

패치별 EGI 분포를 이용한 3D 메쉬 모델 워터마킹

이석환[†], 김태수^{**}, 김병주^{***}, 김지홍^{****}, 권기룡^{*****}, 이건일^{*****}

요 약

본 논문에서는 패치별 EGI 분포를 이용한 3D 메쉬 모델(mesh model) 워터마킹 알고리즘을 제안하였다. 제안한 알고리즘에서는 기하학적 변형에 견고하기 위하여 3D 메쉬 모델을 6개 패치로 분할한다. 그리고 위상학적 변형에 견실한 특성을 가지는 EGI 분포를 각 패치별로 구한다. 그리고 동일한 워터마크 비트열을 각 패치의 EGI 분포 중에서 길이가 큰 면체에 투영된 메쉬 법선 벡터들에 각각 삽입한다. 그러므로 제안한 알고리즘은 기하학적 및 위상학적 변형에 견고하며, 특히 워터마크 추출시 복잡한 재표본화 과정 및 원 모델이 필요 없다. 본 논문에서 제안한 워터마킹 알고리즘의 성능 평가를 위한 모의 실험에서 워터마크가 삽입된 모델의 비가시성 및 다양한 공격에 대한 견고성이 우수함을 확인하였다.

A Watermarking of 3D Mesh Model using EGI Distributions of Each Patch

Suk-Hwan Lee[†], Tae-Su Kim^{**}, Byung-Ju Kim^{***}, Ji-hong Kim^{****},
Ki-Ryong Kwon^{*****}, Kuhn-Il Lee^{*****}

ABSTRACT

Watermarking algorithm for 3D mesh model using EGI distribution of each patch is proposed. The proposed algorithm divides a 3D mesh model into 6 patches to have the robustness against the partial geometric deformation. Plus, it uses EGI distributions as the consistent factor that has the robustness against the topological deformation. To satisfy both geometric and topological deformation, the same watermark bits for each subdivided patch are embedded by changing the mesh normal vectors. Moreover, the proposed algorithm does not need not only the original mesh model but also the resampling process to extract the watermark. Experimental results verify that the proposed algorithm is imperceptible and robust against geometrical and topological attacks.

Key words: mesh watermarking(메쉬 워터마킹), EGI, patch(패치)

※ 교신저자(Corresponding Author) : 이석환, 주소 : 대구광역시 북구 산격동 1370번지 경북대학교 대학원 전자공학과(702-701), 전화 : 053)940-8812, FAX : 053)950-5505, E-mail : skyllee@palgong.knu.ac.kr

접수일 : 2003년 1월 29일, 완료일 : 2003년 4월 14일

[†] 정회원, 경북대학교 전자공학과 대학원 박사과정

^{**} 정회원, 경북대학교 전자공학과 대학원 박사과정 (E-mail : kts1101@palgong.knu.ac.kr)

^{***} 정회원, 국방과학연구소 연구원 (E-mail : bjleokim@hotmail.com)

^{****} 정회원, 동의대학교 영화영상공학과 조교수 (E-mail : arim@dongeui.ac.kr)

^{*****} 종신회원, 부산외국어대학교 컴퓨터전자공학부 부교수 (E-mail : krkwon@mail.pufs.ac.kr)

^{*****} 정회원, 경북대학교 공과대학 전자공학과 교수 (E-mail : kilee@ee.knu.ac.kr)

※ 본 과제는 정보통신부의 정보통신기술기초연구지원사업(정보통신연구진흥원)으로 수행한 연구결과입니다.

1. 서 론

최근에 정지 영상, 오디오, 및 비디오 등의 디지털 미디어에 대한 워터마킹의 연구가 많이 진행되어지고 있다. 그러나 Web 3D, MPEG4, VRML, 및 3D 기하학적 CAD 데이터 등의 3D 데이터들은 웹 기반 하에서 많이 이용되어지고 있는 반면, 저작권 보호에 대한 효과적인 방법이 거의 없다. 이러한 문제점을 해결하기 위하여 최근에 3D 모델에 대하여 워터마크를 삽입하는 연구가 시작되어지고 있다.

3D 모델 데이터들은 정지 영상, 오디오, 및 비디오 등의 데이터와는 다른 특징을 가지고 있다. 먼저 다양한 기하학적(geometrical) 및 위상학적(topological) 연산에 의하여 편집이 용이하다. 그리고 3D 모델의

절대적인 표현 방법이 없다. 즉 하나의 모델에 대하여 꼭지점, 에지, 및 면들이 다르게 구성되어 있는 여러 메쉬들로 표현이 가능하다. 또한 모델 데이터의 정렬 순서가 절대적이지 않다. 그러나 공간상에서 모델의 위치 및 방위각이 필요하다. 마지막으로 렌더링(rendering) 속도를 높이기 위하여 모델 데이터 양을 메쉬 간단화(mesh simplification)에 의하여 줄일 수 있다.

3D 메쉬 모델의 대표적인 워터마킹 알고리즘으로 Ohbuchi 등[1,2]은 3D 메쉬 모델의 메쉬 분광 영역(mesh spectral domain)에서 워터마크를 삽입하였다. 그러나 이 알고리즘은 메쉬 간단화(mesh simplification) 및 리메쉬(remesh)와 같이 메쉬 연결성을 바꾸는 공격에 대하여 견고하지 못하는 단점을 갖는다. Kanai 등[5]이 제안한 알고리즘에서는 3D 메쉬 모델의 다해상도 웨이블릿 분해를 이용하여 3D 웨이블릿 계수 벡터에 워터마크를 삽입한다. 이 알고리즘에서 적용된 웨이블릿 분해는 4개의 메쉬를 1개의 메쉬로 분할하는 일정 형태의 부분할 기법이므로, 3D 메쉬 모델의 위상(topology)을 4-to-1의 연결성으로 바꾸어야 하는 단점이 있다. Praun 등[6]은 점차적 메쉬(progressive mesh)의 다해상도 해석을 이용하여 메쉬의 꼭지점 상에서 일련의 스칼라 기저함수(scalar basis function) 집합을 구한 후, 기저함수에 가중치를 둔 메쉬의 법선 벡터 방향에 따라 워터마크를 삽입하는 알고리즘을 제안하였다. 그러나, 이 알고리즘에서는 워터마크 검출시 원 모델의 동일한 기하학적 및 위상학적 특성을 가지는 재표본화(resampling) 과정이 필요하다. 즉, 메쉬 간단화, 리메쉬, 및 절단과 같은 공격에 강하면서, 워터마크 검출시 재표본화 과정 및 원 모델이 필요없는 알고리즘이 필요하다. Benedens [3,4]이 제안한 알고리즘에서는 3D 형상을 인식할 수 있는 요소이며 또한 위상학적 변형에 변하지 않는 메쉬 법선 벡터 분포를 이용하였다. 이 알고리즘에서는 미리 정의된 하나의 원점에 대한 메쉬들의 표면 법선 벡터 분포를 이용한다. 즉, 하나의 EGI 분포에서 랜덤하게 선택된 방향에 속해 있는 메쉬들의 법선 벡터 분포에 워터마크를 삽입한다. 그러나 절단과 같은 기하학적 공격이 가해질 경우, 그 방향에 속해 있는 메쉬들이 없어지므로, 삽입된 워터마크 정보가 없어진다. 그러므로 여러 방향에 대한 EGI 분포들의 집합에 대하여 각 방향에 대한

EGI 크기 분포에 따라 동일한 워터마크를 삽입하면 절단 공격이 가해지더라도 나머지 방향에 대한 EGI 분포가 존재하므로 워터마크를 손실없이 검출할 수 있다.

