

3D 형상 데이터의 압축 및 전송을 위한 VRML메쉬의 병합에 관한 연구

장태범*, 문광원**, 정재열***, 김덕수****

Merge of VRML Mesh for 3D Shape Data Compression and Transmission

Jang, T. B.*, Moon, K. W.**, Chung, J. Y.*** and Kim, D. S.****

ABSTRACT

VRML data, which is mainly structural element, is frequently used for modeling and visualizing 3D objects. Although there can be variations, it is a usual practice to represent 3D shapes in VRML format. Ever since the advent of Internet, there have been strong needs to transfer shape data through Internet. Because of this need, it is necessary to transform a data file in VRML or similar format into a more convenient form to transfer through the network. In a VRML file, a model is sometimes divided into a set of triangle meshes due to several practical reasons. However, this causes various demerits for the fast transmission. Therefore, it is more efficient to merge the mesh sets into one mesh set for the transmission. In this paper, we present the problems in the merge process and the techniques to handle the situation.

Key words : VRML, Mesh, Internet, transmission, compression, merge

1. 서 론

제품 정보는 제품의 설계에서부터 제조, 검사, 판매, 보수에 이르는 전 생명 주기(Life Cycle) 동안 그 제품과 관련되어 생성된 정보들을 말한다. 기업에서는 소비자의 요구를 신속하게 만족시키기 위해 제품 정보를 생성하여, 관리하기 위해 노력하고 있으며 그 방법의 하나로써 CAD와 CAM 시스템들을 사용하고 있다. CAD시스템은 개발된 이후로 산업계에서 필수적인 도구로 인식되고 있다. 또한, 2D CAD시스템에서 3D CAD시스템으로 발전하면서 설계자들은 가시화된 3D 형상을 보면서 상호대화적으로 설계를 할 수 있을 뿐만 아니라, 제품 생명 주기 동안 제품 수정이 용이하여 설계 비용이 감소하는 등 여러 효과를 얻을 수 있다^[1].

최근 들어, 제조업에서는 인터넷을 활용하기 위해 노력하고 있다^[2-4]. 그러한 노력들 중의 하나가 인터넷을 통한 원격지의 시스템간에 데이터를 공유하는 것이다. 기업에서는 제품 정보를 인터넷을 통하여 여러 부서에서 효과적으로 공유하여 제품의 생산주기를 단축시키는 등 여러 효과를 얻을 수 있다. 또한, 이렇게 공유되는 정보는 기존의 텍스트나 그림과 같은 2D 정보에 국한되지 않고 3D 형상정보 등 다양한 정보까지 확장되고 있다.

인터넷을 통하여 3D 형상정보를 공유하기 위한 방법 중의 하나가 VRML^[5,6](Virtual Reality Modeling Language)이다. VRML은 인터넷 상에서 3D 형상정보를 포함하는 VR(Virtual Reality)을 구현하기 위한 파일 포맷으로 널리 사용되고 있다. 이러한 이유로 대부분의 상용 CAD시스템들은 인터넷 상에서 3D 형상정보를 쉽게 공유하기 위해 형상 모델을 VRML형식으로 변환하는 기능을 구현하고 있다.

CAD 시스템에서 생성된 VRML파일을 인터넷으로 공유하기 위해서는 많은 어려움이 따른다. 이러한 어려움 중의 하나가 대용량의 데이터이다. CAD 시스템에서는 주로 대용량의 형상 데이터를 다루고 있으며,

*학생회원, 엔지타 기술 연구소
**학생회원, Zionex
***학생회원, 한빛전자통신(주) 중앙연구소
****중신회원, 한양대학교 산업공학과
- 논문투고일: 2001. 6. 12
- 심사완료일: 2001. 11. 20

어는 긴 전송시간을 필요하게 된다. 이와 같은 문제점을 해결하기 위해 해외에서는 3D 형상정보의 압축을 위한 여러 기술들이 연구^[7-11]되어 왔으며 국내에서도 여러 연구^[12-15]가 진행 중에 있다. 그러나, 이러한 연구들을 직접 CAD 시스템에 사용하는 형상정보에 적용하기에는 무리가 있다. 일반적으로 CAD시스템에서는 하나의 3D형상정보를 여러 개의 객체로 분할하여 표현하지만, 국내외에서 진행되고 있는 3D형상정보를 압축하는 기술들은 일반적으로 각 객체에 개별적으로 적용되기 때문에 압축효율이 저하되는 등 여러 문제점이 발생할 수 있다.

