

# Polygonal Mesh로 표현된 3차원 모델의 에러복원 연구

김대용, 류대하, 박성원, 김미자, 장의선

한양대학교 정보통신대학원

## Data Recovery of 3D Polygonal Mesh Model

Dai-yong Kim, Dae-ha Ryu, Sung-won Park, Mi-ja Kim, Euee S. Jang

Graduate School of Information and Communications, Hanyang University

boamania@dmlab.hanyang.ac.kr, moshymoro@dmlab.hanyang.ac.kr, kiwise@dmlab.hanyang.ac.kr,

mimibaby@dmlab.hanyang.ac.kr, esjang@hanyang.ac.kr

### Abstract

3차원 Polygonal Mesh는 그래픽스, 애니메이션, 게임에서 3차원 객체에 대한 표현에 사용되고, 이러한 3차원 모델에 대한 IndexedFaceSet 노드에 3차원 정점정보와 연결정보를 압축하는데 MPEG-4 3DMC를 사용한다. 이러한 연결정보는 다각형의 Mesh 형태로 3차원 모델을 구성하는 정보를 갖는데, 이는 Topological Surgery 라고 하는 방법을 통해서 2차원의 스트립 단위의 데이터로 분해된다. 이러한 3D 데이터는 방송환경과 같은 재전송이 불가능한 네트워크 환경에서 유무선 네트워크 상에서 채널 문제로 인해서 데이터의 손실이 있게 되면, 복호화된 데이터는 데이터의 손상이 발생하게 된다. 이러한 현상은 3D 모델의 좌표의 연결정보에 손상을 주게 되고, 여기서 복호화된 데이터는 스트립 단위로 손상이 발생하게 된다. 이러한 현상은 3차원 모델의 좌표의 연결정보에 손상을 주게 된다. 본 논문은 이러한 3차원 정보의 손상을 효과적으로 복원하기 위한 연구에 관한 것이며, Mesh의 면을 이루는 각 꼭지점의 좌표들의 연결 정보가 손실되지 않는 스트립에서는 약간의 차이는 있을 수 있으나, 완벽한 복원을 하였고, 두 개 이상의 스트립이 붙어서 손상된 경우나, 좌표의 연결 정보가 없는 경우에는 조건에 따라 현저히 좋은 결과를 얻을 수 있었다.

**Keywords** : MPEG(Moving Picture Experts Group), MPEG-4 3DMC(MPEG-4 3D Mesh Coding), VRML(Virtual-Reality Modeling Language)

### I. 서 론

3차원 Polygonal Mesh는 그래픽스, 애니메이션, 게임에서 3차원 객체에 대한 표현에 사용되고, 이러한 3차원 모델에 대한 IndexedFaceSet 노드에 3차원 정점정보와 연결정보를 압축하는데 MPEG-4 3DMC가 사용하게 되는데, 3DMC는 TS(Topological Surgery) 라고 하는 것을 이용하여, triangle tree 와 vertex graph 와 같은 정보로 나뉘져 인코딩 된다. 이러한 데이터는 3차원 데이터를 사과를 깎아 내듯, 스트립 단위로 인코딩 되고, 디코딩 된다. 이러한 3차원 데이터는 유무선 네트워크 상에서 데이터 전송시에 채널 문제로 인해서 데이터의 손실이 있게 되면, 복호화된 데이터는 TS 에 의해서 스트립 단위로 데이터의 손상이 발생하게 된다. 이렇게 손상된 데이터는 겉모양이 깨져있는 3차원 모델이 되게 된다. 하지만 TS 에 의해서 IndexedFaceSet노드에 있는 point 정보는 triangle tree 와 vertex graph 때문에 날라가지 않고, 단지 이들 point 들 사이의 연결 정보인 coordIndex 정보만 손실 되게 된다.

본 논문은 triangle tree 와 vertex graph 와 같이 원래 있던 정보만 이용, 부가적인 추가 정보를 사용하지 않고, 효

과적으로 빠진 부분을 메꾸는데 중점을 두었다. 그리고 좀 더 효과적인 복원 방법을 위해서 추가적인 정보를 이용하는 방법에 대해서 연구하였다.

Mesh 의 면을 이루는 각 꼭지점의 좌표들의 연결 정보가 손실되지 않는 스트립에서는 약간의 차이는 있을 수 있으나, 완벽한 복원을 하였고, 두 개 이상의 스트립이 붙어서 손상된 경우나, 좌표의 연결 정보가 없는 경우에는 조건에 따라 현저히 좋은 결과를 얻을 수 있었다. 그리고 3차원 모델의 크기가 커지면, point 들 사이의 거리도 가깝고, point 들 수가 많아서 좀 더 효과적인 결과를 얻을 수 있었다.