본 논문에서는 3D 메쉬 모델에서 6개로 분할된 패치의 EGI 분포를 이용하여 워터마크를 삽입하는 알고리즘을 제안하였다. EGI (Extended Gaussian Image)[12-14]는 메쉬 법선 벡터들의 방향에 따른 분포를 알 수 있으며, 메쉬 간단화 및 리메쉬 등의 위상학적 변형에 변하지 않는다. 제안한 알고리즘에서는 절단과 같은 기하학적 변형에 견고하기 위하여 3D 메쉬 모델을 6개의 패치로 분할한다. 그리고 각 패치별로 EGI 분포를 구한 후, EGI 분포에서 길이가 큰 면체에 투영된 메쉬 법선 벡터들에 워터마크를 삽입한다. 즉, 꼭지점에 연결된 모든 메쉬 법선 벡터들이 각각의 워터마크 정보를 만족하는 최적의 꼭지점을 구한 후, 이 꼭지점으로 이동시킨다. 그러므로, 제안한 알고리즘은 기하학적 및 위상학적 공격에 견고하며, 특히 워터마크 추출시 복잡한 재표본화 과정 및 원 모델이 필요 없다.

본 논문에서 제안한 알고리즘의 성능을 평가하기 위한 실험 결과에서, 제안한 알고리즘이 메쉬 간단화 및 리메쉬 등의 위상학적 공격과 절단 등의 기하학적 공격에 대하여 우수한 견고성을 가짐을 확인할 수 있었다. 또한 워터마크 삽입시 단계적 탐색 방법을 이용하여 최적의 꼭지점을 구하므로 우수한 비가시성을 확인할 수 있었다.

2. 3D 워터마킹 이론

본 장에서는 3D 모델의 워터마킹에 대한 삽입 조건, 공격 형태, 그리고 대표적인 3D 메쉬 데이터인 VRML에 대하여 살펴본다.

2.1 삽입 필요 조건

일반적인 3D 모델의 워터마크 삽입 시 필요한 기술적 요구조건은 다음과 같다.

1) 비가시성 (Invisibility) : 2D 영상의 비가시성과 동일한 특성으로 워터마크가 삽입된 3D 모델이 원래의 3D 모델과 구분이 없어야 한다. 그러나 3D 모델에서는 2D 영상에서의 PSNR과 같은 객관적 화질 척도가 없다. 그러므로 시각적으로 워터마크의 삽입 여부

를 판단해야 하는 어려움이 없다.

2) 견고성 (Robustness): 삽입된 워터마크가 다양한 공격에 대하여 제거 혹은 파괴가 어렵거나 불가능해야 한다. 3D 메쉬 데이터인 VRML에서는 꼭지점 및 연결성의 정보를 가지고 있다. 이들 데이터의 형상은 꼭지점의 개수가 줄어들거나, 꼭지점의 좌표가 바뀌어지거나 혹은 연결성의 정보가 바뀌어져도 달라지지 않는다. 그러나 이러한 공격에 대하여 워터마크가 손실되어서는 안 된다.

3) 용량성 (Capacity): 저작자 정보나 저작권 보호에 필요한 정보를 기록하기에 충분한 양의 워터마크를 삽입해야 한다.

이러한 조건들은 서로 절충 관계 (trade-off)이므로 동시에 최적화 될 수 없다. 다양한 공격에 대하여 더욱 더 견고해야 한다면 삽입되어야 할 워터마크의 데이터 양이 줄어들어야 한다. 그러므로 워터마크 시스템이 사용되는 응용 분야에 따라 적절한 타협점이 필요하다.

2.2 워터마크가 삽입된 3D 모델에 대한 공격 형태

워터마크가 삽입된 3D 메쉬에 대한 공격은 꼭지점 랜덤화(vertex randomization), 리메쉬(remesh), 메쉬 단순화(mesh simplification), 아핀 변환(affine transformation), 절단(cropping) 등이 있다.

첫째, 꼭지점 랜덤화는 3D 메쉬의 꼭지점 좌표에 랜덤 잡음을 첨가하는 공격이다. 그러므로 꼭지점의 좌표가 이러한 잡음에 의하여 이동하게 된다. 꼭지점 좌표에 워터마크를 삽입하는 알고리즘에서는 부과된 잡음의 크기 및 분포가 커질수록 BER (bit error rates)가 커져서 워터마크를 검출할 수 없다. 둘째, 리메쉬는 꼭지점 위치의 변경 없이 메쉬 연결성을 새로이 바꾸는 공격이다. 그리고 메쉬 단순화는 꼭지점의 수를 줄이면서 해상도를 낮추는 공격으로 에지 붕괴 방법(edge collapse operations) 및 QEM (quadratic error metric) 방법 등이 있다. 리메쉬 및 메쉬 단순화 공격은 꼭지점의 변경뿐만 아니라 연결성이 변경되는 공격 형태로써 단순히 꼭지점이나 연결성 정보에 워터마크를 삽입할 경우 워터마크를 검출할 수 없다. 셋째, 아핀 변환 공격에는 회전 (rotation), 이동 (translation), 비례 축소(scaling) 및 쉬어링(shearing)의 방법이 있다. 아핀 공격된 3D 메쉬에서는 연결성 정보는 바뀌지 않으나 모든 꼭지점들의 좌표가

바뀌게 된다. 따라서 아핀 공격된 메쉬에서는 워터마크를 추출하기 위하여 원 모델로 재정렬(realignment)하는 과정이 필요하다. 넷째, 절단 공격은 꼭지점들의 일부분을 잘라내는 공격 형태이다. 전체 3D 메쉬에 대하여 워터마크를 삽입할 경우 워터마크 삽입된 정보의 일부분이 없어지므로 워터마크를 검출할 수 없다.

2.3 삽입 대상 모델

최근 웹 상에서는 평면 이미지뿐만 아니라 3차원 그래픽을 구현할 수 있는 Web 3D 기술에 대한 관심이 증가되고 있다. 이들 중 가상 현실 구현 언어 (virtual reality modeling language, VRML)는 여러 가지 장점들로 인하여 특히 주목받고 있다. 첫째, 국제 표준 기구인 ISO/IEC가 웹 상에서 3차원 그래픽을 표현하는 표준으로 VRML을 공인하고 있다. 둘째, VRML은 소스 자체가 공개되어 있고 라이선스를 요구하지 않아 원천 기술 개발 뿐만 아니라 누구나 자유로이 웹 상에서 3차원 그래픽을 표현할 수 있다.