본 논문에서는 CAD 시스템에서 생성된 3D 형상정보를 형상 압축 알고리즘을 적용하기 위해 필요한 전처리 과정에 대해서 논의하고자 한다. 기존의 형상 압축 알고리즘을 CAD 시스템에 다루는 형상 정보에 효율적으로 적용하기 위해 여러 객체로 분할된 모델들을 병합하는 알고리즘을 제안하고, 그 과정에서 발생할 수 있는 문제점들을 기술하고자 한다.

2. 3차원 형상 데이터의 인터넷을 통한 공유

2.1 VRML 개요

가상현실은 효과적인 정보 전달을 위하여 현실과 가까운 형태로 정보를 전달하는 기술로서, 3D공간이 표현 가능하며 3D 공간에서 사용자가 원하는 방향으로 조작하거나 실행이 가능하다. 이러한 가상현실을 인터넷에서 구현하는 방법 중의 하나가 VRML이다. VRML은 가상현실을 인터넷 상에서 구현하기 위해 개발한 것으로서 다수의 사용자들 간의 상호 작용, 시뮬레이션, 인터넷을 통한 가상 공간을 탐색하기 위한 언어이다.

VRML은 노드를 기본단위로 물체의 형상, 재질, 그리고 향해, 링크 등을 표현하거나 처리할 수 있다. 여러 노드들 중에 Shape노드는 물체의 형상정보를 표현하는데 사용되며, 하위 노드로 geometry 노드와 appearance 노드를 가지고 있다. geometry 노드는 본 논문에서 차후에 다룰 대상으로서, vertex의 좌표 값이나 normal정보 등과 같이 3D 형상을 정의하기 위한 기본적인 데이터를 정의하며, appearance 노드는 물체의 재질, 색깔, 밝기, 투명도 등을 표현한다^[5,6,16,17].

2.2 VRML이용 시 문제점

VRML에서는 주어진 형상모델을 여러 방법으로 표현할 수 있다. 예를 들면 하나의 Shape 노드를 사용하

```
#VRML V2.0 utf8
Shape { geometry{...}
appearance{...}
...
}
Shape { geometry{...}
appearance{...}
...
}
```

Fig. 1. The diagram of a VRML file.

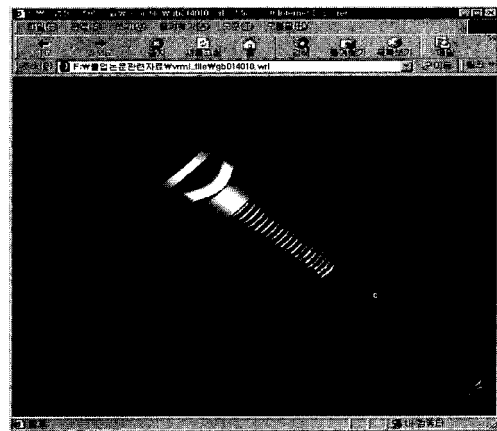


Fig. 2. A VRML file exported from CAD system.

여 표현할 수도 있으며, 여러 개의 shape 노드의 조합으로 표현할 수 있다. 그런데, 대부분의 CAD 모델링 소프트웨어에서 생성된 VRML 파일들은 여러 개의 Shape 노드를 사용하여 물체를 표현하게 된다. Fig. 1은 여러 개의 Shape 노드로 구성된 VRML 파일의 구성도이다.

Fig. 2는 Pro-Engineer CAD 시스템에서 모델링이 된 후 VRML 파일로 변환된 볼트모델^[18]을 보여주고 있다. 이 볼트모델의 VRML파일은 59개의 Shape 노드들로 구성되어 있다. Fig. 3은 Fig. 2의 볼트모델에서 4개의 Shape 노드들을 보여주고 있다. 이러한 Shape 노드는 각각 볼트의 머리, 나사선, 밀면 등을 나타내고 있다. 이처럼, 여러 개의 Shape 노드를 사용하여 하나의 물체를 나타낼 때 다음과 같은 문제점을 발생할 수 있다.