### II. 배경지식

#### A. Topological Surgery

3차원 Mesh는 정점(vertex)과 모서리(edge) 그리고 삼각형(triangles)으로 구성된다. 모서리(edge)는 두 개의 정점(vertex)으로 구성되고, 삼각형은 세 개의 정점으로 구성된다. 3차원 Mesh 부호화에서 3차원 Mesh는 triangular mesh라고 불리우고, 이것은 단지 삼각형(triangles)로 구성된다. 다른 다각형 Mesh들은 다각형을 삼각형으로 재구성

한 후에 MPEG-4 3DMC 톨로 압축될 수 있다. 복호화될 때 삼각형들은 다각형으로 저장된다. 연결정보들은 무손실 전송을 필요로 하고, 때문에 손실 압축은 압축후에 다른 topological model 에 포함된다. 3DMC에서 연결정보는 주어진 모델에 대한 저장을 위해 무손실로 인코딩 된다.

VRML에서 IndexedFaceSet은 다각형에 대한 정점들의 집합으로 3차원의 형태의 연결정보로 묘사된다. 예를 들어, 다각형에 첫 번째, 세 번째, 열 번째 그리고 37번째 정점을 순서대로 표현한다면 "1, 3, 11, 37, -1"로 표현된다. 마지막의 -1은 정점 37번째가 정점 1과 연결된다는 것을 의미한다. 이러한 방법은 쉬운 방법이다, 그러나 썩 효과적이지는 못하다. 정점들 다각형들에 의해 공유되고, 여러번 사용되게 된다. 예를 들어 정점 1은 이전의 예에서 사용되었지만, 다른 형태로 재사용이 가능하다: "1, 3, 7, -1" "10, 1, 3, 8, -1" 기타 등등.

3차원 Mesh를 indexed face(혹은 다각형)의 집합으로 다루는 대신에, 3D Mesh는 connected polygon이나 connected components 로 쉽게 표현 될수 있다. 현재 IndexedFaceSet의 표현인 VRML(and BIFS)를 보게 되면 연속적인 다각형은 다른 다각형과의 연결이 필연적이지는 않다. 일단 다각형의 3D 연결이 정의되면, 그것들은 새로운 정점의 리스트로 표현이 가능하다. 그리고 각각의 3D connected component들에 대한 각각의 맵을 가지고 있다.

3DMC에서 TS(Topological Surgery)는 3차원 Mesh에서 3차원 연결 정보를 분해하고, 2차원의 mesh map, triangle tree, vertex graph를 생성시킨다. 이것은 3차원 Mesh 구조의 연결 정보를 나타낸다 이것은 또한 듀얼그래프(dual graph) 표현이라고 불리는데, triangle tree와 vertex graph는 3차원 객체 형태로 서로 보완적인 관계에 있다.[2]

**B. Error-Resilience Mode**

유무선 환경에서 3차원 Mesh 코딩에서 디코딩 된 데이터에서 패킷 손실과 burst 에러를 최소화하기 위해서 데이터 재동기화 톨인 Error-Resilience Mode를 제공한다. 각각의 3DMC 파티션이나 패킷들의 vertex graph 와 triangle tree 정보들은 각각 독립적으로 표현 될 수 있다. 그림 3.1 에 3차원 Mesh 비트스트림에 대한 파티션 타입에 대한 3가지 모드가 있다.

VG	TT	TD	last_compo nent	....	VG	TT	TD	last_compo nent
----	----	----	--------------------	------	----	----	----	--------------------

(a) Partition Type 0

VG	last_vg	VG	last_vg	....	VG	last_vg
----	---------	----	---------	------	----	---------

(b) Partition Type 1

vg_id	left/right bounding loop index	boundary prediction mode	....	TT	branch_len gth	TD
-------	--------------------------------------	--------------------------------	------	----	-------------------	----

(c) Partition Type 2

그림 2.1 Partition type for 3DMC error resilience.

3DMC의 partitioning은 패킷의 사이즈가 모두 같지 않기 때문에 파티션 사이즈는 논리적이라기보다는 다소 물리적인 것에 가깝다. 그러나 파티션은 복합적인 패킷 사이즈로 설계되었다고 할수 있다. 작은 사이즈의 패킷으로 파티션 전체를 패킹(packing)하는 것은 불가능하다, 하지만 사이트 정보와 산술부호화와 같은 방법으로 많은 데이터 오버헤드를 줄일 수 있다.