VRML 파일 포맷은 그림 1에서와 같이 꼭지점 좌표(point)와 그 꼭지점들의 연결성(coordIndex)으로 구성된다. 이 그림에서 살펴보면, 하나의 꼭지점은 소수 6자리를 가지는 3개의 실수들로 구성되어 있으며, 이들 실수들은 각각 x , y 및 z 축의 좌표를 나타낸다. 하나의 메쉬는 세 개의 꼭지점들의 연결로 이루어지며, 이들에 대한 정보는 coordIndex에 나타나있다. 예를 들면, coordIndex에서 앞에 있는 네 개의 숫자 0 1 2 -1은 하나의 메쉬를 나타낸다. 여기서 0 1 2은 각각 0번째, 1번째, 그리고 2번째 꼭지점을 나타내며, -1은 하나의 메쉬가 끝남을 알리는 표시이

```
#VRML V2 0 utf8
Shape {geometry IndexedFaceSet { solid FALSE
  coord Coordinate {
    point [ -0.134807 0.114232 0.0386105, -0.179442 0.120314 0.0221062,
            -0.328641 0.263905 0.248764, 0.0934687 0.128426 0.159006,
            -0.0370266 0.106107 0.0558309, -0.0530986 0.101265 0.0499631,
            -0.130255 0.111078 0.0214038, 0.321475 0.0162994 0.187689,]}
    coordIndex [0 1 2 -1, 4 5 6 -1, 1 5 6 -1, 0 4 7 -1, 0 1 5 -1, 7 3 2 -1]
    colorPerVertex FALSE
    color Color {color [0 0 1, 0 1 0, 1 1 0, 0 1 1, 1 0 0, 1 0 1]}}}
```

그림 1. VRML 데이터 파일 구조

다. 본 논문의 실험 또한 이들 메쉬로 표현된 VRML 파일을 사용하였다.

3D 모델의 사용 용도에 따라 워터마킹에 요구되는 조건이 다르다. 높은 정밀도를 필요로 하는 메쉬 데이터 경우에는 원형상의 변형 없는 워터마킹 알고리즘이 제안되어야 한다. 예를 들면 캐속조형 시스템에서 사용되는 3D STL 파일에서는 패시의 법선 영역의 시작 위치와 꼭지점 영역의 순서 정보를 바꿈으로써 원 형상을 변경하지 않고 워터마크를 삽입한다 [11]. 여기서 패시 (facet)은 STL 파일에서 3개의 꼭지점이 연결된 메쉬를 나타낸다. 이와 같은 경우에는 패시의 저장 순서를 변경하는 공격에 대한 강인성을 가지기 위해서는 패시의 저장 순서에 종속적이지 않는 워터마크를 삽입하여야 한다. 한편, Web 상에서의 3D 그래픽을 표현하는 VRML 파일에서는 워터마크 삽입 시 높은 정밀도보다는 기하학적인 변형과 위상학적 변형에 대한 강인성을 고려해야 한다. 본 논문에서는 이러한 VRML 메쉬 데이터에 대하여 워터마크 비트에 따라 꼭지점 좌표를 비가시성을 만족하는 범위 내에서 변화시킴으로써 법선 벡터의 분포를 변화시켰다.

3. 제안한 3D 워터마킹 알고리즘

본 논문에서 제안한 3D 메쉬 모델의 워터마크 삽입 알고리즘의 전체 블록도는 그림 2에서와 같다. 3D VRML 데이터의 꼭지점 (vertex) 및 메쉬의 연결성 (mesh connectivity)을 입력으로 하여, 각 꼭지점의 좌표를 워터마크 정보에 따라 이동시킨다. 워터마크 검출시 각 패치의 중점 좌표값 및 각 패치에 대한 EGI 분포의 순위 정보가 필요하다.

3.1 3D 메쉬 모델의 패치 분할

3D 메쉬 모델에서 절단과 같은 기하학적 변형이

가해질 경우, 워터마크 정보를 가지는 꼭지점들의 일부가 없어진다. 그러나 제안한 알고리즘에서는 3D 메쉬 모델을 6개 패치들로 분할한 후, 동일한 정보의 워터마크를 각 패치별로 삽입함으로써, 기하학적 변형과 같은 공격에도 견고성을 가지게 된다. 3D 메쉬 모델의 모든 꼭지점들은 각 패치의 중점들과의 거리를 구한 후, 거리가 최소가 되는 패치로 분류된다.

먼저 패치 P 의 중점 I 을 구하기 위하여 그림 3에서와 같이 모든 방향에 대하여 일정한 거리를 나타내는 6개 방향에 대하여 초기 위치를 선정한다. 즉, 6개의 단위 벡터 방향 $\pm x, \pm y, \pm z$ 에 인접한 꼭지점 중 원점에서 거리가 가장 먼 위치에 있는 6개의 꼭지점을 I 의 초기 위치로 설정한다. 그리고 각 꼭지점 v 을 6개 패치의 중점 I 와의 거리가

$$P_i = \{ v: d(v, I_i) < d(v, I_j), \text{ all } j \neq i, 1 \leq j \leq 6 \} \quad (1)$$

와 같이 최소가 되는 패치 P_i 로 분류한다. 여기서, $d(v, I)$ 는 $\|v - I\|^2$ 이다.

각 패치 $P_{i \in [1, 6]}$ 에 속해 있는 꼭지점들의 무게중심을 구한 후,

$$I_i = E[v | v \in P_i], 1 \leq i \leq 6 \quad (2)$$

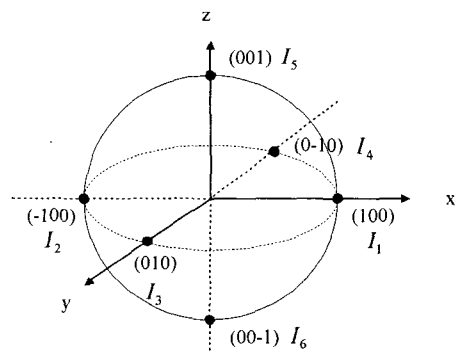


그림 3. 6개 패치 P 로 구분하기 위한 초기 위치 설정의 6개 방향 벡터

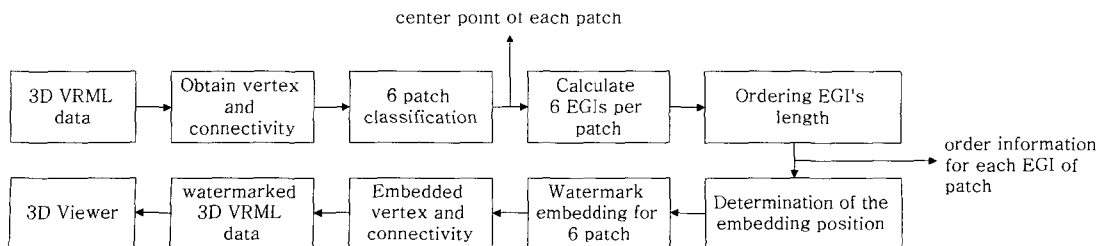


그림 2. 제안한 3D 워터마크 삽입 알고리즘의 전체 블록도

와 같이 I_i 에 대입한다. 그리고 I_i 에 대하여 (1) 식을 이용하여 각 꼭지점을 다시 분류한다. 이러한 과정을 거리 감소 비율 $(D_m - D_{m+1}) / D_m$ 이 문턱값 보다 작을 때까지 반복 수행한다. 여기서 D_m 은

$$D_m = \sum_{i=1}^6 \sum_{v \in P_i} \|I_i - v\|^2 \quad (3)$$

이며, m 은 반복 횟수를 나타낸다.