두 개의 Shape 노드에서 생성된 두 면이 인접할 때 각각의 경계를 표현하기 위하여 동일한 vertex들이 각 Shape 노드에서 개별적으로 저장될 수 있다. 이러한 경우에 동일한 vertex가 중복되어 저장될 뿐만 아니라 수치적 오차 등의 이유로 하나의 점이 기하학적으로 다

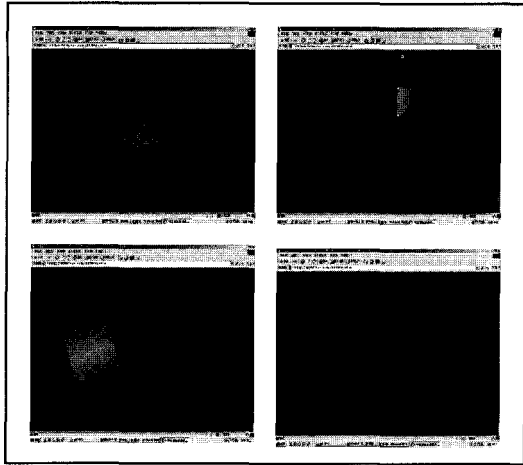


Fig. 3. Shape nodes of a VRML file.

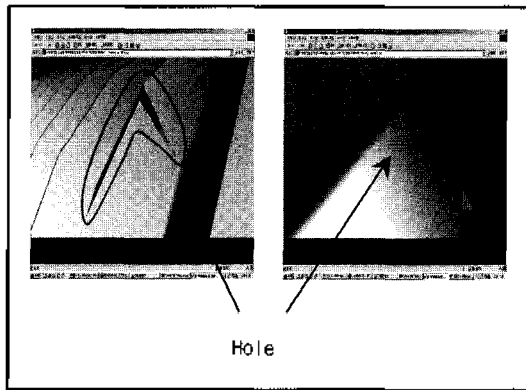


Fig. 4. Examples of holes.

른 점들로 저장되기도 한다. 이는 3D 형상정보를 대형화할 뿐 아니라 Fig. 4와 같이 내부적으로 틈새(Hole)가 발생시키는 등 형상정보를 효과적인 이용을 어렵게 한다.

3. 메쉬 병합

VRML 파일에서 하나의 Shape 노드를 사용하여 표현된 물체를 메쉬라 하겠다. 여러 개의 메쉬를 단일한 메쉬로 병합하는 과정을 설명하고자 한다.

3.1 중복vertex 탐색

VRML 파일 포맷에서 Shape 노드는 여러 가지 방법으로 정의된다. 그 중 많이 사용되는 방법이 IndexedFaceSet을 이용하는 것이다. IndexedFaceSet은

우선 vertex들을 저장한 후, vertex들의 색인을 조합하여 면을 정의하고, 이러한 면들의 집합으로 물체를 표현하는 방법이다. IndexedFaceSet을 이용한 많은 VRML 파일에서는 하나의 메쉬안에 여러 vertex가 중복 정의되는 경우가 빈번하게 발생한다.

이와 같은 IndexedFaceSet으로 정의된 여러 Shape 노드를 병합하기 위해서, 우선 형상을 정의하는 메쉬의 중복 저장된 vertex를 탐색하였다. 이렇게 함으로써, 메쉬를 병합하기 위한 메쉬의 연결정보를 얻을 수 있을 뿐만 아니라, 형상을 표현하는 메쉬 내에 존재할 수 있는 자체적인 중복성을 제거하여 형상정보의 크기를 줄일 수 있다.

3.2 중복 Vertex 병합

여러 메쉬로 표현된 형상정보를 하나의 메쉬로 병합하기 위해서는, 우선 중복된 vertex를 제거하고 이를 일치화 하였다. 즉, 여러 메쉬로 표현된 형상정보는 기하학적으로 하나의 형상으로 보이지만, 위상학적으로는 서로 개별적인 형상들로 표현되어 있어 이를 병합해야 한다. 이러한 문제점을 해결하기 위해, 우선 VRML 파일에서 동일한 기하정보이지만 서로 다른 vertex 색인으로 되어 있는지를 탐색하였다. 탐색된 중복 vertex들에 대해서는 중복성을 제거하고 face에 사용되는 vertex 색인을 일치화하였다.