에러 내성 모드(Error resilience mode) 없이, 진부한 비트 스트림들은 파티션 타입 0번의 형태이다. 이러한 에러 내성 모드에서 다중 파티션에 있는 TT와 TD로부터 VG를 식별 할수 있는데, VG는 전체 비트 스트림의 5~10%의 비트율을 차지한다. 그러나 VG 없이는 3차원 Mesh를 재구성하기는 불가능하다. 이러한 partitioning의 형태는 관련정보들의 우선순위에 의해서 유용하게 사용되어질수 있다. 남아있는 TT와 TD 데이터는 파티션 타입2에 포함되는데, 이것은 3차원 Mesh가 하나의 파티션 타입 1과 다중 파티션 타입 2를 가지고 있다고 할수 있다. 파티션 타입 0은 하나나 그이상의 connected component로 구성 된다. 만약 현재 connected component가 파티션에서 마지막이라면 last\_component 필드는 1-bit 플래그가 된다. VG, TT, 와 TD 데이터는 connected component로 구성 되어질수 있다. 파티션 타입 M에서 다중 VG들은 파티션에 묶여질수 있다. TT와 TD데이터에 각각의 VG 짝들은 비트스트림 안에서 순차적으로 디코딩 되게 되는데, 첫 번째 VG는 첫 번째 TT와 TD와 연결된다. 만약 현재 VG가 마지막 파티션이면 last\_vg 필드가 결정 된다.

파티션 타입 2에서 TT전에, 몇몇의 헤더 정보는 triangle tree(left/right bounding loop index)의 traversal을 쉽게하고, boundary prediction mode 를 인식하는데 첨부된다. 만약 파티션 타입 2가 존재하면, connected component는 여러 조각으로 분할 되게 된다. 이러한 것들은 nonmanifold models(repetitive coding of the same vertices) 과 비슷한 문제를 기한다. 다음은 파티션 사이의 경계선(boundary) 부호화의 두가지 모드가 있다.[2]

☞ Restricted boundary prediction mode : 이러한 모드는 파티션 사이의 꼭지점을 복제할수 없다. 이전의 파티션에서 예측된 꼭지점들은 이용할수 없다, 예측은 단지 꼭지점만 이용가능하고, 현재 파티션에서만 예측되어진다.

☞ Extended boundary prediction mode : 또 다른 방법으로, 이 모드는 다른 파티션과 공유된 꼭지점을 복제한다. 그러므로, 확장된 예측은 제한된 예측보다 부호화 효율은 덜 효과적이다. 그러나, 이전의 파티션이 손실 되었을때 제한된 예측의 경우는 triangle들은 이전의 파티션과 공유된 꼭지점들은 복원되지 않는다.

이와 같은 에러 내성 모드는 에러에 강인한 3차원 모델을 제공하지만, 데이터가 손실 되면 이들은 스트립 단위로 깨져서 나타나게 된다. 현재까지는 스트립 단위로 깨진 부분을 그대로 보여주는 방법만 사용되어 왔으나, 본 논문에서는 이 깨진 부분을 메꿔주는 데 중점을 두어서 연구 하였다.

### C. M.E.S.H

본 논문에서는 깨진 데이터와 복원된 데이터 사이의 오차를 계산하기 위하여 MESH(Measuring Errors between Surfaces using the Hausdorff distance)라는 프로그램을 사용하였다.

이 오차에 대한 분석을 통해서 데이터의 복원 정도를 판단하게 된다. MESH는 삼각형으로 이루어진 3차원 Mesh로 표현되는 서로 다른 3차원 평면 사이의 거리를 측정하는 효과적인 방법이다. 측정 기준은 불필요한 계산과 메모리 사용을 줄이기 위해 적절하게 사용된 Hausdorff 거리의 근사치에 근거를 두고 있으며, 다른 유사한 도구들과 비교해 볼 때 메모리나 속도 면에서 월등한 성능을 보여준다.

디지털 이미지 기술에서는 원본 데이터와 변형된 데이터 사이의 일대일 매핑을 요구하게 되는데 이러한 제약은 3차원 모델의 경우 실질적으로 너무 제한 적이게 된다.

3차원 모델에 적용된 많은 방법들은 단순히 기하학적인 왜곡 외에도 위상 정보의 변화 또한 내포하게 되는데, 바로 이점이 이러한 형태의 데이터에 대한 측정에 있어서 간단한 point to point 측정 방식 대신에 Hausdorff 측정 방법을 이용하는 것에 정당성을 부여하게 된다.