3.2 각 패치별 EGI 분포

3차원 공간상에서의 법선 벡터 분포는 방향 히스토그램 (orientation histogram)인 EGI 분포에서 알 수 있다. 본 논문에서는 EGI 분포를 구하기 위하여 그림 4에서와 같이 길이 1인 구를 동일한 면적을 가지는 240면체로 분할하여, 각 메쉬의 표면 법선 벡터를 240면체 중 가장 근접한 방향을 가지는 면체로 투영시킨다. 먼저 각 패치별 메쉬의 법선 벡터를 구한다. 여기서 표면 법선 벡터는 면에 대하여 안쪽 방향 및 바깥쪽 방향의 두 가지 방향을 가진다. 정확한 방향을 판단하기 위하여 원점에서의 벡터가 아닌 각 패치의 중점 I 에 대한 꼭지점들의 벡터에 대한 메쉬의 법선 벡터 및 면적을 구한다. 즉, i 번째 패치의 j 번째 메쉬의 법선 벡터 \vec{n}_{ij} 및 면적 A_{ij} 은

$$\vec{n}_{ij} = \frac{1}{\|\vec{n}_{ij}\|^2} (\vec{I_i v_1} - \vec{I_i v_2}) \times (\vec{I_i v_1} - \vec{I_i v_3}) \quad (4)$$

$$A_{ij} = \|(\vec{I_i v_1} - \vec{I_i v_2}) \times (\vec{I_i v_1} - \vec{I_i v_3})\| / 2 \quad (5)$$

이다. 여기서 메쉬의 법선 벡터는 그림 5에서와 같이 바깥 방향 및 안쪽 방향에 대하여 \vec{n}_{ij} 및 $-\vec{n}_{ij}$ 가 된다. 두 벡터 중, 패치의 중점 I_i 과 메쉬의 중점 m_j 을 잇는 벡터 $\vec{I_i m_j}$ 와의 내적이 큰 값을 메쉬의 법선 벡터로 결정한다.

각 패치별로 메쉬의 모든 법선 벡터 \vec{n} 를 위와 같

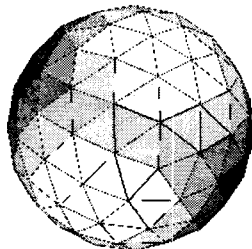


그림 4. 동일한 면적을 가지는 240면체로 분할된 단위 구

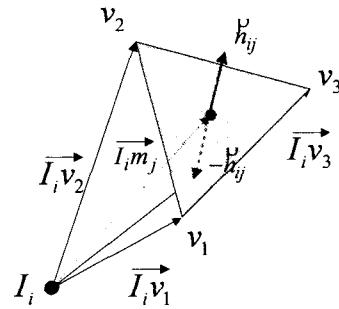


그림 5. i 번째 패치 P_i 로 분류된 j 번째 메쉬의 법선 벡터

이 구한 후, 이 벡터를 240면체 중 가장 가까운 방향에 있는 면체로 투영시킨다. 즉, \vec{n} 와 240면체 B 의 단위 법선 벡터 \vec{BC}_j 와의 내적이

$$B_i = \{ \vec{n} : \|\vec{BC}_i \cdot \vec{n}\|^2 < \|\vec{BC}_j \cdot \vec{n}\|^2 \text{ all } j \neq i, 1 \leq j \leq 240 \} \quad (6)$$

와 같이 최소가 되는 면체 B_i 로 투영시킨 후, 이 벡터의 면적을 투영된 B_i 의 길이에 더한다. 여기서 B_i 의 길이는 B_i 에 투영된 모든 법선 벡터의 면적의 합을 나타낸다. EGI 분포에서 B_i 의 길이가 크다는 것은 이 방향에 대한 메쉬들이 많이 분포함을 나타낸다. 즉, 길이가 큰 B_i 의 법선 벡터들은 다양한 공격에도 변하지 않는다. 제안한 방법에서는 이러한 B_i 에 속하는 모든 법선 벡터들의 평균 사이각을 이용하여 워터마크를 삽입한다. B_i 에 투영된 모든 메쉬 법선 벡터 \vec{n} 와 B_i 의 중점 벡터 \vec{BC}_i 와의 사이각 $\theta = \cos^{-1}(\vec{n} \cdot \vec{BC}_i)$ 은 $0 \leq \theta \leq \theta_m$ 의 범위를 갖는다. 여기서 θ_m 은 서로 인접한 B_i 및 B_{i+1} 들의 중점 벡터 \vec{BC}_i 및 \vec{BC}_{i+1} 와의 사이각의 0.5배이다. θ_m 근처에 있는 \vec{n} 는 공격시 다른 B 으로 투영될 수 있으므로, 제안한 알고리즘에서는 그림 6에서와 같이 실험적으로 구한 θ 범위

$$0 \leq \cos^{-1}(\vec{n} \cdot \vec{BC}_i) \leq \theta_{\max} \quad \theta_{\max} < \theta_m \quad (7)$$

$$\theta_{\max} = 8.6$$

안에 있는 \vec{n} 만을 B 로 투영시킨다.

3.3 EGI 분포에 따른 워터마크 위치 결정

각 패치별 EGI 분포에서 B 의 길이의 크기에 따라 순위를 정한 후, 워터마크 길이만큼 B 를 선택한다.

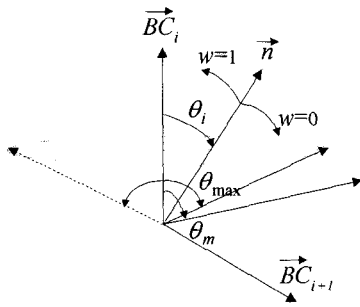


그림 6. B_i 에 투영된 메쉬 법선 벡터 \vec{n} 와 B_i 의 중점 벡터 \vec{BC}_i 와의 사이각.

즉, n 개의 워터마크 삽입시 각 패치별 n 개의 순위 정보가 필요하며, 총 $n \times B$ 개의 면체에 투영된 모든 법선 벡터들이 워터마크에 의하여 이동하게 된다. 제안한 방법에서는 1 비트의 워터마크를 6개 패치별 EGI 분포 중 동일한 순위에 있는 6개의 B 에 삽입한다. 즉, j 번째 워터마크 w_j 를 그림 7에서와 같이 i 번째 패치의 EGI 분포에서 j 번째 길이 크기 순위를 가지는 B_{ij} 에 삽입한다.

watermark	w_1	w_2	w_3	...	w_N
1st. patch	B_{11}	B_{12}	B_{13}	...	B_{1N}
2nd. patch	B_{21}	B_{22}	B_{23}	...	B_{2N}
3rd. patch	B_{31}	B_{32}	B_{33}	...	B_{3N}
	\vdots	\vdots	\vdots	...	\vdots
6th. patch	B_{61}	B_{62}	B_{63}	...	B_{6N}

그림 7. 6개 패치별 EGI 분포에서 길이순으로 정렬된 B 에 동일한 워터마크 삽입.