Fig. 5는 본 논문에서 중복된 vertex를 병합하기 위해 제안한 일련의 알고리즘을 보여주고 있다. 제안된 알고리즘은 전체 메쉬의 중복 vertex를 병합하기 위해 2개의 메쉬에 대한 중복 vertex를 병합하고 이를 반복적으로 적용하였다. 두개의 메쉬의 중복 vertex를 병합하기 위해, 우선 각 메쉬에 존재하는 vertex중에 기하학적으로 동일한 vertex를 찾은 후 중복된 vertex를 제거하였다. 또한, 이 과정 중에 발생할 수 있는 위상적

```

S = S0
while (Sk)
  while (all i, j)
    if (vi = vj)
      then vj = vi, vjindex = viindex
  S = S ⊕ Sk
where , k = number of shapes - 1
vi : vertex of S
vj : vertex of Sk
    
```

Fig. 5. The algorithm for merging meshes.

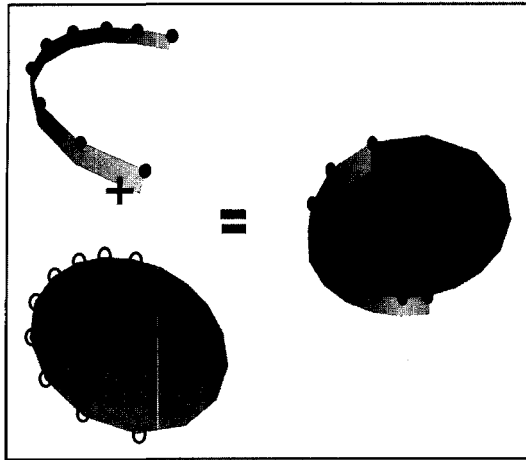


Fig. 6. The merge of meshes.

인 정보를 수정하였다. Fig. 6은 2개의 메쉬로 분할되어 모델링된 나사의 머리부분의 중복 vertex를 병합하는 과정을 보여주고 있다.

3.3 Hole Healing

탐색된 vertex를 병합하면 여러 개의 메쉬에서 하나의 메쉬로 3D 형상정보를 표현할 수 있다. 그러나, 이렇게 병합된 메쉬는 여러 문제점을 발생시킬 수 있다. 그 중의 하나가 수치적 오류 등으로 인하여 병합된 3D 형상정보에 의도하지 않은 hole이 발생한다는 것이다. 이러한 의도하지 않은 hole은 형상압축 알고리즘 적용시 오작동이 발생하는 등 여러 문제점을 유발할 수 있다.

본 절에서는, 주어진 3D 형상정보가 hole이 없다는 가정 하에 hole을 탐색하여 이를 제거하는 알고리즘을 설명하고자 한다.

3.3.1 Hole 탐색

CAD 시스템에서 VRML 변환 시 CAD 모델러가 의도하지 않는 Hole이 수치적인 에러 등과 같은 이유로 발생할 수 있다. 이와 같이 의도하지 않은 Hole을 제거하기 위해서는 먼저 메쉬 내의 hole을 탐색할 필요가 있다.

Vertex병합 알고리즘을 사용하여 병합된 메쉬 내의 hole을 효과적으로 탐색하기 위해 Fig. 7과 같은 winged edge^[19,20] 데이터 구조를 사용하였다. IndexedFaceSet으로 표현된 메쉬를 winged edge 데이터 구조에 저장하기 위해, VRML 파일 안에 있는 vertex 색인에 대해 winged edge 데이터 구조에 vertex를 생성한다. 이러한 과정 후에, VRML 파일 내에서 vertex의 조합으로 표

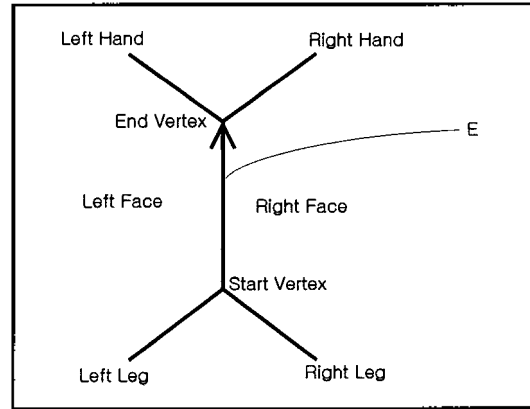


Fig. 7. The winged edge data structure.

현된 face들에 대해 edge와 face를 winged edge데이터 구조에 추가하였다. 특히, 이 과정 중에 edge의 경우 이미 존재 여부를 미리 파악하고, 이미 존재하는 경우에는 기존 edge의 정보를 수정하였다.