다음은 Hausdorff 거리에 대한 간략한 설명이다. [3]

평면 S에 속한 점 p 와 평면 S'사이의 거리  $d(p, S')$ 는 다음과 같이 정의된다.

$$d(p, S') = \min_{p' \in S'} \|p - p'\|_2 \quad \text{식 1}$$

$\|\cdot\|_2$  이 일반적인 Euclidean norm을 나타낸다고 할 때, 평면 S와 평면 S' 사이의 Hausdorff 거리는 다음과 같이 주어진다.

$$d(S, S') = \max_{p \in S} d(p, S')$$

식 1에서 정의된 점 대 면 거리는 두 평면 S와 S'

사이의 평균 오차  $d_m$  을 정의하는데 사용된다.

$$d_m(S, S') = \frac{1}{|S|} \iint_{p \in S'} d(p, S') dS \quad \text{식 2}$$

$|S|$  가 S의 면적을 나타낸다고 할 때, RMS 오차는 다음과 같다.:

$$d_m(S, S') = \sqrt{\frac{1}{|S|} \iint_{p \in S'} d(p, S')^2 dS} \quad \text{식 3}$$

아래의 그림 2.3은 M.E.S.H 프로그램의 실시 예이다.

본 논문에 대한 이해를 돕기 위하여 3차원 모델이 부분적으로 깨진 것을 선택 하였다.

그림의 왼쪽에 있는 것이 오차를 나타내는 색상표인데, 여기서는 파란색이 나타날수록 오차가 적으며 빨간색이 나타날수록 오차가 큰 것을 나타낸다.

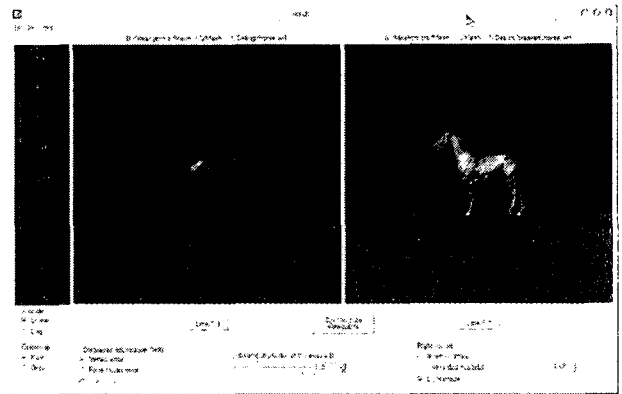


그림 2.3 M.E.S.H 프로그램 실행결과

### III. 복원 알고리즘

#### (1) Triangulation Problem

깨진 3차원 Mesh 모델을 복원하기 위해서는 빠진 부분을 메꿔주는 알고리즘을 필요로 하는데, 그것은 3차원의 데이터를 2차원으로 바꿔주는 문제와 같다. 만약 삼각형 하나가 손실이 되면 그것을 2차원으로 바꿔주는 방법은 한가지가 되는데, n 개 이상이 손실이 되면, 이것을 메꿔주는 방법의 가지 수는 n가지 방법 이상이 나오게 된다.

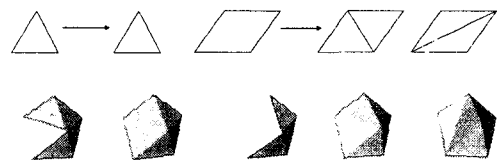
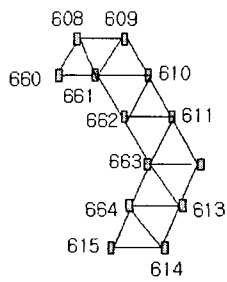


그림 3.1 Triangulation Problem

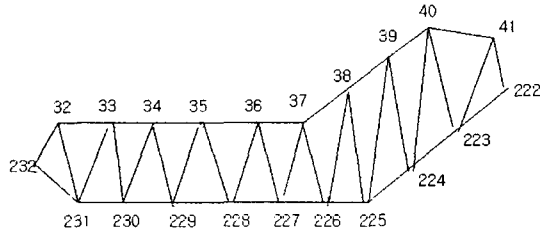
#### (2) Mesh의 순서와 Root Triangle을 결정

그림 3.2와 같이 스트립 단위가 가운데 부분을 절단 해나가는 경우에는 위 두 가지 방법을 사용하더라도 쉽게 복원되지 않는다. 그래서 이들이 가지고 있는 Mesh의 순서를 고려하여 Root Triangle을 결정하는 방법이다. 여기서 Root Triangle는 Triangulation 하기 위한 첫 번째 삼각형을 임의로 Root Triangle 이라고 명하였다.