3.4 워터마크 삽입

제안한 알고리즘에서는 B 에 투영된 모든 메쉬 법선 벡터들을 1비트의 랜덤 시퀀스(random sequence)으로 이루어진 워터마크 정보에 의하여 이동시킨다. 즉, i 번째 패치의 EGI 분포에서 j 번째 길이 순위인 B_j 에 투영된 모든 메쉬 법선 벡터 \vec{n} 와 B_j 의 중점 벡터 \vec{BC}_j 와의 사이각 θ 가 j 번째 워터마크 w_j 에 따라 바뀌어지도록 각 꼭지점 v 의 위치를 이동시킨다. 여기서, 워터마크 w_j 가 1일 경우, 그림 6에서와 같이 θ 를 0의 방향으로 이동시키고, 워터마크 w_j 가 0일 경우 θ 를 θ_{max} 방향으로 이동시킨다. 그러나 하나의 꼭지점에는 여러 개의 메쉬가 연결되어 있고,

각 메쉬의 법선 벡터들이 다른 면체로 투영될 수 있다. 그러므로 하나의 꼭지점에 연결된 메쉬의 법선 벡터들이 워터마크 정보를 모두 만족하는 꼭지점의 위치를 찾아야 한다. 이를 위하여 제안한 알고리즘에서는 단계적 검색 방법을 이용하여 각 꼭지점의 탐색 범위 내에 비용 함수가 최소가 되는 최적의 위치를 찾는다.

3.4.1 꼭지점의 탐색 범위

워터마크 삽입시 비가시성을 고려하기 위하여 현재 꼭지점 v 의 탐색 범위 내에서 최적의 꼭지점 v^* 를 찾는다. 여기서 탐색 범위는 현재 꼭지점 v 에 연결된 모든 꼭지점 $v = valence(v)$ 들 중 각 축의 최소값을 넘지 않도록 설정된다. 현재 꼭지점 v 의 좌표가 (x, y, z) 일 때, 각 축에 대한 탐색 범위는 $x \pm \Delta x, y \pm \Delta y$ 및 $z \pm \Delta z$ 이다. 여기서 $\Delta x, \Delta y$ 및 Δz 는 각각

$$\begin{aligned} \Delta x &= 0.5 \times \min |x - v_x^i|_{v^i \in valence(v)}, \\ \Delta y &= 0.5 \times \min |y - v_y^i|_{v^i \in valence(v)}, \\ \Delta z &= 0.5 \times \min |z - v_z^i|_{v^i \in valence(v)} \end{aligned} \quad (8)$$

이며, v_x^i, v_y^i 및 v_z^i 는 v 에 연결된 꼭지점 v^i 의 각 축에 대한 좌표값이다. 현재 꼭지점 v 에 대한 탐색 범위 내에서 v 에 연결된 모든 메쉬 법선 벡터들이 각각의 워터마크 정보를 만족하는 최적의 꼭지점 v^* 를 구한 후, 이 꼭지점으로 이동된다. 그리고 최적의 꼭지점 v^* 는 현재 꼭지점 v 에 연결된 모든 꼭지점들의 좌표를 벗어나지 않는다.

3.4.2 단계적 검색 방법

현재 꼭지점 v 는 탐색 범위 내에서 비용 함수 $cost(v)$ 가 최소가 되는 최적의 꼭지점 v^* 위치로 이동한다. 그러나 탐색 범위 내에서 존재하는 모든 꼭지점 위치에 대한 비용 함수를 구하려면 처리 시간이 길어진다. 즉, 꼭지점의 좌표가 소수 6자리의 실수인 VRML 파일에서는 탐색 범위 내에 무수히 많은 꼭지점의 위치가 존재한다. 그러므로 제안한 알고리즘에서는 단계적 검색 방법을 이용하여 최적의 꼭지점을 구한다. 첫 번째 단계에서는 현재 꼭지점 $v = (x, y, z)$ 에서 탐색 범위의 최종 위치에 있는 27개의 꼭지점 v'

$$v' = \{ (x', y', z') \mid \begin{aligned} x' &\in \{x - \Delta x, x, x + \Delta x\}, \\ y' &\in \{y - \Delta y, y, y + \Delta y\}, \\ z' &\in \{z - \Delta z, z, z + \Delta z\} \end{aligned} \} \quad (9)$$

들의 비용 함수 값을 각각 구한다. 그리고 현재 꼭지점 v 를 v' 위치에 대한 비용 함수 값들 중에서 최소의 값을 가지는 위치의 꼭지점 v^*

$$v^* = \arg[\min_{v'} \{\cos t(v')\}] \quad (10)$$

으로 그림 8에서와 같이 이동시킨 후, Δx , Δy 및 Δz 를 각각 $\Delta x/2$, $\Delta y/2$ 및 $\Delta z/2$ 으로 줄인다. 두 번째 단계에서는 줄어든 범위 내에서의 27개 꼭지점 중 최소의 비용 함수를 가지는 꼭지점으로 다시 이동시킨다. 이러한 과정을 모든 꼭지점들에 대하여 세 단계까지 반복적으로 수행한다.

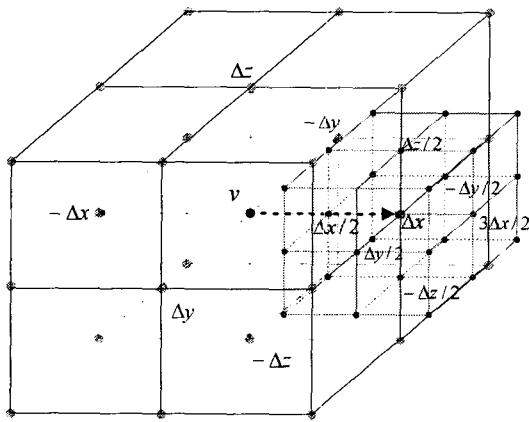


그림 8. 최적의 꼭지점을 구하기 위한 단계적 검색 방법

3.4.3 비용 함수

제안한 알고리즘에서는 각 단계별로 꼭지점의 탐색 범위 내의 27개 위치에 따라 이에 연결된 메쉬들의 비용 함수를 각각 구한다. 여기서 비용 함수는 두 가지의 워터마크 삽입 조건을 만족한다. 첫 번째로는 꼭지점에 연결된 메쉬의 법선 벡터가 워터마크 삽입 대상 면체에 투영되지 않으면, 이 메쉬는 비용 함수 계산에서 제외된다. 두 번째로는 꼭지점에 연결된 메쉬의 법선 벡터가 워터마크 비트 1이 삽입되는 면체에 투영되면, 이 면체의 중점 벡터와의 사이각 θ 가 0에 가까운 방향으로 이동되어야 한다. 그러나 워터마크 비트가 0일 경우에는 사이각 θ 가 θ_{max} 에 가까운 방향으로 이동되어야 한다.