위와 같이 Shape 내의 3 차원 형상정보를 winged edge데이터 구조에 담은 후, 두개의 면과 접하지 않은 edge들을 탐색하고 그러한 edge들을 연결하여 hole을 탐색하였다.

3.3.2 유사 Vertex들의 탐색 및 병합

Hole을 구성하고 있는 vertex들은 수치오차 등 여러 요인으로 인해 실제로는 동일한 vertex가 서로 다른 vertex들로 존재할 수 있다. 이러한 점들을 대해 tolerance개념을 도입하여 동일한 vertex로 만들면 형상정보의 큰 변화없이 hole이 제거될 수도 있으며 정보의 용량도 또한 줄어들 수 있다.

Hole을 구성하는 vertex들 중에 유사한 vertex들을 탐색하기 위해 vertex간의 거리를 이용한 절대적인 tolerance를 도입하였다. 상대적인 tolerance 또한 고려해 볼 수 있으나, 이를 결정하기 위해서는 여러 요소를 고려해야 하고 각 상황에 맞추어 달리 적용해야 하는 어려움이 있다. 또한, 수치 오차를 이유로 상이하게 표현된 vertex를 병합하는 것이 목적이므로 절대적인 tolerance를 채택하였다.

여러 절대적인 tolerance값들을 적용하여 vertex 병합을 시도하였다. 이 때 tolerance가 10^{-4} 보다 작은 경우에 vertex를 병합하였더니 약 97%의 정확성을 나타냈고, 10^{-5} 보다 작은 경우에 vertex 병합을 하였더니 약 61%의 정확성을 나타내었다. 따라서 본 논문에서는 hole을 구성하고 있는 vertex들 중에서 거리가 10^{-4} 보다 작은 경우에 두 vertex를 유사한 vertex라고 판단하여

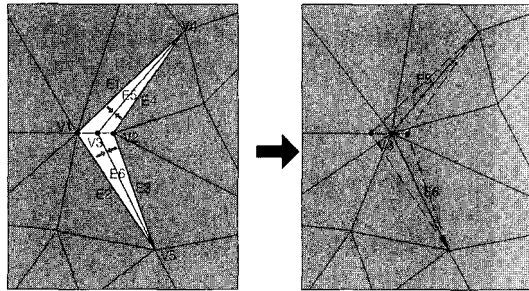


Fig. 8. The merge of similar vertices.

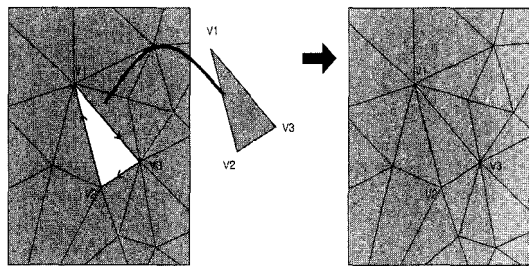


Fig. 9. The insertion of a new triangle.

이들을 하나의 vertex로 병합하였다.

3.3.3 Insertion of new triangles

유사한 vertex들을 탐색하여 병합해도 hole이 제거되지 않은 경우가 발생할 수 있다. 이러한 경우에는 Fig. 9와 같이 새로운 삼각형들을 생성, 추가하여 hole을 제거하였다. 이를 위해 hole을 형성하는 다각형을 생성하고 이것에 triangulation을 적용하여 새로운 삼각형을 생성, 추가하여 hole을 제거하였다.

4. 적용 예

본 논문에서 제시된 알고리즘을 수행하여, 상용CAD 시스템에서 생성된 VRML파일 안의 여러 메쉬를 하나의 메쉬로 병합하였다. Fig. 10은 Fig. 2의 형상정보의 중복된 vertex를 병합한 후에 나타난 hole과 hole이 제거된 결과를 보여주는 것이다.

Table 1은 메쉬를 병합하기 이전의 파일 용량과 병합 후의 파일 용량을 비교한 것이다. 이 표에서 보면 메쉬 병합 알고리즘을 적용하면 파일 용량이 약 40% 이상 감소함을 알 수 있다. 이와 같이, 메쉬 병합 알고리즘은 메쉬를 병합하여 3차원 형상정보를 안정적으로 다룰 수 있게 해줄 뿐만 아니라 파일 용량 자체 또한 감소시킴으로써 3차원 형상정보를 인터넷에 쉽게 공유할 수 있게 해준다.