우선 이들의 리스트인 608 609 610 611 612 613 614 615 664 663 662 661 660를 구하고, 이들이 순서를 가지고 있다는 것을 알수 있다. 그리고 대부분의 이들의 순서는 각각 값의 차이가 1인 것을 착안하여, 값의 차이가 1인 것 중 길이가 가장 긴 부분의 첫 값을 Root Triangle 이라고 고려하였다. 그리고 처음부터 순서대로 Triangulation 한다.



(a)



(b)

그림 3.2 Triangulation Algorithm

그림 3.2(a)(b)는 이 스트립이 하나, 두 개가 빠진 것을 나타낸 것이다.

그리고 그림 3.3을 보면 대부분 원래 데이터와 비슷하게 되었지만, 아직 끝 부분에서 잘려 나간 부분이 완벽하게 복원되지 못했다. 하지만 그림 3.4에서 보면 비교적 큰 3차원 모델에 대해서는 좋은 결과를 얻을 수 있었다.

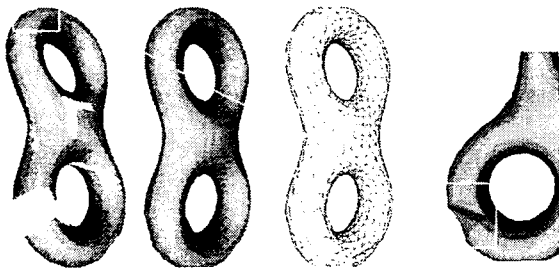


그림 3.3 Result of Algorithm method 3

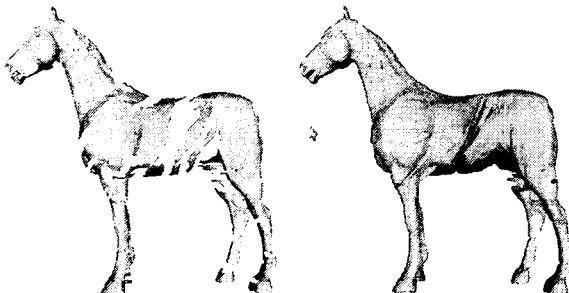


그림 3.4 Result of Algorithm method 3

#### IV. 결론 및 향후 과제

##### (1) Test Data

이 논문에서는 객관적인 실험 결과를 얻기 위해서 삼각형으로 이루어진 3차원 Mesh로 표현되는 서로 다른 3차원

평면사이의 거리를 측정하는 방법으로 사용하는 'M.E.S.H'라고 하는 프로그램을 사용하였고, 테스트 데이터는 SNHC(<http://sait.samsung.co.kr/snhc>)에 올라와 있는 VRML 파일을 사용하였다.

##### (2) 실험결과

알고리즘을 적용하여 실험한 결과 30% Packetlosses에서는 평균 31.56% 향상되었고, 10% Packetlosses에서는 평균 36.69% 향상 됨을 알 수 있었다.

test data	before(RMS)	after(RMS)	향상률(%)
eight	1.1069	0.5125	53.69
femur	1.3711	1.2606	8.06
horse	0.9619	0.8337	13.33
pieta	1.7979	1.1968	33.44
shape	0.6106	0.4306	29.48
skull	0.9831	0.4779	51.39

표 1 30% Packetlosses RMS 비교

test data	before(RMS)	after(RMS)	향상률(%)
eight	0.6085	0.1085	82.16
femur	0.4353	0.2845	34.65
horse	0.2153	0.1583	26.48
pieta	0.5814	0.4063	30.12
shape	0.7101	0.63144	11.07
skull	0.4779	0.3075	35.65

표 2 10% Packetlosses RMS 비교

#### V. 결론

위의 복원기법은 추가적인 데이터 없이, 주어진 데이터의 정보만 가지고 복원을 실시하였다. 그리고 Packetlosses가 적을수록 RMS 차이값의 향상율이 높아짐을 알 수 있었다.

#### 참고문헌

- [1] Taubin G., and J. Rossignac, Geometry Compression through Topological Surgery. *ACM Transactions on Graphics*, pp 84115, April 1998.
- [2] Pereira, Fernando/ Ebrahimi, Touradj, The Mpeg-4 Book, Prentice Hall Publishing 2002.
- [3]