위의 조건을 만족하는 꼭지점 v 에서의 비용 함수 $\cos t(v)$ 는

$$\cos t(v) = \sum_{m_i \in S(v)} a_{m_i} |\theta_{w_i} - \theta_{m_i}| \quad (11)$$

와 같다. 여기서 $S(v)$ 는 v 에 연결된 모든 메쉬들을 나타내며, θ_{m_i} 는 $S(v)$ 의 i 번째 메쉬 m_i 의 법선 벡터 \vec{n}_i 와 m_i 이 속해 있는 패치의 EGI 분포 중 이 벡터가 투영된 B_i 의 \vec{BC}_i 와의 사이각을 나타낸다. \vec{n}_i 가 투영된 B_i 이 워터마크가 삽입되는 면체이면 a_{m_i} 는 1이고, 워터마크 삽입 면체에서 제외되면 a_{m_i} 는 0이다. 그리고 \vec{n}_i 가 투영된 B_i 에 삽입될 워터마크 비트가 1이면 θ_{w_i} 가 0이고, 워터마크 비트가 0이면 θ_{w_i} 가 θ_{max} 이다.

모든 꼭지점들이 이에 연결된 메쉬 법선 벡터들의 워터마크 정보를 각각 만족하는 위치로 이동한다. 그러므로, 6개 패치의 EGI 분포 중에서 워터마크가 삽입되는 면체로 투영되는 모든 법선 벡터들의 분포가 워터마크 정보에 따라 모두 이동하게 된다.

3.5 워터마크 추출

제안한 알고리즘에 의하여 삽입된 워터마크의 검출 과정은 삽입 과정과 유사하며, 기존의 알고리즘과는 달리 원 모델이 필요 없다. 워터마크가 삽입되었다고 추정되는 모델에 대하여 워터마크 삽입 과정에서 얻은 각 패치의 중점 I 를 이용하여 6개 패치로 분류한다. 그리고 각 패치별 메쉬의 법선 벡터 및 EGI 분포를 구한다. 각 패치별 워터마크가 삽입된 B 의 정렬 순서를 이용하여 B 에 투영되는 모든 메쉬 법선 벡터들과 B 의 중점 벡터와의 평균 사이각에 의하여 워터마크를 검출한다.

j 번째 비트의 워터마크 w_j 는 각 패치별 EGI 분포에서 j 번째 순위를 가지는 B_{ij} 에서의 평균 사이각 θ_{ij} 들의 전체 평균값 $\bar{\theta}_j$ 가

$$\bar{\theta}_j = 1/6 \sum_{i=1}^6 \theta_{ij} \quad (12)$$

특정 문턱값 θ_{th} 보다 작으면, w_j 은 1이고, θ_{th} 보다 크면 w_j 은 0으로 검출된다. 여기서 i 번째 패치의 j 번째 순위를 가지는 B_{ij} 에서 B_{ij} 에 투영된 메쉬의 법선 벡터들과 B_{ij} 의 중점 벡터 \vec{BC}_{ij} 와의 평균 사이각 θ_{ij} 은

$$\theta_{ij} = 1/N \sum_{k=0}^N \cos^{-1}(\vec{BC}_{ij} \cdot \vec{n}_k) \quad (13)$$

이며, N 은 B_{ij} 에 투영되는 메쉬 법선 벡터들의 수를 나타낸다.

4. 실험 결과 및 고찰

제안한 3D 워터마킹 알고리즘의 성능을 평가하기 위하여 컴퓨터 모의 실험을 행하였다. 본 실험에서는 펜티엄 IV PC (Windows 2000) 환경 하에서 MS Visual C++ 6.0을 사용하여 구현했고, 메쉬 간단화 공격은 MeshToSS Version 1.0.1의 프로그램을 사용하였다. 그리고 35,947개의 꼭지점 및 69,451개의 패쉬를 가지는 Stanford Bunny 모델의 VRML 데이터를 사용하였다. 그리고 제안한 알고리즘에서 사용되는 워터마크는 길이가 50인 가우시안 랜덤 시퀀스 (Gaussian random sequence)를 1비트로 변환하여 사용하였다. 그러므로 각 패치별 EGI 분포에서 50개의 면체가 필요하며, 한 모델에서 총 300개의 면체에 워터마크를 삽입한다. 워터마크 검출시 사용되는 문턱값 θ_{th} 는 실험적으로 4.8 degree로 결정하였다. 그리고 본 실험에서는 동일한 워터마크의 길이 및 법선 벡터 분포를 이용하는 Benedens의 알고리즘과 성능 비교를 행하였다.

Stanford Bunny 모델 및 제안한 알고리즘에 의하여 워터마크가 삽입된 모델을 그림 9에 나타내었다. 이 그림에서 볼 수 있듯이 시각적으로 워터마크의 삽입 여부를 구분하기 어렵다. 제안한 워터마킹 알고리즘의 견고성을 평가하기 위하여 본 실험에서는 워터마크가 삽입된 모델에 대하여 표면 간단화 (simplification), 절단 (cropping), 및 꼭지점 랜덤화 등의

공격을 행하였다. 다양한 공격에 대한 실험 결과는 표 1에서와 같다. 이 표에서 살펴보면 6개 패치에 대한 300 면체 (bin)에 각각 워터마크 1 비트를 삽입하므로, bin 비트 에러 수는 300 비트 중에서 손실이 발생한 비트 수를 나타내며, bin 비트의 BEP는 300 비트에 대한 비트 에러 수의 퍼센트를 나타낸다. 그리고 1 비트의 워터마크는 6개 패치의 면체에 삽입되므로, 300 비트 중 추출된 워터마크는 총 50 비트이다. 그러므로 워터마크 비트 에러 수는 50 비트 중에서 손실이 발생한 비트 수를 나타내며, 이에 대한 퍼센트는 워터마크 비트의 BEP으로 나타내었다. 제안한 알고리즘에서는 패치별로 워터마크를 삽입하므로, 여러 면체에 대한 비트 손실이 발생하더라도 워터마크 비트 손실은 거의 발생되지 않음을 표에서 알 수 있다.