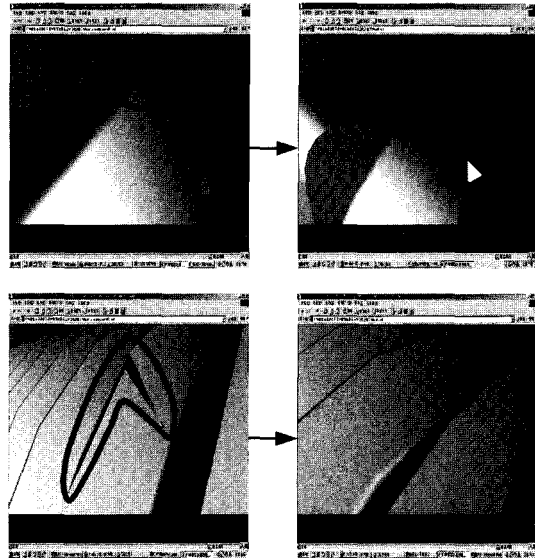


Fig. 10. Healing holes.

Table 1. The file size after merging mesh

파일명	원래 용량 (A)	병합 후 용량 (B)	비율 (B/A)
Bolt_1	660 KB	286 KB	57%
Bolt_2	1003 KB	414 KB	59%
Bolt_3	1252 KB	526 KB	58%
Bolt_4	665 KB	266 KB	60%
Bolt_5	561 KB	201 KB	64%
Bolt_6	498 KB	208 KB	58%
Bolt_7	461 KB	194 KB	58%
Bolt_8	860 KB	354 KB	59%

5. 결 론

인터넷과 컴퓨터 하드웨어의 발달로 3D 물체를 표현하는 대용량의 3D형상정보를 인터넷을 통해 공유하는 기술들이 계속해서 개발, 발전되고 있다.

VRML은 인터넷상에서 3D 물체를 표현하는 표준의 언어로서 자리를 잡아가고 있다. 이러한 추세에 발맞추어, 여러 상용 CAD 시스템들은 각자의 시스템에서 제작된 3D 모델들을 인터넷상에서 보여주기 위해 VRML 파일포맷으로 변환하는 기능을 구현되어 있다. 상용 CAD 시스템에서 VRML 파일을 생성할 때, 여러 개의 Shape 노드를 사용하여 하나의 물체를 나타내기 때문에 다음과 같은 문제점이 발생할 수 있다.

먼저, 두 개의 Shape 노드에 인접한 면에서는 동일

한 점들이 각 노드에 중복적으로 저장될 수 있다. 이렇게 중복되어 저장된 vertex는 파일의 용량을 크게 하며, 전송 시간에 영향을 준다. CAD 시스템에서 VRML 파일을 만들 때, 동일한 점들이 서로 다른 Shape 노드에 미세한 오차를 가지고 저장될 수 있으며, 이러한 오류의 결과로서 hole이 발생할 수 있다. 이러한 hole들은 압축 알고리즘 적용 시에 오류를 발생시키기도 하며 압축효율을 저하시키는 등 여러 문제를 발생시키기도 한다.

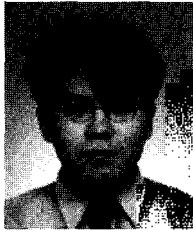
본 논문에서는 이러한 문제점을 직시하고, hole이 제거된 메쉬 병합 알고리즘을 제안하였다. 제안한 알고리즘은 먼저 중복된 vertex를 제거하고, 3D 형상정보에 포함되어 있는 hole을 탐색, 제거함으로써, 3D 형상정보를 인터넷으로 공유하기 위한 여러 압축 및 전송 알고리즘의 효율적인 적용을 가능하게 할 뿐만 아니라 3차원 형상정보의 용량을 축소시킬 수 있다.

감사의 글

이 논문은 2001년 한양대학교 교내연구비 지원으로 연구되었음.

