002.
- [9] Aaron E. Walsh and Mikael Bourges-Sevenier, *MPEG-4 Jump-Start*, Prentice Hall, 2002.
- [10] G. Taubin and J. Rossignac, "Geometric Compression through Topological Surgery," *ACM Transactions on Graphics*, 1998.
- [11] O. Sorkine, D. Cohen-Or, and S. Toldeo, "High-Pass Quantization for Mesh Encoding," *In Proc. of Eurographics Symp. on Geometry Proc.*, 2003.
- [12] B. Kronrod and C. Gotsman, "Optimized Compression of Triangle Mesh Geometry Using Prediction Trees," *In Proc. of 1st Int'l Symp. on 3D Data Proc., Visualization and Transmission*, 2002.
- [13] H. Hoppe, "Progressive Meshes," *In Proc. of ACM SIGGRAPH 96*, 1996.
- [14] "Information Technology - Coding of Audio-Visual Objects - Part2: Visual," ISO/IEC JTC1/SC29/WG11 N5546, Mar. 2003.
- [15] Eun-Young Chang, Daiyong Kim, Byeongwook Min, Sunyoung Lee, Namho Hur, Soo In Lee, and Euee S. Jang, "Vertex and Face Permutation Order Compression for Efficient Animation Support," *In Proc. of Electronic Imaging 2006*, 2006.
- [16] Eun-Young Chang, Daiyong Kim, Sunyoung Lee, Namho Hur, Jinwoong Kim, Soo In Lee, and Euee S. Jang, "Status Report of 3DMC Ref. SW," ISO/IEC JTC1/SC29/WG11 M12858, Bangkok, Thailand, Jan. 2006.
- [17] Sunyoung Lee, Byeongwook Min, Daiyong Kim, Eun-Young Chang, Namho Hur, Soo In Lee, and Euee S. Jang, "An Adaptive Quantization Scheme for Efficient Texture Coordinate Compression in MPEG 3DMC," *Lecture Notes in Computer Science(LNCS)*, Vol.3767, 2005, pp.73-83.