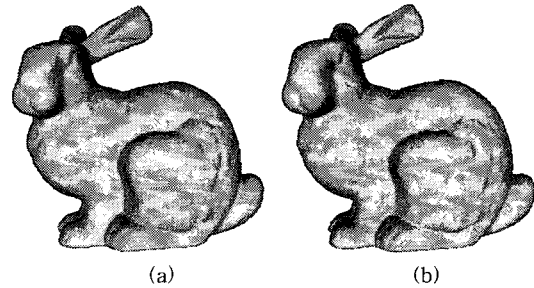


그림 9. 모의 실험에 사용된 (a) Stanford Bunny 모델 및 (b) 제안한 알고리즘에 의하여 워터마크가 삽입된 Stanford Bunny 모델

표 1. 다양한 공격에 대한 견고성 실험 결과(정점의 개수가 35,947인 Stanford Bunny 모델 사용)

Attacks		Proposed algorithm				Benedens	
		# of bin bit error	BEP of bin bit [%]	# of watermark bit error	BEP of watermark bit [%]	# of watermark bit error	BEP of watermark bit [%]
Simplify	28,237 vertex (78.5%)	11	3.66	0	0	2	4
	18,769 vertex (52.2%)	23	7.66	0	0	3	6
	12,157 vertex (33.8%)	44	14.66	1	2	9	18
	6,218 vertex (17.3%)	87	29	8	16	11	22
Random noise 50%		20	6.6	0	0	3	6
Random noise 100%		56	18.67	0	0	8	16
Cropping		13	5.2	0	0	9	18
Simplify (52.2%)+Cropping		38	15.2	1	2	13	26

표면 간단화에 대한 실험에서는 MeshToSS^[8]를 이용하여 워터마크가 삽입된 모델의 꼭지점의 개수를 다양한 비율로 줄였다. 표에서 살펴보면, 30%까지 간단화된 모델에서는 워터마크 비트 손실이 거의 없음을 알 수 있다. 그리고 17%까지 매우 많이 간단화된 모델에서도 워터마크의 84%가 손실없이 검출되었음을 알 수 있다. 그러나 Benedens의 알고리즘에서는 제안한 알고리즘의 워터마크 비트 손실보다 더욱 많음을 알 수 있다. 메쉬 간단화에서는 꼭지점의 개수가 줄어지면서 연결성 정보가 바뀌어진다. 그러므로 원 모델의 메쉬가 없어지거나 새로운 메쉬가 생성된다. 그러나 메쉬 간단화된 모델에서는 원 모델의 형상을 유지하므로, 이에 대한 EGI 크기 분포는 원 모델의 EGI 크기 분포와 유사하다. 하나의 EGI 크기 분포를 이용하는 Benedens의 알고리즘에서는 33%로 간단화된 모델에 대하여 18% 정도의 워터마크 비트 손실이 있다. 즉, 이 모델의 EGI 크기 분포가 원 모델의 EGI 크기 분포와 82% 정도 유사함을 알 수 있다. 제안한 알고리즘에서는 전체 bin 비트 손실이 15% 정도로 Benedens의 알고리즘과 유사한 EGI 크기 분포를 가진다. 그러나 6개 패치의 EGI 크기 분포 집합에 워터마크 정보를 조합적으로 삽입하므로, 워터마크 비트 손실이 Benedens의 알고리즘의 손실보다 매우 적음을 알 수 있다.

꼭지점 랜덤화에 대한 실험에서는 랜덤하게 표본화된 꼭지점 v 을 $v \times (1 + \alpha \times \text{uniform}())$ 로 바꾸었다. 여기서 변조도 α 는 0.008이며, $\text{uniform}()$ 는 [-0.5, 0.5]의 범위를 가지는 균등한 랜덤 함수이다. 전체 꼭지점에 대하여 50% 및 100%로 랜덤하게 표본화된 꼭지점에 대하여 랜덤 잡음을 첨가한 모델에서는 대하여 견고성 실험을 행하였다. 메쉬 간단화 실험에서와 같이 꼭지점 랜덤화 공격에서는 메쉬의 연결성은 유지되지만, 꼭지점의 위치가 바뀌어지므로 메쉬의 표면 법선 벡터가 바뀌어진다. Benedens의 알고리즘에서는 각각 6% 및 16% 정도의 워터마크 비트 손실이 있다. 그러나 제안한 알고리즘에서는 전체 bin 비트 손실이 각각 6% 및 19% 정도이지만, 6개 패치의 EGI 크기 분포에서 삽입된 워터마크 비트가 손실없이 검출됨을 알 수 있다.

절단에 대한 실험에서는 워터마크가 삽입된 모델의 임의의 일정 부분을 제거하였다. 제안한 알고리즘에서는 패치별로 워터마크를 삽입하므로, 남아있는

부분에 속하는 패치에서는 워터마크가 여전히 남아 있다. 그러므로, 절단된 모델에서는 워터마크가 손실 없이 검출됨을 표에서 알 수 있다. 그러나 Benedens의 알고리즘에서는 절단된 부분에 워터마크를 삽입한 메쉬들이 없어진다. 그러므로 그 부분에 삽입된 워터마크 비트의 손실이 있음을 알 수 있다. 워터마크가 삽입된 모델에 대하여 꼭지점의 개수를 33.2%까지 간단화된 모델, 모든 꼭지점에 대하여 랜덤 잡음을 첨가한 모델, 그리고 임의의 일정 부분을 제거한 모델은 그림 10에서와 같다. 이 결과로부터 제안한 3D 워터마킹 알고리즘이 삽입한 워터마크에 대한 비가시성 및 다양한 공격에 대한 우수한 견고성을 가짐을 확인할 수 있었다.

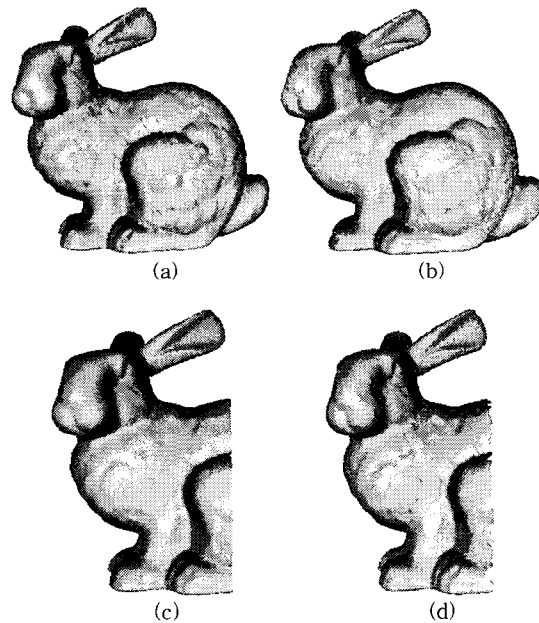


그림 10. 제안한 알고리즘에 의하여 워터마크가 삽입된 모델에 대하여 (a) 꼭지점의 개수를 33.8%로 간단화된 모델, (b) 모든 꼭지점에 대하여 랜덤 잡음을 첨가한 모델, (c) 절단된 모델, 및 (d) 절단된 모델에 대하여 꼭지점의 개수를 52.2%로 간단화된 모델

5. 결 론

본 논문에서는 패치별 EGI 분포를 이용한 3D 메쉬 모델의 워터마킹 알고리즘을 제안하였다. 제안한 알고리즘에서는 기하학적 및 위상학적 변형에 대한 우수한 견고성을 가지기 위하여 동일한 워터마크 비트열을 6개로 분할된 패치의 EGI 분포에 각각 삽입

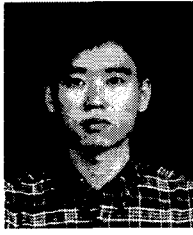
한다. 그리고 워터마크 삽입에서는 인접한 꼭지점들 간의 최소 거리 내에 단계적 탐색 방법을 이용하여 최적의 꼭지점을 찾으므로 비가시성이 향상된다. 또한 워터마크 추출시 복잡한 재표본화 과정 및 원 모델이 필요 없다.