참고문헌

1. McMahon, C. and Browne, J., *CAD/CAM principles, practice and manufacturing management*, 2nd Edition, Addison Wesley, 1998.
2. Cha, J. H., Lee, S. J. and Jeoun, H. Y., "Internet-Based Remote Control System Using Virtual Reality," *Transactions of the Society of CAD/CAM Engineers*, Vol. 5, No. 1, pp. 88-94, 2000.
3. Chung, J. H. and Lee, K. W., "Development of Collaboration Framework for Injection Model Design," *Transactions of the Society of CAD/CAM Engineers*, Vol. 5, No. 1, pp. 33-41, 2000.
4. Yang, S. and Choi, Y., "Co-Des: Real Time Collaborative Design System," *Transactions of the Society of CAD/CAM Engineers*, Vol. 5, No. 1, pp. 42-49, 2000.
5. Ames, A. L., Nadeau, D. R. and Moreland, J. L., *VRML 2.0 Source Book*, Wiley, 1997.
6. Carey, Rikk and Bell, Gavin, *The Annotated VRML2.0 Reference Manual*, A-W Developers Press, 1997.
7. Chow, M. M., "Optimized Geometry Compression for Real-time Rendering," *IEEE Visualization*, 1997.
8. Deering, Michael, "Geometry compression," *Proceedings SIGGRAPH 95*, pp.13-20, ACM SIGGRAPH, 1995.
9. Garland, M. and Heckbert, P. S., "Surface Simplification using quadric error metrics," *SIGGRAPH 97*, pp. 209-216, 1997.
10. Hoppe, H. DeRose, T., Duchamp, T., McDonald, J. and Stuetzle, W., "Mesh optimization," *SIGGRAPH 93*, pp. 19-26, 1993.
11. Rossignac, Jarek, "Edgebreaker : Compressing the incidence graph of triangle meshes," *Technical Report GIT-GVU*, 1998.
12. Kim, Y.-S., Park, D.-G., Jung, H.-Y., and Cho, H.G., "An improved TIN compression using Delaunay triangulation," *Proceedings of Pacific Graphics '99*, pp.118-125, 1999.
13. Lee, E.-S. and Ko, H.-S., "Vertex data compression for triangle meshes," *Eurographics 2000*, Vol. 19, No. 3, pp. 1-10, 2000.
14. Joo, S. W., Lee, S. H. and Park, K. H., "Mesh Decimation for Polygon Rendering Based Real-Time 3-Axis NC Milling Simulation," *Transactions of the Society of CAD/CAM Engineers*, Vol. 5, No. 4, pp. 347-358, 2000.
15. Kang, S. C., Lee, K. Y. and Kim, T. W., "Normal Meshes for Multiresolution Analysis on Irregular Meshes with a Boundary," *Transactions of the Society of CAD/CAM Engineers*, Vol. 6, No. 3, pp. 184-192, 2001.
16. VRML 2.0 Specification, at <http://vrml.sgi.com/moving-worlds/index.html>, 1996.
17. VRML 97 Specification, at <http://www.vrml.org/Specifications/VRML97/DIS/index.html>, 1997.
18. VRML file of bolt at <http://www.partdb.co.kr>, 2000.
19. Lee, K., *Principles of CAD/CAM/CAE Systems*, Addison Wesley, 1998.
20. Mantyla, M., *An Introduction to Solid Modeling*, Computer Science Press, 1988.



장 태 범

1997년 한양대학교 산업공학과 학사
 1999년 한양대학교 산업공학과 석사
 2002년 한양대학교 산업공학과 박사 과정 수료
 2002년~현재 (주)엔지티 부설 엔지티기술 연구소 연구원
 관심분야: Geometric Modeling, Optimization, Computer Graphics and Computational Geometry



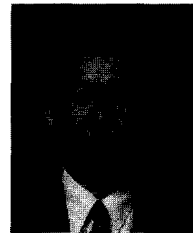
문 광 원

1999년 한양대학교 산업공학과 학사
 2001년 한양대학교 산업공학과 석사
 2001년~현재 Zionex T³ plan 전임연구원
 관심분야: Computer Graphics and Internet Applications



정 재 열

2000년 한양대학교 산업공학과 학사
 2002년 한양대학교 산업공학과 석사
 2002~현재 한빛전자통신(주) 중앙연구소 연구원
 관심분야: Computational Geometry, 3D Data Compression, Web Application, Computer Graphics



김 덕 수

1982년 한양대학교 산업공학과 학사
 1985년 New Jersey Institute of Technology 산업공학과 석사
 1990년 The University of Michigan 산업공학과 박사
 1989년~1991년 Schlumberger Technology CAD/CAM Co. Senior Software Engineer
 1991년~1995년 삼성종합기술원 선임연구원
 1995년~현재 한양대학교 산업공학과 부교수
 관심분야: Geometric Modeling, Computational Geometry, STEP and Internet Applications