본 논문에서 제안한 알고리즘의 성능을 평가하기 위하여 VRML 3D 데이터의 Stanford Bunny 모델에 대하여 컴퓨터 모의 실험을 행하였다. 이 실험에서 제안한 워터마킹 알고리즘이 비가시성 및 견고성 면에서 우수함을 확인할 수 있었다.

현재 회전 및 이동과 같은 아핀 변환 공격에 대한 연구가 진행 중이다. 즉, 아핀 변환된 모델에서는 워터마크 삽입시 저장된 6개 패치의 중점에 대한 재배열 (realignment) 과정을 이용하면 원 모델에 대한 회전 각도 및 이동 위치를 알 수 있을 것이다. 그리고, 여러 가지 공격에 대하여 각 면체에 대한 사이각들의 변화 분포를 이용하여 워터마크 검출에 필요한 판단 조건인 최적의 문턱치를 구할 수 있을 것이다. 이를 이용하면 워터마크 비트 손실을 줄일 수 있을 것이다. 제안한 알고리즘에서는 전체 형상이 비틀어지는 쉬어링 등의 공격에 대하여 견고성이 매우 약하다. 이와 같은 공격에 대하여 견고성이 우수한 알고리즘이 필요하다. 마지막으로 본 논문에서는 3D VRML 데이터에 적용되는 워터마킹 알고리즘을 제안하였다. 그러므로 CAD, STL, PLY 등의 3D 모델의 데이터에 대한 워터마킹 알고리즘 연구가 진행 중이다.

참 고 문 헌

- [1] R. Ohbuchi, H. Masuda, and M. Aono, "Watermarking Three-Dimensional Polygonal Models Through Geometric and Topological Modification," *IEEE JSAC*, pp. 551-560, 1998.
- [2] R. Ohbuchi, S. Takahashi, T. Miyazawa, and A. Mukaiyama, "Watermarking 3D Polygonal Meshes in the Mesh Spectral Domain," *Proc. Graphics Interface 2001*, pp. 9-17, 2001.
- [3] O. Benedens, "Geometry-Based Watermarking of 3D Models," *IEEE CG&A*, pp. 46-55, Jan./Feb. 1999.
- [4] Oliver Benedens, "Two high capacity methods for embedding public watermarks into 3D polygonal models," *Proceedings of SPIE: Security and Watermarking of Multimedia Contents, SPIE*, pp. 329~340, 1999.
- [5] S. Kanai, H. Date, and T. Kishinami, "Digital Watermarking for 3D Polygons using Multiresolution Wavelet Decomposition," *Proc. Sixth IFIP WG 5.2 GEO-6*, pp. 296-307, Dec. 1998.
- [6] E. Praun, H. Hoppe, A. Finkelstein, "Robust Mesh Watermarking," *Proc. SIGGRAPH 99*, pp. 49-56, 1999.
- [7] M. G. Wagner, "Robust Watermarking of Polygonal Meshes," *Proc. Geometric Modeling & Processing 2000*, pp. 201-208, Apr. 2000.
- [8] B-L. Yeo and M. M. Yeung, "Watermarking 3D Objects for Verification," *IEEE CG&G*, pp. 36-45, Jan./Feb. 1999.
- [9] T. Kanai, MeshToSS Version 1.0.1, <http://graphics.sfc.keio.ac.jp/MeshToSS/indexE.html>.
- [10] Boon-Lock, Y., Minerva, M., "Watermarking 3D Objects for Verification," *IEEE Computer Graphics, Special Issue on Image Security*, pp.36-45, Jan/Feb. 1999,
- [11] 김기석, 천인국, "폐쇄조형 시스템을 위한 3차원 기하학적 형상인 STL의 디지털 워터마킹," *멀티미디어학회 논문지 제5권 제5호*, pp. 552-561, 10. 2002.
- [12] B. K. P. Horn, *Robot Vision*, The MIT Electrical Engineering and Computer Series, MIT Press, Cambridge, Mass., 1986.
- [13] B. K. P. Horn, "Extended Gaussian Image," *Proc. IEEE*, vol. 72, no. 12, pp. 1671-1686, Dec. 1984.
- [14] S. B. Kang and K. Ikeuchi, "The Complex EGI: A New Representation for 3-D Pose Determination," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 15, no. 7, July 1993.



이 석 환

1999년 2월 경북대학교 전자공학과 졸업
 2001년 2월 경북대학교 전자공학과 석사
 2001년 3월~현재 경북대학교 전자공학과 박사과정

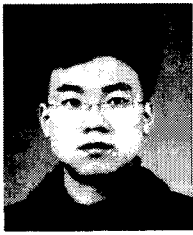
관심분야 : 영상처리, 영상통신, 정보보호



김 태 수

1998년 2월 경북대학교 전자공학과 졸업
 1998년 4월~2001년 10월 LG전자 DID 설계실 연구원
 2003년 2월 경북대학교 전자공학과 석사 졸업
 2003년 3월~현재 경북대학교 전자공학과 박사과정

관심분야 : 영상처리, 정보보호, 3D 그래픽스



김 병 주

1997년 2월 경북대학교 전자공학과 졸업
 1999년 2월 경북대학교 전자공학과 석사
 2003년 8월 경북대학교 전자공학과 박사과정
 2003년 9월~현재 국방과학연구소 연구원

관심분야 : 영상처리, 영상통신, 정보보호



김 지 흥

1986년 2월 경북대학교 전자공학과 졸업(공학사)
 1988년 2월 경북대학교 대학원 전자공학과 졸업(공학석사)
 1996년 8월 포항공과대학교 대학원 전자전기공학과 졸업(공학박사)

1988년 2월~1997년 2월 한국전자통신연구원 선임연구원
 1997년 3월~2002년 2월 부산외국어대학교 컴퓨터공학과 조교수
 2002년 3월~현재 동의대학교 영화영상공학과 조교수
 관심분야 : 영상처리, 컴퓨터그래픽스, 워터마킹 등



권 기 통

1986년 2월 경북대학교 전자공학과 졸업 (공학사)
 1990년 2월 경북대학교 대학원 전자공학과 졸업(공학석사)
 1994년 8월 경북대학교 대학원 전자공학과 졸업(공학박사)
 2000년 7월~2001년 8월 Univ. of Minnesota, Post-Doc. 과정

1996년 3월~현재 부산외국어대학교 컴퓨터전자공학부 부교수
 2002년~현재 한국멀티미디어학회 총무이사
 관심분야 : 멀티미디어 정보보호, 멀티미디어 통신, 웨이브릿 변환



이 건 일

1965년 2월 경북대학교 물리학과 졸업
 1970년 2월 경북대학교 물리학과 이학석사
 1984년 2월 부산대학교 물리학과 이학박사
 1969년~1984년 경북대학교 공과대학 전자공학과 조교-부교수

1982년~1983년 미국 RPI 공과대학 교환교수
 1984년~현재 경북대학교 공과대학 전자공학과 교수
 1995년~현재 산업표준심의회 전자부회 전자관(IEC/TC39) 전문위원회 위원
 관심분야 : 비디오 및 오디오 공학, 전자계측, 의용전자 공학